A unified approach to congestion control and node-based multipath routing

Fernando Paganini, *Senior Member, IEEE,* and Enrique Mallada

***Abstract*—The paper considers a TCP/IP-style network with flow control at end-systems based on congestion feedback, and routing decisions at network nodes on a per-destination basis. The main generalization with respect to standard IP is to allow routers to split their traffic in a controlled way between the outgoing links.**

We formulate global optimization criteria, combining those used in the congestion control and traffic engineering, and propose decentralized controllers at sources and routers to reach these optimal points, based on congestion price feedback. We first consider adapting the traffic splits at routers to follow the negative price gradient; we prove this is globally stabilizing when combined with primal congestion control, but can exhibit oscillations in the case of dual congestion control. We then propose an alternative anticipatory control of routing, proving its stability for the case of dual congestion control.

We present a concrete implementation of such algorithms, based on queueing delay as congestion price. We use TCP-FAST for congestion control, and develop a multipath variant of the distance vector routing protocol RIP. We demonstrate through ns2-simulations the collective behavior of the system, in particular that it reaches the desired equilibrium points.

I. Introduction

The congestion present in a packet-switched network at any given time is a function of the amount of traffic introduced by the transport layer, and of the routes chosen by the network layer to carry this traffic to destination. Ideally, both rates and routes should be controlled to ensure the most efficient and fair utilization of the available bandwidth. However, while TCP congestion control is in widespread use, it has been traditionally difficult to adapt the network layer to congestion, except at the very slow time-scales, where traffic engineering is used for congestion planning. A large part of the difficulty lies in the use by IP routers of single paths to destination. Attempting to adapt such paths to instantaneous congestion results in routing instabilities, observed since the early days of the Arpanet, and well documented in academic studies [2], [24]. In contrast, multipath routing can more easily reach equilibrium: instead of drastic switches of large bulks of traffic, it can gradually adapt the traffic mix between different routes.

Mathematically, the distinction between single path and multipath routing reveals itself when we consider the optimization of a convex congestion cost to serve a matrix of end-to-end demands. This problem is non-convex when one optimizes over single-path routes, but its relaxation to multipath amounts to convex multicommodity optimization. References on these problems and their use for traffic engineering are [2], [21], [6], [5].

The optimization interpretation is particularly useful if one seeks to combine multipath routing with congestion control, since the latter has also been framed in terms of convex optimization of utility [10], [14], [22]. Indeed, a multipath proposal is already present in Kelly's original work [10] (see also [8], [12], [23]): here one defines a rate variable for each *end-to-end path* from source to destination, and the sources control all these variables to optimize overall utility. This proposal, while mathematically well behaved, implies essentially to transfer functionality from the network layer to the transport layer, which must now be aware of paths inside the network, and has scalability problems since the number of such paths is exponential.

A more scalable, node-centric alternative is to have routers take charge of the multipath function, by controlling the split of traffic to each destination among their outgoing links. This idea goes back to Gallager [7] (followed by [1]) for the case of fixed input traffic; in that work the traffic split is adapted following the gradient of an overall cost function, interpreted as network delay. Following some ideas in [2], this approach can also be adapted to include "primal" flow control, as shown in [25], which also combines it with a gradient algorithm for power control in the case of a wireless network. Other cross-layer work with the node-centric view for wireless networks is [4], [13], where instead of the smooth adaptation of traffic splits, the object of optimization is the scheduling of wireless transmissions at each time slot. In [18], we proposed a combination of gradient based multipath routing with primal and dual congestion control, identifying the criteria that are optimized in each case, and giving partial stability results.

In this paper we extend and develop the work of [18] in many respects. In Section II we present the formulation from [18] and in Section III we review the stability theory for primal laws with gradient-based multipath adaptation as in [7]. We find, however, that this method is unable to provide dynamic stability in the case of dual congestion control: oscillatory instabilities can occur due to the second order nature of the dynamics. In response to this, we propose in Section IV an anticipatory approach to route adaptation, where the control

Electrical and Telecommunications Engineering, Universidad ORT, Uruguay. e-mail: {paganini,mallada}@ort.edu.uy.Research supported in part by PDT-Uruguay, and by AFOSR-US, grant FA9550-06-1-0511.

of traffic splits includes a “derivative action” term that induces stability. We prove that this control achieves the global, social welfare equilibrium, with local stability under any network. Some proofs are relayed to the Appendices.

We follow the theoretical work with a discussion on implementation in Section V, leading to a proposal which demonstrates the practical feasibility of the approach and its moderate requirements with respect to current practice in TCP-IP networks. Of the different alternatives for congestion price, we focus here on queueing delay, used by TCP variants such as Vegas [3] and FAST [9]. The information requirements between nodes are similar to those in a distance vector protocol, here with congestion price as a measure of distance; we develop a variant of RIP [19] that suits this purpose. We also introduce modifications to TCP-FAST to suit the multipath setting. In Section VI we present simulation work in ns2 that demonstrates the properties of this implementation, and in particular validates the analytical studies. Conclusions are given in Section VII.

II. Problem formulation

We consider a network made up of a set of nodes $\mathcal{N}$, and a set of directed links $\mathcal{L}$ between them. Nodes, denoted by the indices i and j, can be sources or destinations of packets, or intermediate router nodes. We describe the links either by a single index l, or by the directed pair (i, j) of nodes they connect.

The network supports various flows between source-destination pairs of nodes. We use the index $k \in \mathcal{K}$ to denote an individual flow or “commodity”, and $s(k)$, $d(k)$ denote respectively the corresponding source and destination nodes. While these are unique for each k, we allow the traffic to follow multiple paths between source and destination. This is modeled through the following variables for each k:

- x^k, external flow in packets per second entering the network at the source;
- y_l^k, flow through link l;
- x_i^k, total flow coming into node i.

At the source node the incoming flow is only external,

$$x_{s(k)}^k = x^k. \tag{1}$$

The inflow balance equation for node j is

$$x_j^k = \sum_{i:(i,j)\in\mathcal{L}} y_{i,j}^k, \quad j \neq s(k), \tag{2}$$

and the outflow balance of node i is

$$x_i^k = \sum_{j:(i,j)\in\mathcal{L}} y_{i,j}^k, \quad i \neq d(k). \tag{3}$$

The total flow on link l is given by

$$y_l = \sum_k y_l^k. \tag{4}$$

A. Optimization problems

Following [10], we associate with each commodity k an increasing, concave utility function $U_k(x^k)$ that specifies the flow’s demand for rate. We formulate a multipath counterpart of the “system problem” in [10].

Problem 1 (WELFARE): Maximize $\sum_k U_k(x^k)$, subject to link capacity constraints $y_l \leq c_l$, and flow balance constraints (1-4).

The solution of this convex program gives the maximum achievable utility over all sources if traffic is allowed to follow multiple routes between source and destination. Our main objective in this paper is to devise a decentralized control system at routers and sources to achieve this optimum.

A second problem can be formulated replacing capacity constraints with barrier functions $\phi_l(y_l)$ that specify the congestion cost at the link. We assume $\phi_l(y_l)$ is increasing and convex in y_l.

Problem 2 (SURPLUS): Maximize

$$S := \sum_k U_k(x^k) - \sum_l \phi_l(y_l), \tag{5}$$

subject to flow balance constraints (1-4) .

In economic terms, S above is the *aggregate surplus* (see e.g. [16]). As a special case when demand is inelastic, it includes the optimal routing problem of minimizing $\sum_l \phi_l(y_l)$ for a fixed traffic matrix. This was the problem studied in [7] for ϕ_l interpreted as delay; for that case, the generalization (5) was developed in [25].

B. Control variables

By appropriate redefinition of the variables, the above problems can be shown to be equivalent to those considered in [10], [12], [23], in terms of rates per route or path across the network. As argued before, for reasons of scalability as well as preserving layer separation, we prefer to use the following set of variables:

- The transport layer at sources should control only the total amount of rate x^k they input to the network, similar to current TCP congestion control.
- As in IP, the network layer at routers makes forwarding decisions based on destination; the only change is that multiple next hops by destination can be used. The variable $\alpha_{i,j}^d$ controls the fraction of traffic to destination d that uses outgoing link (i, j).

Specifically, we can impose the split in each commodity,

$$y_{i,j}^k = \alpha_{i,j}^{d(k)} x_i^k, \qquad (i,j) \in \mathcal{L}; \tag{6}$$

or alternatively the weaker condition

$$\sum_{k:d(k)=d} y_{i,j}^k = \alpha_{i,j}^d \sum_{k:d(k)=d} x_i^k, \qquad (i,j) \in \mathcal{L}. \tag{7}$$

The latter imposes the split on total rates per destination, the factor that impacts congestion, but allows more freedom in routing individual flows. Indeed, it may be advantageous in practice to keep individual flows single path, still achieving or (or approximating) the overall multipath traffic split. We will mostly use (6) for simplicity, but the theory extends to (7).

It is possible to infer from (6) an overall relationship between the vector $x = \{x^k\}$ of input source rates, and the vector $y = \{y_l\}$ of total link rates. Given a set of split ratios $\alpha = \{\alpha^d_{i,j}\}$, under some mild assumptions on connectivity, the relationship $y = R(\alpha)x$ follows from (1-4) and (6), see Appendix A. The *routing matrix* $R(\alpha)$ generalizes the single-path routing case, where it is simply a matrix of zeros and ones.

If α is fixed, replacing the constraints in Problem 1 and Problem 2 by the relationship $y = R(\alpha)x$ yields the optimization problems over input rates considered in the congestion control literature [10], [22], generalized to fixed multipath routes. Our main concern, however, is to use the routing splits as *variables*, and control them to solve the original problems with unconstrained routing.

C. Feedback signals

As in congestion control, the primary feedback signal is a congestion measure or *price* p_l for each link $l \in \mathcal{L}$; we assume p_l depends only the total traffic y_l; there is no "service differentiation" between commodities. Later on we discuss choices on how to define this price.

In our multipath setting, different paths from a node to destination will have their own congestion prices at any time. We do not require routers to be aware of such paths; rather, they can work with local and neighbor information to infer their price-to-destination q^d_i, $i \in \mathcal{N}$, representing the average price of sending packets from node i to destination d, using the current routing patterns. Node prices are thus defined to satisfy

$$\begin{aligned} q^d_d &= 0, \\ q^d_i &= \sum_{j:(i,j)\in\mathcal{L}} \alpha^d_{i,j}[p_{i,j} + q^d_j], \quad i \neq d. \end{aligned} \tag{8}$$

Given link prices $p_{i,j}$, under mild conditions there exist unique solutions q^d_i to the above recursive equations. More details are given in Appendix A. At the source node of commodity k, the node price summarizes the congestion cost of the network. We denote it by

$$q^k := q^{d(k)}_{s(k)}.$$

To determine the q^d_i in a decentralized network requires communication across neighboring nodes, and recursive updates that take time to converge. We do not model these dynamics; further comments are given in Section V. We are also not modeling the *delays* incurred in propagation of rates through the network and of congestion prices in feedback, considered in e.g. [15], [22] for congestion control. This simplification is done for mathematical tractability. As partial justification we mention that our main focus is the much slower time-scale in which routing adaptation can take place.

Remark: Congestion prices from node to destination have also been considered in the literature on "back-pressure" scheduling in wireless networks (e.g., [13], [4]), motivated by the Lagrangian dual of Problem 1 with respect to the node balance constraints. In fact, we show in Appendix B that under our proposed control, the *equilibrium* values of node prices correspond to these Lagrange multipliers. The *dynamics* of both proposals are, however, very different, as are the resulting implementations. In the backpressure work, q^d_i is dynamically controlled, and routing is deduced from it, by scheduling at each time slot the commodity with the highest price differential. Around equilibrium, these price differentials will equalize, and routing "chatters" between paths; the traffic split is never explicitly found, it can only be interpreted in a mean sense as emerging from such fluctuations. In this paper we take, in a sense, the opposite path: the routing splits $\alpha^d_{i,j}$ will be explicitly controlled as a "primal" variable, and the evolution of q^d_i will follow from (8) as a consequence. This induces a different dynamics of these prices, and also enables different forwarding implementations, as discussed later.

D. Basic relationships

The following basic lemma relates the price and flow variables defined so far.

Lemma 1: For each commodity k,

$$x^k q^k = \sum_{l\in\mathcal{L}} y^k_l p_l. \tag{9}$$

Proof: We write the sequence of identities

$$\begin{aligned} \sum_{i\in\mathcal{N}} x^k_i q^{d(k)}_i &= \sum_{i\in\mathcal{N}\setminus d(k)} x^k_i \sum_{j:(i,j)\in\mathcal{L}} \alpha^{d(k)}_{i,j}(p_{i,j} + q^{d(k)}_j) \\ &= \sum_{(i,j)\in\mathcal{L}} y^k_{i,j}(p_{i,j} + q^{d(k)}_j) \\ &= \sum_{l\in\mathcal{L}} y^k_l p_l + \sum_{j\in\mathcal{N}\setminus s(k)} q^{d(k)}_j \sum_{i:(i,j)\in\mathcal{L}} y^k_{i,j} \\ &= \sum_{l\in\mathcal{L}} y^k_l p_l + \sum_{j\in\mathcal{N}\setminus s(k)} q^{d(k)}_j x^k_j. \end{aligned}$$

The first identity is from (8), the second uses (6); the third step follows by grouping terms by the end-nodes of the links, and the last step uses (2). Now cancelling node terms, only the source term $x^k q^k$ remains on the left-hand side.

∎

The following identity is obtained by aggregating over the various commodities:

$$\sum_k x^k q^k = \sum_{l\in\mathcal{L}} y_l p_l. \tag{10}$$

In an analogous way, one can establish the following dynamic relationship, which holds regardless of the chosen control laws, to be defined later:

$$\dot{x}^k q^k = \sum_{l\in\mathcal{L}} \dot{y}_l^k p_l - \sum_{(i,j)\in\mathcal{L}} x_i^k \dot{\alpha}_{i,j}^{d(k)} [p_{i,j} + q_j^{d(k)}]. \quad (11)$$

III. Gradient route adaptation with congestion control

Having defined the controlled variables (input rates and traffic splits) and the feedback variables (link and node prices), what remains is to choose the control laws that map between them, to run at sources and routers.

Focusing on a router i, for each d we must define a control law for the vector $\alpha_i^d := \{\alpha_{i,j}^d\}_{j:(i,j)\in\mathcal{L}}$ of routing splits as a function of

$$\pi_i^d := \{p_{i,j} + q_j^d\}_{j:(i,j)\in\mathcal{L}}, \quad (12)$$

the vector of prices to destination d seen from node i. Both vectors have the dimension of L_i, the number of outgoing links at i, and α_i^d belongs to the unit simplex

$$\Delta_i = \{\alpha_{i,j}^d \geq 0 : \sum_{j:(i,j)\in\mathcal{L}} \alpha_{i,j}^d = 1\}. \quad (13)$$

Our first choice for the control of $\alpha_{i,j}^d$ is to follow the negative price gradient: to transfer traffic in gradual steps from more expensive to cheaper routes. This includes the proposal from [7] when prices are interpreted as marginal costs, as seen below. In continuous time, we impose the following conditions on the derivative $\dot{\alpha}_i^d$:

(i) $\dot{\alpha}_i^d$ is negatively correlated with π_i^d, i.e.

$$\sum_{j:(i,j)\in\mathcal{L}} \dot{\alpha}_{i,j}^d (p_{i,j} + q_j^d) \leq 0, \quad (14)$$

with equality only if $\dot{\alpha}_i^d = 0$.

(ii) $\dot{\alpha}_i^d$ is constrained so that the trajectory remains on the simplex. In particular, it must satisfy

$$\sum_{j:(i,j)\in\mathcal{L}} \dot{\alpha}_{i,j}^d = 0. \quad (15)$$

(iii) $\dot{\alpha}_i^d = 0$ if and only if for each $j : (i,j) \in \mathcal{L}$,

$$\begin{aligned} &\text{either } \; q_i^d = p_{i,j} + q_j^d, \\ &\quad \text{or } \; \alpha_{i,j}^d = 0 \text{ and } q_i^d < p_{i,j} + q_j^d. \end{aligned} \quad (16)$$

In other words, split ratios per node only settle when prices of routes that carry traffic have equalized (and thus are equal to the node price) and the remaining unused routes have higher price.

There may be more than one choice of control satisfying these restrictions. A specific one is

$$\dot{\alpha}_i^d = \beta_i E_{\alpha_i^d}[-\pi_i^d], \quad (17)$$

where $\beta_i > 0$ and $E_{\alpha_i^d}$ denotes a projection operation required to keep the trajectory within the simplex Δ_i. In the special case when α_i^d is *interior* to Δ_i ($\alpha_{i,j}^d > 0 \; \forall j$) the projection must simply enforce (15), so in this case it is given by the matrix

$$E = I - \frac{1}{L_i}\mathbf{1}\cdot\mathbf{1}^T, \quad (18)$$

where the identity matrix and the vector of ones $\mathbf{1}$ have the dimension L_i. E applied to a vector subtracts the mean from each component. So, for an interior α_i^d, (17) is simply $\dot{\alpha}_{i,j}^d = \beta_i(\overline{\pi_i^d} - \pi_{i,j}^d)$; namely, increase routing in links with lower-than-average prices, decrease it in the rest.

For α_i^d on the boundary of Δ_i, we want to allow the motion of $\alpha_{i,j}^d$ away from zero (to explore new routes), but restrict it to be non-negative, so as to remain in the simplex Δ. So, if the drift vector v points outside of Δ_i, we will replace it by its best approximation within Δ_i. We formalize this as follows: for $a \in \mathbb{R}^{L_i}$, let $\Psi_{\Delta_i}(a)$ denote the point in Δ_i closest to a, i.e.

$$\Psi_{\Delta_i}(a) = \operatorname{argmin}_{b\in\Delta_i}|a-b|.$$

Now define

$$E_{\alpha_i^d}[v] := \lim_{\epsilon\to 0+} \frac{\Psi_{\Delta_i}(\alpha_i^d + \epsilon v) - \alpha_i^d}{\epsilon}. \quad (19)$$

Since the boundary of Δ_i is piecewise linear, the limit in (19) is in fact achieved for small enough $\epsilon > 0$, for which $\alpha_i^d + \epsilon E_{\alpha_i^d}[v]$ becomes the point in the simplex closest to $\alpha_i^d + \epsilon v$.

Remark: Adapting routes gradually means routing loops could appear during a transient phase. To avoid this, a *blocking method* was proposed in [7] which checks information on prices further downstream before initiating transmission to a neighbor. Details on this procedure are given in Section V. Blocking still respects the first two conditions above but weakens the third. While (16) is still sufficient for $\{\dot{\alpha}_{i,j}^d\} = 0$, the necessity is no longer true: a certain route may remain blocked despite being cheaper.

We will now combine the route adaptation defined above, with two choices of congestion control algorithms studied in the literature. We use the notation

$$[w]_z^+ := \begin{cases} w, & \text{if } w > 0 \text{ or } z > 0; \\ 0 & \text{otherwise.} \end{cases}$$

A. Primal congestion control and global stability

Consider the scenario in which source rates are updated as in [10] by the *primal* equations

$$\dot{x}^k = \kappa(x^k)[U_k'(x^k) - q^k]_{x^k}^+, \quad (20)$$

where $\kappa(x^k) > 0$, and link prices follow the static law

$$p_l := \phi_l'(y_l), \quad (21)$$

i.e. the price is the marginal cost of the link.

The state variables of the system are the source rates and node split ratios. We study the asymptotic behavior.

Theorem 2: Under (20-21), and assumptions (i)-(iii) on the control of α, the system rates converge globally to a solution of Problem 2.

Proof: We take the derivative of the surplus along system trajectories,

$$\begin{aligned}\dot{S} &= \sum_k U_k'(x^k)\dot{x}^k - \sum_l \phi_l'(y_l)\dot{y}_l \\ &= \sum_k [U_k'(x^k) - q^k]\dot{x}^k + \sum_k q^k \dot{x}^k - \sum_l p_l \dot{y}_l \\ &= \sum_k \kappa(x^k)[U_k'(x^k) - q^k][U_k'(x^k) - q^k]^+_{x^k} \\ &\quad - \sum_k \sum_{(i,j)\in\mathcal{L}} x_i^k \dot{\alpha}_{i,j}^{d(k)}(p_{i,j} + q_j^{d(k)}), \end{aligned} \tag{22}$$

where we have invoked (20-21) and (11), added over k. Both of the above sums are non-negative, using (14); so $\dot{S} \geq 0$, the surplus increases along trajectories. While $\dot{S} = 0$ may occur outside equilibrium, a careful application of Lasalle's invariance principle implies the stability result, see Appendix B.

■

B. Dual congestion control

We now consider the "dual" congestion control algorithm first proposed in [14], where link prices follow

$$\dot{p}_l = \gamma_l [y_l - c_l]^+_{p_l}. \tag{23}$$

Based on the received price q^k, the sources choose a rate that instantaneously maximizes $U_k(x^k) - q^k x^k$, i.e. $x^k = f_k(q^k)$ satisfying

$$U_k'(x^k) = q^k, \text{or } x^k = 0 \text{ and } U_k'(x^k) < q^k. \tag{24}$$

Proposition 3: Consider an equilibrium of the dynamics (23), with rates satisfying (24) and routing splits α satisfying the equilibrium condition (16). Then the rates are a solution to Problem 1.

Proof: see Appendix B.

Remark: The equilibrium is not necessarily unique, but all equilibria are optimal.

Can we claim convergence to equilibrium under the chosen dynamics? This fact is more delicate than in the primal case. To gain insight, consider the restriction of Problem 1 to fixed routing splits α:

$$\psi(\alpha) := \max \sum_k U_k(x^k), \tag{25}$$
$$\text{subject to } y = R(\alpha)x \leq c.$$

Introduce the Lagrangian of this problem with respect to the capacity constraints, and use (10):

$$\begin{aligned} L(\alpha, p, x) &= \sum_k U_k(x^k) + \sum_l p_l(c_l - y_l) \\ &= \sum_k [U_k(x^k) - q^k x^k] + \sum_l p_l c_l. \end{aligned} \tag{26}$$

Its maximum over x for fixed α and p is achieved precisely by the source law (24), let us denote it by

$$W(\alpha, p) := \max_x L(\alpha, p, x).$$

From convex duality, the minimum of the above over $p \geq 0$ is $\psi(\alpha)$, so the solution to Problem 1 is

$$\psi^* = \max_\alpha \psi(\alpha) = \max_\alpha \min_p W(\alpha, p), \tag{27}$$

a *saddle point* of the function W. From this perspective, we can give an interpretation for the dual dynamics. Applying the envelope theorem (see [16]), the partial derivatives of W can be computed on the Lagrangian L, at the maximizing x, leading to:

$$\frac{\partial W}{\partial p_l} = c_l - y_l,$$
$$\frac{\partial W}{\partial \alpha_{i,j}^d} = -\sum_k \frac{\partial q^k}{\partial \alpha_{i,j}^d} x^k = -\sum_{k:d(k)=d} x_i^k [p_{i,j} + q_j^d].$$

We omit the derivation of the last identity, which follows from (8) and a similar argument to Lemma 1. The implication is that the dynamics (23) is a gradient projection algorithm for the minimization over p in (27), whereas from (14) we have

$$\langle \dot{\alpha}_i^d, \frac{\partial W}{\partial \alpha_i^d} \rangle \geq 0 \quad \text{for each } i,$$

where $\langle\ ,\ \rangle$ denotes Euclidean inner product. Thus α moves in the direction of the *maximization* in (27). The fact that both controls produce opposite effects on W makes it difficult to conclude something about the joint dynamics. In [18] we performed a *two time-scale* analysis, obtaining convergence results under the assumption that α varies much more slowly than p. It is, unfortunately, not true that the dynamics will make the equilibrium stable when the separation of time-scales is not ideal as assumed in [18].

Example 1: Consider a simple network with two nodes (source and destination) and two parallel links, of capacity c_1, c_2. Each link generates a price according to (23). The traffic split can be described in this case by a single parameter $\alpha := \alpha_1$, with $\alpha_2 = 1 - \alpha$. An update that follows the negative price gradient has the form

$$\dot{\alpha} = \beta(p_2 - p_1),$$

with saturation to the interval $[0, 1]$. The equilibrium is $x^* = c_1 + c_2$, $\alpha^* = c_1/x^*$, with $p_1^* = p_2^* = q^*$ depending on the chosen utility function. To simplify the analysis, let us temporarily replace the source by an inelastic one with rate $x \equiv x^*$. Also, consider a trajectory for which the saturation constraints on α, p_1, p_2 remain inactive. Denoting $\delta\alpha = \alpha - \alpha^*$, $\delta p_i = p_i - p_i^*$, the dynamics becomes *linear:*

$$\begin{bmatrix} \delta\dot{\alpha} \\ \delta\dot{p}_1 \\ \delta\dot{p}_2 \end{bmatrix} = \begin{bmatrix} 0 & -\beta & \beta \\ \gamma_1 x^* & 0 & 0 \\ -\gamma_2 x^* & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta\alpha \\ \delta p_1 \\ \delta p_2 \end{bmatrix}. \tag{28}$$

The eigenvalues of the preceding matrix are 0 and $\pm j\sqrt{\beta(\gamma_1 + \gamma_2)x^*}$. The 0 eigenvalue is a consequence of having introduced the inelastic source, which makes the equilibrium price indeterminate. The purely imaginary mode is of more concern: it reveals a harmonic oscillation of the price and split dynamics, which could have as large an amplitude as the saturation constraints allow. Essentially, we are seeing the "route flaps" of congestion-based single path routing turning up again, in a smoother form, in our multipath algorithm.

If the elastic source is introduced back in the problem, the dynamics is no longer linear. Nevertheless, we can say the following: for the important case where $\gamma_l = 1/c_l$ (i.e., price represents queueing delay), the linearization around equilibrium replaces the mode at zero with a stable eigenvalue, but the imaginary modes remain. Moreover, through a Lyapunov analysis similar to Proposition 4 below we find that asymptotically the source rate must converge to x^* as above, with dynamics of α and p approaching the one in (28), and thus exhibiting possibly large oscillations.

Remark: The above example raises a modeling question that has been a subject of debate. Should delay be modeled as a *static* function of link rate, or by the "fluid tank" model of (23) (with $\gamma_l = 1/c_l$)? The former follows from classical queueing theory in steady state, the latter captures the transient dynamics but not the stochastic effects. In this example, this question has central importance: if the static model is adopted, as is done in [7], [1], [25], then Theorem 2 implies asymptotic convergence of the traffic splits. If, instead, the model (23) holds true, the above analysis predicts oscillations. Which is correct? We will see in packet simulations in Section VI that indeed oscillations are observed, giving support to the second model, and indicating that the stability of delay-based multipath routing requires a different approach.

IV. Anticipative control of traffic splits and its stability

The preceding example reveals a limitation with the adaptation of multipath routing based on the gradient of congestion price. Oscillatory instabilities may appear, and these are not avoided by making the adaptation "slow": if we reduce the parameter β, the frequency of oscillation is reduced, but the oscillations remain. In practice, the traffic slowly drifts between one link and the other, and back, but does not settle in the right place.

In control terms, the culprit is the second order dynamics: the α's integrate the price changes, and these in turn integrate the variations in link rates, functions of α: this amounts to a mass-spring kind of dynamics with no damping. How, then, do we introduce damping in this loop? A classical idea is to use "proportional-derivative" control[1], i.e. to introduce some anticipation of future prices in the control of routing splits. We will thus replace (17) by

$$\dot{\alpha}_i^d = \beta_i E_{\alpha_i^d}[-(\pi_i^d + \nu_i \dot{\pi}_i^d)], \quad \nu_i > 0. \tag{29}$$

[1]We acknowledge discussions with Jeff Shamma who has recently promoted the use of derivative action in dynamic games [20].

Remarks:

- The equilibrium properties of the control law remain unchanged, since the derivative terms vanish at equilibrium. In particular, if we combine this control with dual congestion control, it is still true through Proposition 3 that an equilibrium point must be a solution to the WELFARE problem.
- The fact that derivatives appear on both sides of (29) makes it a differential equation in implicit form. In particular, the price derivatives $\dot{q}_j^d$ that appear in $\dot{\pi}_i^d$ will in turn depend on traffic split derivatives through (8). The question arises as to whether this equation is non-singular, i.e. if it can be solved into an ordinary differential equation (ODE) at all times. A partial answer is the following: if the routing remains loop free at all times, then $\dot{q}_j^d$ is only a function of $\dot{\alpha}_{i',j'}$ for nodes *downstream* of i; by successive substitutions we can turn this into an ODE. This loop-free condition will automatically be satisfied by local dynamics around equilibrium, which is optimal hence loop free, or by the global dynamics if the blocking method of [7], described below, is included as a modification to (29).

We now study the behavior of the above anticipatory control of routes in combination with dual congestion control. We first consider a simple case, slightly more general than the example: a network of L parallel links between a single source and destination. Let

$$c = \begin{bmatrix} c_1 \\ \vdots \\ c_L \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_L \end{bmatrix}, \quad p = \begin{bmatrix} p_1 \\ \vdots \\ p_L \end{bmatrix}, \quad \alpha = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_L \end{bmatrix}$$

be the vectors of link capacities, rates, prices and split ratios, and q, x the scalar source variables. We have

$$y = x\alpha, \qquad q = \alpha^T p, \qquad x = f(q), \tag{30}$$

where f is the decreasing demand curve determined by (24). The equilibrium of (23-24) and (29) is

$$x^* = \sum_{l=1}^{L} c_l = f(q^*), \quad \alpha_l^* = \frac{c_l}{x^*}, \quad p^* = q^* \mathbf{1}. \tag{31}$$

Proposition 4: The equilibrium (31) is locally asymptotically stable under the dynamics (23-24), (29).

Before tackling the proof, we write the dynamics in incremental variables around equilibrium, $\delta x = x - x^*$ and so on. Without loss of generality take $q^* > 0$ and small δp under which no price saturation occurs. Then

$$\begin{aligned} \delta \dot{p}_l &= \gamma_l \delta y_l \\ &= \gamma_l [(x - x^*)\alpha_l + x^*(\alpha_l - \alpha_l^*)] \\ &= \gamma_l [\delta x \alpha_l + x^* \delta \alpha_l]. \end{aligned} \tag{32}$$

If α is interior to the unit simplex Δ (which happens locally since α^* is interior), the projections in (29) are simply given by the matrix E in (18). Also note $Ep^* = 0$, so we locally rewrite (29) as

$$\delta \dot{\alpha} = -\beta E(\delta p + \nu \delta \dot{p}). \tag{33}$$

Also, noting that $\delta\alpha \perp \mathbf{1}$ and $p^* = q^*\mathbf{1}$, (30) yields

$$\delta q = \alpha^T \delta p + \delta\alpha^T p^* = \alpha^T \delta p. \tag{34}$$

Proof of Proposition 4: Define the Lyapunov function candidate $V \geq 0$, vanishing only at equilibrium:

$$V(\delta\alpha, \delta p) = \frac{x^*}{2\beta}\|\delta\alpha + \beta\nu E\delta p\|^2 + \sum_{l=1}^{L} \frac{(\delta p_l)^2}{2\gamma_l}.$$

Here $\|\cdot\|$ is the Euclidean norm. The derivative of $\delta\alpha + \beta\nu E\delta p$ is equal to $-\beta E\delta p$ from (33); therefore the first term in V has derivative

$$x^*(\delta\alpha+\beta\nu E\delta p)^T(-E\delta p) = -x^*\delta\alpha^T\delta p - x^*\beta\nu\|E\delta p\|^2.$$

Note for the above that $E\delta\alpha = \delta\alpha$. Now, using (32), the derivative of the second term in V is

$$\sum_{l=1}^{L} \delta p_l[\alpha_l \delta x + x^*\delta\alpha_l] = (\alpha^T\delta p)\delta x + x^*\delta\alpha^T\delta p.$$

Combining both terms and using (34) yields

$$\dot{V} = -x^*\beta\nu\|E\delta p\|^2 + \delta q\delta x.$$

Now since the demand curve $f(q)$ is decreasing we have

$$\delta q\delta x = (q - q^*)(f(q) - f(q^*)) \leq 0,$$

so V decreases along trajectories.

The Lasalle principle [11] implies convergence to an invariant set where $\dot{V} \equiv 0$. This implies $x \equiv x^*$ and $q \equiv q^*$. Also, due to the first term in $\dot{V}$ we have $E\delta p \equiv 0$ which means δp is parallel to $\mathbf{1}$, $\delta p(t) = \delta q(t)\mathbf{1}$. But since $\delta q \equiv 0$ we have $\delta p \equiv 0$. Finally, (32) implies $\delta\alpha \equiv 0$ so the invariant set is the equilibrium. ■

Remarks:

- If we set $\nu = 0$ in the above, i.e. there is no anticipatory term in the dynamics of α, we still must have convergence to $x \equiv x^*$, as claimed in the example above for $L = 2$, however there is no guarantee that prices and α's will converge.
- The incremental equations (32-34) are not linearizations, they are exact as long as prices remain positive and α is interior to the simplex. Therefore the Lyapunov proof extends to show that the basin of attraction of the equilibrium includes any sub-level set of $V(\delta\alpha, \delta p)$ that does not touch these boundaries. If the boundaries are reached, the resulting hybrid dynamics are more involved, with possible discontinuities of V, which makes it difficult to give a global result. For more extensive treatment of these global stability issues we refer to [17].

We now state a local dynamic stability result for a general network, with arbitrary topology and multiple commodities.

Theorem 5: For any network, under the congestion control (23-24), and (29), trajectories converge locally to an equilibrium (optimum of Problem 1).

The proof is given in Appendix C.

V. Implementation

The theory described in the previous sections can be taken as a basis for more than one implementation, depending on the choice of the link congestion price, the source utility function, and the method for sharing congestion information between routers and with traffic sources. In this section we discuss these issues, and describe one such design.

A. Discussion

1) Routing protocol and node price formation: Prevailing methods for computing single-path routing tables in IP routers are shortest-path algorithms, with hop-count the default metric. For this computation, routers disseminate metric information, either globally (in link-state protocols such as OSPF) or to neighbors (in distance-vector protocols such as RIP), see e.g. [19]. The latter alternative is very well suited for our purposes, by making the node price q_i^d the metric used in announcements. Router i generates prices $p_{i,j}$ of its own outgoing links, and receives announcements of downstream prices q_j^d, so it can periodically update q_i^d to the right-hand side of (8). This iteration converges under the same conditions that ensure node prices are well defined, as shown in Appendix A.

2) Update of split ratios, blocking, and forwarding: The control of $\{\alpha_{i,j}^d\}$ can follow either the gradient (17) or anticipatory (29) dynamics, we will favor the second. We now discuss how to avoid the transient formation of routing loops. Thinking of the node price as a potential, traffic should tend to flow "downhill", and will do so at equilibrium: node i will not permanently use link (i,j) if $q_j^d > q_i^d$. However, since our dynamics of α are continuous the above *improper* routing could occur transitorily, and with it, routing loops. To avoid them, [7] proposed to start from a loop-free configuration, and to **block** the start of the use of a new link if there is an improper routing in its downstream path. This is signalled through a flag that accompanies routing announcements, see details below.

Having explicit split variables $\alpha_{i,j}^d$ allows for fine-grained multipath forwarding, tracking these proportions at the packet time-scale. In contrast, the backpressure approaches [13], [4] only change paths when price changes are observed, inherently a slower-scale phenomenon, implying in practice route oscillations.

3) Communication of prices from routers to sources: A major point of discussion among congestion control implementations is whether it is necessary to introduce *explicit* congestion signals between routers and sources, or it suffices to rely on implicit measures which can be estimated by the transport layer.

To address this issue for the multipath algorithm, it is important to consider the time-scales involved. TCP sources must control congestion quickly, faster than routing updates; hence it is not reasonable to rely only on explicit communication of node prices, which may take time to converge to the solution of (8). On the other hand, a source can estimate its *average* congestion price over the routes it is using, based on the ACK stream as in single-path routing, for certain price signals: loss or ECN marking probability, or queueing delay. In this way, the source could estimate its price faster than the time it takes the access router to make it explicit.

A proposal based on loss or ECN marking was outlined in [18]. From here on we focus on queuing delay as a congestion measure. If p_l is the delay of each link's queue, then q_i^d in (8) is the average queueing delay between node i and destination, and q^k at the source is the average queueing delay experienced by packets over all paths. What sources can explicitly measure is the average round-trip-time of their packets, $\overline{RTT} = \overline{D} + q^k$, where $\overline{D}$ is the propagation/processing delay averaged over all routes. This "BaseRTT" does *not* coincide with the minimum observed RTT used in single-path settings. Therefore, to estimate $\overline{D}$ we will rely slow time-scale explicit signalling, as explained below.

B. Details and ns2 implementation

1) Multipath distance vector protocol: This protocol is based on the Bellman-Ford distance vector algorithm, and its most well-known implementation, RIP (see e.g. [19]). The protocol learns routes to an IP destination address from its own locally connected networks, and from routes received from neighboring routers. But, as compared to RIP, our multipath protocol does not discard a route if it has a shorter (or cheaper) alternative; rather, it maintains in its routing table all possible next hops for a given destination. Each row in the routing table is accompanied with its metric, $\pi_{i,j}^d = p_{i,j} + q_j^d$, and its routing split variable, $\alpha_{i,j}^d$. Here $p_{i,j}$ is the queueing delay of the link, measured as the link queue divided by its capacity, and q_j^d is the metric learned from the downstream router.

When the algorithm starts, it learns the routes from directly connected destinations, and assigns them cost $q_j^d = 0$. Since these are the first routes to be learned, they are assigned $\alpha_{i,j}^d = 1$: all traffic for this destination will initially be routed through this path. Analogously, every time a new destination is discovered it is assigned $\alpha_{i,j}^d = 1$; on the other hand, new routes learned for an already known destination are assigned $\alpha_{i,j}^d = 0$.

Routing announcements of the form (destination, metric, flag) are sent from each node to its neighbors. These are sent every δ_t^r seconds, or also asynchronously if the node has received a notification that changes its routing table or metric. The first two fields are similar to RIP's, the metric is the weighted average q_i^d from (8). The additional flag indicates whether this is a proper or improper route, used for blocking loops: a route is announced as improper if at least one of the next hops j with $\alpha_{i,j}^d > 0$ satisfies either (i) $q_j^d > q_i^d$, or (ii) node j's last announcement had the improper flag on.

Upon reception of an announcement from node j with the improper flag on, if the current $\alpha_{i,j}^d = 0$, node i will block node j for this d (denoted $j \in B_i^d$), and forbid the α_i^d dynamics from increasing this component.

2) Split updates and projections: Updates to $\alpha_{i,j}^d$ are made periodically, with period δt^a, according to

$$\alpha_{i,t+\delta t^a}^d = \alpha_{i,t}^d + \beta_i E_{\alpha_i^d} \circ E_{B_i^d}[-(\pi_{i,t}^d + \nu_i \Delta \pi_{i,t}^d)]. \tag{35}$$

This amounts to a discretization of (29), with an additional projection that implements blocking. We now describe how such projections are computed. Given a subset Φ of next-hops, and a drift vector v, let

$$(E_\Phi[v])_j := \begin{cases} v_j - \overline{v^\Phi}, & j \notin \Phi; \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

Here $\overline{v^\Phi}$ is the average of $\{v_j, j \notin \Phi\}$. Thus $E_\Phi[v]$ prevents motion in all components $j \in \Phi$, and keeps the overall average motion at zero. To implement blocking, we simply set $\phi = B_i^d$. We can also implement $E_{\alpha_i^d}$ this way, only in this case Φ must be constructed iteratively.

We use Ω as an auxiliary set for this construction.

1) Set $\Phi = \varnothing$ and $\Omega = \varnothing$. Note that the first implies $E_\Phi = E$ as in (18).
2) Update $\Phi := \Phi \cup \Omega$.
3) Set $\Omega = \{j : v_j - \overline{v^\Phi} < 0, \alpha_{i,j}^d = 0\}$.
4) If $\Omega \neq \varnothing$ repeat from step 2; otherwise, finish.

Note that $\overline{v^\Phi}$ increases with each iteration: the coordinates in Ω, removed from the average, are smaller than the previous average. When no more coordinates can be removed, the set Φ contains all coordinates of v that take α_i^d outside Δ_i, consistently with the definition of $E_{\alpha_i^d}$.

Finally, due to the finite step used in (35), the possibility exists that α_i^d could escape through another boundary of Δ_i. If this happens we adjust step size β_i down so that α_i^d reaches exactly that boundary.

3) Forwarding: To forward packets in a way that matches $\alpha_{i,j}^d$ in the mean, we add the auxiliary variable $\hat{e}_{i,j}^d$, updated after each packet forwarding by $\hat{e}_{i,j}^d = (1 - c^d)\hat{e}_{i,j}^d + c^d \frac{\delta_j}{\alpha_{i,j}^d}$ with $c^d \in (0,1)$ and $\delta_j = 1$ for the chosen link, $\delta_j = 0$ for the rest. This assumes equal size packets, otherwise weights can be added. Thus $\hat{e}_{i,j}^d$ tracks the ratio between the actual rate fraction through link (i,j) and $\alpha_{i,j}^d$. Forwarding decisions are made by choosing the link (i,j) with minimum $\hat{e}_{i,j}^d$.

4) Sources: tracking of $\overline{D}$, TCP-FAST modifications: Associated with source nodes are TCP-FAST agents. These estimate average RTT and BaseRTT by running, for each received ACK, the updates

$$\overline{RTT} := (1 - a) * \overline{RTT} + a * currentRTT, \tag{37}$$

$$BaseRTT := (1 - b) * BaseRTT + b * (\overline{RTT} - q^k). \tag{38}$$

The RTT averaging equation (37) is standard: the parameter a can be made inversely proportional to the current window size. This makes the time constant of the filter correspond to a certain number of RTTs.

Equation (38) for $BaseRTT$ is modified from FAST, it is not based on the minimum RTT. The reason is that, as mentioned above, $BaseRTT$ must track the average propagation delay $\overline{D}$ across all used paths. The idea of (38) is to use prices q^k, explicitly communicated at the slower routing time-scale, by the IP agent co-located with the source. These occasional messages indicate the correct average queueing delay (congestion price); by lowpass filtering the difference $(\overline{RTT} - q^k)$, with parameter $b << a$, we are able to track the slow variations of $\overline{D}$ in the variable $BaseRTT$. Then, the instantaneous difference $RTT - BaseRTT$ used by TCP-FAST will track the fast variations of the source congestion price q^k.

Another modification required on TCP-FAST is that, consistently with multipath routing, it no longer makes sense to consider the ordering of packet arrivals in decisions about congestion control. In particular, the duplicate ACK feature should be removed, and RTT averaging should be performed on all packets, not just those which come in order.

VI. Simulations

A. Gradient vs anticipatory control

Our first simulation is intended to support our discussion on the need for anticipatory control of traffic splits. We simulate a simple network with two parallel routes between source and destination, with bottlenecks of respectively 50Mbps and 100Mbps. A single TCP-FAST source uses the network, and the input router performs the two-way split. Figure 1 below shows the behavior of the prices (queueing delays) on both paths, for the cases $\nu = 0$ (no derivative action) and $\nu > 0$. In both cases, the input rate stabilizes to 150Mbps, but in the absence of derivative action, we observe oscillatory instabilities, consistent with the second order dynamic model discussed in Section III-B. Once we introduce enough damping in the system, the queues settle down around a common equilibrium price.

B. Dynamic example, 4-node topology

We now turn to a richer example that exhibits various features of the protocol. The topology is depicted in Figure 2. There are three groups (Grp 0, Grp 1 and Grp 2) of sources at nodes 0, 1 and 2 with 5, 10 and 10 TCP-FAST connections each, with a common destination at node 3 and parameters $K_{G_0} = K_{G_1} = K_{G_2} = 100$. This parameter represents the number of packets to be stored in network queues in equilibrium. All links have the same capacity (1 Gbps) and propagation delay of 10ms. Capacities and delays are the same in the reverse path. Packet size is 1040 bytes. The following parameters were used in routers: $\beta = 0.25$, $\nu = 15$, $\delta_t^r = 5ms$, $\delta_t^a = 10ms$. In TCP sources, we used $a^{-1} = cwnd$, $b^{-1} = 30{*}cwnd$ for the averaging of RTT and BaseRTT.

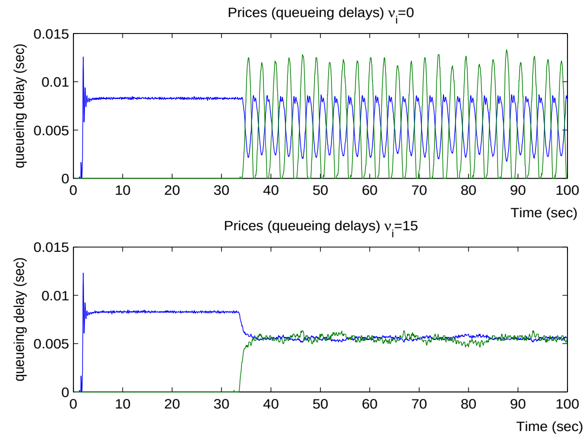


Fig. 1. Dynamics of multipath routing.

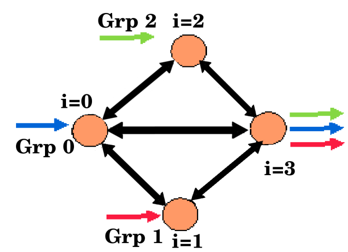


Fig. 2. Simulated network.

The simulation results are depicted below. Figures 3-5 contain split ratios and metrics (prices) for nodes 0, 1 and 2. Figure 6 contains (x^k, q^k) for all source groups.

In the initialization process, all nodes (0,1,2) discover first the direct route to destination node 3. Grp 0 starts first, and node 0 begins sending all traffic directly to node 3. This is also the default route in single path protocols since it is the shortest path. When link (0,3) saturates, node 0 starts splitting the rate gradually between the other two paths and the TCP-FAST sources in Grp 0 react to the lowering in the average queueing delay by increasing the rate. After 18 seconds node 0 reaches equilibrium $\alpha_0^3 = (\alpha_{0,1}^3, \alpha_{0,2}^3, \alpha_{0,3}^3)^T = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^T$, and Grp 0 sources reach their fair rate $x^{k,*} = (c_{0,1} + c_{0,2} + c_{0,3})/5 = 600$ Mbps.

At 20 seconds, Grp 1 starts sending traffic through link (1,3), which becomes more congested, increasing the metric $\pi_{0,1}^3$ seen by node 0 and making it eventually stop sending traffic through link (0,1). Notice that until this happens, node 1 cannot start using link (1,0), although it sees a cheaper price through that link. Only when node 0 has completed its transfer of routing to links (0,2) and (0,3), node 1 unblocks the cheaper route (1,0), after which the system reaches a new equilibrium $\alpha_0^3 = (0, \frac{1}{2}, \frac{1}{2})^T$, $\alpha_1^3 = (\alpha_{1,0}^3, \alpha_{1,3}^3)^T = (\frac{1}{2}, \frac{1}{2})^T$ and $x^{k,*} = (c_{0,3} + c_{1,3} + c_{2,3})/15 = 200$ Mbps.

After 60 seconds, Grp 2 sources start sending traffic

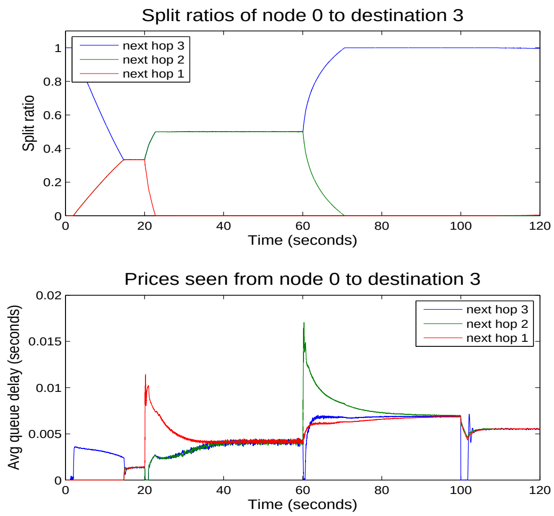


Fig. 3. Split ratios and prices from node 0.

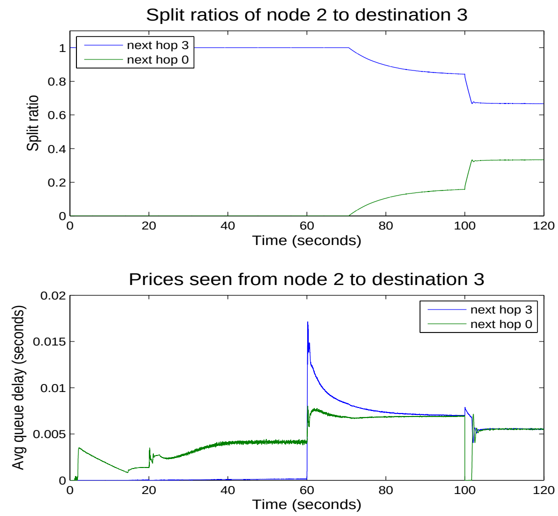


Fig. 5. Split ratios and prices from node 2.

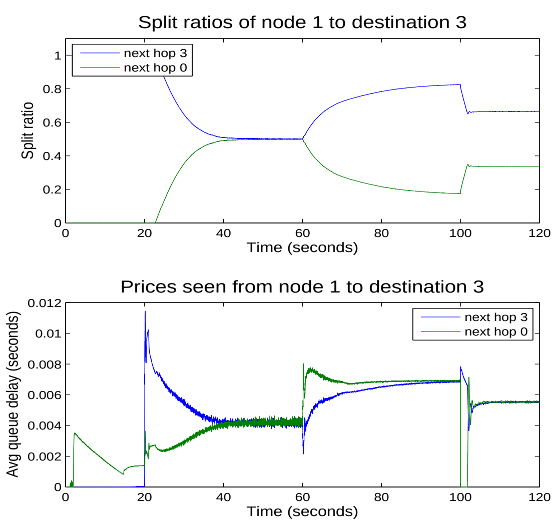


Fig. 4. Split ratios and prices from node 1.

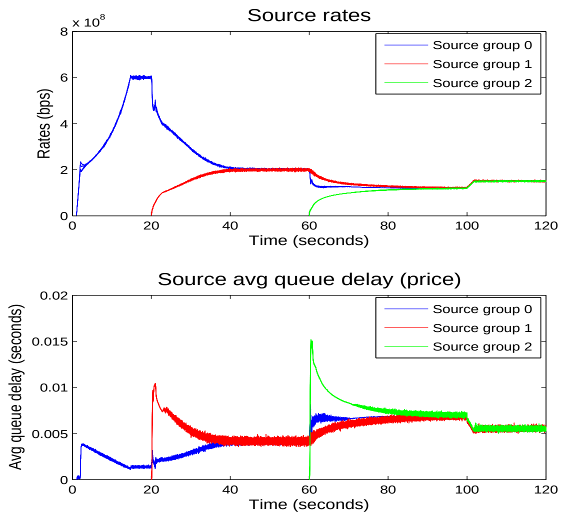


Fig. 6. Source rates and average queueing delays.

from node 2 to destination 3. This reactivates the routing algorithm at node 0 and makes it change its routing configuration to $\alpha_0^3 = (0,0,1)^T$ in spite of the fact that the link (0,3) is not the cheapest (node 1 is blocked). After node 0 reaches its new equilibrium, both nodes 1 and 2 start changing their splits until they get to $\alpha_1^3 = (\alpha_{1,0}^3, \alpha_{1,3}^3)^T = (\frac{1}{6}, \frac{5}{6})^T$, $\alpha_2^3 = (\alpha_{2,0}^3, \alpha_{2,3}^3)^T = (\frac{1}{6}, \frac{5}{6})^T$, and $x^{k,*} = (c_{0,1} + c_{0,2} + c_{0,3})/25 = 120$ Mbps.

Finally, at 100 seconds, Grp 0 disconnects its TCPs and the system reaches the last equilibrium shown in the simulations. Node 0 stays at $\alpha_0^3 = (0,0,1)^T$, sending all traffic directly to destination 3; nodes 1 and 2 increase the traffic fractions sent to node 0, reaching $\alpha_1^3 = \alpha_2^3 = (\frac{1}{3}, \frac{2}{3})^T$. All flows achieve a rate of $x^{k,*} = (c_{0,1} + c_{0,2} + c_{0,3})/20 = 150$ Mbps.

VII. Conclusions and Future Work

We have proposed a framework in which multipath routing and congestion control work in unison to pursue a common objective: the maximization of aggregate utility or surplus over the network. The control of input rates and routing splits is decentralized, relying on a common congestion "currency" for its decisions. We have studied mathematically the equilibrium and dynamic properties of various control laws; in particular, we have proposed a new anticipatory control of traffic splits which stabilizes the maximum welfare allocation when combined with dual congestion control.

The theory has assumed persistent TCP flows. Given the relatively slow dynamics of routing, it is important to extend this work to consider the effect of finite TCP flows that come in and out of the network. This remains open for future research. Another interesting future topic is combining the above control of network and transport layers with the lower layers, particularly for wireless networks, possibly offering alternatives to the backpressure scheduling approach [13], [4].

We have presented a packet implementation based on queueing delay as a congestion price; routers measure local prices and exchange information with neighbors, following a multipath variant of a distance-vector routing protocol. Fast-TCP sources estimate this delay from their RTT measurements in real time, calibrating their propagation delay through periodic interactions with the IP layer. Our ns2 simulations verify the expected behavior from the theory. One could alternatively consider implementations based on loss or marking as a congestion price; we will explore these in future work.

Appendix A: Split ratios and node prices

In this Appendix we study the recursive relationships that define node prices in terms of split ratios, reproduced here for convenience (recall $q_d^d = 0$):

$$q_i^d = \sum_{j:(i,j)\in\mathcal{L}} \alpha_{i,j}^d [p_{i,j} + q_j^d], \quad i \neq d. \tag{39}$$

We analyze first a single destination d, taken for simplicity to be node n, and define the matrices

$$A = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} \cdots & \alpha_{1,n-1} \\ \alpha_{2,1} & \alpha_{2,2} \cdots & \alpha_{2,n-1} \\ \vdots & \vdots \; \ddots & \vdots \\ \alpha_{n-1,1} & \cdots & \alpha_{n-1,n-1} \end{bmatrix}, \quad a = \begin{bmatrix} \alpha_{1,n} \\ \alpha_{2,n} \\ \cdots \\ \alpha_{n-1,n} \end{bmatrix}.$$

Here we dropped the superscript from the variables $\alpha_{i,j}$, and defined these for every pair of nodes; $\alpha_{i,j} = 0$ if there is no link (i,j); in particular, the diagonal of A is zero. The identity $[A|\, a]\mathbf{1}_n = \mathbf{1}_{n-1}$ expresses the balance of mass per node; here $\mathbf{1}_m$ is the vector of ones of length m. We also define matrix B of dimensions $(n-1)\times L$, where

$$b_{i,l} = \begin{cases} \alpha_{i,j} & \text{if } l = (i,j), \\ 0 & \text{otherwise }. \end{cases}$$

With this notation, the node price equations (39) can be written in vector form as

$$q = Aq + Bp,$$

where q is the vector $\{q_i\}_{i=1}^{n-1}$, and p is the vector $\{p_l\}_{l\in\mathcal{L}}$. We will study conditions under which the above equations can be solved for

$$q = (I - A)^{-1}Bp, \tag{40}$$

a vector with non-negative elements. The most useful case is when the routing contains no loops.

Proposition 6: Assume there is no closed loop with $\alpha_{i,j} > 0$ in all its links. Then the matrix A is nilpotent, $A^{n-1} = 0$.

Proof: Think of $\alpha_{i,j}$ as the probability that a packet starting at i arrives at j after one hop. Assuming independent routing decisions, the probability that the packet arrives in j after two hops is $\sum_{j'} \alpha_{i,j'}\alpha_{j',j}$, the (i,j) element of the matrix A^2. Analogously, the (i,j) element of A^{n-1} is the probability that the packet arrives at j after $(n-1)$ hops. If this probability were positive, it would imply the packet could stay with positive probability in the set of nodes $\{1,\ldots,n-1\}$ for $(n-1)$ hops; since staying in that set requires looping, this implies positive probability of a loop, a contradiction. ■

As a consequence, if the routing disallows loops, then

$$(I - A)^{-1} = I + A + A^2 + \ldots A^{n-2} \geq 0,$$

therefore there exist non-negative prices q satisfying (39). Furthermore, the iteration

$$q^+ := Aq + Bp \tag{41}$$

which represents the price propagation converges in at most $n-1$ steps.

In Gallager [7], a more general argument is given that implies $\rho(A) < 1$ provided there is a path of positive probability from any node to destination. Under these conditions, q satisfying (39) is uniquely defined; in this general case the convergence of (41) is asymptotic.

An additional use of the matrix notation is to make explicit the relationship between node and link variables, for fixed routing. For a commodity k with source node $s(k)$ and destination $d(k) = n$, let

$$\tilde{x}^k = \left[x_1^k \ldots x_{n-1}^k\right]^T$$

be the vector of rates of commodity k entering all nodes except d. If y^k is the vector of link rates, with links ordered as before, then the split equation (6) is represented in matrix form by

$$y^k = B^T \tilde{x}^k.$$

Also, the node balance equations at all nodes except d can be represented by

$$A^T \tilde{x}^k + e_{s(k)} x^k = \tilde{x}^k,$$

where $e_{s(k)}$ is the canonical basis vector at the source node. Under the same conditions as before on the routing, the above equation can be solved for $\tilde{x}^k$, yielding

$$y^k = B^T (I - A^T)^{-1} e_{s(k)} x^k.$$

The above procedure can clearly be generalized to any commodity, with destination not necessarily equal to n, by appropriate definition of the matrices A,B, which are destination-dependent. We obtain in general

$$y^k = (B^{d(k)})^T [I - (A^{d(k)})^T]^{-1} e_{s(k)} x^k =: R_k(\alpha) x^k,$$

with $R_k(\alpha)$ a column vector. Adding over k we have

$$y = \sum_k R_k x^k = R(\alpha) x;$$

we thus have an expression for the routing matrix $R(\alpha)$ that maps source to link rates, for fixed α. We can also describe with the same matrix, the dual relationship between link and source prices. This is done by extending (40) to any destination, and picking out of the vector q of node prices, the source node price:

$$q^k = e_{s(k)}^T (I - A^{d(k)})^{-1} B^{d(k)} p = R_k(\alpha)^T p.$$

From here we have the global relationship

$$q = R(\alpha)^T p$$

between the link prices and the vector of source prices, which extends the single path case [15], [22].

We conclude the appendix with a proposition to be used below, which refers to the gradient price dynamics.

Proposition 7: Suppose $p_{i,j}$ are constant. If $\dot\alpha$ satisfies (14) with q specified by (39), then q converges asymptotically to an equilibrium value.

Proof: Taking a derivative in (39) for constant $p_{i,j}$ (dropping the superscript d), we have

$$\dot q_i = \sum_{j:(i,j)\in\mathcal{L}} \dot\alpha_{i,j}[p_{i,j} + q_j] + \sum_{j:(i,j)\in\mathcal{L}} \alpha_{i,j}\dot q_j.$$

Denoting the first term by $v_i(t)$, which satisfies $v_i \le 0$ due to (14), we write in matrix form $\dot q - A\dot q = v$. Under the previous conditions for $\rho(A) < 1$, we solve

$$\dot q = (I - A)^{-1} v = (I + A + A^2 + \ldots) v \le 0;$$

since q is lower bounded we conclude that it converges to an equilibrium. ■

APPENDIX B: GLOBAL OPTIMA

We characterize optimality conditions in our optimization problems using Lagrangian duality, and use this to establish optimality of equilibria.

SURPLUS Problem

We start with Problem 2, rewriting it as follows:

$$\text{Maximize } S(x,y) = \sum_k U_k(x^k) - \sum_l \phi_l\left(\sum_k y_l^k\right) \tag{42}$$

in the variables $x = \{x^k\}_{k\in\mathcal{K}}$, and $y = \{y_l^k\}_{k\in\mathcal{K}, l\in\mathcal{L}}$, subject to the constraints for every k, and $i \neq d(k)$:

$$g_i^k(x,y) := \sum_{j:(i,j)\in\mathcal{L}} y_{i,j}^k - x^k = 0, \quad i = s(k),$$

$$g_i^k(x,y) := \sum_{j:(i,j)\in\mathcal{L}} y_{i,j}^k - \sum_{j:(j,i)\in\mathcal{L}} y_{j,i}^k = 0, \quad i \neq s(k). \tag{43}$$

We introduce the Lagrangian

$$L^{SUR}(x,y,\lambda) = S(x,y) + \sum_k \sum_{i\neq d(k)} \lambda_i^k g_i^k(x,y)$$

with Lagrange multipliers λ_i^k $k \in \mathcal{K}$, $i \neq d(k)$. From convex duality the optimization

$$\min_{\lambda} \max_{x,y\ge 0} L^{SUR}(x,y,\lambda)$$

will give the same result as Problem 2. The variables λ, x, y are at a saddle point of the dual if they satisfy (43) and for each k we have:

$$\frac{\partial L^{SUR}}{\partial x^k} = U_k'(x^k) - \lambda_{s(k)}^k = 0, \quad (\text{or} < 0 \text{ and } x^k = 0), \tag{44}$$

$$\frac{\partial L^{SUR}}{\partial y_{i,j}^k} = -\phi_l' + \lambda_i^k - \lambda_j^k = 0, \quad (\text{or} < 0,\ y_{i,j}^k = 0), \quad j \neq d(k); \tag{45}$$

$$\frac{\partial L^{SUR}}{\partial y_{i,j}^k} = -\phi_l' + \lambda_i^k = 0, \quad (\text{or} < 0 \text{ and } y_{i,j}^k = 0), \quad j = d(k), \tag{46}$$

where ϕ_l' is evaluated at $y_l = \sum_k (y_l^k)$.

Primal stability proof

We complete here the proof of Theorem 2. Invoking the Lasalle invariance principle [11], the state trajectories will converge to an invariant set inside $\{(x,\alpha) : \dot S = 0\}$. We show that this such trajectory achieves optimal rates.

In reference to (22), we see that $\dot S = 0$ implies (24) for each commodity k. Also, from (22) and the restriction for equality in (14), for each d, $i \neq d$, we have

$$\text{either} \quad \sum_{d(k)=d} x_i^k = 0,$$

$$\text{or} \quad \dot\alpha_{i,j}^d = 0 \ \forall j : (i,j) \in \mathcal{L}. \tag{47}$$

Consider a trajectory satisfying $\dot S \equiv 0$. In particular, $\dot x^k \equiv 0$ from (20), the external rate is constant, and due to (47) we see that the only destinations for which the split ratios are allowed to vary are those for which the node is carrying no traffic. This means that while $\dot S = 0$, all *link* flows are constant and thus by (21) so are link prices $p_{i,j}$. Node prices can continue to vary at nodes which carry no destination traffic, as studied in Proposition 7.

The dynamics indeed allows for $\dot S = 0$ to hold for a finite interval of time, during which all link rates are constant, and come out of this state later when the evolving node prices provide a cheaper, currently unused route. However, for a trajectory moving entirely within the set $\dot S = 0$, as stipulated in the Lasalle principle, link rates y_l and node prices $p_l = \phi_l'(y_l)$ must remain constant for all time. Invoking Proposition 7 denote

$$\lambda_i^k := \lim_{t\to\infty} q_i^{d(k)}.$$

For all nodes that receive traffic $x_i^k > 0$, we have the second alternative in (47) and so (16) implies conditions (45-46). Therefore we are asymptotically at an optimum of Problem 2.

■

Remark: If blocking is included in the dynamics, (16) need not apply at any given time. However, it is not difficult to see that in the limit for $t \to \infty$ under the conditions of Proposition 7, there can be no improper routing and thus no blocking. So the conditions (16) will hold for the asymptotic q^d, as required.

WELFARE PROBLEM

Moving now to Problem 1, we can use the same set of variables $\{x^k\}$, y_l^k, and write it as

$$\text{Maximize } \sum_k U_k(x^k),$$
$$\text{subject to (43) and } \sum_k y_l^k \le c_l \text{ for each } l.$$

Its Lagrangian has now additional multipliers $\mu_l \ge 0$ for the capacity constraints:

$$L^{WEL}(x,y,\lambda,\mu) = \sum_k U_k(x^k) + \sum_k \sum_{i\neq d(k)} \lambda_i^k g_i^k(x,y) + \sum_l \mu_l(c_l - \sum_k y_l^k).$$

Writing the saddle point conditions for this problem gives equations analogous to (44-46), except that we substitute ϕ_l' by μ_l, and we have the additional condition

$$\sum_k y_l^k = c_l, \text{ (or } < c_l \text{ and } \mu_l = 0). \tag{48}$$

Proof of Proposition 3

At an equilibrium of the dual, we have

$$y_l = c_l, \text{ or } y_l < c_l \text{ and } p_l = 0, \tag{49}$$

together with (24), and (16). Taking $\mu_l = p_l$, and $\lambda_i^k = q_i^{d(k)}$, we see these rates and multipliers are a saddle point of L^{WEL}, therefore an optimum of the system problem.

■

APPENDIX C: STABILITY OF ANTICIPATORY CONTROL

We study the dual congestion control (23-24) with anticipatory control of routes (29), locally around an equilibrium point (denoted by superscript $\star$). Let $J_i^d := \{j : \alpha_{i,j}^{*,d} > 0\}$ be the set of neighbors used by node i to route to d in equilibrium. For simplicity, assume

$$p_{i,j}^* + q_j^{*,d} > q_i^{*,d} \quad \forall j \notin J_i^d,$$

i.e. the remaining links are *strictly* more expensive. Then for sufficiently small deviations from equilibrium prices, only links in J_i^d will be used; denote by

$$\alpha_i^d := \{\alpha_{i,j}^d\}_{j\in J_i^d},\ p_i^d := \{p_{i,j}\}_{j\in J_i^d},\ \mathbf{q}_i^d := \{q_j^d\}_{j\in J_i^d},$$

the vectors of splits, link prices and downstream neighbor prices of this reduced dimensionality at node i. Note the distinction between $\mathbf{q}_i^d$ and the scalar node price $q_i^d = (\alpha_i^d)^T(p_i^d + \mathbf{q}_i^d)$; by analogous reasoning as in (34) we have the incremental relationship

$$\delta q_i^d = (\alpha_i^d)^T(\delta p_i^d + \delta \mathbf{q}_i^d). \tag{50}$$

The equilibrium $\alpha_i^{*,d}$ is *interior* to the simplex of this dimension. Thus, the projection in (29) is locally achieved by a matrix E_i^d analogous to E in (18), which projects on $\ker \mathbf{1}^T$ for $\mathbf{1}$ of the simplex dimension.

$$\delta\dot\alpha_i^d = -\beta_i E_i^d[\delta p_i^d + \delta\mathbf{q}_i^d + \nu_i \delta\dot p_i^d + \nu_i \delta\dot{\mathbf{q}}_i^d]. \tag{51}$$

Denote also by $x_i^d = \sum_{k:d(k)=d} x_i^k$ the total rate destined to d reaching node i, and by $y_i^d = x_i^d\alpha_i^d$ the corresponding vector of rates over links in J_i^d. We have the incremental relationship

$$\delta y_i^d = \delta x_i^d \alpha_i^d + x_i^{d,*}\delta\alpha_i^d. \tag{52}$$

Note that the above relationship only requires (7), the weaker assumption on splitting by destination instead of flow. This fact implies that the proof below readily extends to that situation.

Proof of Theorem 5

Generalizing Proposition 4, define for node i the Lyapunov term

$$V_i = \sum_d \frac{x_i^{d,*}}{2\beta_i}\|\delta\alpha_i^d + \beta_i\nu_i E_i^d(\delta p_i^d + \delta\mathbf{q}_i^d)\|^2 + \sum_{j:(i,j)\in\mathcal{L}} \frac{(\delta p_{i,j})^2}{2\gamma_{i,j}}.$$

Differentiating the second term and using (23) yields

$$\sum_{j:(i,j)\in\mathcal{L}} \delta p_{i,j}\frac{\delta\dot p_{i,j}}{\gamma_{i,j}} \le \sum_{j:(i,j)\in\mathcal{L}} \delta p_{i,j}\delta y_{i,j} = \sum_d (\delta y_i^d)^T\delta p_i^d;$$

the inequality is due to the case where the price projection in (23) is active. Differentiating also the first term and using (51) leads to

$$\dot V_i \le -\beta_i\nu_i \sum_d x_i^{d,*}\|E_i^d(\delta p_i^d + \delta\mathbf{q}_i^d)\|^2 - \sum_d x_i^{d,*}(\delta\alpha_i^d)^T(\delta p_i^d + \delta\mathbf{q}_i^d) + \sum_d (\delta y_i^d)^T\delta p_i^d; \tag{53}$$

Focus on the second line of (53); we transform each term for fixed d using (52) and (50):

$$\begin{aligned} -x_i^{d,*}(\delta\alpha_i^d)^T(\delta p_i^d + \delta\mathbf{q}_i^d) + (\delta y_i^d)^T\delta p_i^d \\ =&(\delta x_i^d\alpha_i^d - \delta y_i^d)^T(\delta p_i^d + \delta\mathbf{q}_i^d) + (\delta y_i^d)^T\delta p_i^d \\ =&\delta x_i^d \delta q_i^d - (\delta y_i^d)^T\delta\mathbf{q}_i^d. \end{aligned} \tag{54}$$

Now define the Lyapunov function candidate $V := \sum_i V_i$. Clearly $V_i \ge 0$, and $V = 0$ implies $\delta p_{i,j} = 0$ for every link; under analogous conditions as in Appendix A the recursive equation (50) leads to the unique solution $\delta q_i^d = 0$. The first term in V_i then implies $\delta\alpha_{i,j}^d = 0$. Therefore V vanishes only at equilibrium. Compute its derivative along trajectories:

$$\dot V = \sum_i \dot V_i \le -\sum_{i,d}\beta_i\nu_i x_i^{d,*}\|E_i^d(\delta p_i^d + \delta\mathbf{q}_i^d)\|^2 + \sum_d\left\{\sum_i\left[\delta x_i^d\delta q_i^d - (\delta y_i^d)^T\delta\mathbf{q}_i^d\right]\right\}. \tag{55}$$

Now write

$$
\begin{aligned}
\sum_i (\delta y_i^d)^T \delta \mathbf{q}_i^d &= \sum_{(i,j)\in\mathcal{L}} \delta q_j^d \delta y_{i,j}^d \\
&= \sum_j \delta q_j^d \sum_{i:(i,j)\in\mathcal{L}} \delta y_{i,j}^d \\
&= \sum_j \delta q_j^d \Big(\delta x_j^d - \sum_{\substack{k\,:\, d(k)=d \\ s(k)=j}} \delta x^k\Big) \\
&= \sum_j \delta q_j^d \delta x_j^d - \sum_{k:d(k)=d} \delta q^k \delta x^k. \quad (56)
\end{aligned}
$$

The third step above follows from noticing that the (incremental) total rate δx_j^d entering node j, destined to d is the sum of components from inside and outside the network. Substituting (56) in the term in braces of (55), after cancellations we have the overall expression

$$
\dot{V} \leq -\sum_{i,d} \beta_i \nu_i x_i^{d,*} \|E_i^d(\delta p_i^d + \delta \mathbf{q}_i^d)\|^2 + \sum_k \delta q^k \delta x^k.
$$

From the demand curve of each commodity k, we have $\delta q^k \delta x^k \leq 0$, so V decreases along trajectories. Once more, the Lasalle principle implies that trajectories converge to an invariant set within $\{\dot{V} \equiv 0\}$. This condition implies $\delta x^k \equiv 0$, $\delta q^k \equiv 0$, and $E_i^d(\delta p_i^d + \delta \mathbf{q}_i^d) \equiv 0$ for each d, i. From the latter, $\dot{\alpha}_i^d \equiv 0$ and the routing is constant in time, therefore so are the link rates since the input rates are fixed. These constant link rates cannot exceed capacity, or prices would grow without bound, contradicting the fact that q^k is constant for all k. So prices must also be at an equilibrium. The invariant trajectories inside $\{\dot{V} \equiv 0\}$ are equilibrium points. ∎

REFERENCES

[1] D. Bertsekas, E. Gafni and R. Gallager, "Second derivative algorithms for minimum delay distributed routing in networks", *IEEE Trans. on Comm.*, Vol Com-32 (8), pp. 911-919, 1984.

[2] D. Berstekas and R. Gallager, *Data Networks*, Prentice-Hall 1992.

[3] L. S. Brakmo and L. Peterson, "TCP Vegas: end to end congestion avoidance on a global Internet", *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 8, October 1995.

[4] L. Chen, S. H. Low, M. Chiang and J.C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad-hoc wireless networks", IEEE INFOCOM 2006.

[5] A. Elwalid, C. Jin, S. Low, and I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering", IEEE INFOCOM 2001.

[6] B. Fortz and M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights", IEEE INFOCOM 2000.

[7] R. G. Gallager, "A minimum delay routing algorithm using distributed computation", *IEEE Trans. on Communicactions*, Vol Com-25 (1), pp. 73-85, 1977.

[8] H. Han, S. Shakkottai, C.V. Hollot, R. Srikant and D. Towsley, Overlay TCP for Multi-Path Routing and Congestion Control. *ENS-INRIA ARC-TCP Workshop, Paris, France, Nov. 2003.*

[9] C. Jin, D. X. Wei and S. H. Low, "FAST TCP: motivation, architecture, algorithms, performance"; *IEEE INFOCOM 2004.*

[10] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability", *Jour. Oper. Res. Society*, vol. 49(3), pp 237-252, 1998.

[11] H. K. Khalil, *Nonlinear systems*, Prentice-Hall, 1996.

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

[13] X. Lin and N. Shroff, "Joint rate control and scheduling in multihop wireless networks", *Proc. 43rd IEEE Conference on Decision and Control,* December 2004

[14] S. H. Low and D. E. Lapsley, "Optimization flow control, I: basic algorithm and convergence", *IEEE/ACM Transactions on Networking*, vol.7, no.6,pp861-874, December 1999.

[15] S. Low, F. Paganini, J. Doyle, "Internet Congestion Control", *IEEE Control Systems Magazine*, February 2002.

[16] A. Mas-Colell, M. Whinston, J. Green, *Microeconomic Theory*, Oxford University Press, 1995.

[17] E. Mallada, F. Paganini, "Stability of node-based multipath routing and dual congestion control", Proc. *IEEE Conf. on Decision & Control*, Cancún, Mexico, 2008.

[18] F. Paganini, "Congestion control with adaptive multipath routing based on optimization", *Conf. on Information Sciences and Systems*, Princeton, NJ, Mar 2006.

[19] L. L. Peterson and B. S. Davie, *Computer Networks: a Systems Approach*, Morgan-Kauffman, San Francisco, 2003.

[20] J. S. Shamma and G. Arslan, "Dynamic fictitious play, dynamic gradient play, and distributed convergence to Nash equilibria", *IEEE Trans. on Automatic Control*, Vol.50(3), 2005, pp. 312-327.

[21] A. Sridharan, R.Guerin, C. Diot. "Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/ISIS Networks". IEEE/ACM Transactions on Networking. March 2005

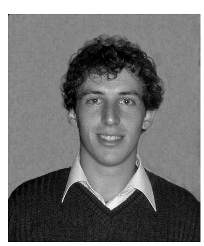
[22] R. Srikant, *The Mathematics of Internet Congestion Control,* Birkhauser, 2004.

[23] T. Voice, "Stability of multipath dual congestion control algorithms", *IEEE/ACM Trans. on Networking*, Dec. 2007.

[24] J. Wang, L. Li, S. H. Low and J. C. Doyle, "Cross-layer Optimization in TCP/IP Networks" *IEEE/ACM Trans. on Networking*, 13(3):582-268, June 2005.

[25] Y. Xi and E. Yeh, "Node-Based Distributed Optimal Control of Wireless Networks", *Conf. on Information Sciences and Systems*, Princeton, NJ, Mar 2006.

Fernando Paganini Fernando Paganini (M'90–SM'05) received his Ingeniero Electricista and Licenciado en Matemática degrees from the Universidad de la Republica, Montevideo, Uruguay, in 1990, and his M.S. and PhD degrees in Electrical Engineering from the California Institute of Technology, Pasadena, in 1992 and 1996 respectively. From 1996 to 1997 he was a postdoctoral associate at MIT. Between 1997 and 2005 he was on the faculty the Electrical Engineering Department at UCLA, reaching the rank of Associate Professor. Since 2005 he is Professor of Electrical and Telecommunications Engineering at Universidad ORT, Uruguay. Dr. Paganini is the recipient of the 1994 O. Hugo Schuck best paper award at the American Control Conference, the Wilts and Clauser Prizes for his PhD Thesis at Caltech in 1996, the 1999 NSF CAREER Award and the 1999 Packard Fellowship, and the 2004 George S. Axelby Award. His research interests are control and networks.



Enrique Mallada Enrique Mallada received the degree of Telecommunications Engineer from Universidad ORT, Uruguay in 2005. Since then, he has been a Teaching and Research Assistant in the Department of Telecommunications at Universidad ORT, and a member of the MATE research group. He also holds the position of Engineer at the Traffic Engineering Department of ANTEL, the main telecommunications operator in Uruguay. From 2004 to 2007 he was IT-Specialist at IBM and collaborated in an ITSO redpaper in Raleigh, NC, 2006. His research interests are on the use of optimization and control to model telecommunications networks.