Optimal Distributed Stochastic Routing Algorithms for Wireless Multihop Networks

Alejandro Ribeiro, *Member, IEEE,* Nikolaos D. Sidiropoulos, *Senior Member, IEEE,* and Georgios. B. Giannakis, *Fellow, IEEE*

***Abstract*—A novel framework was introduced recently for stochastic routing in wireless multihop networks, whereby each node selects a neighbor to forward a packet according to a probability distribution. Generalizing (deterministic) shortest path routing, stochastic routing offers greater flexibility that matches the random nature of wireless links. Consider the pairwise reliability matrix $\mathbf{R}$, whose (i,j)-th entry R_{ij} represents the probability that a packet transmitted from the j-th user U_j is correctly received by the i-th user U_i. Using $\mathbf{R}$ to capture physical layer aspects of the wireless medium, several rate-oriented stochastic routing formulations can be reduced to *centrally* solvable convex optimization problems. The present paper, introduces *distributed* algorithms that find optimal routing probabilities without the burden of collecting $\mathbf{R}$ at a central node and then percolating the resulting routing probabilities through network nodes. The resultant schemes are distributed in the sense that: (i) terminal U_j has access only to the j-th row and column of $\mathbf{R}$; and (ii) U_j interchanges variables only with those single-hop neighbors having positive probability of decoding its packets. The distributed algorithms are built by recasting the optimization problems and applying dual decomposition techniques. Since iterates obtained via dual decomposition do not always converge to centralized optimal routing probabilities, two known regularization approaches are further invoked, namely the method of multipliers (MoM) and the alternating-direction MoM. Convergence to the optimal routing matrix is then guaranteed under mild conditions. Many rate-oriented optimality criteria of practical interest can be addressed by the distributed framework, including maximization of: (i) the minimum rate; (ii) a weighted sum of rates; (iii) the product of rates; and (iv) the source's rate in a relay network. Robustness of the distributed algorithms is tested with respect to "topological" changes, communication errors and node mobility.**

***Index Terms*—Routing, wireless multihop networks, distributed network optimization, convex optimization, dual decomposition.**

Manuscript received November 8, 2006; revised May 15, 2007 and October 21, 2007; accepted December 8, 2007. The associate editor coordinating the review of this paper and approving it for publication was D. Wu. A summary of the main results in this paper was presented in the *IEEE ICASSP Conf.*, Honolulu, HI, April 2007. Work in this paper was prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon. The work of N. D. Sidiropoulos was partially supported by the EU under FP6 FET project CoopCom.

A. Ribeiro and G. B. Giannakis are with the Dept. of Electrical and Computer Engineering, University of Minnesota, 200 Union Street SE, Minneapolis, MN 55455 (e-mail: {aribeiro, georgios}@ece.umn.edu).

N. D. Sidiropoulos is with the Dept. of Electronic and Computer Engineering, Technical University of Crete, Chania - Crete, Greece 731 00 (e-mail: nikos@telecom.tuc.gr).

Digital Object Identifier 10.1109/T-WC.2008.070511

I. Introduction

MULTIHOP routing for wireless networks has recently attracted much interest, see e.g., [6], [8], due in part to its potential for reducing energy consumption (and thus increasing network lifetime) in energy-limited settings, such as sensor networks. The potential for energy-savings of multi-hop routing fundamentally stems from reducing the effective average distance between communicating nodes. Properly exploiting this potential requires addressing the challenge of finding multi-hop routes according to suitable optimality criteria; see e.g., [12], [18], [20]. Current multihop routing protocols for wireless networks are based on accumulated knowledge about routing in wired networks. Consequently, the usual approach is to: i) define a communication radius for each node; ii) draw the corresponding connectivity graph; and iii) utilize network optimization tools, e.g., shortest path routing, to find the optimal routes. While valuable as a first-order approach, a connectivity graph is not necessarily an accurate model of a wireless network [9], [10], [14]. In a recent paper we introduced a framework to design stochastic routing algorithms based on the reliability (pairwise packet-success-probability) matrix $\mathbf{R}$ whose (i,j)-th entry R_{ij} represents the probability that a packet transmitted from the j-th user U_j is correctly received by the i-th user U_i [17]; see also [5] and Fig. 1. Using $\mathbf{R}$ to capture the essential characteristics of the inherently unreliable wireless broadcast channel is still a rather coarse abstraction. Unlike the crude connectivity graph, however, $\mathbf{R}$ takes link reliability and radio interference from other nodes into account. The usefulness of a model based on $\mathbf{R}$ hinges on the relative communication efficiency and algorithmic complexity of finding optimal routes. Enticingly, many interesting optimality criteria lead to optimal routing algorithms in the form of convex optimization problems [17].

A. Stochastic routing in wireless multihop networks

Consider a wireless network with $J+1$ user nodes $\{U_j\}_{j=1}^{J+1}$ in which the first J users $\{U_j\}_{j=1}^{J}$ participate in routing packets to a destination $D \equiv U_{J+1}$. The physical and medium access layers are such that (s.t.) if a packet is transmitted by U_j it is correctly *received* by U_i with probability R_{ij} that we arrange in the matrix $\mathbf{R}$. Packets are stochastically routed, i.e., transmitted, according to probabilities T_{ij} collected in a matrix $\mathbf{T}$. When user terminal U_j decides to transmit a packet it selects a random terminal as the intended next hop with U_i chosen with probability T_{ij}. If the transmission is successfully received – something that happens with probability R_{ij} –

1536-1276/08$25.00 © 2008 IEEE

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

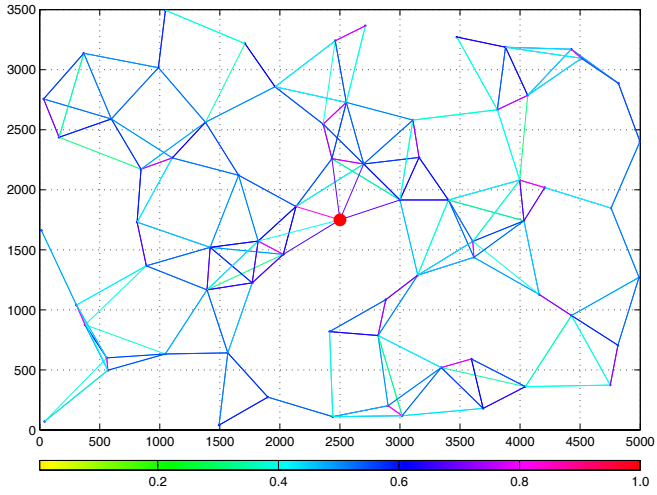

Fig. 1. Schematic representation of $\mathbf{R}$ for a network with 70 terminal nodes. The color index (gray level) represents the value of R_{ij}. To generate R_{ij} we consider the per-packet SINRs as in (45) and determine the per-packet reliabilities with a $(23, 12)$ Golay code, [16, pp. 457]. The value of R_{ij} is then computed as the time average over 10^3 packet transmissions ($\mu_j = 0.2$, $\sigma_j = -90$db $\forall j$, $c = 1$ and $\alpha = 3.4$).

the packet moves to U_i's queue; otherwise it is kept by U_j that attempts transmission, possibly to a different node, at a later time. To capture the evolution of packets through the network we define a matrix $\mathbf{K}$ whose elements K_{ij} represent the probability that a packet moves from U_j's queue to U_i's queue. For $i \neq j$ the packet moves from U_j to U_i if and only if it is routed through U_i *and* is correctly decoded; since these two events are independent, we have (^{T} denotes transposition and $\mathbf{1}$ the all-ones column vector)

$$K_{ij} = T_{ij}R_{ij} \quad \text{for } i \neq j, \qquad \mathbf{K}^T\mathbf{1} = \mathbf{1}, \qquad \mathbf{T}^T\mathbf{1} = \mathbf{1} \quad (1)$$

where the last two constraints come from the fact that $\mathbf{K}$ and $\mathbf{T}$ are stochastic matrices, i.e., columns add to one.

To complete the formulation let each terminal maintain a queue to store packets to be transmitted. Exogenous packet arrivals at U_j can be described by a stationary stochastic process with average rate ρ_j; let $\boldsymbol{\rho} := [\rho_1, \ldots, \rho_J]^T$. Further denote as $\boldsymbol{\mu} := [\mu_1, \ldots, \mu_J]^T$ the transmission probabilities and $\boldsymbol{\lambda} := [\lambda_1, \ldots, \lambda_J]^T$ the average rate of packet departures. For queue stability it suffices to have $\mathbf{0} \preceq \boldsymbol{\lambda} \preceq \boldsymbol{\mu}$, where $\preceq$ should be interpreted element-wise and $\mathbf{0}$ denotes the all-zero vector. Defining $\mathbf{K}_D$ as the $J \times J$ upper left submatrix of $\mathbf{K}$, it is not difficult to see that $\boldsymbol{\rho}$ and $\boldsymbol{\lambda}$ are related by [17]

$$\boldsymbol{\rho} = (\mathbf{I} - \mathbf{K}_D)\boldsymbol{\lambda}. \quad (2)$$

With $\mathbf{R}$ available at a central location, the stochastic routing protocols outlined here yield routes maximizing a measure of the arrival rate vector $\boldsymbol{\rho}$. Specifically, letting $f(\boldsymbol{\rho}) : \mathbb{R}^J \to \mathbb{R}$ be a utility function used to compare arrival rate vectors $\boldsymbol{\rho}$, the optimal routing matrix $\mathbf{T}^*$ is given as the solution of the generic optimization problem

$$\begin{aligned} \mathbf{T}^* = \arg\max \quad & f[(\mathbf{I} - \mathbf{K}_D)\boldsymbol{\lambda}] \\ \text{s.t.} \quad & K_{ij} = R_{ij}T_{ij} \quad \text{for } i \neq j, \\ & \mathbf{K}^T\mathbf{1} = \mathbf{1}, \quad \mathbf{T}^T\mathbf{1} = \mathbf{1}, \quad \mathbf{0} \preceq \boldsymbol{\lambda} \preceq \boldsymbol{\mu}. \end{aligned} \quad (3)$$

Finding efficient methods to solve (3) is challenging for a general $f(\boldsymbol{\rho})$. Remarkably though, for any $f(\boldsymbol{\rho})$ that is concave and monotonically non-decreasing in each component[1] (3) can be transformed into an equivalent convex optimization problem for which globally convergent solution methods are available [17].

In fact, the basic result in [17] is that for functions $f(\boldsymbol{\rho})$ that are monotonically non-decreasing in each component there exists an optimal solution of (3) with $\boldsymbol{\lambda} = \boldsymbol{\mu}$; hence (3) can be rewritten as

$$\begin{aligned} \mathbf{T}^* = \arg\max \quad & f[(\mathbf{I} - \mathbf{K}_D)\boldsymbol{\mu}] \\ \text{s.t.} \quad & K_{ij} = R_{ij}T_{ij} \quad \text{for } i \neq j, \\ & \mathbf{K}^T\mathbf{1} = \mathbf{1}, \quad \mathbf{T}^T\mathbf{1} = \mathbf{1}. \end{aligned} \quad (4)$$

The concavity of $f(\boldsymbol{\rho})$ further implies that the argument in (4) is concave, which together with the fact that the constraints are linear equalities imply that (4) is a convex optimization problem that can be solved in polynomial time using interior point methods [3, Ch. 11]. It is worth stressing that $\boldsymbol{\lambda} = \boldsymbol{\mu}$ is not the unique optimal solution of (3) but among the set of optimal pairs $(\mathbf{T}^*, \boldsymbol{\lambda}^*)$ there exists one with $\boldsymbol{\lambda}^* = \boldsymbol{\mu}$.

Requiring $f(\boldsymbol{\rho})$ to be monotonically non-decreasing in each component is a mild condition ensuring that an increase in the rate of one user does not decrease the value of the objective function to be maximized. Many practical rate-maximizing criteria rely on concave functions $f(\boldsymbol{\rho})$ that are monotonically non-decreasing in each component [17]. These include "workhorse criteria" such as optimal $\boldsymbol{\alpha}$-weighted sum-rate with $f(\boldsymbol{\rho}) = \boldsymbol{\alpha}^T\boldsymbol{\rho}$, max-min rate with $f(\boldsymbol{\rho}) = \min_{j\in[1,J]} \rho_j$, and max-product rate with $f(\boldsymbol{\rho}) = \prod_{j\in[1,J]} \rho_j$.

At this point it is important to clarify that when solving (4), the matrix $\mathbf{R}$ is considered fixed. Its entries depend on fading and interference through the signal-to-interference-plus-noise ratios (SINRs) and the joint queue occupancy distribution, which in turn depends on the arrival rates, scheduling and routing decisions. Ultimately, routing and scheduling over wireless links should be optimized jointly. But since this appears analytically intractable, similar to [14], [18], [20] where fixed SINRs are adopted as link metrics, we assume that $\mathbf{R}$ is fixed during route optimization[2]. Note also that a conservative (worst-case) fixed $\mathbf{R}$ corresponding to continuously backlogged queues can be justified through a dominant system argument [17]. Indeed, using such a fixed $\mathbf{R}$ amounts to maximizing throughput under a condition that is sufficient for stability. Furthermore, in a heavily loaded system in which queues rarely empty, $\mathbf{R}$ is essentially independent of routing.

Finding optimal routes as solutions of (4) incurs manageable complexity; yet, it requires $\mathbf{R}$ to be available at the access point (AP) – or any designated node for that matter – so that (4) can be solved and the optimal routing matrix $\mathbf{T}^*$ can then be distributed throughout the network. This entails: i) a large communication cost to collect $\mathbf{R}$ and percolate $\mathbf{T}^*$;

[1]We say a function $g(\mathbf{v})$ is monotonically non-decreasing in each component if for vectors $\mathbf{v}^{(1)} := [v_1^{(1)}, \ldots, v_J^{(1)}]^T$ and $\mathbf{v}^{(2)} := [v_1^{(2)}, \ldots, v_J^{(2)}]^T$ with $v_j^{(1)} \preceq v_j^{(2)}$ and $v_i^{(1)} = v_i^{(2)}$ for $i \neq j$, we have that $g[\mathbf{v}^{(1)}] \preceq g[\mathbf{v}^{(2)}]$.

[2]Alternately, recent approaches which account for the coupling between routing and SINRs, e.g., [7], render the problem tractable by sacrificing rate optimality and assuming that the links are binary valued (good or bad).

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

ii) considerable delay to compute $\mathbf{T}$ in a "batch" mode; and iii) lack of resilience to changes in $\mathbf{R}$, a problem particularly important in highly dynamic (e.g., mobile) scenarios.

Distributed on-line routing algorithms, whereby nodes operate in adaptive mode and iteratively exchange variables only with one-hop neighbors tackle precisely these problems. Indeed, in a distributed iterative algorithm it is assumed that U_j has access only to the link reliabilities for transmission to and from other nodes, i.e., the j-th row and column of $\mathbf{R}$, respectively. Consequently, distributed algorithms neither require $\mathbf{R}$ to be available at a central node, nor percolate the routing matrix $\mathbf{T}^*$. Thus, they can afford reduced communication cost, and gain robustness to changes in topology due to fading and/or mobility; see e.g., [7], [13], [19], [22]. The main goal of this paper is to show that the optimization problem in (4) can be solved by an iterative distributed algorithm whereby: i) node U_j has access only to the j-th row and column of $\mathbf{R}$; ii) U_j interchanges messages with one-hop neighbors, defined as the set of terminals with positive probability of decoding U_j's packets; and iii) as time progresses U_j computes its optimal routing probabilities, i.e., the j-th column of $\mathbf{T}$.

The rest of the paper is organized as follows. In Section II we work with max-min optimal routing ($f(\boldsymbol{\rho}) = \min_{j\in[1,J]} \rho_j$) and reformulate (4) into an equivalent problem that is amenable to distributed implementation. We argue that this reformulation is also applicable to a fairly broad class of optimality criteria, thus motivating the generic problem formulation of Section II-A. We then show how dual decomposition techniques can be applied to yield a distributed iterative algorithm converging to the (dual) optimal solution of the (convex) routing problem in Section III. After discussing convergence properties of dual decomposition in Section III-A, we point out that convergence of the algorithm in Section III cannot be always guaranteed, thus motivating the introduction of the method of multipliers and the alternating direction method of multipliers in Section IV. We finally present simulations in Section V. We also illustrate the effect of routing decisions on link reliabilities and consider the applicability of the proposed protocols to mobile environments.

Notation: For a vector $\mathbf{v} := [v_1, \ldots, v_J]^T$ and a set of indices $c = (i_1, \ldots, i_c)$ with $1 \leq i_1 < \ldots < i_c \leq J$ define the vector $\mathbf{v}_c := [v_{i_1}, \ldots, v_{i_c}]^T$. Likewise, for the matrix $\mathbf{M} := (M_{ij})$ define the vectors $\mathbf{M}_{cj} := [M_{i_1 j}, \ldots, M_{i_c j}]^T$ and $\mathbf{M}_{jc} := [M_{j i_1}, \ldots, M_{j i_c}]^T$ containing subsets of the j-th column and row of $\mathbf{M}$ respectively. Note that even if $\mathbf{M}_{jc}$ contains a subset of $\mathbf{M}$'s j-th row it is defined as a column vector. The componentwise product of vectors $\mathbf{v}^{(1)} := [v_1^{(1)}, \ldots, v_J^{(1)}]^T$ and $\mathbf{v}^{(2)} := [v_1^{(2)}, \ldots, v_J^{(2)}]^T$ is denoted as $\mathbf{v}^{(1)} \cdot \mathbf{v}^{(2)} := [v_1^{(1)} v_1^{(2)}, \ldots, v_J^{(1)} v_J^{(2)}]^T$.

II. A Separable Problem

The optimization problem in (4) is not in a form that facilitates a distributed solution. Towards this end, we first outline in this section equivalent reformulations, whose solution coincides with (4) for a given $f(\boldsymbol{\rho})$. The reformulated problems can be separated via a dual decomposition, and lend themselves to a distributed solution. For specificity, consider as optimality criterion the rate of the worst user $f(\boldsymbol{\rho}) = \min_{j\in[1,J]} \rho_j$, leading to the problem

$$\begin{aligned} \mathbf{T}^* = \arg\max \quad & \min_{j\in[1,J]} [(\mathbf{I} - \mathbf{K}_D)\boldsymbol{\mu}]_j \\ \text{s.t.} \quad & K_{ij} = R_{ij} T_{ij} \quad \text{for } i \neq j, \\ & \mathbf{K}^T \mathbf{1} = \mathbf{1}, \quad \mathbf{T}^T \mathbf{1} = \mathbf{1}. \end{aligned} \tag{5}$$

In order to reduce the number of variables we will eliminate the equality constraints in (5). To this end, define the set $c(j) := \{i : R_{ij} > 0; i \neq j, i \in [1, J+1]\}$ containing the indices of terminals U_i that can decode U_j's transmission with non-zero probability. Likewise, define $r(j) := \{i : R_{ji} > 0; i \neq j, i \in [1, J+1]\}$ as the set of nodes that U_j decodes with non-zero probability. Using these definitions we can write the rate of the j-th user as

$$\begin{aligned} \rho_j &= [(\mathbf{I} - \mathbf{K}_D)\boldsymbol{\mu}]_j \\ &= \mu_j (1 - K_{jj}) - \sum_{i\in r(j)} \mu_i K_{ji} \\ &= \mu_j \sum_{i\in c(j)} K_{ij} - \sum_{i\in r(j)} \mu_i K_{ji} \end{aligned} \tag{6}$$

where in the second equality we used the constraint $\mathbf{K}^T \mathbf{1} = \mathbf{1}$. Upon substituting $K_{ij} = R_{ij} T_{ij}$, (6) becomes

$$\rho_j = \sum_{i\in c(j)} \mu_j R_{ij} T_{ij} - \sum_{i\in r(j)} \mu_i R_{ji} T_{ji}. \tag{7}$$

For a more compact notation let vectors $\mathbf{t}_j := \mathbf{T}_{c(j)j}$ and $\mathbf{t}'_j = \mathbf{T}_{jc(j)}$ contain the non-zero elements of the j-th column and row of $\mathbf{T}$, respectively. We further define the vectors $\mathbf{r}_j := \mu_j \mathbf{R}_{c(j)j}$ and $\mathbf{s}_j := \boldsymbol{\mu}_{c(j)} \cdot \mathbf{R}_{jc(j)}$ so that

$$\rho_j = \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{t}'_j. \tag{8}$$

Vectors $\mathbf{r}_j$ and $\mathbf{s}_j$ are constant and known at node U_j. Indeed, $\mathbf{R}_{c(j)j}$ contains the probabilities of other nodes $U_i \neq U_j$ decoding correctly U_j's packets. This U_j can easily estimate by counting acknowledgments of packets sent to these terminals. The probabilities of U_j decoding correctly transmissions from other nodes (required to construct $\mathbf{R}_{jc(j)}$) can be fed back from the corresponding (one-hop) neighbors. We assume that estimation of success probabilities and associated feedback among neighboring nodes is perfect.

Using (8) and noting that the constraint $\mathbf{T}^T \mathbf{1} = \mathbf{1}$ is equivalent to the set of constraints $\{\mathbf{t}_j^T \mathbf{1} = 1\}_{j=1}^J$, we can rewrite the max-min optimal routing problem in (5) as

$$\begin{aligned} \mathbf{T}^* = \arg\max \quad & w \\ \text{s.t.} \quad & w \preceq \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{t}'_j = \rho_j, \ \mathbf{t}_j^T \mathbf{1} = 1, \ \mathbf{0} \preceq \mathbf{t}_j. \end{aligned} \tag{9}$$

Even though (9) is written in terms of local variables ($\mathbf{t}_j$), local constants ($\mathbf{r}_j$, $\mathbf{s}_j$) and neighboring variables ($\mathbf{t}'_j$), it is not yet in a separable form. Indeed, note that: i) the variable w is constrained to be smaller than the rates ρ_j of the J terminals and in that sense its optimization requires access to all the variables; and ii) computing ρ_j requires access to the local variables $\mathbf{t}_j$ and neighboring variables $\mathbf{t}'_j$. While $\mathbf{t}_j$ contains U_j's transmission probabilities (the variable that U_j is interested to optimize), $\mathbf{t}'_j$ contains the probabilities of other

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

terminals U_i routing their packets through U_j, a variable that U_j's (one-hop) neighbors are interested to optimize.

To overcome these hurdles we introduce local variables w_j and $\mathbf{u}_j$ that can be viewed as U_j's estimates of (the global variable) w and (the neighboring variable) $\mathbf{t}'_j$. We then introduce equality constraints $\mathbf{u}_j = \mathbf{t}'_j$ and $w = w_j$, $\forall j \in [1, J]$. Using these (local) variables we can write the constraint in (9) as

$$w_j \leq \rho_j = \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{u}_j; \quad \mathbf{u}_j = \mathbf{t}'_j, \quad w = w_j. \tag{10}$$

The last step is to replace w in (9) by the weighted sum $w = (\sum_{j=1}^J w_j)/J$. Note that if there is a non-zero probability for a multi-hop route connecting any pair of nodes, the set of constraints $\{w_j = w_i \ \forall i \in c(j)\}_{j=1}^J$ is equivalent to requiring $w_i = w_j$, $\forall\ i, j \in [1, J]$. We can now reformulate (9) as

$$\begin{aligned} \mathbf{T}^* = \arg\max \ & \frac{1}{J} \sum_{j=1}^J w_j \\ \text{s.t.} \ & w_j \leq \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{u}_j, \ \ \mathbf{t}_j^T \mathbf{1} = 1, \ \ \mathbf{0} \preceq \mathbf{t}_j \\ & \mathbf{t}'_j = \mathbf{u}_j, \ \ w_j = w_i \ \forall i \in c(j) \end{aligned} \tag{11}$$

where the maximization is over $\{\mathbf{t}_j\}_{j=1}^J$ (or, equivalently, $\mathbf{T}$), $\{\mathbf{u}_j\}_{j=1}^J$, and $\mathbf{w} := [w_1, \ldots, w_J]^T$.

We summarize the equivalence of (9) and (11) in the following proposition.

Proposition 1 *If there exists a non-zero probability multihop route between any pair of nodes, the matrix* $\mathbf{T}^*$ *is a solution of* (9) *if and only if it is a solution of* (11).

Comparing (9) with (11) we recognize that the latter does not contain any intrinsically global variable and that the sole coupling between terminals is through the equality constraints $\mathbf{t}'_j = \mathbf{u}_j$ and $w_j = w_i$ for all $i \in c(j)$. An important feature of (11) is that the constraints on the problem variables can be classified into i) local constraints involving only variables kept at the j-th terminal; and ii) coupling constraints enforcing the equality with neighboring variables of interest. Indeed, note that $w_j \leq \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{u}_j$, $\mathbf{t}_j^T \mathbf{1} = 1$, and $\mathbf{0} \preceq \mathbf{t}_j$ involve the variables $\mathbf{x}_j := (w_j, \mathbf{t}_j, \mathbf{u}_j)$ only. Thus, they can be locally enforced, meaning that it is possible for U_j to find values of $\mathbf{x}_j$ satisfying these constraints. The equality constraints cannot be enforced locally but it is important to note that they relate neighboring variables only. Readers familiar with dual decomposition techniques – see e.g., [2, Sec. 3.4.2], [11], [15] – may notice that the form of (11) lends itself to distributed optimization of the type we will elaborate on in Section III.

Remark 1 The terms $\mathbf{r}_j^T \mathbf{t}_j$ and $\mathbf{s}_j^T \mathbf{t}'_j$ in (8) respectively correspond to packets successfully transmitted from and to U_j. Their difference is the rate ρ_j available to U_j's own packets. This interpretation of (8) is reminiscent of the one encountered in flow control optimization [2, Sec. 5.1]. Different from flow control, the optimization here over the probabilities T_{ij} has to account for the joint constraint $\mathbf{t}_j^T \mathbf{1} = 1$ that outgoing flows from U_j must adhere to. Notwithstanding, flow control is about deterministic splitting of traffic for load balancing purposes, and the optimization of network flows is implemented at the transport layer where optimal routes are assumed available.

A. General problem formulation

The equivalence between (5) and (11) is not unique to max-min optimal routing since the same steps can be applied to reformulate many optimization problems. To clarify this point consider a given packet success probability matrix $\mathbf{R}$ of which node U_j only knows the non-zero elements of its j-th column and row $\mathbf{r}_j := \mathbf{R}_{c(j)j}$ and $\mathbf{s}_j := \mathbf{R}_{jc(j)}$. Terminal U_j is interested in finding the vector $\mathbf{t}_j := \mathbf{T}_{c(j)j}$ that determines its probability of routing packets through neighboring nodes. Introduce the matrix $\mathbf{U}$ with the same sparsity pattern as $\mathbf{T}$ and let $\mathbf{u}_j := \mathbf{U}_{jc(j)}$ denote the non-zero components of the j-th row of $\mathbf{U}$. Node U_j maintains locally the variables w_j, $\mathbf{t}_j$, and $\mathbf{u}_j$ that we arrange in the vector $\mathbf{x}_j := [w_j, \mathbf{t}_j^T, \mathbf{u}_j^T]^T$. Also, recall that $\mathbf{w} := [w_1, \ldots, w_J]^T$ contains the variables w_j of all terminals and let $\mathbf{v}_j := \mathbf{w}_{c(j)}$ collect the variables w_j of U_j's neighbors. Finally, abbreviate by $\mathbf{X} := (\mathbf{w}, \mathbf{T}, \mathbf{U})$ the triplet of problem variables.

Our goal in this paper is to find distributed algorithms converging to the optimal solution of the problem

$$\begin{aligned} \mathbf{X}^* = \arg\max_{\mathbf{X}} \ & \mathbf{w}^T \mathbf{1} \\ \text{s.t.} \ & \mathbf{x}_j \in \mathcal{X}_j, \ \mathbf{t}'_j = \mathbf{u}_j, \ \mathbf{v}_j = w_j \mathbf{1} \end{aligned} \tag{12}$$

where we defined $\mathbf{X}^* := (\mathbf{T}^*, \mathbf{U}^*, \mathbf{w}^*)$ and $\mathcal{X}_j$ is a set specifying the chosen routing optimality criterion. Note that the constraint $\mathbf{t}'_j = \mathbf{u}_j$ implies that $\mathbf{T}^* = \mathbf{U}^*$ so that after obtaining the optimal solution U_j knows the (optimal) probabilities $\mathbf{t}_j$ with which to route its packets through its neighbors and the probabilities $\mathbf{t}'_j = \mathbf{u}_j$ with which its neighbors route packets through U_j.

To find the optimal solution to (12) we require the following operational conditions to hold:

- **(a1)** The set $\mathcal{X}_j$ is convex.
- **(a2)** There is a non-zero probability multi-hop route connecting any pair of nodes.
- **(a3)** Node U_j can only communicate with its one hop neighbors $\{U_i : i \in c(j)\}$; hence, it has no access to variables of other nodes.
- **(a4)** The probability that U_j decodes U_i is non-zero if and only if the probability that U_i decodes U_j is non-zero. This implies $c(j) = r(j)$ for all $j \in [1, J]$.

Assumption (a1) ensures that the problem in (12) is convex; (a2) is required so that the constraints $\{w_j = w_i \ \forall i \in c(j)\}_{j=1}^J$ imply $w_i = w_j$, $\forall i, j \in [1, J]$; (a3) is in line with the distributed setup; and (a4) guarantees that if U_j has access to U_i's variables then U_i has access to U_j's variables, which is natural in a peer-to-peer setting, and will be exploited later on.

The formulation in (12) encompasses all the routing problems introduced in [17], with the set $\mathcal{X}_j$ specifying the corresponding optimality criterion. In particular, we have:

Max-min optimal rate. This is the problem considered in detail in Section II and can be obtained from (12) by defining the set

$$\mathcal{X}_j^1 := \left\{ \mathbf{x}_j : w_j \leq \mathbf{r}_j^T \mathbf{t}_j - \mathbf{s}_j^T \mathbf{u}_j, \ \mathbf{0} \preceq \mathbf{t}_j, \ \mathbf{t}_j^T \mathbf{1} = 1 \right\}. \tag{13}$$

Additional convex constraints can be added to the definition of $\mathcal{X}_j$. Since we know that $\mathbf{u}_j$ is a vector of probabilities,

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

using the set $\mathcal{X}_j = \mathcal{X}_j^1 \cap \{\mathbf{x}_j : \mathbf{0} \preceq \mathbf{u}_j \preceq \mathbf{1}\}$ is equivalent to using $\mathcal{X}_j^1$ because the constraint $\mathbf{0} \preceq \mathbf{u}_j \preceq \mathbf{1}$ is implicit in $\mathbf{u}_j = \mathbf{t}_j'$. Preventing the components of $\mathbf{u}_j$ to become too large enhances the numerical stability of the problem.

Optimal weighted sum-rate. Here the goal is to maximize a weighted sum of average rates, i.e., $f(\boldsymbol{\rho}) = \boldsymbol{\alpha}^T\boldsymbol{\rho}$ with $\boldsymbol{\alpha} := [\alpha_1, \ldots, \alpha_J]^T \succeq \mathbf{0}$. In this case we define the set

$$\mathcal{X}_j^2 := \left\{\mathbf{x}_j : w_j = \alpha_j(\mathbf{r}_j^T\mathbf{t}_j - \mathbf{s}_j^T\mathbf{u}_j),\ \mathbf{0} \preceq \mathbf{t}_j,\ \mathbf{t}_j^T\mathbf{1} = 1\right\} \tag{14}$$

and consider the optimization problem

$$\begin{aligned}\mathbf{T}^* = \arg\max\ & \mathbf{w}^T\mathbf{1} \\ \text{s.t.}\ & \mathbf{x}_j := (w_j, \mathbf{t}_j, \mathbf{u}_j) \in \mathcal{X}_j^2,\ \mathbf{t}_j' = \mathbf{u}_j\end{aligned} \tag{15}$$

which amounts to dropping the constraint $\mathbf{v}_j = w_j\mathbf{1}$ in (12). Note that for this criterion $w_j = \alpha_j\rho_j$.

Extra convex constraints can be dealt with by modifying the set $\mathcal{X}_j^2$ in (14). A case of interest is to consider a minimum acceptable rate $\rho_j^{\min}$ for terminal U_j that can be accommodated by considering the set $\mathcal{X}_j := \mathcal{X}_j^2 \cap \{w_j/\alpha_j \geq \rho_j^{\min}\}$. A solution $\mathbf{T}^*$ to (15) with a minimum rate constraint may not exist for some values of $\boldsymbol{\rho}^{\min}$ – in such cases interior point methods return an infeasibility certificate. When it exists, $\mathbf{T}^*$ ensures the minimum acceptable rate $\rho_j^{\min}$ to every user with the excess traffic distributed to the most favored users having large weights α_j and/or reliable connections to the AP.

Optimal product of rates. Maximizing the product of rates constitutes a fairer alternative to the weighted sum-rate criterion in (15) since it prevents solutions in which users with less reliable links receive a very small packet delivery rate. The function to be maximized in this case is $f(\boldsymbol{\rho}) = \prod_{j=1}^J \rho_j$. Equivalently, since the logarithm is monotonically increasing the concave function $f(\boldsymbol{\rho}) = \sum_{j=1}^J \log(\rho_j)$ can be used instead. To cast this problem under the distributable formulation in (12) it suffices to replace $\mathcal{X}_j^2$ in (15) by

$$\mathcal{X}_j^3 := \left\{\mathbf{x}_j : w_j \leq \log[\mathbf{r}_j^T\mathbf{t}_j - \mathbf{s}_j^T\mathbf{u}_j],\ \mathbf{0} \preceq \mathbf{t}_j,\ \mathbf{t}_j^T\mathbf{1} = 1\right\}. \tag{16}$$

The local variables w_j denote the logarithm of the local rate.

Another example of a convex constraint is a cooperation limit whereby terminals require their own rate to be at least a certain percentage $\beta_j \in [0,1]$ of their total outgoing rate $\mathbf{r}_j^T\mathbf{t}_j$. The set effecting this constraint is $\mathcal{X}_j := \mathcal{X}_j^3 \cap \{\mathbf{x}_j : \mathbf{r}_j^T\mathbf{t}_j - \mathbf{s}_j^T\mathbf{u}_j \geq \beta_j(\mathbf{r}_j^T\mathbf{t}_j)\}$. This constraint guarantees that at least β_j of the packets U_j transmits were generated at U_j.

Optimal rate with relays. In a relay network a group of terminals collaborate in relaying traffic on behalf of a designated active user. Let U_{j_0} denote this active user and terminals $\{U_j\}_{j=1,j\neq j_0}^J$ be willing to serve as relays. The optimal relay network maximizing the rate ρ_{j_0} can be found by solving (12) with

$$\begin{aligned}\mathcal{X}_j^3 &= \left\{\mathbf{x}_j : 0 = \mathbf{r}_j^T\mathbf{t}_j - \mathbf{s}_j^T\mathbf{u}_j,\ \mathbf{0} \preceq \mathbf{t}_j,\ \mathbf{t}_j^T\mathbf{1} = 1\right\},\ j \neq j_0 \\ \mathcal{X}_{j_0}^3 &= \left\{\mathbf{x}_{j_0} : w_{j_0} = \mathbf{r}_{j_0}^T\mathbf{t}_{j_0} - \mathbf{s}_{j_0}^T\mathbf{u}_{j_0},\ \mathbf{0} \preceq \mathbf{t}_{j_0},\ \mathbf{t}_{j_0}^T\mathbf{1} = 1\right\}.\end{aligned} \tag{17}$$

In this example, w_j is the local estimate of the source's rate ρ_{j_0} at terminal U_j.

III. Distributed Implementation via Dual Decomposition

Problems of the general form (12) or (15) can be solved using the so called dual decomposition methods [2, Sec. 3.4.2], [15]. Since (a1) guarantees convexity of the problem the basic idea is to optimize the dual function that, as we will show in this section, exhibits a separable structure. Associate, thus, Lagrange multipliers $\mathbf{p}_j$ with the constraints $\mathbf{t}_j' - \mathbf{u}_j = \mathbf{0}$ and $\mathbf{q}_j$ with the constraints $\mathbf{v}_j - w_j\mathbf{1} = \mathbf{0}$ to form the Lagrangian

$$\mathcal{L}(\mathbf{X}, \mathbf{P}, \mathbf{Q}) = -\mathbf{w}^T\mathbf{1} + \sum_{j=1}^J \left[(\mathbf{t}_j' - \mathbf{u}_j)^T\mathbf{p}_j + (\mathbf{v}_j - w_j\mathbf{1})^T\mathbf{q}_j\right] \tag{18}$$

which is defined over the feasible region of the primal variables $\{\mathbf{x}_j \in \mathcal{X}_j\}_{j=1}^J$. Matrices $\mathbf{P}$ and $\mathbf{Q}$ are defined to have the same sparsity pattern as $\mathbf{T}$ (and thus $\mathbf{U}$); the dual variables (multipliers) in (18) are respectively given by the non-zero elements of the j-th row of $\mathbf{P}$ and the j-th column of $\mathbf{Q}$; i.e., $\mathbf{p}_j = \mathbf{P}_{jc(j)}$ and $\mathbf{q}_j = \mathbf{Q}_{c(j)j}$. Vectors $\mathbf{p}_j$ and $\mathbf{q}_j$ are maintained at terminal U_j.

The Lagrangian in (18) is used to obtain the dual function

$$g(\mathbf{P}, \mathbf{Q}) = \min_{\{\mathbf{x}_j \in \mathcal{X}_j\}_{j=1}^J} \mathcal{L}(\mathbf{X}, \mathbf{P}, \mathbf{Q}) \tag{19}$$

which in turn leads to the dual problem defined as the unconstrained maximization of $g(\mathbf{P}, \mathbf{Q})$ – note that we do not impose non-negativity constraints on the multipliers because this is an equality-constrained problem. For convex optimization problems strong duality holds, implying that the maximum in (12) coincides with the negative of the maximum of the dual function $g(\mathbf{P}, \mathbf{Q})$; i.e.,

$$\mathbf{1}^T\mathbf{w}^* = -\max_{\mathbf{P},\mathbf{Q}} g(\mathbf{P}, \mathbf{Q}). \tag{20}$$

The problem in (20) is an unconstrained optimization problem that can be solved with a gradient ascent algorithm. However, since the dual function $g(\mathbf{P}, \mathbf{Q})$ is not always differentiable a generalization of the gradient, the so called *subgradient*, is used instead.

Definition 1 *Consider a concave function $f(\mathbf{P}) : \mathbb{R}^M \to \mathbb{R}$. If $\nabla_{\mathbf{P}}(\mathbf{P})$ satisfies*

$$f(\tilde{\mathbf{P}}) \leq f(\mathbf{P}) + \nabla_{\mathbf{P}}(\mathbf{P})(\tilde{\mathbf{P}} - \mathbf{P}) \tag{21}$$

for all $\tilde{\mathbf{P}} \in \mathbb{R}^M$ we say that $\nabla_{\mathbf{P}}(\mathbf{P})$ is a subgradient of $f(\mathbf{P})$ at $\mathbf{P}$. Given a subset of $\mathbf{P}$ entries collected in a vector $\mathbf{p}_j$ let $\nabla_{\mathbf{p}_j}(\mathbf{P})$ denote the corresponding entries of $\nabla_{\mathbf{P}}(\mathbf{P})$.

The subgradient is any vector $\nabla_{\mathbf{P}}$ defining a supporting hyperplane of the concave function $f(\mathbf{P})$. When a gradient exists, i.e., when $f(\mathbf{P})$ is differentiable, it is the unique subgradient of $f(\mathbf{P})$.

A subgradient of $g(\mathbf{P}, \mathbf{Q})$ is presented in the next proposition; see also [2, Sec. 3.4.2].

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

Proposition 2 *For given multipliers* $\mathbf{P}$ *and* $\mathbf{Q}$*, let* $\mathbf{X}^\dagger(\mathbf{P},\mathbf{Q})$ *denote the optimal argument of the Lagrangian, i.e.,*

$$\mathbf{X}^\dagger(\mathbf{P},\mathbf{Q}) := \arg \min_{\{\mathbf{x}_j \in \mathcal{X}_j\}_{j=1}^J} \mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q}) \tag{22}$$

with $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q})$ *given by* (18)*. Then, a subgradient* $\nabla_{\mathbf{P},\mathbf{Q}}$ *of* $g(\mathbf{P},\mathbf{Q})$ *has entries*

$$\begin{aligned}\nabla_{\mathbf{p}_j}(\mathbf{P},\mathbf{Q}) &= \mathbf{t}_j'^{\dagger}(\mathbf{P},\mathbf{Q}) - \mathbf{u}_j^\dagger(\mathbf{P},\mathbf{Q}) \\ \nabla_{\mathbf{q}_j}(\mathbf{P},\mathbf{Q}) &= \mathbf{v}_j^\dagger(\mathbf{P},\mathbf{Q}) - w_j^\dagger(\mathbf{P},\mathbf{Q})\mathbf{1}.\end{aligned} \tag{23}$$

Proof: Consider the value of the dual function for arbitrary multiplier matrices $(\tilde{\mathbf{P}},\tilde{\mathbf{Q}})$

$$g(\tilde{\mathbf{P}},\tilde{\mathbf{Q}}) = \min_{\{\mathbf{x}_j \in \mathcal{X}_j\}_{j=1}^J} -\mathbf{w}^T\mathbf{1} \tag{24}$$

$$+\sum_{j=1}^J \left[(\mathbf{t}_j' - \mathbf{u}_j)^T\tilde{\mathbf{p}}_j + (\mathbf{v}_j - w_j\mathbf{1})^T\tilde{\mathbf{q}}_j\right]$$

$$\leq -\mathbf{w}^{\dagger T}\mathbf{1} + \sum_{j=1}^J \left[(\mathbf{t}_j'^{\dagger} - \mathbf{u}_j^\dagger)^T\tilde{\mathbf{p}}_j + (\mathbf{v}_j^\dagger - w_j^\dagger\mathbf{1})^T\tilde{\mathbf{q}}_j\right] \tag{25}$$

where for notational simplicity we omit the arguments of $\mathbf{t}_j'^{\dagger}$, $\mathbf{u}^\dagger$, $\mathbf{v}_j^\dagger$, and $w_j^\dagger$. The equality in (24) follows from the definitions of the dual function in (19) and the Lagrangian in (18); and the inequality in (25) is true since $\{\mathbf{x}^\dagger\}_{j=1}^J = \mathbf{X}^\dagger$ cannot yield a value smaller than the optimal argument of (24).

Subtracting $g(\mathbf{P},\mathbf{Q}) = -\mathbf{w}^{\dagger T}\mathbf{1} + \sum_{j=1}^J \left[(\mathbf{t}_j'^{\dagger} - \mathbf{u}_j^\dagger)^T\mathbf{p}_j + (\mathbf{v}_j^\dagger - w_j^\dagger\mathbf{1})^T\mathbf{q}_j\right]$ from both sides of the inequality in (25) yields

$$g(\tilde{\mathbf{P}},\tilde{\mathbf{Q}}) - g(\mathbf{P},\mathbf{Q}) \leq \sum_{j=1}^J \left[(\mathbf{t}_j'^{\dagger} - \mathbf{u}_j^\dagger)^T(\tilde{\mathbf{p}}_j - \mathbf{p}_j) + (\mathbf{v}_j^\dagger - w_j^\dagger\mathbf{1})^T(\tilde{\mathbf{q}}_j - \mathbf{q}_j)\right]. \tag{26}$$

Comparing (26) with (21) we recognize that the constraint violations in (23) satisfy the definition of a subgradient of $g(\mathbf{P},\mathbf{Q})$ [cf. (21)]. ∎

Proposition 2 asserts that for general multipliers $(\mathbf{P},\mathbf{Q})$ the Lagrangian is optimized by variables $\mathbf{X}^\dagger(\mathbf{P},\mathbf{Q})$ which violate the equality constraints in (12). Interestingly, the amount by which the equality constraints are violated is a subgradient of the dual function. Indeed, the multiplier $\mathbf{p}_j$ (respectively $\mathbf{q}_j$) is associated with the constraint $\mathbf{t}_j' - \mathbf{u}_j = \mathbf{0}$ (respectively $\mathbf{v}_j - w_j\mathbf{1} = \mathbf{0}$); the optimal arguments of the Lagrangian violate this constraint by an amount $\nabla_{\mathbf{p}_j}(\mathbf{P},\mathbf{Q}) = \mathbf{t}_j'^{\dagger}(\mathbf{P},\mathbf{Q}) - \mathbf{u}_j^\dagger(\mathbf{P},\mathbf{Q})$ (respectively $\nabla_{\mathbf{q}_j}(\mathbf{P},\mathbf{Q}) = \mathbf{v}_j^\dagger(\mathbf{P},\mathbf{Q}) - w_j^\dagger(\mathbf{P},\mathbf{Q})\mathbf{1}$).

An important property of the optimal arguments of the Lagrangian is that they can be computed locally at each node. To be precise, define the vectors $\mathbf{p}_j' = \mathbf{P}_{c(j)j}$ and $\mathbf{q}_j' = \mathbf{Q}_{jc(j)}$ containing the dual variables of the one-hop neighbors $\{U_i : i \in c(j)\}$, and construct the *local* Lagrangian $\mathcal{L}_j(\mathbf{x}_j;\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j')$ by grouping the terms that depend only on the local variable $\mathbf{x}_j$ [cf. (18)]

$$\mathcal{L}_j(\mathbf{x}_j;\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j') = -w_j + \mathbf{t}_j^T\mathbf{p}_j' - \mathbf{u}_j^T\mathbf{p}_j + w_j\mathbf{1}^T(\mathbf{q}_j' - \mathbf{q}_j). \tag{27}$$

By construction $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q}) = \sum_{j=1}^J \mathcal{L}_j(\mathbf{x}_j;\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j')$ [cf. (18) and (27)]. If we further note that the primal variables $\mathbf{x}_j$ appear only in $\mathcal{L}_j(\mathbf{x}_j;\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j')$, we deduce that the optimal arguments in (22) can be found as

$$\mathbf{x}_j^\dagger := \arg \min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{L}_j(\mathbf{x}_j,\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j'). \tag{28}$$

The ultimate reasons enabling a distributed implementation of a subgradient ascent algorithm can be read out from Proposition 2 and (28): i) a subgradient of the dual function is obtained from the arguments optimizing the Lagrangian $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q})$ [cf. (23)]; ii) the subgradients $\nabla_{\mathbf{p}_j}(\mathbf{P},\mathbf{Q})$ and $\nabla_{\mathbf{q}_j}(\mathbf{P},\mathbf{Q})$ depend only on local and neighboring variables [cf. (23)]; and iii) the optimization of the Lagrangian $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q})$ separates into the optimization of J local Lagrangians $\mathcal{L}_j(\mathbf{x}_j,\mathbf{p}_j,\mathbf{q}_j,\mathbf{p}_j',\mathbf{q}_j')$. Furthermore, these local Lagrangians depend only on local and neighboring variables [cf. (27) and (28)].

Consequently, subgradient ascent for $g(\mathbf{P},\mathbf{Q})$ can be implemented by the following distributable iteration:

[I1] *Compute subgradient*. Given local multipliers $\mathbf{p}_j(n)$ and $\mathbf{q}_j(n)$, and neighboring multipliers $\mathbf{p}_j'(n)$ and $\mathbf{q}_j'(n)$, minimize the local Lagrangian with respect to the local primal variables (n denotes iteration index)

$$\begin{aligned}\mathbf{x}_j(n) &= \arg \min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{L}_j[\mathbf{x}_j,\mathbf{p}_j(n),\mathbf{q}_j(n),\mathbf{p}_j'(n),\mathbf{q}_j'(n)] \\ &:= \arg \min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{L}_j(\mathbf{x}_j,n)\end{aligned} \tag{29}$$

where we defined $\mathcal{L}_j(\mathbf{x}_j,n) := \mathcal{L}_j[\mathbf{x}_j,\mathbf{p}_j(n),\mathbf{q}_j(n),\mathbf{p}_j'(n),\mathbf{q}_j'(n)]$ and the primal iterates are $\mathbf{x}_j(n) := [w_j(n),\mathbf{t}_j(n)^T,\mathbf{u}_j(n)^T]^T$.

[I2] *Subgradient ascent step*. Using local primal variables $[w_j(n),\mathbf{t}_j(n),\mathbf{u}_j(n)]$ and neighboring primal variables $[\mathbf{v}_j(n),\mathbf{t}_j'(n),\mathbf{u}_j'(n)]$ update local multipliers (c_n is a properly selected step size)

$$\begin{aligned}\mathbf{p}_j(n+1) &= \mathbf{p}_j(n) + c_n[\mathbf{t}_j'(n) - \mathbf{u}_j(n)] \\ \mathbf{q}_j(n+1) &= \mathbf{q}_j(n) + c_n[\mathbf{v}_j(n) - w_j(n)\mathbf{1}].\end{aligned} \tag{30}$$

Algorithm 1 details the distributed implementation of [I1]-[I2]. Given the local multipliers $\mathbf{p}_j(n)$ and $\mathbf{q}_j(n)$, and the one-hop-neighbors' multipliers $\mathbf{p}_j'(n)$ and $\mathbf{q}_j'(n)$, user terminal U_j solves a (local) convex optimization problem to find the primal variables $\mathbf{x}_j(n)$ that optimize the (local and global) Lagrangian in step 3. In turn, these primal variables are used in the gradient ascent steps 6 and 7 to obtain the updated multipliers $\mathbf{p}_j(n+1)$ and $\mathbf{q}_j(n+1)$. Steps 6 and 7 represent the subgradient ascent step for the dual function $g(\mathbf{P},\mathbf{Q})$ and as such are the steps guaranteeing convergence of the iterates $\{\mathbf{p}_j(n)\}_{j=1}^J$ and $\{\mathbf{q}_j(n)\}_{j=1}^J$ obtained from (29)-(30) to $\{\mathbf{p}^*,\mathbf{q}^*\}_{j=1}^J := \arg\max g(\mathbf{P},\mathbf{Q})$ as $n \to \infty$ (convergence of (29)-(30) requires some qualifications that we discuss in the next subsection). The remaining steps ensure that the variables are properly communicated. Steps 8 and 2 ensure

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

Algorithm 1 Dual decomposition solver

Require: Packet success probabilities to and from neighbors $\mathbf{R}_{c(j)j}$ and $\mathbf{R}_{jc(j)}$

Ensure: Optimal multipliers $\mathbf{p}_j^*$ and $\mathbf{q}_j^*$

1: **for** $n = 1$ to ∞ **do** {repeat for the life of the network}
2: Receive multipliers $p_{ij}(n)$ and $q_{ji}(n)$ from one hop neighbors $\{U_i : i \in c(j)\}$
3: Minimize Lagrangian [cf. (29)]: $\mathbf{x}_j(n) = \arg\min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{L}_j[\mathbf{x}_j, \mathbf{p}_j(n), \mathbf{q}_j(n), \mathbf{p}'_j(n), \mathbf{q}'_j(n)]$
4: Transmit $w_j(n)$, $t_{ij}(n)$, and $u_{ji}(n)$ to neighbor U_i; repeat for all $\{U_i : i \in c(j)\}$.
5: Receive $w_i(n)$, $t_{ji}(n)$, and $u_{ij}(n)$ from one hop neighbors $\{U_i : i \in c(j)\}$
6: Subgradient ascent iteration for $\mathbf{p}_j$ [cf. (30)]: $\mathbf{p}_j(n+1) = \mathbf{p}_j(n) + c_n[\mathbf{t}'_j(n) - \mathbf{u}_j(n)]$
7: Subgradient ascent iteration for $\mathbf{q}_j$ [cf. (30)]: $\mathbf{q}_j(n+1) = \mathbf{q}_j(n) + c_n[\mathbf{v}_j(n) - w_j(n)\mathbf{1}]$
8: Transmit multipliers $p_{ji}(n+1)$ and $q_{ij}(n+1)$ to neighbor U_i; repeat for all $\{U_i : i \in c(j)\}$
9: **end for**

that the updated multipliers are sent to and received by the corresponding neighboring node, while steps 4 and 5 guarantee the same for the primal variables.

Remark 2 Strictly speaking the problem formulation in (15) is not a special case of (12) since it involves dropping the constraints $\mathbf{v}_j = w_j\mathbf{1}$. However, the Lagrangian for the optimization problem in (15) can be obtained from (18) by setting $\mathbf{q}_j = \mathbf{0}$. In that sense the subsequent treatment leading to Algorithm 1 can be reproduced by eliminating all references to the multipliers $\mathbf{q}_j$. To obtain an algorithm solving the dual problem of (15) remove all references to dual variables $q_{ij}(n)$ and $\mathbf{q}_j(n)$ from Algorithm 1.

A. Discussion of convergence properties

The goal of Algorithm 1 is for U_j to obtain the optimal routing probabilities $\mathbf{t}_j$. We are thus interested in having $\lim_{n\to\infty} \mathbf{t}_j(n) = \mathbf{t}_j^*$, with $\mathbf{t}_j(n)$ obtained from the iteration (29)-(30) and $\mathbf{t}_j^*$ the solution of (12). Since the iteration (29)-(30) implements subgradient ascent for the dual function, convergence of the primal variables cannot be always guaranteed. Relevant convergence properties of subgradient descent are summarized next (see e.g., [2, Sec. 3.4.3]).

Property 1 *Consider the iteration* (29)-(30) *and let* $\{\mathbf{p}^*, \mathbf{q}^*\}_{j=1}^J := \arg\max g(\mathbf{P}, \mathbf{Q})$ *denote the optimal solution of the dual problem in* (20)*. We then have that:*

(a) *if the step size is constant, i.e.,* $c_n = c\ \forall n$*, then*

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} \mathbf{p}_j(n) = \mathbf{p}_j^* \qquad \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} \mathbf{q}_j(n) = \mathbf{q}_j^* \tag{31}$$

implying that the average value of the dual iterates converges to the optimal dual variables; and

(b) *if the step size sequence is non-summable,* $\sum_{n=0}^{\infty} c_n = \infty$ *but square summable,* $\sum_{n=0}^{\infty} c_n^2 < \infty$*,*

$$\lim_{n\to\infty} \mathbf{p}_j(n) = \mathbf{p}_j^* \qquad \lim_{n\to\infty} \mathbf{q}_j(n) = \mathbf{q}_j^* \tag{32}$$

implying that the sequence of dual iterates converges to the optimal dual variables.

Primal optimal variables $\{\mathbf{x}_j^*\}_{j=1}^J$ cannot always be recovered from dual optimal variables $\{\mathbf{p}_j^*\}_{j=1}^J$ and $\{\mathbf{q}_j^*\}_{j=1}^J$ [3, Sec. 5.5.5]. If, e.g., (12) amounts to a linear program, the optimal dual variables $\{\mathbf{p}_j^*\}_{j=1}^J$ and $\{\mathbf{q}_j^*\}_{j=1}^J$ render the Lagrangian $\mathcal{L}(\mathbf{X}, \mathbf{P}^*, \mathbf{Q}^*)$ in (18) independent of the primal variables $\mathbf{X}$ [3, Sec. 5.5.5]. Since we determine $\mathbf{x}_j(n)$ as $\min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{L}_j\left[\mathbf{x}_j; \mathbf{p}_j(n), \mathbf{q}_j(n), \mathbf{p}'_j(n), \mathbf{q}'_j(n)\right]$, a sequence of dual iterates converging to optimal dual variables, i.e., $\mathbf{p}_j(n) \to \mathbf{p}_j^*$ and $\mathbf{q}_j(n) \to \mathbf{q}_j^*$, does not imply $\mathbf{x}_j(n) \to \mathbf{x}_j^*$. In practice, $\lim_{n\to\infty} \mathbf{t}_j(n) \neq \mathbf{t}_j^*$ for many practical optimality criteria including max-min optimal rate, optimal weighted sum-rate, and optimal rate with relays as defined in Section II-A – for these problems the sets $\mathcal{X}_j$ are convex polygons and (12) is a linear program. Regularization approaches are known to guarantee convergence of the primal iterates $\mathbf{x}_j(n)$ to the primal optima $\mathbf{x}_j^*$. One of them, the method of multipliers, is pursued in the next section.

Remark 3 To avoid a large variance of the iterates (i.e., large fluctuations around the mean) when $c_n = c\ \forall n$ as in Property 1-(a) requires a small value of c. However, this results in a slow convergence rate. This can be alleviated by adjusting c_n as per Property 1-(b), but this is difficult to implement in a distributed setting. These complementary drawbacks provide another motivation for the approach in Section IV.

IV. The Method of Multipliers

While useful as a first approach, the dual decomposition method summarized in Algorithm 1 does not always lead to a satisfactory solution of (12). As discussed previously, when the dual function is non-differentiable and the step size c_n is fixed the iteration defined by (29) and (30) converges only on an average sense. Perhaps more important, recovering the primal variables optimizing (12) from the dual variables optimizing (20) cannot always be guaranteed.

A common regularization approach is the so called method of multipliers (MoM). The MoM is based on modifying the optimization argument in (12) by adding (hence the term *regularization*) a quadratic term corresponding to the squared norm of the equality constraints; i.e.,

$$\mathbf{T}^* = \arg\min_{\mathbf{X}} \ -\mathbf{w}^T\mathbf{1} + \frac{c}{2} \sum_{j=1}^{J} \left[\|\mathbf{t}'_j - \mathbf{u}_j\|^2 + \|\mathbf{v}_j - w_j\mathbf{1}\|^2\right]$$
$$\text{s.t.} \quad \mathbf{x}_j := (w_j, \mathbf{t}_j, \mathbf{u}_j) \in \mathcal{X}_j;\ \mathbf{t}'_j = \mathbf{u}_j;\ \mathbf{v}_j = w_j\mathbf{1}. \tag{33}$$

Due to the triangle inequality norms are convex functions of their arguments; and consequently the problem in (33) is convex. Furthermore, the solutions of (12) and (33) coincide since the terms $\|\mathbf{t}'_j - \mathbf{u}_j\|^2$ and $\|\mathbf{v}_j - w_j\mathbf{1}\|^2$ are null at any feasible point. The Lagrangian associated with (33) is known

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

as the *augmented* Lagrangian of (12) and with $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q})$ as in (18), it is given by

$$\mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q}) = \mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q}) + \frac{c}{2}\sum_{j=1}^{J}\left[\|\mathbf{t}_j' - \mathbf{u}_j\|^2 + \|\mathbf{v}_j - w_j\mathbf{1}\|^2\right]. \tag{34}$$

Mimicking steps (19) and (20) we can define the dual function $h(\mathbf{P},\mathbf{Q}) := \min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q})$ and deduce that finding the optimal value of (33) – which coincides with the optimal value of (12) – is equivalent to solving the corresponding dual problem

$$-\mathbf{1}^T\mathbf{w}^* = \max_{\mathbf{P},\mathbf{Q}} h(\mathbf{P},\mathbf{Q}) := \max_{\mathbf{P},\mathbf{Q}} \min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q}). \tag{35}$$

Recalling Proposition 2, we can obtain a subgradient of $h(\mathbf{P},\mathbf{Q})$ from the arguments minimizing the augmented Lagrangian. Upon (re-) defining

$$\begin{aligned}\{\mathbf{x}_j^\dagger(\mathbf{P},\mathbf{Q})\}_{j=1}^J &= \arg\min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q}) \\ &= \arg\min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q}) \\ &\quad + \frac{c}{2}\sum_{j=1}^{J}\left[\|\mathbf{t}_j' - \mathbf{u}_j\|^2 + \|\mathbf{v}_j - w_j\mathbf{1})\|^2\right]\end{aligned} \tag{36}$$

we have that the subgradient components $\nabla_{\mathbf{p}_j}(\mathbf{P},\mathbf{Q})$ and $\nabla_{\mathbf{q}_j}(\mathbf{P},\mathbf{Q})$ of $h(\mathbf{P},\mathbf{Q})$ are given as in (23).

Recapitulating the key steps enabling the distributed implementation of the subgradient ascent for $g(\mathbf{P},\mathbf{Q})$, we see that: i) the arguments minimizing the augmented Lagrangian $\mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q})$ lead to a subgradient of $h(\mathbf{P},\mathbf{Q})$ [cf. (23) and (36)]; and ii) the subgradients $\nabla_{\mathbf{p}_j}(\mathbf{P},\mathbf{Q})$ and $\nabla_{\mathbf{q}_j}(\mathbf{P},\mathbf{Q})$ depend only on local and neighboring variables [cf. (23)].

Different from $\mathcal{L}(\mathbf{X},\mathbf{P},\mathbf{Q})$, the minimization of $\mathcal{A}(\mathbf{X},\mathbf{P},\mathbf{Q})$ cannot be separated into local independent optimizations due to the coupling between $\mathbf{t}_j'$ and $\mathbf{u}_j$ and $\mathbf{v}_j$ and w_j introduced by the quadratic terms [cf. (36)]. Note however, that the coupling is between neighboring variables only, and consequently we can again devise a distributed algorithm to solve the minimization in (36). Specifically, our goal is a distributed algorithm that for given multipliers $\mathbf{P}(n)$ and $\mathbf{Q}(n)$ at the n-th iteration converges to the optimal value of the augmented Lagrangian

$$\begin{aligned}\{\mathbf{x}_j(n)\}_{j=1}^J &:= \arg\min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},\mathbf{P}(n),\mathbf{Q}(n)) \\ &:= \arg\min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},n).\end{aligned} \tag{37}$$

A separable iteration converging to $\{\mathbf{x}_j(n)\}_{j=1}^J$ can be obtained using a coordinate descent iteration as described in the following proposition.

Proposition 3 *For fixed n consider iterations over a second index m. With $\mathcal{L}_j(\mathbf{x}_j,n)$ as in* (29)*, define the local augmented Lagrangian at the (n,m)-th iteration as*

$$\mathcal{A}_j(\mathbf{x}_j,n,m) = \mathcal{L}_j(\mathbf{x}_j,n) + \frac{c}{2}\Big[\|\mathbf{t}_j'(n,m) - \mathbf{u}_j\|^2 + 2\|\mathbf{v}_j(n,m) - w_j\mathbf{1})\|\|^2 + \|\mathbf{t}_j - \mathbf{u}_j'(n,m)\|^2\Big] \tag{38}$$

and consider iterates $\{\mathbf{x}_j(n,m+1)\}_{j=1}^J$ satisfying

$$\mathbf{x}_j(n,m+1) = \arg\min_{\mathbf{x}_j\in\mathcal{X}_j} \mathcal{A}_j(n,m). \tag{39}$$

Then, $\mathbf{x}_j(n,m)$ converges to the optimal value of (37)*, i.e., $\lim_{m\to\infty}\mathbf{x}_j(n,m) = \mathbf{x}_j(n)$, $\forall\ j\in[1,J]$.*

Proof: The only terms of $\mathcal{A}(\mathbf{X},n)$ in (37) that depend on local primal variables $\mathbf{x}_j$ are those contained in the local augmented Lagrangian $\mathcal{A}_j(\mathbf{x}_j,n,m)$ in (38). Thus, the optimal arguments in (39) are such that

$$\mathbf{x}_j(n,m+1) = \arg\min_{\mathbf{x}_j\in\mathcal{X}_j} \mathcal{A}\Big[\mathbf{x}_1(n,m),\ldots,\mathbf{x}_{j-1}(n,m),\mathbf{x}_j,\mathbf{x}_{j+1}(n,m),\ldots,\mathbf{x}_J(n,m),n\Big]. \tag{40}$$

That is, (40) minimizes $\mathcal{A}(\mathbf{X},n)$ along the coordinates corresponding to $\mathbf{x}_j$. By definition this is a coordinate descent algorithm for minimizing $\mathcal{A}(\mathbf{X},n)$ and we thus have

$$\lim_{m\to\infty}\mathbf{x}_j(n,m) = \arg\min_{\{\mathbf{x}_j\in\mathcal{X}_j\}_{j=1}^J} \mathcal{A}(\mathbf{X},n) =: \mathbf{x}_j(n) \tag{41}$$

where the first equality follows from convergence results for coordinate descent, see e.g., [2, Sec. 3.2.1]; and the second one from the definition in (37). ■

The coordinate descent iteration (38)-(39) depends only on local and neighboring variables and can thus be implemented in a distributed fashion to obtain the arguments $\{\mathbf{x}_j(n)\}_{j=1}^J$ minimizing the augmented Lagrangian [cf. (37) and (41)]. In turn, these optimal $\{\mathbf{x}_j(n)\}_{j=1}^J$ can be used to implement the subgradient ascent iteration in (30).

The resulting Algorithm 2 embeds an outer iteration (indexed by n) implementing subgradient descent as per (30), and an inner iteration (indexed by m for fixed n) implementing coordinate descent as per (38)-(39) to minimize the augmented Lagrangian. Indeed, steps 4-8 implement (38)-(39) while steps 6 and 7 represent the interchange of primal variables among neighbors. Strictly speaking, step 9 is only true as $M\to\infty$. But even for finite M it provides a reasonable approximation to (37) that can be used to find the subgradients in (23) and implement the gradient ascent iteration in steps 10 and 11. Steps 12 and 2 communicate the dual variables and step 3 initializes the coordinate descent (inner) iteration.

Different from the dual decomposition in Algorithm 1 convergence of the primal iterates $\{\mathbf{x}_j(n)\}_{j=1}^J$ to the optimal primal arguments $\{\mathbf{x}_j^*\}_{j=1}^J$ as $n\to\infty$ can be guaranteed for the MoM in Algorithm 2 as we summarize in the following property; see e.g., [2, Sec. 3.4.4].

Property 2 *Consider implementation of the MoM in Algorithm 2 to solve the optimization problem in* (12) *and let $\{\mathbf{x}_j^*\}_{j=1}^J$ denote the arguments minimizing* (12)*. Then, for any*

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

Algorithm 2 Method of multipliers

Require: Packet success probabilities to and from neighbors $\mathbf{R}_{c(j)j}$ and $\mathbf{R}_{jc(j)}$

Ensure: Routing probabilities $\mathbf{t}_j$

1: **for** $n = 1$ to ∞ **do** {repeat for the life of the network}
2: Receive multipliers $p_{ij}(n)$ and $q_{ji}(n)$ from one hop neighbors $\{U_i : i \in c(j)\}$
3: Initial value for coordinate descent: $\mathbf{x}_j(n,0) = \mathbf{x}_j(n-1)$
4: **for** $m = 1$ to M **do**
5: Coordinate descent iteration for $\mathbf{x}_j(n,m)$: $\mathbf{x}_j(n,m) = \arg\min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{A}_j(n,m)$
6: Transmit $w_j(n,m)$, $t_{ij}(n,m)$, and $u_{ji}(n,m)$ to neighbor U_i. Repeat for all $\{U_i : i \in c(j)\}$.
7: Receive $w_i(n,m)$, $t_{ji}(n,m)$, and $u_{ij}(n,m)$ from one-hop neighbors $\{U_i : i \in c(j)\}$
8: **end for**
9: Argument minimizing augmented Lagrangian: $\mathbf{x}_j(n) = \mathbf{x}_j(n,M)$
10: Subgradient ascent iteration for $\mathbf{p}_j$: $\mathbf{p}_j(n) = \mathbf{p}_j(n-1) + c[\mathbf{t}'_j(n) - \mathbf{u}_j(n)]$
11: Subgradient ascent iteration for $\mathbf{q}_j$: $\mathbf{q}_j(n) = \mathbf{q}_j(n-1) + c[\mathbf{v}_j(n) - w_j(n)\mathbf{1}]$
12: Transmit multipliers $p_{ji}(n+1)$ and $q_{ij}(n+1)$ to neighbor U_i. Repeat for all $\{U_i : i \in c(j)\}$.
13: **end for**

value of M, we have $\lim_{n\to\infty} \mathbf{x}_j(n) = \mathbf{x}_j^*$; in particular it holds that

$$\lim_{n\to\infty} \mathbf{t}_j(n) = \mathbf{t}_j^*. \tag{42}$$

Property 2 guarantees that the optimal routing probabilities can be obtained by running Algorithm 2. It also establishes that the convergence in (42) holds for any number of inner iterations M. A particularly interesting algorithm becomes available for $M = 1$ which leads to the so-called alternating direction MoM. For this algorithm, we define the local augmented Lagrangian at time n as

$$\mathcal{A}_j(\mathbf{x}_j, n) = \mathcal{L}_j(\mathbf{x}_j, n) + \frac{c}{2}\Big[\|\mathbf{t}'_j(n) - \mathbf{u}_j\|^2 + 2\|\mathbf{v}_j(n) - w_j\mathbf{1})\|\|^2 + \|\mathbf{t}_j - \mathbf{u}'_j(n)\|^2\Big] \tag{43}$$

and define the iteration of the primal variables as

$$\mathbf{x}_j(n+1) = \arg\min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{A}_j(\mathbf{x}_j, n) \tag{44}$$

with the iteration of the dual variables (multipliers) given as in (30). Due to Property 2 the iteration (43)-(44) and the corresponding Algorithm 3 converge, as $n \to \infty$, to the optimal routing probabilities $\mathbf{t}_j^*$.

Remark 4 The algorithms of this section assume that packets exchanged for computing $\mathbf{T}$ are received error-free. While it is certainly possible to protect the critical routing information (using e.g., error control coding) so that this is approximately true, it is not fully true with the problem setup in Section I, wherein packets are correctly decoded according to the probabilities in $\mathbf{R}$. An alternative assumption is to suppose that routing variables sent from U_j are correctly received by U_i with probability R_{ij} as would be the case if they were included in packet headers. This falls beyond the scope of the present paper, but it is worth mentioning that there exist asynchronous distributed optimization results that may be used to prove convergence of Algorithms 1-3 even in this case, under certain conditions [2, Ch. 6].

Algorithm 3 Alternating direction method of multipliers

Require: Packet success probabilities to and from neighbors $\mathbf{R}_{c(j)j}$ and $\mathbf{R}_{jc(j)}$

Ensure: Routing probabilities $\mathbf{t}_j$

1: **for** $n = 1$ to ∞ **do** {repeat for the life of the network}
2: Receive multipliers $p_{ji}(n)$ and $q_{ji}(n)$ from one hop neighbors $\{U_i : i \in c(j)\}$
3: Coordinate descent iteration for $\mathbf{x}_j(n)$: $\mathbf{x}_j(n) = \arg\min_{\mathbf{x}_j \in \mathcal{X}_j} \mathcal{A}_j(n)$
4: Transmit $w_j(n)$, $t_{ij}(n)$, and $u_{ji}(n)$ to neighbor U_i. Repeat for all $\{U_i : i \in c(j)\}$.
5: Receive $w_i(n)$, $t_{ji}(n)$, and $u_{ij}(n)$ from one hop neighbors $\{U_i : i \in c(j)\}$
6: Subgradient ascent iteration for $\mathbf{p}_j$: $\mathbf{p}_j(n+1) = \mathbf{p}_j(n) + c[\mathbf{t}'_j(n) - \mathbf{u}_j(n)]$
7: Subgradient ascent iteration for $\mathbf{q}_j$: $\mathbf{q}_j(n+1) = \mathbf{q}_j(n) + c[\mathbf{v}_j(n) - w_j(n)\mathbf{1}]$
8: Transmit multipliers $p_{ij}(n+1)$ and $q_{ij}(n+1)$ to neighbor U_i. Repeat for all $\{U_i : i \in c(j)\}$.
9: **end for**

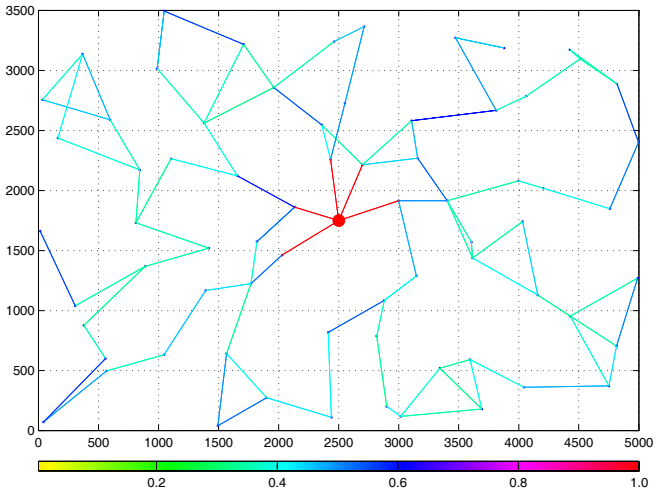

Fig. 2. Max-min optimal routes obtained after solving (12) with the set $\mathcal{X}_j^1$ defined in (13). Note how terminals in compromised spots divide their traffic between more than one neighbor.

Remark 5 In Algorithms 1-3 node U_j solves a convex optimization problem to minimize the (augmented) Lagrangian per iteration [cf. (29), (39), and (44)]. The number of variables in these problems is the sum of the dimensions of w_j, $\mathbf{t}_j$, and $\mathbf{u}_j$ which amounts to $1 + 2|c(j)|$, with $|c(j)|$ denoting the cardinality of $c(j)$. If, as expected, the number of neighbors $|c(j)|$ is small, the minimizations in (29), (39), or (44) incur low computational burden.

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

V. Simulations

For the simulations in this section we consider a network with $J = 70$ terminal users randomly placed on a rectangle of 3.5 km $\times$5 km at whose center is the common access point U_{J+1}; see Fig. 1. To determine the success probability matrix $\mathbf{R}$ we let terminals transmit at random with probability $\mu_j = 0.2$. In every slot, consider the indicator variable $e_j(n) = 1$ if U_j transmitted in the n-th slot and $e_j(n) = 0$ otherwise. Letting p_j denote the transmission power of U_j and $h_{ij}(n)$ the gain in the channel $U_j \to U_i$ at the n-th time slot, we have

$$\gamma_{ij}(n) = \frac{h_{ij}(n)p_j}{\sigma_i + (1/S)\sum_{k=1,k\neq j}^{J} e_k(n)h_{ik}(n)p_k} \tag{45}$$

where $\gamma_{ij}(n)$ denotes the instantaneous SINR in the $U_j \to U_i$ link for the slot n, σ_i the noise power at U_i, and S the spreading gain common to all nodes in the network. The channels $h_{ij}(n)$ are assumed Rayleigh distributed with mean $\bar{h}_{ij}$, known at the receiver end, and independent across terminals and time. The mean channel power obeys an exponential pathloss law $\bar{h}_{ij} = \kappa d(U_i, U_j)^{\alpha}$ where $d(U_i, U_j)$ denotes the distance between U_i and U_j, and κ and α are constants. By convention, $h_{jj}(n) = +\infty$ to ensure that U_j does not transmit and receive simultaneously.

The SINR $\gamma_{ij}(n)$ in (45) determines the instantaneous probability $R_{ij}(n)$ of the packet sent by U_j to be correctly decoded by U_i in the n-th time slot. Terminals further utilize a $(23, 12)$ Golay code whose error probability as a function of SINR can be found in [16, pp. 457]. Each R_{ij} element of the matrix $\mathbf{R}$ is the time average of $R_{ij}(n)$, i.e., $R_{ij} = \lim_{N\to\infty}(1/N)\sum_{n=1}^{N} R_{ij}(n)$. A schematic representation of the resulting $\mathbf{R}$ matrix is shown in Fig. 1. The corresponding max-min optimal routes $\mathbf{T}$, obtained after solving (12) with the set $\mathcal{X}_j^1$ defined in (13) – or the equivalent (4) with utility $f(\boldsymbol{\rho}) = \min_j(\rho_j)$ – are depicted in Fig. 2. The algorithm run by individual nodes in these simulations is the alternating direction MoM outlined in Algorithm 3.

Alternating direction MoM. We start by testing Algorithm 3 in ideal operating conditions. Therefore, we assume that: i) the matrix $\mathbf{R}$ is fixed, i.e., we ignore changes brought about by, e.g., mobility; ii) $\mathbf{R}$ is independent of routing decisions, as is the case in heavily loaded networks; and iii) the vectors $\mathbf{r}_j$ and $\mathbf{s}_j$ are known by U_j, e.g., they were acquired during a training phase. The purpose of this exercise is to assess convergence properties of Algorithm 3 to provide a benchmark for more realistic simulations. In Fig. 3-(top) depicts the evolution of the variable $w_j(n)$ that estimates the rate of the worst user and the instantaneous rate of the worst user $\rho_j(n) := \mathbf{r}_j^T\mathbf{t}_j(n) - \mathbf{s}_j^T(n)\mathbf{t}'_j(n)$. We can see that these two variables approach each other and become more or less equal after approximately $n = 80$ iterations. This is also the approximate number of iterations it takes for Algorithm 3 to reach steady state. Fig. 3-(top) further illustrates fairness of the max-min criterion by showing the normalized sum rate $(1/J)\sum_{j=1}^{J}\rho_j(n)$. We see that the normalized sum rate is about twice the minimum rate. This means that on average terminals get twice the rate of the most compromised user. We also plot in Fig. 3-(bottom) the path followed by the rate of 10 different representative users with similar conclusions.

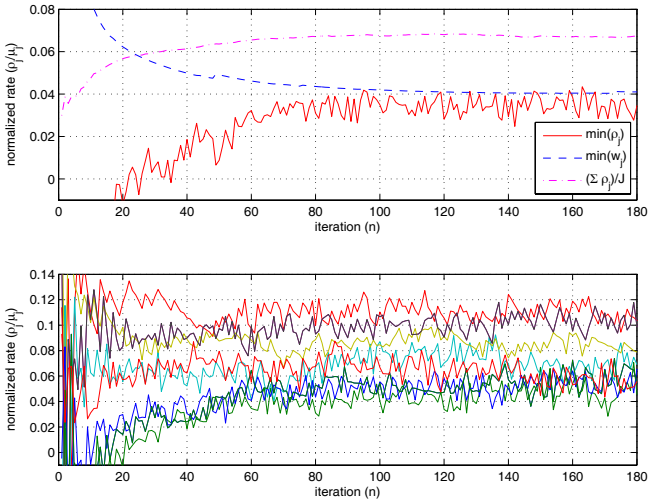

Fig. 3. Convergence of Algorithm 3 to the max-min optimal routes in Fig. 2. In the top we see that it takes about 80 iterations for the minimum rate to converge to the optimal minimum rate. Convergence is typically much faster as illustrated by the random selection of 10 representative terminals (bottom).

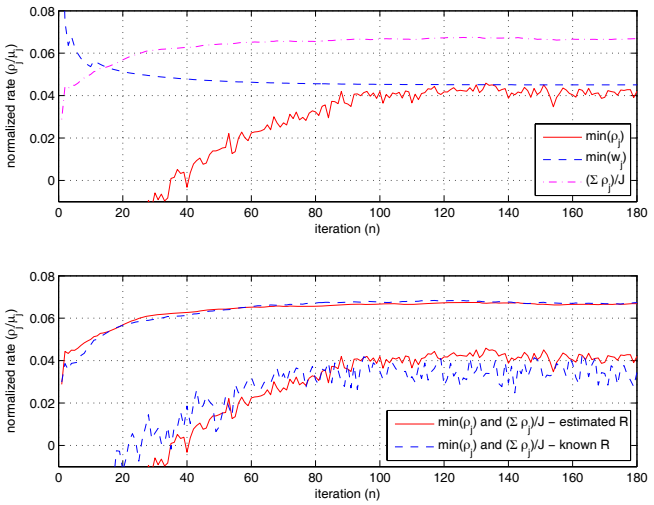

Fig. 4. Terminals keep running estimates $\hat{R}_{ij}(n)$ that they update as $\hat{R}_{ij}(n+1) = \beta + (1-\beta)\hat{R}_{ij}(n)$ whenever a packet is successfully decoded and $\hat{R}_{ij}(n+1) = (1-\beta)\hat{R}_{ij}(n)$ when it is not. Convergence slows down to about 100 iterations (top). Comparing with the case when R_{ij} is known we observe a slight increase in the minimum rate. This is because the latter case underestimates channel reliability by assuming that all queues are backlogged when in fact only some of them are.

Online estimation of reliability. We now lift the assumption that $\mathbf{R}$ is known beforehand and independent of $\mathbf{T}^*$ and consider the effect of estimating $\mathbf{R}$ elements and the interaction between $\mathbf{R}$ and the optimal routing matrix $\mathbf{T}^*$. For these simulations we consider slots separately and determine the instantaneous SINRs $\gamma_{ij}(n)$ in (45) that in turn determine the instantaneous success probability of individual packets. Terminals do not know $\mathbf{r}_j$ and $\mathbf{s}_j$ beforehand but keep running estimates $\hat{R}_{ij}(n)$ of their elements. When packets are transmitted, i.e., when $e_j(n) = 1$, estimates are updated as $\hat{R}_{ij}(n+1) = \beta + (1-\beta)\hat{R}_{ij}(n)$ whenever a packet

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

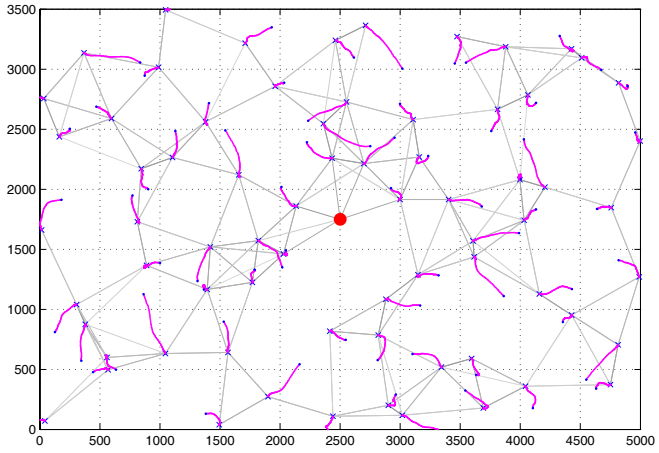

Fig. 5. Adaptability of Algorithm 3 to terminal mobility. Each terminal moves up to 2 meters in a random direction. We show paths followed by terminals in 700 realizations of these random movements.

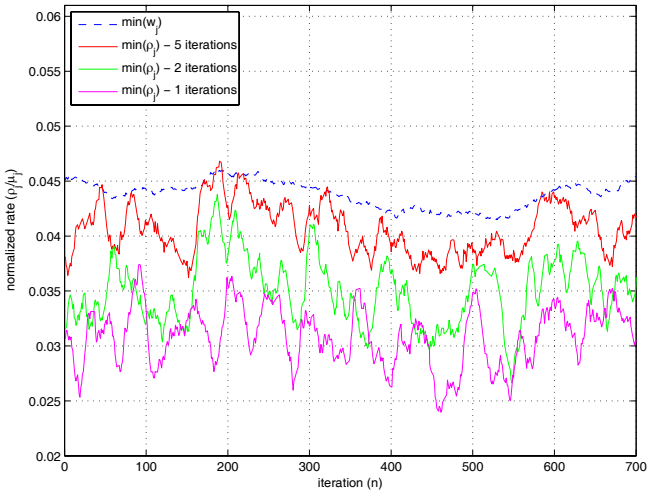

Fig. 6. Response of Algorithm 3 to user mobility. After each movement we let each terminal transmit 1, 2, or 5 packets corresponding to an equal number of iterations of Algorithm 3. The losses with respect to optimal performance are 32%, 17% and 6%, respectively.

is successfully decoded and $\hat{R}_{ij}(n+1) = (1-\beta)\hat{R}_{ij}(n)$ when it is not. The routing variables are updated only when packets are correctly decoded. With reference to Fig. 4-(top) we see that taking these realistic effects into account slows down the convergence rate of Algorithm 3 to about $n = 100$ iterations. Note that since the optimal routes $\mathbf{T}^*$ determine which queues tend to be frequently backlogged and which ones not, $\mathbf{R}$ is in practice affected by $\mathbf{T}^*$. Given that we find $\mathbf{T}^*$ using $\mathbf{R}$, in practice there exists a closed loop that we opened for tractability purposes. The effect of this interaction on the optimal utility is illustrated in Fig. 4-(bottom). We see that the optimal utility when the interaction between $\mathbf{R}$ and $\mathbf{T}^*$ is taken into account is a little larger than when the interaction is not considered. This is because the latter case underestimates channel reliability by assuming that all queues are backlogged when in fact only some of them are.

Mobility. We finally consider adaptability of Algorithm 3 to changes in $\mathbf{R}$ brought about by node mobility. For this purpose, we let each terminal move up to 2 meters in a random direction, and consider 700 repetitions of these random movements. The paths followed by the terminals are plotted in Fig. 5. After each random movement we let terminals take D transmission decisions; i.e, for D times we let each terminal decide whether to transmit a packet or not. If they transmit a packet the reliability estimates are updated as in the previous subsection. Whenever a packet is correctly decoded the optimal routing probabilities are updated as per Algorithm 3. We consider $D = 1$, $D = 2$ and $D = 5$ transmission decisions per movement instance. For benchmark purposes we also let terminals take transmission decisions until Algorithm 3 converges. Fig. 6 depicts the ability of Algorithm 3 to pursue the optimal routes computed by running Algorithm 3 until convergence after each movement. We see that with only one iteration run, i.e., $D = 1$, there is a 32% loss in utility performance with respect to the optimal. This is reduced to about 17% with $D = 2$ iterations and about 6% with $D = 5$ iterations.

VI. Conclusions

Stochastic multihop routing holds great promise to improve performance of wireless networks. Building on recent results that formulate stochastic routing problems as convex optimization programs based on the pairwise error probability matrix $\mathbf{R}$, this paper developed distributed routing algorithms to find rate-optimal routes. Since routing algorithms developed in [17] cannot be implemented in a distributed fashion, we introduced equivalent problems amenable to distributed implementation. Many problems can be cast in the latter formulation including max-min rate, sum-rate, maximum product-rate, and rate-optimal relay networks. In all of these problems, additional convex constraints, e.g., minimum acceptable rate or cooperation limit, can be easily incorporated.

Distributed routing algorithms were obtained via dual decomposition, requiring iterations based on communication with one-hop neighbors only. Since in many cases of interest dual decomposition iterates do not necessarily converge to the optimal routing matrix, we adopted two well-known regularization approaches, namely the method of multipliers (MoM) and the alternating direction MoM. Convergence of these algorithms to the optimal routing matrix is guaranteed under mild conditions. Of particular practical importance is the guaranteed convergence in the presence of communication errors.

Simulations corroborated that the MoM is a robust algorithm quickly converging to the optimal routes. We further demonstrated that the resulting algorithms respond well to changes in the pairwise error probability matrix arising due to, e.g., node mobility. While the analytic development was based on the assumption that $\mathbf{R}$ is independent of $\mathbf{T}^*$, the interplay between $\mathbf{R}$ and $\mathbf{T}^*$ can be captured and even exploited in on-line versions that alternate between estimating $\mathbf{R}$ and computing $\mathbf{T}^*$ in a distributed fashion. This is important because it closes the loop that we opened for tractability considerations.

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.

REFERENCES

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," *ACM SIGCOMM Computer Commun. Rev.*, vol. 34, pp. 121–132, Oct. 2004.

[2] D. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 2nd ed., 1997.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[4] C.-S Chang, "Stability, queue length, and delay of deterministic and stochastic queueing networks," *IEEE Trans. Autom. Control*, vol. 39, pp. 913–931, May 1994.

[5] D. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proc. Int. ACM Conf. Mobile Computing, Networking*, pp. 134–146, San Diego, CA, Sept. 2003.

[6] A. Ephremides, "Energy concerns in wireless networks," *IEEE Wireless Commun.*, vol. 9, pp. 48–59, Aug. 2002.

[7] A. Eryilmaz and R. Srikant, "Joint congestion control, routing, and MAC for stability and fairness in wireless networks," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 1514–1524, Aug. 2006.

[8] A. J. Goldsmith and S. B. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Wireless Commun.*, vol. 9, pp. 8–27, Aug. 2002.

[9] M. Haenggi, "On routing in random Rayleigh fading networks," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 1553–1562, July 2005.

[10] M. Haenggi and D. Puccinelli, "Routing in ad hoc networks: a case for long hops," *IEEE Commun. Mag.*, vol. 43, pp. 93–101, Oct. 2005.

[11] B. Johansson, P. Soldati, and M. Johansson, "Mathematical decomposition techniques for distributed cross-layer optimization of data networks," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 1535–1547, Aug. 2006.

[12] J. Kuruvila, A. Nayak, and I. Stojmenovic, "Hop count optimal position based packet routing algorithms for ad hoc wireless networks with a realistic physical layer," *IEEE J. Select. Areas Commun.*, vol. 23, pp. 1267–1275, June 2005.

[13] S. H. Low, F. Paganini, and J. C. Doyle, "Internet congestion control," *IEEE Control Syst. Mag.*, vol. 22, pp. 28–43, Feb. 2002.

[14] H. Lundgren, E. Nordstrom, and C. Tschudin, "The gray zone problem in IEEE 802.11b based ad hoc networks," *ACM SIGMOBILE Mobile Computing and Commun. Rev.*, vol. 3, pp. 104–105, July 2002.

[15] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 1439–1451, Aug. 2006.

[16] J. G. Proakis, *Digital Communications*. McGraw Hill, 4th ed., 2001.

[17] A. Ribeiro, Z.-Q. Luo, N. D. Sidiropoulos, and G. B. Giannakis, "A general optimization framework for stochastic routing in wireless multihop networks," *IEEE Trans. Signal Processing*, May 2007 (revised).

[18] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad-hoc mobile wireless networks," *IEEE Personal Commun.*, vol. 6, pp. 46–55, Apr. 1999.

[19] G. Scutari, D. P. Palomar, and S. Barbarossa, "Optimal multiplexing strategies for wideband meshed networks based on game theory–part II: algorithms," *IEEE Trans. Signal Processing*, June 2007 (to appear).

[20] I. Stojmenovic, A. Nayak, and J. Kuruvila, "Design guidelines for routing protocols in ad hoc and sensor networks with a realistic physical layer," *IEEE Commun. Mag.*, vol. 43, pp. 101–106, Mar. 2005.

[21] A. Tsirigos and Z. Haas, "Analysis of multipath routing–part I: the effect on the packet delivery ratio," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 138–146, Jan. 2004.

[22] X. Wang and K. Kar, "Cross-layer rate optimization for proportional fairness in multihop wireless networks with random access," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 1548–1559, Aug. 2006.

Alejandro Ribeiro (M'07) received his B.Sc. degree in Electrical Engineering from Universidad de la Republica Oriental del Uruguay, Montevideo, Uruguay in 1998. He is currently a research associate in the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN where he received his M.Sc. and Ph. D. degrees in Electrical Engineering in May 2005 and December 2006. From 1998 to 2003 he was a member of the Technical Staff at Bellsouth Montevideo. His research interests lie in the areas of communication theory, signal processing and networking. His current research focuses on wireless ad hoc and sensor networks, distributed signal processing, and fundamental properties of networks. Mr. Ribeiro is a Fulbright Scholar.

Nikolaos D. Sidiropoulos (SM) received the Diploma in Electrical Engineering from the Aristotelian University of Thessaloniki, Greece, and M.S. and Ph.D. degrees in Electrical Engineering from the University of Maryland at College Park (UMCP), in 1988, 1990 and 1992, respectively. After military service, he joined the Institute for Systems Research at UMCP as Postdoctoral Fellow (1994-1995) and Research Scientist (1996-1997). In 1997, he joined the Department of Electrical Engineering at the University of Virginia as Assistant Professor, and in 2000 the Department of Electrical and Computer Engineering at the University of Minnesota - Minneapolis as Associate Professor. Since 2002, he is a Professor in the Department of Electronic and Computer Engineering at the Technical University of Crete, Chania - Crete, Greece, and Adjunct Professor at the University of Minnesota. His research interests are primarily in signal processing for communications and networking, and multi-way factor analysis. He is currently chair (2007 - 2008) of the Signal Processing for Communications and Networking Technical Committee (SPCOM-TC) of the IEEE Signal Processing Society (SPS), where he has served as member (2000 - 2005) and vice-chair (2005 - 2006). He is also a member of the Sensor Array and Multichannel processing Technical Committee (SAM-TC) of the IEEE SPS (2004 - 2009). He has served as Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING (2000 - 2006), and IEEE SIGNAL PROCESSING LETTERS (2000 - 2002). Dr. Sidiropoulos received the NSF/CAREER award in June 1998, and the IEEE Signal Processing Society Best Paper Award in 2001 and 2007. He is a Distinguished Lecturer of the IEEE SPS for 2008-2009, and an active consultant for industry in the areas of frequency hopping systems and signal processing for xDSL modems.

Georgios B. Giannakis (F'97) received his Diploma in Electrical Engr. from the Ntl. Tech. Univ. of Athens, Greece, 1981. From 1982 to 1986 he was with the Univ. of Southern California (USC), where he received his MSc. in Electrical Engineering, 1983, MSc. in Mathematics, 1986, and Ph.D. in Electrical Engr., 1986. Since 1999 he has been a professor with the ECE Department at the Univ. of Minnesota, where he now holds an ADC Chair in Wireless Telecommunications.

His general interests span the areas of communications, networking and statistical signal processing - subjects on which he has published more than 275 journal papers, 450 conference papers, two edited books and two research monographs. Current research focuses on complex-field and network coding, multicarrier, cooperative wireless communications, cognitive radios, cross-layer designs, mobile ad hoc networks and wireless sensor networks.

G. B. Giannakis is the (co-) recipient of six paper awards from the IEEE Signal Processing (SP) and Communications Societies including the G. Marconi Prize Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award and the G. W. Taylor Award for Distinguished Research from the University of Minnesota. He has served the IEEE in a number of posts, and is currently a Distinguished Lecturer for the IEEE-SP Society.

Authorized licensed use limited to: University of Minnesota. Downloaded on December 11, 2008 at 15:42 from IEEE Xplore. Restrictions apply.