

# A Class of Convergent Algorithms for Resource Allocation in Wireless Fading Networks

Nikolaos Gatsis, Alejandro Ribeiro, and Georgios B. Giannakis, *Fellow, IEEE*

**Abstract**—Optimal and reduced-complexity near-optimal algorithms are developed for the design of wireless networks in the presence of fading. The physical layer is interference-limited, whereby network terminals treat interference as noise. Optimal wireless network design amounts to joint optimization of application-level rates, routes, link capacities, power consumption, and power allocation across frequency tones, neighboring terminals, and fading states. The present contribution shows how recent results establishing the optimality of layered architectures can be realized in practice by developing physical layer resource allocation algorithms that are seamlessly integrated into layered architectures without loss of optimality. Specifically, the provably convergent algorithms yield (near-)optimal end-to-end rates, multicommodity flows, link capacities, and average powers. These design variables are obtained offline, and are subsequently used for control during network operation.

**Index Terms**—Cross-layer design, interference, multi-hop, optimization methods, resource management.

## I. INTRODUCTION

IN recent years, optimization has been fruitfully brought into the design of wireless networks; see e.g., [1]–[3]. Optimal wireless network design emanates from the formulation of an optimization problem involving variables such as end-to-end rates, link rates, and communication resources, that is, power and bandwidth. The best operating point is defined as the solution of such an optimization problem, and protocols follow from algorithms used for its solution; see e.g., [1] and [2].

This paper is concerned with optimal wireless network design in the presence of fading when the physical layer is interference-limited. All network terminals are allowed to transmit over the available frequency bands and non-intended transmissions are treated as noise. The goal is to find the optimal operating point that maximizes a network-wide performance metric. Finding such an optimal operating point requires determination of (optimal) end-to-end rates, routes,

link capacities, average power consumption, as well as power allocation across frequency tones, neighboring terminals, and fading states. The large number of variables involved hints at the difficulty of finding such an operating point. This difficulty is confirmed by the fact that the optimization problem associated with this definition of optimal wireless network is *not* convex.

This complication is typically addressed through the introduction of suitable simplifications. Joint optimization of transport-level rates and physical layer power allocation is considered using a high-signal-to-interference-plus-noise-ratio (SINR) approximation with deterministic channels [4, Sec. 3.4]. A high-SINR assumption is also used to lend tractability to a more general joint congestion control, routing, link rate, and power control problem [5]. The more accurate  $\log(1 + \text{SINR})$  objective is also treated in [5] using a trick whereby the total instantaneous transmission power at each node is kept constant, but a way to optimally pick that constant is needed. A heuristic algorithm for the physical layer power allocation, where terminals transmit either at full power or not at all, is pursued in [6]. The strategy of on-off power control—where terminals transmit at their spectral masks or not at all—is shown to be optimal under a low-SINR assumption in [7], but efficient algorithms to find the optimal power allocations are still lacking.

Alternatively, simplifications and insights can be obtained from the Lagrangian dual problem. Interestingly, it is possible to show that layers emerge naturally from the decomposition of Lagrangians associated with optimal networking formulations [3]. However, these layered architectures might in principle be suboptimal for wireless networks, as the lack of convexity entails a positive duality gap. A recent development has shown that, regardless of assumptions on the physical layer, the duality gaps of non-convex wireless network optimization problems are null in the presence of fading [8]. The optimality of separating wireless network design into layers follows easily from the latter result [9]. A consequence of the optimality of layered architectures is that algorithms for optimal resource allocation at the physical layer can be leveraged with relative simplicity to design optimal networks. And in that sense, it upholds the importance of solving physical layer resource allocation problems—a task not addressed in [8] or [9].

The present paper develops algorithms to solve physical layer resource allocation problems, and discusses their optimal integration within layered architectures. The proposed algorithms implement subgradient descent on the dual function (Section III-A). Such descent algorithms are guaranteed to find optimal dual variables, but do not necessarily converge

Manuscript received March 2, 2009; revised December 2, 2009 and March 10, 2010; accepted March 17, 2010. The associate editor coordinating the review of this paper and approving it for publication was E. Hossain.

Work in this paper was supported by NSF grants CCF-0830480 and ECCS-0824007, and also 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. Part of this paper was presented at the IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Taipei, Taiwan, Apr. 2009.

N. Gatsis and G. B. Giannakis are with the Dept. of Electrical and Computer Engineering, Univ. of Minnesota (e-mail: {gatsisn, georgios}@umn.edu).

A. Ribeiro is with the Dept. of Electrical and Systems Engineering, Univ. of Pennsylvania (e-mail: aribeiro@seas.upenn.edu).

Digital Object Identifier 10.1109/TWC.2010.05.10360

to the optimal primal variables. This often neglected subtlety is particularly important in the present context because of the non-convexity of the associated optimization problem. The first contribution of this paper is to show that, despite this lack of convexity, it is possible to obtain arbitrarily close to optimal primal variables (Section III-B). These near-optimal primal variables are obtained by forming weighted running averages of the primal sequence obtained as a byproduct of the subgradient descent iteration. This offers a valuable generalization of ergodic convergence results for convex optimization problems; see e.g., [10], [11], and references therein. The proposed optimization methodology is then used to obtain a simple algorithm for network control (Section III-C). The resultant algorithm comprises two phases: (a) an offline phase, where near-optimal primal variables and Lagrange multipliers are obtained; and (b) an online phase, where power is allocated based on the current channel realization, and routing is performed.

Among other simpler subproblems, each step of the subgradient algorithm necessitates the solution of a sum-rate power allocation problem. This problem has exponential complexity [12] motivating the use of approximate solutions, which constitute the second contribution of the present work (Section IV). Suitable approximations render the high-SINR scenario tractable, but the low-SINR scenario remains intractable. Successive convex approximations are also developed, using the condensation method, see e.g., [4, Sec. 2.2], and more sophisticated approaches adapted from the digital subscriber lines (DSL) literature [13], [14]. Note that while in e.g., [4], [5], [7], approximations are used to modify the primal problem, here they are used to approximately solve subgradient descent iteration steps. The performance of these different approximations is confirmed using numerical results (Section V). Moreover, a simulation of the network control strategy is provided in Section V, and conclusions are drawn in Section VI.

## II. MODELING AND PROBLEM STATEMENT

This section formulates the cross-layer resource allocation problem for wireless fading networks. Its solution and the resulting network control strategy will be the themes of Sections III and IV.

Consider a wireless network as in [2] and [8], comprising a set of terminals (nodes) denoted by  $\mathcal{V}$ . Two nodes  $i, j$  form a link when they can communicate with each other; here, the communication between nodes is bidirectional. The set of nodes that a node  $i \in \mathcal{V}$  can communicate with, forms the neighborhood of  $i$  that is denoted by  $\mathcal{N}(i)$ . The network operates in a time-slotted fashion.

Packets generated exogenously at each node correspond to possibly different applications (such as video, voice, file transfer), and are destined for different nodes. Such packet streams are called commodities, and are generically indexed by  $k$ . The destination node associated with each commodity  $k$ , is denoted by  $\text{dest}(k) \in \mathcal{V}$ . Each node serves commodities that have other nodes as destination. Clearly, commodities that have as destination node  $i$  are never actually generated at node  $i$ . Hence, the set of commodities that can be generated

exogenously at node  $i$  is  $\mathcal{K}_i := \{k | i \neq \text{dest}(k)\}$ . The transport layer protocol of the network is responsible for maintaining the end-to-end flow of packets from each node  $i$  to the destination of every commodity  $k$  that is generated exogenously at node  $i$ . The average rate of such a flow is denoted by  $a_i^k$ .

Moreover, there are flows endogenous to the network layer. Specifically, each node  $i$  will transmit to its neighbors  $j \in \mathcal{N}(i)$  packets of commodity  $k \in \mathcal{K}_i$  at an average rate denoted by  $r_{ij}^k$ , called network-layer rate or multicommodity flow. Rates  $r_{ij}^k$  are essentially routing variables, because they dictate how packets of various commodities are forwarded to the outgoing links of node  $i$ . As such, they pertain to the network layer. Hence, at node  $i$  there are exogenous packet arrivals with rate  $a_i^k$  from the transport layer, and endogenous packet arrivals with rates  $r_{ji}^k$  from neighbors  $j \in \mathcal{N}(i)$ . Packets also leave node  $i$  with rates  $r_{ij}^k$  towards neighbors  $j \in \mathcal{N}(i)$ . Queue stability considerations dictate these rates to satisfy a *flow conservation* constraint

$$a_i^k \leq \sum_{j \in \mathcal{N}(i)} r_{ij}^k - \sum_{j \in \mathcal{N}(i), j \neq \text{dest}(k)} r_{ji}^k \quad \forall i, k \in \mathcal{K}_i. \quad (1)$$

In addition, link  $(i, j)$  ‘‘carries’’ total average rate  $\sum_{k \in \mathcal{K}_i} r_{ij}^k$  that cannot exceed the average (i.e., ergodic) capacity of the link, denoted by  $c_{ij}$ . This gives rise to the *link capacity* constraint

$$\sum_{k \in \mathcal{K}_i} r_{ij}^k \leq c_{ij} \quad \forall i, j \in \mathcal{N}(i). \quad (2)$$

Link capacities  $c_{ij}$  in wireless networks are not fixed, but vary with time and depend on the specific physical and medium access control (MAC) layers, as well as on the allocation of network resources. In the considered network model, terminals have a set of tones  $\mathcal{F} := \{1, \dots, F\}$  available for transmission. It is further assumed that link  $(i, j)$  over tone  $f$  has power gain coefficient  $h_{ij}^f$ , which captures fading effects. The focus of the present work is on *non-orthogonal* medium access, where different terminals are allowed to use the same frequency to transmit, treating other terminals’ transmissions as noise. Network resources here are power allocations  $p_{ij}^f(\mathbf{h})$  for link  $(i, j)$  over tone  $f$  as a function of the power gains  $h_{ij}^f$  collected in the vector  $\mathbf{h}$ . The channel vector  $\mathbf{h}$  may change from slot to slot or even at a faster scale, depending on the coherence time. The fading process is stationary and ergodic, but otherwise has arbitrary correlation, as is typically the case in practice. Let  $\mathbb{E}_{\mathbf{h}}$  denote expectation over the stationary distribution of  $\mathbf{h}$ .

Link capacities are dictated by the SINR; the instantaneous SINR of link  $(i, j)$  over tone  $f$  is

$$\gamma_{ij}^f := \frac{h_{ij}^f p_{ij}^f(\mathbf{h})}{\sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kl}^f p_{kl}^f(\mathbf{h})} \quad (3)$$

where  $\sigma_j^f$  is the noise variance at node  $j$  over tone  $f$ , and  $\mathcal{I}_{ij}$  denotes the set of links causing interference to  $(i, j)$ . This set consists of the links carrying: (i) incoming transmissions for  $j$  over  $f$  from nodes other than  $i$ ; (ii) outgoing transmissions from  $j$ ; and (iii) transmissions originating from nodes in  $\mathcal{N}(j)$  intended for nodes other than  $j$ . Specifically, this set

takes the form

$$\mathcal{I}_{ij} := \{(k, l) : k \in \mathcal{N}(j) \setminus \{i\}, l \in \mathcal{N}(k); \\ (i, l) : l \in \mathcal{N}(i) \setminus \{j\}; (j, l) : l \in \mathcal{N}(j)\}. \quad (4)$$

Based on (4), the term  $h_{jj}^f \sum_{l \in \mathcal{N}(j)} p_{jl}^f$  in the denominator in (3) represents self-interference to receiving node  $j$  from transmissions originating from  $j$ . In order to discourage this self-interference, and thus ensure half-duplex operation,  $h_{jj}^f$  is set to a high value. Furthermore, interference from “far-away” links, corresponding to  $(k, l)$  with  $k \in \mathcal{N}(j)$ ,  $k \neq j$  and  $l \in \mathcal{N}(k)$ ,  $l \neq j$ , is neglected. Note that  $\gamma_{ij}^f$  depends on power allocations  $p_{ij}^f(\mathbf{h})$  as well as on  $\mathbf{h}$ . A related FDMA network model was considered in [15], but interference caused by these  $\mathcal{I}_{ij}$  links to  $(i, j)$  was not accounted for.

Supposing capacity-achieving codebooks, the ergodic capacity  $c_{ij}$  of link  $(i, j)$  is

$$c_{ij} = \mathbb{E}_{\mathbf{h}} \left[ \sum_{f \in \mathcal{F}} \ln(1 + \gamma_{ij}^f) \right]. \quad (5)$$

Two kinds of power constraints are considered. The first ones are instantaneous spectral mask constraints, expressed as  $0 \leq p_{ij}^f(\mathbf{h}) \leq p_{ij \max}^f$ . The second kind pertains to the average power consumed by a node in the network. Specifically, the average power per node  $i$  is defined as

$$p_i := \mathbb{E}_{\mathbf{h}} \left[ \sum_{j \in \mathcal{N}(i)} \sum_{f \in \mathcal{F}} p_{ij}^f(\mathbf{h}) \right] \quad (6)$$

and is assumed constrained by a power budget  $p_{i \max}$ , i.e.,  $0 \leq p_i \leq p_{i \max}$  for all  $i$ .

Average exogenous rates  $a_i^k$  in practical networks lie within application-specific bounds expressed as  $a_{i \min}^k \leq a_i^k \leq a_{i \max}^k$ . Furthermore, the network designer may wish to impose upper bounds on link capacities  $c_{ij}$  and multicommodity flows  $r_{ij}^k$ , that is,  $0 \leq c_{ij} \leq c_{ij \max}$  and  $0 \leq r_{ij}^k \leq r_{i \max}^k$ . The notation  $\mathbf{x}$  will be used to denote the collection of all average variables, i.e.,  $a_i^k, r_{ij}^k, c_{ij}, p_i$  for all  $i \in \mathcal{V}, j \in \mathcal{N}(i), k \in \mathcal{K}_i$ . Then, all previously mentioned average and instantaneous “box” constraints can be conveniently written using the set notation

$$\mathcal{B} := \{\mathbf{x}, \mathbf{p}(\mathbf{h}) : 0 \leq p_{ij}^f(\mathbf{h}) \leq p_{ij \max}^f, 0 \leq p_i \leq p_{i \max}, \\ a_{i \min}^k \leq a_i^k \leq a_{i \max}^k, 0 \leq c_{ij} \leq c_{ij \max}, 0 \leq r_{ij}^k \leq r_{i \max}^k\}. \quad (7)$$

Higher exogenous arrival rates  $a_i^k$  are in general more desirable, a fact captured by utility functions  $U_i^k(a_i^k)$  that are selected strictly increasing and concave. Moreover, average power consumption will be penalized by cost functions  $V_i(p_i)$ , selected strictly increasing and convex. The optimal operating point of the wireless network refers to average variables  $a_i^k, r_{ij}^k, c_{ij}, p_i$  and instantaneous power allocations  $p_{ij}^f(\mathbf{h})$ , such that constraints (1), (2), (5), (6), and (7) are respected, while the total network utility is maximized and the total cost

minimized; that is,

$$P = \max_{\{\mathbf{x}, \mathbf{p}(\mathbf{h})\} \in \mathcal{B}} \sum_{i, k \in \mathcal{K}_i} U_i^k(a_i^k) - \sum_i V_i(p_i) \quad (8a)$$

$$\text{subj. to } a_i^k \leq \sum_{j \in \mathcal{N}(i)} r_{ij}^k - \sum_{\substack{j \in \mathcal{N}(i) \\ j \neq \text{dest}(k)}} r_{ji}^k \quad \forall i, k \in \mathcal{K}_i \quad (8b)$$

$$\sum_{k \in \mathcal{K}_i} r_{ij}^k \leq c_{ij} \quad \forall i, j \in \mathcal{N}(i) \quad (8c)$$

$$c_{ij} \leq \mathbb{E}_{\mathbf{h}} \left[ \sum_f \ln(1 + \gamma_{ij}^f) \right] \quad \forall i, j \in \mathcal{N}(i) \quad (8d)$$

$$\mathbb{E}_{\mathbf{h}} \left[ \sum_{j \in \mathcal{N}(i)} \sum_{f \in \mathcal{F}} p_{ij}^f(\mathbf{h}) \right] \leq p_i \quad \forall i. \quad (8e)$$

where inequalities (8d) and (8e) replaced the equalities (5) and (6). Note that variables  $a_i^k, r_{ij}^k$ , and  $c_{ij}$  have units nats/(sec · Hz) [cf. (5)], while variables  $p_i$  and  $p_{ij}^f(\mathbf{h})$  have units W/Hz.

It is worth stressing at this point that in practice, the generated traffic traverses the network in bits or packets, which are stored at every node in queues. In particular, each node keeps a queue at the network layer for every commodity that can enter the node, and a queue at the physical layer for every outgoing link. For instance, the average arrival rate to the queue of node  $i$  for commodity  $k \in \mathcal{K}_i$  is  $a_i^k + \sum_{j \in \mathcal{N}(i), j \neq \text{dest}(k)} r_{ji}^k$ , while the average departure rate is  $\sum_{j \in \mathcal{N}(i)} r_{ij}^k$ . The constraints in (8) guarantee that all the queues in the network are stable [2, Ch. 3].

Algorithmic solutions to (8) are developed in the ensuing Sections III-A and III-B. A more detailed description of the queueing operations and the network functionality are given in Section III-C, where the solution of (8) is utilized in order to obtain a simple strategy for network control.

### III. SUBGRADIENT METHOD

In this section, a solution of the constrained optimization task in (8) is sought via its Lagrange dual. First, the dual of problem (8) is specified. Then a subgradient algorithm for its solution is developed in Section III-A. As the subgradient algorithm returns Lagrange multipliers, an issue of interest is how to recover network variables  $a_i^k, r_{ij}^k, c_{ij}, p_i, p_{ij}^f(\mathbf{h})$  from Lagrange multipliers; this is addressed in Section III-B. The convergent algorithms of Sections III-A and III-B yield the (near-)optimal operating point of wireless fading networks. Section III-C deals with implementation of these algorithms, leading to a simple strategy for network control.

Let  $\nu_i^k, \xi_{ij}, \lambda_{ij}, \mu_i$  be Lagrange multipliers for constraints (8b), (8c), (8d), and (8e), respectively. The box constraints (7) are kept implicit. Also let  $\mathbf{\Lambda}$  collectively denote all Lagrange multipliers. The Lagrangian function of (8) reduces after straightforward re-arrangements to

$$L(\mathbf{\Lambda}, \mathbf{x}, \mathbf{p}(\mathbf{h})) = \sum_{i, k \in \mathcal{K}_i} (U_i^k(a_i^k) - \nu_i^k a_i^k) + \sum_i (\mu_i p_i - V_i(p_i)) \\ + \sum_{\substack{i, j \in \mathcal{N}(i) \\ f \in \mathcal{F}}} \mathbb{E}_{\mathbf{h}} [\lambda_{ij} \ln(1 + \gamma_{ij}^f) - \mu_i p_{ij}^f(\mathbf{h})] + \sum_{i, j \in \mathcal{N}(i)} (\xi_{ij} - \lambda_{ij}) c_{ij}$$

$$+ \sum_{\substack{i,k \in \mathcal{K}_i, j \in \mathcal{N}(i) \\ j \neq \text{dest}(k)}} (\nu_i^k - \nu_j^k - \xi_{ij}) r_{ij}^k + \sum_{\substack{i,k \in \mathcal{K}_i, j \in \mathcal{N}(i) \\ j = \text{dest}(k)}} (\nu_i^k - \xi_{ij}) r_{ij}^k. \quad (9)$$

It should be noted that the last sum in (9) may not be present. The dual function and the dual problem are, respectively,

$$q(\mathbf{\Lambda}) := \max_{(\mathbf{x}, \mathbf{p}(\mathbf{h})) \in \mathcal{B}} L(\mathbf{\Lambda}, \mathbf{x}, \mathbf{p}(\mathbf{h})) \quad (10)$$

$$D = \min_{\mathbf{\Lambda} \geq \mathbf{0}} q(\mathbf{\Lambda}). \quad (11)$$

Due to constraint (8d), problem (8) is non-convex. Remarkably though, [8] has established that problem (8) has zero duality gap for wireless networks with continuous fading channels  $\mathbf{h}$ , and hence, for all practical fading models. The implication is that the dual problem (11) can be solved without loss of optimality. This is desirable because the dual problem is a convex optimization problem [16, Sec. 5.2], and also its solution can effect separation of the primal problem into conventional layers (see [8], [9], and the discussion in Section III-A).

On the other hand, a task needed in the solution of the dual problem is maximization of the Lagrangian [cf. (10)]. Such maximization is computationally intractable (in terms of seeking an exact maximizer), due to the term  $\ln(1 + \gamma_{ij}^f)$  in (9). One of the present paper's contributions is to tackle the solution of (11) (and hence of (8)) for a network whose physical layer is characterized by link capacities as in (5), where terminals use the simple strategy of treating non-intended transmissions as noise. This fills the need for an algorithmic solution to problems (11) and (8), which was not provided in [8]. In what follows, the solution of (11) and the recovery of optimal primal variables from it are developed (Sections III-A and III-B). Suitable approximation algorithms aiming at evaluating (10) are detailed in Section IV.

In order to facilitate the development, (8) is expressed in a more compact and generic form, which nevertheless captures all its essential features, as will be explained shortly. Upon rewriting all constraints in (8) with 0 on the right-hand side of the inequalities, (8) can be expressed as

$$P = \max_{(\mathbf{x}, \mathbf{p}(\mathbf{h})) \in \mathcal{B}} f(\mathbf{x}) \quad (12a)$$

$$\text{subj. to } \mathbf{g}_1(\mathbf{x}) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}), \mathbf{h})] \leq \mathbf{0} \quad (12b)$$

where the association of functions  $f(\mathbf{x})$ ,  $\mathbf{g}_1(\mathbf{x})$  and  $\mathbf{g}_2(\mathbf{p}(\mathbf{h}), \mathbf{h})$  in (12) with the objective and constraints in (8) is evident. Regarding problem (12), the following assumptions are made.

**AS1.** Function  $f$  is concave,  $\mathbf{g}_1$  is convex, and  $\mathbf{g}_2$  is continuous.

**AS2.** Constraint set  $\mathcal{B}$  is convex, closed, bounded, and can be written in a decoupled form as  $\mathcal{B} = \mathcal{B}_{\mathbf{x}} \times \mathcal{B}_{\mathbf{p}}$ , where  $\mathcal{B}_{\mathbf{p}}$  is independent of  $\mathbf{h}$ .

**AS3.** Every entry of  $\mathbf{h}$  is a continuous random variable.

**AS4.** Problem (12) is strictly feasible (Slater constraint qualification).

The previous assumptions are satisfied by the setup of problem (8) and were utilized in [8] to show that the duality gap is zero, i.e.,  $P = D$ . Note that problem (12) is not

convex in general. Keeping the box constraints implicit, the Lagrangian function is

$$L(\mathbf{\Lambda}, \mathbf{x}, \mathbf{p}(\mathbf{h})) = f(\mathbf{x}) - \mathbf{\Lambda}^T (\mathbf{g}_1(\mathbf{x}) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}), \mathbf{h})]). \quad (13)$$

When (12) takes the particular form (8), the Lagrangian function (13) clearly reduces to (9).

#### A. Iterative algorithm

The dual problem (11) is solved via subgradient iterations [17, Sec. 8.2], [9]. Let  $t$  be the iteration index. The sequence  $\mathbf{\Lambda}(t)$  obtained from the subgradient method, with initial  $\mathbf{\Lambda}(0) \geq \mathbf{0}$ , is

$$(\mathbf{x}(t), \mathbf{p}(\mathbf{h}; t)) \in \arg \max_{(\mathbf{x}, \mathbf{p}(\mathbf{h})) \in \mathcal{B}} L(\mathbf{\Lambda}(t), \mathbf{x}, \mathbf{p}(\mathbf{h})) \quad (14a)$$

$$\mathbf{\Lambda}(t+1) = \left[ \mathbf{\Lambda}(t) + \epsilon_t (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]) \right]^+ \quad (14b)$$

where the inclusion symbol ( $\in$ ) in (14a) covers the possibility of multiple maximizers;  $\epsilon_t$  is the stepsize (varying with  $t$  in general); and  $[\cdot]^+$  denotes componentwise projection onto the nonnegative reals. Note that the norm of the subgradient,  $\|\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]\|$ , is bounded, because function  $\mathbf{g}_1$  is convex, hence continuous; function  $\mathbf{g}_2$  is continuous; and the set  $\mathcal{B}$  is closed and bounded (cf. AS1 and AS2). Thus, there exists a constant  $G$  such that  $\|\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]\| \leq G$  for all  $t \geq 0$  ( $\|\cdot\|$  denotes Euclidean norm).

Now apply the  $\max_{(\mathbf{x}, \mathbf{p}(\mathbf{h})) \in \mathcal{B}}$  operator to (9), and note that (9) is a sum of terms where each term depends on different variables, and (7) are box constraints. Then eq. (14a) becomes

$$a_i^k(t) \in \arg \max_{a_i^k \min \leq a_i^k \leq a_i^k \max} [U_i^k(a_i^k) - \nu_i^k(t) a_i^k] \quad (15a)$$

$$r_{ij}^k(t) \in \arg \max_{0 \leq r_{ij}^k \leq r_{ij}^k \max} \begin{cases} (\nu_i^k(t) - \nu_j^k(t) - \xi_{ij}(t)) r_{ij}^k & \text{if } j \neq \text{dest}(k) \\ (\nu_i^k(t) - \xi_{ij}(t)) r_{ij}^k & \text{if } j = \text{dest}(k) \end{cases} \quad (15b)$$

$$c_{ij}(t) \in \arg \max_{0 \leq c_{ij} \leq c_{ij} \max} [(\xi_{ij}(t) - \lambda_{ij}(t)) c_{ij}] \quad (15c)$$

$$p_i(t) \in \arg \max_{0 \leq p_i \leq p_i \max} [\mu_i(t) p_i - V_i(p_i)] \quad (15d)$$

$$\mathbf{p}(\mathbf{h}; t) \in \arg \max_{\substack{0 \leq p_{ij}^f \leq p_{ij}^f \max \\ \forall i, f, j \in \mathcal{N}(i)}} \sum_{i, f} [\lambda_{ij}(t) \ln(1 + \gamma_{ij}^f) - \mu_i(t) p_{ij}^f]. \quad (15e)$$

Eq. (15e) is obtained by noting that the part of (9) which involves the  $\mathbb{E}_{\mathbf{h}}[\cdot]$  operator can be maximized if the term inside the expectation is maximized for each fading state  $\mathbf{h}$ . The optimization variables  $p_{ij}^f$  in the right hand side of (15e) are not functions of  $\mathbf{h}$ ; but the maximizer in the left hand side depends on  $\mathbf{h}$ , because the SINR depends on  $\mathbf{h}$ .

The decomposition of the Lagrangian maximization [cf. (14a)] into (15a)–(15e) can be understood as separation of the solution of the wireless networking problem into conventional layers [8]. In particular, (15a) solves the flow control problem at the transport layer; (15b) performs routing at the network layer; (15c) and (15d) address the link rate control and

(average) power control at the data link layer; and (15e) solves the power allocation at the physical layer.

In order to expand (14b) in an explicit form, note that  $\Lambda$  is a vector with entries the Lagrange multipliers  $\nu_i^k$ ,  $\xi_{ij}$ ,  $\lambda_{ij}$ ,  $\mu_i$  for all  $i, j \in \mathcal{N}(i)$ ,  $k \in \mathcal{K}_i$ ; and the functions  $\mathbf{g}_1$  and  $\mathbf{g}_2$  in (12b) correspond to (8b)–(8e), written also in vector form. Then the subgradient updates (14b) become

$$\nu_i^k(t+1) = \left[ \nu_i^k(t) + \epsilon_t \left( a_i^k(t) - \sum_{j \in \mathcal{N}(i)} r_{ij}^k(t) + \sum_{\substack{j \in \mathcal{N}(i) \\ j \neq \text{dest}(k)}} r_{ji}^k(t) \right) \right]^+ \quad (16a)$$

$$\xi_{ij}(t+1) = \left[ \xi_{ij}(t) + \epsilon_t \left( \sum_{k \in \mathcal{K}_i} r_{ij}^k(t) - c_{ij}(t) \right) \right]^+ \quad (16b)$$

$$\lambda_{ij}(t+1) = \left[ \lambda_{ij}(t) + \epsilon_t \left( c_{ij}(t) - \mathbb{E}_{\mathbf{h}} \left[ \sum_f \ln(1 + \gamma_{ij}^f(t)) \right] \right) \right]^+ \quad (16c)$$

$$\mu_i(t+1) = \left[ \mu_i(t) + \epsilon_t \left( \mathbb{E}_{\mathbf{h}} \left[ \sum_{j \in \mathcal{N}(i), f} p_{ij}^f(\mathbf{h}; t) \right] - p_i(t) \right) \right]^+ \quad (16d)$$

Two stepsize rules are of interest: (a) constant stepsize  $\epsilon_t = \epsilon > 0$ ; and (b) nonsummable but square-summable stepsize:  $\epsilon_t > 0$ ,  $\sum_{t=0}^{\infty} \epsilon_t = \infty$ , and  $\sum_{t=0}^{\infty} \epsilon_t^2 < \infty$  (which implies  $\lim_{t \rightarrow \infty} \epsilon_t = 0$ ). If a constant stepsize is used, then iterations (16) converge in the following sense. Letting  $q_{\text{best}}(t) := \min_{0 \leq s \leq t} q(\Lambda(s))$  denote the best dual value up to iteration  $t$ , it holds that  $\lim_{t \rightarrow \infty} q_{\text{best}}(t) \leq D + \epsilon G^2/2$  [17, Prop. 8.2.3], [9, Th. 6]. Moreover, the running average of the dual iterates,  $\bar{\Lambda}(s) := \frac{1}{s} \sum_{t=0}^{s-1} \Lambda(t)$ , converges to an  $\epsilon G^2/2$ -suboptimal point, i.e.,  $\limsup_{t \rightarrow \infty} q(\bar{\Lambda}(t)) \leq D + \epsilon G^2/2$ . On the other hand, if a nonsummable but square-summable stepsize sequence is used, i.e.,  $\sum_{t=0}^{\infty} \epsilon_t = \infty$  and  $\sum_{t=0}^{\infty} \epsilon_t^2 < \infty$ , iterations (16) will converge to the optimal dual variables [17, Prop. 8.2.6], [9, Th. 5]. Note though that the sequence of primal variables [cf. (14a) or (15)] does not necessarily converge in general under either stepsize rule.

In order to perform iterations (16), the solution of (15) is required. Each of the problems (15a)–(15d) involves a single variable, concave objective, and box constraints; thus, their solution as a function of the Lagrange multipliers is straightforward. But the solution of (15e) poses significant challenges. In particular, (15e) carries similarities to the problem of power control in deterministic DSL channels, see e.g., [13], [14] and references therein. This problem may have exponential complexity in the number of links [12]. Section IV develops an approximate solution of (15e). One remark is now due on the dual updates (16).

**Remark 1.** The subgradients in (16a) and (16b) are easily determined, once the solution of (15) is found. On the other hand, (16c) and (16d) involve the expectation  $\mathbb{E}_{\mathbf{h}}[\cdot]$ . This can be evaluated efficiently through Monte Carlo methods using two possible approaches. The first method assumes that the distribution of  $\mathbf{h}$  is known (e.g., Rayleigh or Rician), and independent realizations of  $\mathbf{h}$  are drawn from this distribution offline. For each realization, (15e) is solved and the quantities

inside the expectations in (16c)–(16d) are formed. Then the expectations are estimated through the sample averages of the quantities inside the expectations. The second method assumes that measurements of the channel vector  $\mathbf{h}$  are available, and uses them in exactly the same way as the first method. The second method works even if the measurements are correlated—which may happen in the present context when the fading process is correlated across time.

In order to complete the design, it is also important to obtain the optimal (primal) solution of (8), apart from the optimal dual variables. This is the theme of the ensuing Section III-B.

### B. Convergence of primal variables

The sequence  $\{\mathbf{x}(t), \mathbf{p}(\mathbf{h}; t)\}$  obtained as a byproduct of the subgradient method [cf. (14a)] does not converge in general. Surprisingly, it is possible to recover optimal or approximately optimal primal variables from the sequence  $\{\mathbf{x}(t), \mathbf{p}(\mathbf{h}; t)\}$  for the nonconvex primal problem in (12).

A key element for this purpose is that as  $\mathbf{p}(\mathbf{h})$  varies in  $\mathcal{B}_{\mathbf{p}}$ , the range of  $\mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]$  is a convex set under AS1–AS4. A slightly weaker statement is shown as part of [8, Lemma 1], but it can be verified that the proof there also shows the statement in the general form just given.

Now, let  $K_s := \sum_{t=0}^{s-1} \epsilon_t$ ,  $s \geq 1$ ; and define the sequence of weighted running averages of  $\mathbf{x}(t)$

$$\bar{\mathbf{x}}(s) := \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{x}(t), \quad s \geq 1 \quad (17)$$

where  $\bar{\mathbf{x}}(s) \in \mathcal{B}_{\mathbf{x}}$  due to the convexity of  $\mathcal{B}_{\mathbf{x}}$ , and the fact that  $\sum_{t=0}^{s-1} \epsilon_t / K_s = 1$ . Also define  $\check{\mathbf{p}}(\mathbf{h}; s) \in \mathcal{B}_{\mathbf{p}}$  such that

$$\mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\check{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] = \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})], \quad s \geq 1. \quad (18)$$

Such a  $\check{\mathbf{p}}(\mathbf{h}; s)$  exists because the range of  $\mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]$  is a convex set.

The main result asserts that the sequence  $\{\bar{\mathbf{x}}(s)\}$  converges to the optimal solution of (12). Formally stated, the properties of sequence  $\{\bar{\mathbf{x}}(s), \check{\mathbf{p}}(\mathbf{h}; s)\}$  are characterized in the following propositions for the considered stepsize rules. (The proofs are given in the Appendix.)

**Proposition 1.** *Let AS1–AS4 hold and the stepsize be constant,  $\epsilon_t = \epsilon > 0$ . Then, sequence  $\{\bar{\mathbf{x}}(s), \check{\mathbf{p}}(\mathbf{h}; s)\}$  satisfies*

- (i)  $\lim_{s \rightarrow \infty} \left\| \left[ \mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\check{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \right]^+ \right\| = 0$ ;
- (ii)  $\liminf_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \geq P - \frac{\epsilon G^2}{2}$ , and  $\limsup_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \leq P$ .

**Proposition 2.** *Let AS1–AS4 hold and the stepsize be non-summable but square summable, that is,  $\epsilon_t > 0$ ,  $\sum_{t=0}^{\infty} \epsilon_t = \infty$ ,  $\sum_{t=0}^{\infty} \epsilon_t^2 < \infty$ . Then, sequence  $\{\bar{\mathbf{x}}(s), \check{\mathbf{p}}(\mathbf{h}; s)\}$  satisfies*

- (i)  $\lim_{s \rightarrow \infty} \left\| \left[ \mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\check{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \right]^+ \right\| = 0$ ;
- (ii)  $\lim_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) = P$ .

Part (i) of the previous results means that the running average  $\bar{\mathbf{x}}(s)$  is asymptotically feasible, and there exists some feasible  $\check{\mathbf{p}}(\mathbf{h}; s)$  associated with it. In more precise terms, the constraint violation caused by  $\{\bar{\mathbf{x}}(s), \check{\mathbf{p}}(\mathbf{h}; s)\}$  converges to zero. Moreover,  $\bar{\mathbf{x}}(s)$  with constant stepsize is asymptotically

optimal within margin  $\epsilon G^2/2$ , while with vanishing stepsize it is optimal. It should be stressed at this point that Propositions 1 and 2 establish the optimality of the average network variables only, i.e., end-to-end rates, multicommodity flows, link capacities, average powers. They do not provide a way to obtain the instantaneous  $\mathbf{p}(\mathbf{h}; s)$  associated with those.

General results somewhat similar to Propositions 1 and 2 appear in [10] for the subgradient method with constant stepsize, and in [11] for vanishing stepsize. These are derived for convex optimization problems only; recall though that (8) is non-convex. Nevertheless, some of the methods in [10] are adapted in the proof of Propositions 1 and 2, given in the Appendix. Moreover, it should be noted that primal averaging has also been used in particular in the context of network utility maximization, but again for convex problems; see e.g., [3].

Iterations (15) and (16) together with (17) solve problem (8) and its dual (11). The flow of the subgradient algorithm is summarized in Table I, where the solution of (15a)–(15d) is shown explicitly. The utilities  $U_i^k(a_i^k)$  and  $V_i(p_i)$  are assumed nonlinear, e.g.,  $U_i^k(a_i^k) = \ln a_i^k$ , for which the inverses  $(U_i^{k'})^{-1}(\cdot)$  and  $(V_i')^{-1}(\cdot)$  exist. Algorithmic solutions to the power allocation at PHY [cf. (15e) or line 12 in Table I] are given in Section IV. In Section III-C, a simple strategy for network control based on the optimal solution of (8) is described.

### C. A strategy for network control

This section describes a network control strategy that utilizes the solution of problem (8)—that is, the optimal  $a_i^{k*}$ ,  $r_{ij}^{k*}$ ,  $c_{ij}^*$ ,  $p_i^*$ ,  $\mathbf{p}^*(\mathbf{h})$ . The network operates in time slots, which are indexed by  $\ell = 1, 2, 3, \dots$ , and network control amounts to determining how the various flows and powers are allocated per time slot. The overall algorithm proceeds in two phases: (a) an offline phase, where near-optimal primal variables and Lagrange multipliers are obtained; and (b) an online phase, where power is allocated based on the current channel realization, and routing is performed. Both phases are run at a central network controller. The two phases are detailed next.

1) *Offline phase:* During this phase, the subgradient algorithm is implemented along with the running averages, as explained in Sections III-A and III-B and listed in Table I. The purpose is to obtain near-optimal primal variables and Lagrange multipliers. In practice, a specific number ( $N$ ) of subgradient iterations will be run. Hence, what is referred to as (near-)optimal  $a_i^{k*}$ ,  $r_{ij}^{k*}$ ,  $c_{ij}^*$ , and  $p_i^*$  are in fact the values of the running averages  $\bar{a}_i^k(N)$ ,  $\bar{r}_{ij}^k(N)$ ,  $\bar{c}_{ij}(N)$ , and  $\bar{p}_i(N)$  at the last iteration. Similarly, the values of  $\Lambda(N)$  (for diminishing stepsize) or  $\bar{\Lambda}(N)$  (for constant stepsize) at the last iteration will yield the (near-)optimal Lagrange multipliers to be used during the online phase. Note also that Propositions 1 and 2 do not provide a method to obtain power allocations  $\mathbf{p}^*(\mathbf{h})$ . These will be obtained during the online phase.

The offline phase is run before communication takes place. The central controller must either know the channel distribution or have channel measurements, in order to estimate the expectations in (16c) and (16d), as explained in Remark 1.

TABLE I  
SUBGRADIENT ALGORITHM.  $N$  IS THE MAXIMUM NUMBER OF SUBGRADIENT ITERATIONS;  $[\cdot]_a^b$  DENOTES PROJECTION ONTO  $[a, b]$ ; AND AT STEP 12, ANY POWER ALLOCATION ALGORITHM CAN BE USED. IF A CONSTANT STEPSIZE IS USED, RUNNING AVERAGES OF LAGRANGE MULTIPLIERS MAY ALSO BE FORMED AT STEP 20.

1:	Initialize Lagrange multipliers $\nu_i^k(0) = \xi_{ij}(0) = \lambda_{ij}(0) = 0, \mu_i(0) = 0$
2:	Initialize running averages of primal variables $\bar{a}_i^k(0) = \bar{r}_{ij}^k(0) = \bar{c}_{ij}(0) = 0, \bar{p}_i(0) = 0$
3:	<b>for</b> $t = 0, 1, 2, \dots, N$ <b>do</b>
4:	<b>for all</b> nodes $i \in \mathcal{V}$ , neighbors $j \in \mathcal{N}(i)$ , and flows $k \in \mathcal{K}_i$ <b>do</b>
5:	Update arrival rates $a_i^k(t) \leftarrow \left[ (U_i^{k'})^{-1}(\nu_i^k(t)) \right]_{a_i^k \min}^{a_i^k \max}$
6:	Update routing variables $r_{ij}^k(t) \leftarrow \begin{cases} 0 & \text{if } j \neq \text{dest}(k) \text{ and } \nu_i^k(t) - \nu_j^k(t) - \xi_{ij}(t) \leq 0 \\ & \text{or if } j = \text{dest}(k) \text{ and } \nu_i^k(t) - \xi_{ij}(t) \leq 0 \\ r_{i \max} & \text{if } j \neq \text{dest}(k) \text{ and } \nu_i^k(t) - \nu_j^k(t) - \xi_{ij}(t) \geq 0 \\ & \text{or if } j = \text{dest}(k) \text{ and } \nu_i^k(t) - \xi_{ij}(t) \geq 0 \end{cases}$
7:	Update link capacities $c_{ij}(t) \leftarrow \begin{cases} 0 & \text{if } \xi_{ij}(t) - \lambda_{ij}(t) \leq 0 \\ c_{ij \max} & \text{if } \xi_{ij}(t) - \lambda_{ij}(t) \geq 0 \end{cases}$
8:	Update average power $p_i(t) \leftarrow \left[ (V_i')^{-1}(\mu_i(t)) \right]_0^{p_i \max}$
9:	<b>end for</b>
10:	Obtain $R$ samples of $\mathbf{h}$ , $\{\mathbf{h}^{(r)}\}_{r=1}^R$ ; Draw $R$ independent realizations of $\mathbf{h}$ from its distribution, or use $R$ available measurements
11:	<b>for</b> channel realizations $r = 1, \dots, R$ <b>do</b>
12:	Power allocation at PHY: Obtain $\mathbf{p}(\mathbf{h}^{(r)}; t)$ as $\arg \max_{\substack{0 \leq p_{ij}^f \leq p_{ij \max}^f \\ \forall i, j, f \in \mathcal{N}(i)}} \sum_{i, f} \left[ \lambda_{ij}(t) \ln \left( 1 + \gamma_{ij}^f(r) \right) - \mu_i(t) p_{ij}^f(\mathbf{h}^{(r)}) \right]$
13:	<b>end for</b>
14:	<b>for all</b> nodes $i \in \mathcal{V}$ , neighbors $j \in \mathcal{N}(i)$ , and flows $k \in \mathcal{K}_i$ <b>do</b>
15:	Approximate ergodic capacity $\hat{C}_{ij} \leftarrow \frac{1}{R} \sum_{r=1}^R \sum_f \ln \left( 1 + \gamma_{ij}^f(r) \right)$
16:	Approximate average power $\hat{P}_i \leftarrow \frac{1}{R} \sum_{r=1}^R \sum_{j \in \mathcal{N}(i), f} p_{ij}^f(\mathbf{h}^{(r)}; t)$
17:	Update Lagrange multipliers $\nu_i^k(t+1) \leftarrow \left[ \nu_i^k(t) + \epsilon_t \left( a_i^k(t) - \sum_{j \in \mathcal{N}(i)} r_{ij}^k(t) + \sum_{\substack{j \in \mathcal{N}(i) \\ j \neq \text{dest}(k)}} r_{ji}^k(t) \right) \right]^+$ $\xi_{ij}(t+1) \leftarrow \left[ \xi_{ij}(t) + \epsilon_t \left( \sum_{k \in \mathcal{K}_i} r_{ij}^k(t) - c_{ij}(t) \right) \right]^+$ $\lambda_{ij}(t+1) \leftarrow \left[ \lambda_{ij}(t) + \epsilon_t \left( c_{ij}(t) - \hat{C}_{ij} \right) \right]^+$ $\mu_i(t+1) \leftarrow \left[ \mu_i(t) + \epsilon_t \left( \hat{P}_i - p_i(t) \right) \right]^+$
18:	<b>end for</b>
19:	<b>for all</b> nodes $i \in \mathcal{V}$ , neighbors $j \in \mathcal{N}(i)$ , and flows $k \in \mathcal{K}_i$ <b>do</b>
20:	Update running averages $\bar{a}_i^k(t+1) \leftarrow \frac{1}{K_{t+1}} [K_t \bar{a}_i^k(t) + \epsilon_t a_i^k(t)]$ $\bar{r}_{ij}^k(t+1) \leftarrow \frac{1}{K_{t+1}} [K_t \bar{r}_{ij}^k(t) + \epsilon_t r_{ij}^k(t)]$ $\bar{c}_{ij}(t+1) \leftarrow \frac{1}{K_{t+1}} [K_t \bar{c}_{ij}(t) + \epsilon_t c_{ij}(t)]$ $\bar{p}_i(t+1) \leftarrow \frac{1}{K_{t+1}} [K_t \bar{p}_i(t) + \epsilon_t p_i(t)]$
21:	<b>end for</b>
22:	<b>end for</b>

Note that even when the offline phase requires considerable computation, this does not imply overhead during the network operation. Moreover, the offline phase needs to be re-run only when the network setup or the channel statistics change.

2) *Online phase:* This phase pertains to the network operation.

Each node  $i$  keeps a queue for each commodity  $k \in \mathcal{K}_i$  at

the network layer, and a queue for each neighbor  $j \in \mathcal{N}(i)$  at the physical layer; see Fig. 1. Every network layer queue accepts exogenous traffic—from the transport layer—with instantaneous rate  $\check{a}_i^k(\ell)$ . Every physical layer queue sends to the corresponding neighbor bits with instantaneous rate  $\check{c}_{ij}(\ell)$ , which depends on the instantaneous power allocation and the fading; this effect will be described in detail later. Each queue operates in a first-in-first-out (FIFO) fashion, and has unlimited storage space.

There is an interface connecting the network layer and physical layer queues at node  $i$ . This interface is responsible for routing, because it takes bits from each network layer queue, and places them into the physical layer queues. The individual rate from the  $k$ th network layer queue to the  $j$ th physical layer queue is denoted by  $\check{r}_{ij}^k(\ell)$ . Then, bits leave the  $k$ th network layer queue with instantaneous rate  $\sum_{j \in \mathcal{N}(i)} \check{r}_{ij}^k(\ell)$ , and arrive at the  $j$ th physical layer queue with rate  $\sum_{k \in \mathcal{K}_i} \check{r}_{ij}^k(\ell)$ . Note that the data in the physical layer queues are labeled with the commodity index  $k$ , which determines their destination. The bits  $\check{c}_{ij}(\ell)$  from node  $i$  to node  $j$  are placed into the corresponding network layer queues of node  $j$  according to their label, except those with  $\text{dest}(k) = j$ , which have arrived at their destination and are not placed in any queue. Now, consider this operation for the bits  $\check{c}_{mi}(\ell)$  arriving to  $i$  from its neighbors  $m \in \mathcal{N}(i)$ ; the endogenous arrivals at the  $k$ th network layer queue will be  $\sum_{m \in \mathcal{N}(i), m \neq \text{dest}(k)} \check{c}_{mi}^k(\ell)$ , where the  $\check{c}_{mi}^k(\ell)$  are determined by splitting  $\check{c}_{mi}(\ell)$ . Note that if the time slot duration is normalized to one, all the aforementioned instantaneous rates also play the role of arrival or service processes for their respective queues.

A network control algorithm must determine  $\check{a}_i^k(\ell)$ ,  $\check{r}_{ij}^k(\ell)$ , and  $\check{c}_{ij}(\ell)$  per time slot  $\ell$ . Variables  $\check{c}_{ij}(\ell)$  also depend on the fading, which of course cannot be controlled. Here,  $\check{a}_i^k(\ell)$ ,  $\check{r}_{ij}^k(\ell)$ , and  $\check{c}_{ij}(\ell)$  will be determined by the optimal solution of (8), that is,  $a_i^{k*}$ ,  $r_{ij}^{k*}$ , and  $\mathbf{p}^*(\mathbf{h})$ . Specifically, since an optimal  $\mathbf{p}^*(\mathbf{h})$  is not readily available, the solution of (15e) with the optimal Lagrange multipliers  $\lambda_{ij}^*$  and  $\mu_i^*$  will be used whenever  $\mathbf{p}^*(\mathbf{h})$  is mentioned.

In the online operation, the central controller knows the current realization of  $\mathbf{h}$ , and has the  $a_i^{k*}$ ,  $r_{ij}^{k*}$ ,  $\lambda_{ij}^*$ , and  $\mu_i^*$  available from the offline phase. It can then obtain  $\mathbf{p}^*(\mathbf{h})$  for the current channels—using the solvers of (15e) detailed in Section IV. These quantities are used per time slot as described next.

The network operates under the condition that the random traffic arrival process has the optimal long-term average, i.e.,  $\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{\ell=1}^M \check{a}_i^k(\ell) = a_i^{k*}$ . This operating condition for  $\check{a}_i^k(\ell)$  is adopted because  $a_i^{k*}$  is the optimal operating point of the network, determined by problem (8). Supposing as usual that there are always enough packets available at the transport layer of every node (infinitely backlogged transport layer [2]), this condition is ensured if the controller admits packets with rate  $\check{a}_i^k(\ell)$  having the correct long-term average  $a_i^{k*}$ . As a special case, one can set  $\check{a}_i^k(\ell) = a_i^{k*}$  for all  $\ell$ .

Moreover, the routing variables are set to the optimal multicommodity flows  $r_{ij}^{k*}$ , i.e.,

$$\check{r}_{ij}^k(\ell) = r_{ij}^{k*}, \quad \ell = 1, 2, \dots \quad (19)$$

The power allocation  $\mathbf{p}^*(\mathbf{h})$  will be used whenever the

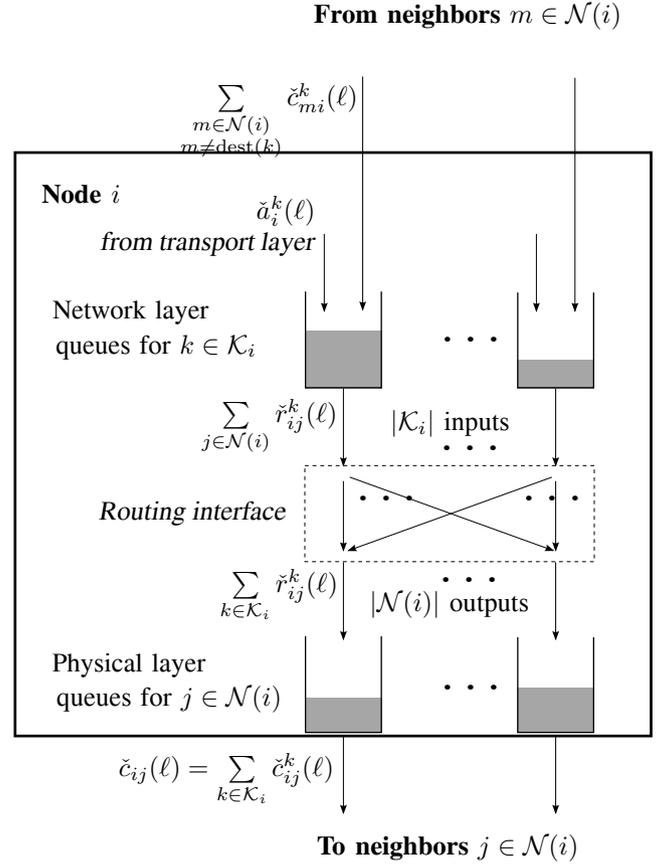


Fig. 1. Queues at node  $i$  and connections to neighbors.

fading state is  $\mathbf{h}$ . The physical layer rate  $\check{c}_{ij}(\ell)$  at each time slot depends on whether the fading is slow or fast. In the case of slow fading, the channels  $\mathbf{h}$  will be (approximately) constant over the time slot, and will have value denoted by  $\mathbf{h}(\ell)$ . The instantaneous physical layer rate  $\check{c}_{ij}(\ell)$  is

$$\check{c}_{ij}(\ell) = \sum_{f \in \mathcal{F}} \ln[1 + \gamma_{ij}^f(\mathbf{h}(\ell), \mathbf{p}^*(\mathbf{h}(\ell)))] \quad \ell = 1, 2, \dots \quad (20)$$

where the notation  $\gamma_{ij}^f(\mathbf{h}, \mathbf{p}(\mathbf{h}))$  means that the channels have value  $\mathbf{h}$  in (3), and the power allocation is  $\mathbf{p}(\mathbf{h})$ . In the case of fast fading, the channels  $\mathbf{h}$  change over the time slot. If the time slot duration is much larger than the coherence time of  $\mathbf{h}$ , then the instantaneous physical layer rate  $\check{c}_{ij}(\ell)$  can be approximated by

$$\check{c}_{ij}(\ell) = \mathbb{E}_{\mathbf{h}} \left[ \sum_{f \in \mathcal{F}} \ln(1 + \gamma_{ij}^f(\mathbf{h}, \mathbf{p}^*(\mathbf{h}))) \right] \quad \ell = 1, 2, \dots \quad (21)$$

It is possible that the queues do not have as many bits as required by the service processes  $\sum_{j \in \mathcal{N}(i)} \check{r}_{ij}^k(\ell)$  and  $\check{c}_{ij}(\ell)$ ; then, they simply transmit as many as available. Furthermore, it is important to stress the role of the physical layer queues in the interaction between the routing decisions  $\check{r}_{ij}^k(\ell)$  and the instantaneous capacity supported by the physical layer  $\check{c}_{ij}(\ell)$ . These buffers effectively store the bits that cannot be transmitted if the instantaneous capacity  $\check{c}_{ij}(\ell)$  drops below the rate that the layer above wants to “pump out”, that is,  $\sum_{k \in \mathcal{K}_i} \check{r}_{ij}^k(\ell)$ .

To build intuition about the algorithm, note that the long-term average rates  $\frac{1}{M} \sum_{\ell=1}^M \check{a}_i^k(\ell)$  and  $\frac{1}{M} \sum_{\ell=1}^M \check{r}_{ij}^k(\ell)$  converge to the optimal  $a_i^{k*}$  and  $r_{ij}^{k*}$ . Moreover, the endogenous rates  $\check{c}_{ji}^k(\ell)$  will satisfy  $\sum_{\ell=1}^M \check{c}_{ji}^k(\ell) \leq \sum_{\ell=1}^M \check{r}_{ji}^k(\ell)$ , because all packets placed endogenously into the network layer queues of node  $i$  must have been routed to  $i$  from its neighbors. Hence, (8b) will be satisfied by the long-term averages of the respective processes; that is,  $\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{\ell=1}^M \check{a}_i^k(\ell) + \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{\ell=1}^M \check{c}_{ji}^k(\ell) \leq \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{\ell=1}^M \check{r}_{ij}^k(\ell)$ . A similar conclusion is true for (8c) with  $c_{ij} = c_{ij}^*$ . Recall that  $a_i^{k*}$ ,  $r_{ij}^{k*}$ ,  $c_{ij}^*$ ,  $p_i^*$ ,  $\mathbf{p}^*(\mathbf{h})$  are optimal solutions of (8). Also, as long as the solution of (15e) with the optimal Lagrange multipliers is close to the optimal power allocation, the long-term average link capacities  $\frac{1}{M} \sum_{\ell=1}^M \check{c}_{ij}^k(\ell)$  and long-term average power consumptions converge, respectively, to their expected values  $\mathbb{E}_{\mathbf{h}}[\sum_{f \in \mathcal{F}} \ln(1 + \gamma_{ij}^f(\mathbf{h}, \mathbf{p}^*(\mathbf{h})))]$  and  $\mathbb{E}_{\mathbf{h}}[\sum_{f \in \mathcal{F}} \sum_{j \in \mathcal{N}(i)} p_{ij}^{f*}(\mathbf{h})]$  under both fading scenarios, since the fading process is stationary and ergodic. Then, (8d) and (8e) are satisfied with  $c_{ij} = c_{ij}^*$  and  $p_i = p_i^*$ .

**Remark 2.** The present framework has certain practical limitations, which are to some degree common to several works developing online algorithms for network optimization and control. Specifically, a central controller is not desirable in large or ad hoc networks, yet it is commonly utilized; see e.g., [2], [3], [6], and references therein. Another limitation is that the network controller needs access to the current fading state without delay, and is again typical; see for instance, the framework of [2] and [6]. Developing efficient methods for acquiring channel estimates accurately and fast poses a challenge. Furthermore, solving for the power allocation per fading state using (15e) may incur high computational burden, even when the machinery of convex optimization is employed, as discussed in Section IV. Most online cross-layer resource allocation algorithms deal with a similar task at the physical and link layers; see e.g., [1], [2], [6]. This task is usually referred to as scheduling and is well known to be difficult for several physical and link layer models, such as the one here; approximate solutions that are suboptimal but considerably less computationally demanding are welcome.

#### IV. POWER ALLOCATION AT THE PHY

This section deals with the solution of (15e). Let  $\mathbf{p}^f$  denote the power allocations  $p_{ij}^f$  of all links over tone  $f$ ; also define  $\mathcal{P}_{\text{mask}}^f := \{\mathbf{p}^f | 0 \leq p_{ij}^f \leq p_{ij}^f \text{max} \forall j \in \mathcal{N}(i), \forall i\}$ . To start, note that (15e) can be decomposed into separate problems, one per tone  $f$ , i.e.,

$$\max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} [\lambda_{ij}(t) \ln(1 + \gamma_{ij}^f) - \mu_i(t) p_{ij}^f]. \quad (22)$$

Note that  $t$  and the Lagrange multipliers are fixed now. Next, certain methods for the solution of (22) are described.

1) *Exact methods:* Exhaustive search in the space of  $\mathbf{p}^f$  can be used in principle to find the global maximizer of (22). The major limitation however, is that exhaustive search incurs exponential complexity in the number of variables in  $\mathbf{p}^f$ . Another approach is to use a branch-and-bound method. Such a method is pursued in [18, Sec. 5], and its adaptation for the solution of (22) is straightforward. The worst-case

complexity of this method is also exponential, but in practice it may work well, in the sense that it can find the global maximizers with much lower computation requirements than the exhaustive search.

2) *Low-SINR and high-SINR approximations:* The low-SINR approximation uses  $\ln(1 + \gamma_{ij}^f) \approx \gamma_{ij}^f$  in (22). The resulting optimization problem can be written after straightforward manipulations as a signomial program, which is an intractable non-convex problem (see [4, Sec. 2.2.5] for definitions). Its solution will not be further pursued here, because it requires extensive computation, while it will only be valid for a limited range of SINRs. Moreover, results on the optimality of on-off power allocation in the low-SINR region (see e.g., [7] and references in [1, Sec. III-E]) may not be directly applicable here, because (22) is formally different than the problems in those works.

On the other hand, the high-SINR approximation uses  $\ln(1 + \gamma_{ij}^f) \approx \ln \gamma_{ij}^f$  in (22). Using the transformation  $p_{ij}^f = e^{y_{ij}^f}$ , the problem reduces to a convex one in the variables  $y_{ij}^f$ . This transformation is standard in geometric programming, as well as in power control problems [4]. In the case of (22) however, the resulting problem is not a geometric program. Nonetheless, it can be solved efficiently by any algorithm used for convex problems, e.g., gradient descent or interior-point methods [16]. Moreover, the solution of the high-SINR approximation may be a step within successive approximation methods, which are described next. High-SINR approximations have been used for cross-layer network optimization in [4, Sec. 3.4] and [5].

3) *Successive convex approximations:* The premise here is that a sequence of power allocations  $\mathbf{p}^f(\tau)$ ,  $\tau \geq 1$ , is obtained, where each  $\mathbf{p}^f(\tau)$  is the solution of an optimization problem that approximates (22). Since this problem must be easy to solve, it is constructed to ensure convexity. This is done essentially by substituting the objective function in (22) with a convex function. The resulting problem can be solved efficiently by any algorithm used for convex problems, e.g., gradient descent or interior-point methods; or even by an algorithm developed specifically for that problem. Three methods to obtain problems approximating (22) are described here, inspired from the DSL literature [13], [14], and the condensation method [4, Sec. 2.2]. In all these cases the function replacing the objective in (22) is a lower bound of the objective for all values of  $p_{ij}^f$ . Moreover, the convex function selected at step  $\tau$  is constructed using the previous solution  $\mathbf{p}^f(\tau - 1)$ . The idea is that the final power allocation obtained from this procedure offers an approximate solution to (22). The details of each method are described next.

First, a solution to (22) is pursued based on the SCALE algorithm [13]. The lower bound for the objective function in (22) is based on the inequality  $\alpha_{ij}^f \ln \gamma_{ij}^f + \beta_{ij}^f \leq \ln(1 + \gamma_{ij}^f)$  for all  $\gamma_{ij}^f$ , where  $\alpha_{ij}^f = \tilde{\gamma}/(1 + \tilde{\gamma})$  for some arbitrary  $\tilde{\gamma} > 0$ , and  $\beta_{ij}^f$  is properly selected [13, eq. (2)]. The inequality becomes equality for  $\gamma_{ij}^f = \tilde{\gamma}$ . The approximation to (22) at step  $\tau$  amounts to

$$\max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} [\lambda_{ij}(t) (\alpha_{ij}^f(\tau) \ln \gamma_{ij}^f + \beta_{ij}^f(\tau)) - \mu_i(t) p_{ij}^f]. \quad (23)$$

TABLE II

POWER ALLOCATION ALGORITHM BASED ON SCALE. T IS THE MAXIMUM NUMBER OF SUCCESSIVE APPROXIMATIONS AFTER THE 1ST (HIGH-SINR) APPROXIMATION ( $T \geq 1$ ); AND AT STEPS 7 AND 13, ANY ALGORITHM CAN BE USED.

```

1: for all tones  $f \in \mathcal{F}$  do
    High-SINR approximation
2: Set  $\tau = 1$ 
3: for all nodes  $i \in \mathcal{V}$  and neighbors  $j \in \mathcal{N}(i)$  do
4: Initialize weights  $\alpha_{ij}^f(1) = 1$ 
5: Initialize power allocations  $0 < p_{ij}^f(0) \leq p_{ij}^f \text{max}$ 
6: end for
7: Solve high-SINR approximation, using  $\mathbf{p}^f(0)$  as initialization
 $\mathbf{p}^f(1) \in \arg \max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} [\lambda_{ij}(t) \ln \gamma_{ij}^f - \mu_i(t) p_{ij}^f]$ 
    Successive approximations
8: repeat
9:  $\tau \leftarrow \tau + 1$ 
10: for all nodes  $i \in \mathcal{V}$  and neighbors  $j \in \mathcal{N}(i)$  do
11: Update weights using  $\mathbf{p}^f(\tau - 1)$ ,  $\alpha_{ij}^f(\tau) \leftarrow \frac{\gamma_{ij}^f(\tau-1)}{1 + \gamma_{ij}^f(\tau-1)}$ 
12: end for
13: Solve the approximation at step  $\tau$ , using  $\mathbf{p}^f(\tau - 1)$  as initialization
 $\mathbf{p}^f(\tau) \in \arg \max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} [\lambda_{ij}(t) \alpha_{ij}^f(\tau) \ln \gamma_{ij}^f - \mu_i(t) p_{ij}^f]$ 
14: until  $\mathbf{p}^f(\tau)$  converges or  $\tau > T$ 
15: end for

```

Because the  $\beta_{ij}^f$  are constants, hereafter they will not be taken into account in problem (23). Using the transformation  $p_{ij}^f = e^{y_{ij}^f}$ , (23) can be recast as a convex optimization problem.

The solution of (22) based on (23) proceeds as follows. In each step  $\tau \geq 2$ , the weights are computed as  $\alpha_{ij}^f(\tau) = \gamma_{ij}^f(\tau - 1) / (1 + \gamma_{ij}^f(\tau - 1))$ , where the previous solution  $\mathbf{p}^f(\tau - 1)$  was used in  $\gamma_{ij}^f(\tau - 1)$ . Then (23) is solved exactly with the fixed  $\alpha_{ij}^f(\tau)$  to obtain  $\mathbf{p}^f(\tau)$ ; and this sequence of weight updating and solving (23) is repeated. This procedure ends when the sequence  $\{\mathbf{p}^f(\tau)\}$  obtained as solutions of (23) converges, or practically, after a prescribed number of approximations. At  $\tau = 1$ ,  $\{\mathbf{p}^f(1)\}$  is obtained from (23) with  $\alpha_{ij}^f(1) = 1$ , which coincides with the high-SINR approximation. Any algorithm can be used for solving (23); recall that any locally optimal solution of (23) will be globally optimal, due to the equivalence of (23) to a convex problem. It is also possible to derive an iterative algorithm tailored specifically to (23); see [13], [19] for details.

The successive approximation procedure based on SCALE is summarized in Table II, where Steps 2–7 correspond to the solution of the high-SINR approximation.

The second method to obtain a convex approximation of (22) is based on rewriting  $\ln(1 + \gamma_{ij}^f)$  as  $\ln(h_{ij}^f p_{ij}^f + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f) - \ln(\sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f)$ , and noting that the second summand with its sign is convex in  $\mathbf{p}^f$ . But any convex function can be lower bounded by an affine function [16, Sec. 3.1.3]. Specifically, given a fixed  $\check{\mathbf{p}}^f$ , the following inequality holds for all  $\check{\mathbf{p}}^f \geq \mathbf{0}$  for properly selected

TABLE III

POWER ALLOCATION ALGORITHM BASED ON AFFINE APPROXIMATION TO THE LOGARITHM OF NOISE-PLUS-INTERFERENCE. T IS THE MAXIMUM NUMBER OF SUCCESSIVE APPROXIMATIONS ( $T \geq 1$ ); AND AT STEP 13, ANY ALGORITHM FOR CONVEX PROGRAMS CAN BE USED.

```

1: for all tones  $f \in \mathcal{F}$  do
2: Set  $\tau = 0$ 
3: for all nodes  $i \in \mathcal{V}$  and neighbors  $j \in \mathcal{N}(i)$  do
4: Initialize power allocations  $0 \leq p_{ij}^f(0) \leq p_{ij}^f \text{max}$ 
5: end for
6: repeat
7:  $\tau \leftarrow \tau + 1$ 
8: for all nodes  $i \in \mathcal{V}$  and neighbors  $j \in \mathcal{N}(i)$  do
9: for all links  $(k, l) \in \mathcal{I}_{ij}$  [causing interference to  $(i, j)$ ] do
10: Update coefficients using  $\mathbf{p}^f(\tau - 1)$ ,
 $a_{ij;kl}^f(\tau) \leftarrow \frac{h_{kj}^f}{\sigma_j^f + \sum_{(m,n) \in \mathcal{I}_{ij}} h_{mj}^f p_{mn}^f(\tau-1)}$ 
11: end for
12: end for
13: Solve convex approximation at step  $\tau$ 
 $\mathbf{p}^f(\tau) \in \arg \max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} \left[ \lambda_{ij}(t) \ln \left( h_{ij}^f p_{ij}^f + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f \right) - \sum_{(k,l) \in \mathcal{I}_{ij}} a_{ij;kl}^f(\tau) p_{kl}^f - \mu_i(t) p_{ij}^f \right]$ 
14: until  $\mathbf{p}^f(\tau)$  converges or  $\tau \geq T$ 
15: end for

```

constants  $b_{ij}^f$  [16, eq. (3.2)]:

$$-\ln \left( \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f \right) \geq - \sum_{(k,l) \in \mathcal{I}_{ij}} a_{ij;kl}^f p_{kl}^f + b_{ij}^f \quad (24a)$$

where

$$a_{ij;kl}^f = \frac{h_{kj}^f}{\sigma_j^f + \sum_{(m,n) \in \mathcal{I}_{ij}} h_{mj}^f p_{mn}^f}. \quad (24b)$$

Using (24) and omitting the  $b_{ij}^f$ , (22) becomes at step  $\tau$  the following convex problem in  $\mathbf{p}^f$ :

$$\max_{\mathbf{p}^f \in \mathcal{P}_{\text{mask}}^f} \sum_{i,j \in \mathcal{N}(i)} \left[ \lambda_{ij}(t) \ln \left( h_{ij}^f p_{ij}^f + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f \right) - \sum_{(k,l) \in \mathcal{I}_{ij}} a_{ij;kl}^f(\tau) p_{kl}^f - \mu_i(t) p_{ij}^f \right]. \quad (25)$$

Similar to the method based on SCALE, the solution of (22) based on (24)–(25) proceeds as follows. In each step  $\tau \geq 1$ , the coefficients  $a_{ij;kl}^f(\tau)$  are computed based on (24) using the previous solution  $\mathbf{p}^f(\tau - 1)$ . Then problem (25) is solved exactly with the fixed  $a_{ij;kl}^f(\tau)$  to obtain  $\mathbf{p}^f(\tau)$ , using any algorithm for convex problems. The procedure is initialized with power allocation  $\mathbf{p}^f(0)$  within the spectral mask. The algorithm is summarized in Table III.

The third method is based on rewriting (22) in an equivalent form such that an upper bound of a ratio of posynomials appears as constraint (see [16, p. 160] for definitions). This

form is

$$\min_{\mathbf{p}^f, \mathbf{u}^f, s} \prod_{i,j \in \mathcal{N}(i)} (u_{ij}^f)^{\lambda_{ij}(t)} + s \quad (26a)$$

$$\text{subj. to } \frac{\sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f}{u_{ij}^f \left( h_{ij}^f p_{ij}^f + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f \right)} \leq 1 \quad \forall i, j \in \mathcal{N}(i) \quad (26b)$$

$$e^{\sum_{i,j \in \mathcal{N}(i)} \mu_i(t) p_{ij}^f} \leq s \quad \forall i, j \in \mathcal{N}(i) \quad (26c)$$

$$0 \leq p_{ij}^f \leq p_{ij}^f \text{max} \quad \forall i, j \in \mathcal{N}(i) \quad (26d)$$

where  $u_{ij}^f$ ,  $s$  are auxiliary variables, and  $\mathbf{u}^f$  collects  $u_{ij}^f$  for all links. Constraint (26b) is a ratio of posynomials and is non-convex. The approach is to approximate the posynomial in the denominator with a monomial. This is called condensation of the posynomial, and constitutes a standard method to convexify constraints which involve an upper bound on the ratio of posynomials [4, Sec. 2.2.5]. Specifically, given fixed  $\check{u}_{ij}^f$ ,  $\check{p}_{ij}^f$ , and  $\check{p}_{kl}^f$  for  $(k, l) \in \mathcal{I}_{ij}$ , the monomial approximating (in fact, lower-bounding for all  $p_{ij}^f > 0$ ,  $u_{ij}^f > 0$ ) the denominator in (26b) for link  $(i, j)$  is

$$\left( \frac{h_{ij}^f p_{ij}^f u_{ij}^f}{\delta_{ij}^f} \right)^{\delta_{ij}^f} \cdot \left( \frac{\sigma_j^f u_{ij}^f}{\eta_{ij}^f} \right)^{\eta_{ij}^f} \cdot \prod_{(k,l) \in \mathcal{I}_{ij}} \left( \frac{h_{kj}^f p_{kl}^f u_{ij}^f}{\theta_{ij;kl}^f} \right)^{\theta_{ij;kl}^f} \quad (27a)$$

where

$$\delta_{ij}^f = \frac{h_{ij}^f \check{p}_{ij}^f \check{u}_{ij}^f}{\check{\omega}}, \eta_{ij}^f = \frac{\sigma_j^f \check{u}_{ij}^f}{\check{\omega}}, \theta_{ij;kl}^f = \frac{h_{kj}^f \check{p}_{kl}^f \check{u}_{ij}^f}{\check{\omega}} \quad (27b)$$

$$\check{\omega} = \check{u}_{ij}^f \left( h_{ij}^f \check{p}_{ij}^f + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f \check{p}_{kl}^f \right). \quad (27c)$$

Using the aforementioned approximation, and applying the transformation  $p_{ij}^f = e^{y_{ij}^f}$ ,  $u_{ij}^f = e^{v_{ij}^f}$ ,  $s = e^z$ , the resulting problem is convex in the variables  $y_{ij}^f$ ,  $v_{ij}^f$ ,  $z$  (but not a geometric program).

The algorithm to solve (22) based on (26)–(27) proceeds as follows. In each step  $\tau \geq 1$ , the parameters  $\delta_{ij}^f(\tau)$ ,  $\eta_{ij}^f(\tau)$ , and  $\theta_{ij;kl}^f(\tau)$  are obtained for all links based on (27) using the previous solution  $\mathbf{p}^f(\tau-1)$ ,  $\mathbf{u}^f(\tau-1)$ . Next, problem (26) is solved, where the denominator in (26b) is first substituted by the monomial in (27a) with parameters  $\delta_{ij}^f(\tau)$ ,  $\eta_{ij}^f(\tau)$ , and  $\theta_{ij;kl}^f(\tau)$ . Note that this problem can be solved by any algorithm for programs which are convex in the variables  $y_{ij}^f$ ,  $v_{ij}^f$ ,  $z$ . The procedure is initialized with power allocation  $\mathbf{p}^f(0)$  within the spectral mask. The algorithm is summarized in Table IV.

**Remark 3.** The algorithms presented here involve centralized computation. This is expected, because the objective function and the interference couple the power allocation across terminals [cf. (15e) or (22)], and a (near-)optimal power allocation is sought. Moreover, while problem (22) may in general incur exponential complexity in the number of links, the methods based on successive convex approximations have polynomial complexity.

## V. NUMERICAL TESTS

The algorithms developed in the previous sections will be tested on a wireless network consisting of 8 terminals placed

TABLE IV

POWER ALLOCATION ALGORITHM BASED ON CONDENSATION.  $T$  IS THE MAXIMUM NUMBER OF SUCCESSIVE APPROXIMATIONS ( $T \geq 1$ ); AND AT STEP 16, ANY ALGORITHM FOR CONVEX PROGRAMS CAN BE USED.

- 1: **for all tones**  $f \in \mathcal{F}$  **do**
- 2:   Set  $\tau = 0$
- 3:   **for all nodes**  $i \in \mathcal{V}$  **and neighbors**  $j \in \mathcal{N}(i)$  **do**
- 4:     Initialize power allocations  $0 \leq p_{ij}^f(0) \leq p_{ij}^f \text{max}$
- 5:     Initialize auxiliary variable  $u_{ij}^f(0)$  such that
 
$$\frac{\sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f(0)}{u_{ij}^f(0) \left( h_{ij}^f p_{ij}^f(0) + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f(0) \right)} \leq 1$$
- 6:   **end for**
- 7:   Initialize auxiliary variable  $s(0)$  such that
 
$$e^{\sum_{i,j \in \mathcal{N}(i)} \mu_i(t) p_{ij}^f(0)} \leq s(0)$$
- 8:   **repeat**
- 9:      $\tau \leftarrow \tau + 1$
- 10:    **for all nodes**  $i \in \mathcal{V}$  **and neighbors**  $j \in \mathcal{N}(i)$  **do**
- 11:     Update parameters  $\delta_{ij}^f$ ,  $\eta_{ij}^f$  of monomial approximation (27a) using  $\mathbf{p}^f(\tau-1)$  and  $\mathbf{u}^f(\tau-1)$ 

$$\delta_{ij}^f(\tau) \leftarrow \frac{h_{ij}^f p_{ij}^f(\tau-1) u_{ij}^f(\tau-1)}{u_{ij}^f(\tau-1) \left( h_{ij}^f p_{ij}^f(\tau-1) + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f(\tau-1) \right)}$$

$$\eta_{ij}^f(\tau) \leftarrow \frac{\sigma_j^f u_{ij}^f(\tau-1)}{u_{ij}^f(\tau-1) \left( h_{ij}^f p_{ij}^f(\tau-1) + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f(\tau-1) \right)}$$
- 12:     **for all links**  $(k, l) \in \mathcal{I}_{ij}$  [causing interference to  $(i, j)$ ] **do**
- 13:      Update parameter  $\theta_{ij;kl}^f$  of monomial approximation (27a) using  $\mathbf{p}^f(\tau-1)$ ,  $\mathbf{u}^f(\tau-1)$ 

$$\theta_{ij;kl}^f(\tau) \leftarrow \frac{h_{kj}^f p_{kl}^f(\tau-1) u_{ij}^f(\tau-1)}{u_{ij}^f(\tau-1) \left( h_{ij}^f p_{ij}^f(\tau-1) + \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f(\tau-1) \right)}$$
- 14:     **end for**
- 15:     **end for**
- 16:     Solve convex approximation at step  $\tau$  in the variables  $y_{ij}^f = \ln p_{ij}^f$ ,  $v_{ij}^f = \ln u_{ij}^f$ ,  $z = \ln s$  to obtain  $\mathbf{p}^f(\tau)$ 

$$\min_{\mathbf{p}^f, \mathbf{u}^f, s} \prod_{i,j \in \mathcal{N}(i)} (u_{ij}^f)^{\lambda_{ij}(t)} + s$$

$$\text{subj. to } v(\mathbf{p}^f, \mathbf{u}^f; \tau) \leq 1 \quad \forall i, j \in \mathcal{N}(i)$$

$$e^{\sum_{i,j \in \mathcal{N}(i)} \mu_i(t) p_{ij}^f} \leq s \quad \forall i, j \in \mathcal{N}(i)$$

$$0 \leq p_{ij}^f \leq p_{ij}^f \text{max} \quad \forall i, j \in \mathcal{N}(i)$$
 where  $v(\mathbf{p}^f, \mathbf{u}^f; \tau)$  is defined at the end of the table
- 17:     **until**  $\mathbf{p}^f(\tau)$  converges **or**  $\tau \geq T$
- 18: **end for**

The function  $v(\mathbf{p}^f, \mathbf{u}^f; \tau)$  is

$$\frac{\left( \sigma_j^f + \sum_{(k,l) \in \mathcal{I}_{ij}} h_{kj}^f p_{kl}^f \right) (\delta_{ij}^f(\tau))^{\delta_{ij}^f(\tau)} (\eta_{ij}^f(\tau))^{\eta_{ij}^f(\tau)} \prod_{(k,l) \in \mathcal{I}_{ij}} (\theta_{ij;kl}^f(\tau))^{\theta_{ij;kl}^f(\tau)}}{\left( h_{ij}^f p_{ij}^f u_{ij}^f \right)^{\delta_{ij}^f(\tau)} (\sigma_j^f u_{ij}^f)^{\eta_{ij}^f(\tau)} \prod_{(k,l) \in \mathcal{I}_{ij}} \left( h_{kj}^f p_{kl}^f u_{ij}^f \right)^{\theta_{ij;kl}^f(\tau)}}$$

on a  $300\text{m} \times 100\text{m}$  area, as shown in Fig. 3. In Section V-A, the subgradient algorithm of Table I is used to obtain near-optimal Lagrange multipliers, and near-optimal primal variables  $a_i^k$ ,  $r_{ij}^k$ ,  $c_{ij}$ ,  $p_i$  (offline phase). Then, simulation results from running the network with the strategy of Section III-C are presented in Section V-B (online phase).

### A. Near-optimal design

Two different designs are pursued. One is obtained from the subgradient method together with the algorithm based on SCALE (cf. Tables I and II), while the other uses the high-SINR approximation instead. Each terminal in the network is destination of a single commodity ( $\text{dest}(k) = k$  for  $k = 1, \dots, 8$ ); hence, the set of commodities that enter each terminal is  $\mathcal{K}_i = \mathcal{V} \setminus \{i\}$ . The rest of the parameters are listed in Table V. The values of channel gains and noise have the

implication that, even with no interference and transmission power 1 W/Hz, the terminals operate at low to moderate SINR. First, the results obtained from the subgradient method combined with the algorithm based on SCALE are presented; and then, they are compared with those from the high-SINR approximation. In both cases, MATLAB's implementation of sequential quadratic programming [20] was used to solve the approximating problem (23). The algorithm based on SCALE (Table II) was run with multiple initializations in Step 5. Specifically, each initialization  $\mathbf{p}^f(0)$  had the following form: one entry took the value of the spectral mask, and all other entries took the value  $-100$  dB; this was repeated for every entry. After the algorithm of Table II had run with those initializations, the solution giving the best objective value in Step 13 was used for the next subgradient step. On the other hand, random initialization was used in Step 5 for the case where just the high-SINR approximation was used for the design; in particular, uniform over the interval  $[0, p_{ij}^f \max/5]$ , independently for each link and tone.

The subgradient method with constant stepsize gives near-optimal Lagrange multipliers, by forming running averages of the dual iterates,  $\bar{\Lambda}(s) := \frac{1}{s} \sum_{t=0}^{s-1} \Lambda(t)$ ; and also near-optimal arrival rates, multicommodity flows, link capacities, and average powers, by forming the corresponding running averages,  $\bar{a}_{ij}^k(t)$ ,  $\bar{r}_{ij}^k(t)$ ,  $\bar{c}_{ij}(t)$ , and  $\bar{p}_i(t)$  (cf. Proposition 1). Those are depicted in Figures 2, 3, 4, and 5, as explained next.

Fig. 2a shows an example of the convergence of the averaged dual iterates; not all Lagrange multipliers can be shown here due to space limitations. It is interesting to note that convergence occurs even with a few Monte Carlo realizations. Moreover, Fig. 2b depicts the norm of the violation of the constraints (8b) and (8c) evaluated at the running averages of the respective variables, which is seen to be decreasing [cf. Proposition 1(i)].

Multicommodity flows  $r_{ij}^k$  define routes for each commodity. Consider commodity  $k = 8$  consisting of traffic transmitted to terminal 8. Terminal 8 is depicted in Fig. 3 as a square. The remaining terminals are shown as circles with area proportional to the total (endogenous plus exogenous) incoming rate to that terminal having terminal 8 as destination, i.e.,  $a_i^8 + \sum_{j \in \mathcal{N}(i), j \neq 8} r_{ji}^8$ . Those incoming rates are also given in Fig. 3, and represent the amount of traffic for terminal 8 that each node handles. Observe that packets accumulate as we move from terminal 1 (the furthest from 8) to the closest terminals 6 and 7.

Fig. 4 shows link capacities. The width of each arrow is proportional to the average capacity  $c_{ij}$  of the corresponding link. The color of each arrow is obtained by linearly mapping the capacities on a scale of gray (see bar in Fig. 4), where black corresponds to the largest capacity in the network. Notice that certain links have larger capacity than others. For instance, the links corresponding to the edges of the "squares", e.g., (2,4) and (5,4), have larger capacities than the "diagonal" links such as (3,4). This is expected, because the distance between terminals 3 and 4 is larger than the distance between 2 and 4, or, between 5 and 4.

The average power  $p_i$  consumed per terminal is depicted in Fig. 5, and is seen to be approximately the same for all terminals. However, the "side" nodes, 1 and 8, require less

TABLE V  
PARAMETERS USED IN THE NUMERICAL TESTS

Network parameters and utility functions	
$F$	1
$h_{ij}^f$	Exponential with mean $\bar{h}_{ij}^f = 0.1(d_{ij}/d_0)^{-2}$ for all $(i, j)$ and $f$ ; $d_{ij}$ = distance between nodes $i$ and $j$ in meters; $d_0 = 20$ m; links are reciprocal, i.e., $h_{ij}^f = h_{ji}^f$ for all $(i, j)$ and $f$
$h_{jj}^f$	10 dB, deterministic for all $j$ and $f$
$\sigma_j^f$	$\min_{i \in \mathcal{N}(j)} \bar{h}_{ij}^f$ (W/Hz) for all $j$ and $f$
$p_i \max$	5 W/Hz for all $i$
$p_{ij}^f \max$	5 W/Hz for all $(i, j)$ and $f$
$a_{ij}^k \max$	14 bps/Hz for all $i$ and $k \in \mathcal{K}_i$
$a_{ij}^k \min$	$1.96 \cdot 10^{-4}$ bps/Hz for all $i$ and $k \in \mathcal{K}_i$
$c_{ij} \max$	$\max_{i, j \in \mathcal{N}(i)} \bar{C}_{ij}$ for all $(i, j)$ where $\bar{C}_{ij}$ is the max. capacity of link $(i, j)$ without interference; obtained via waterfilling under $\mathbb{E}_{\mathbf{h}}[\sum_f p_{ij}^f(\mathbf{h})] \leq p_i \max$
$r_i \max$	$c_{ij} \max$ for all $i$
$U_i^k(a_i^k)$	$\ln a_i^k$ for all $i$ and $k \in \mathcal{K}_i$
$V_i(p_i)$	$p_i^2$ for all $i$
Parameters for design based on SCALE	
$\epsilon_t$	1
$R$	3 independent realizations
$T$	5, up to iteration 2000; 6, thereafter
Algorithm for (23)	MATLAB's Sequential Quadratic Programming default parameters, except $\text{ToIFun} = 10^{-2}$ , $\text{ToIFCon} = 10^{-5}$ lower bound $10^{-10}$ was used for all variables
Parameters for design based on high-SINR approximation	
$\epsilon_t$	1
$R$	3 independent realizations
Algorithm for (23)	same as above
Simulation parameters	
Average arrival rates (% of $a_i^{k*}$ for $(i, k)$ )	96% for (1,2), (1,3), (2,1), (2,3), (3,1), (3,2); 90% for (1,4), (1,5), (4,1), (5,1); 77% for (1,6), (1,7); 73% for (1,8) 93% for (2,4), (4,2), (3,5), (5,3); 92% for (2,5), (3,4), (4,3), (5,2); 80% for (2,6), (2,7), (3,6), (6,2), (6,3), (7,2); 95% for (4,5), (5,4); 75% for (6,1), (7,1); similar for pairs not listed, by the symmetry of the network
Algorithm for (23)	same as above
$T$	6

power. This agrees with the fact that these nodes have less neighbors and are likely to handle a smaller share of the overall traffic. The symmetry of the network here, as well as the selected utilities  $V_i(p_i) = p_i^2$  with the relatively high weights, lead to approximately uniform average power consumption across terminals. More generally, different  $V_i(p_i)$  control the average power consumption in different ways.

Consider now comparing the previous results with those obtained when the high-SINR approximation is used for power allocation at the physical layer. Note that (23) with  $\alpha_{ij}^f = 1$  is solved at each step, but the true rate,  $\ln(1 + \gamma_{ij}^f)$ , is used for the subgradient [cf. (16c)]. The capacities  $c_{ij}$  of all links obtained with the algorithm based on SCALE, given in Fig. 4, are larger than the ones obtained using the high-SINR approximation after 5000 subgradient iterations. Specifically, Table VI shows the latter as percentage of the former; this percentage can be as low as 5%. This shows the suboptimality of the high-SINR approximation for the particular design.

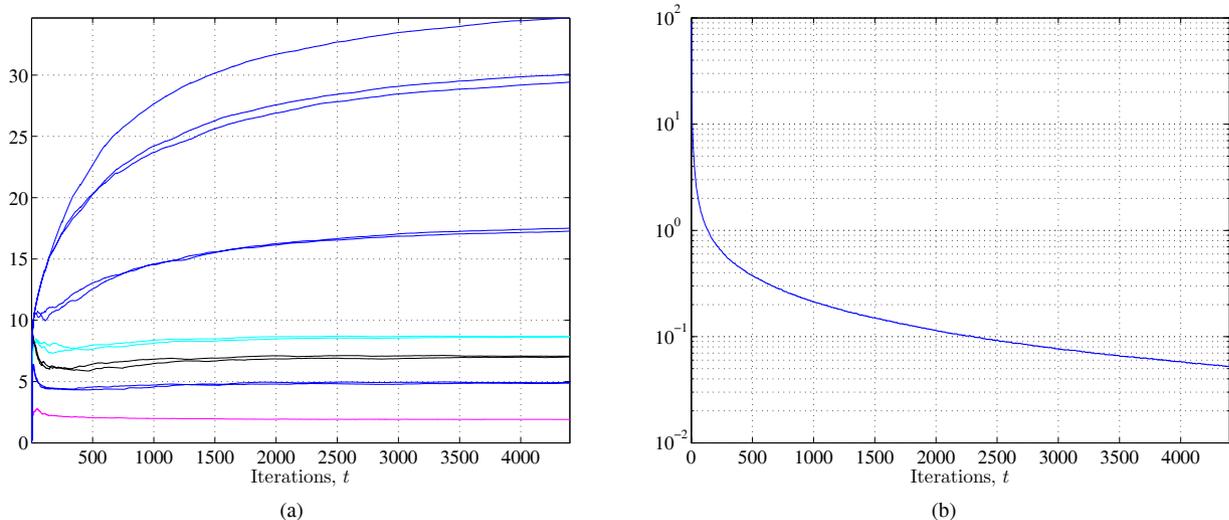


Fig. 2. (a) Convergence of running averages of Lagrange multipliers corresponding to node 1. Averages  $\bar{v}_1^k(t)$ ,  $k = 2, \dots, 8$  are shown in blue;  $\bar{\xi}_{1j}(t)$ ,  $j = 2, 3$  in black;  $\bar{\lambda}_{1j}(t)$ ,  $j = 2, 3$  in cyan; and  $\bar{\mu}_1(t)$  in magenta. (b) Quantity  $\left[ \sum_{i,k \in \mathcal{K}_i} \left( \left[ \bar{a}_i^k(t) - \sum_{j \in \mathcal{N}(i)} \bar{r}_{ij}^k(t) + \sum_{j \neq \text{dest}(k)} \bar{r}_{ji}^k(t) \right]^+ \right)^2 + \sum_{i,j \in \mathcal{N}(i)} \left( \left[ \sum_{k \in \mathcal{K}_i} \bar{r}_{ij}^k(t) - \bar{c}_{ij}(t) \right]^+ \right)^2 \right]^{1/2}$  capturing the violation of constraints (8b) and (8c).

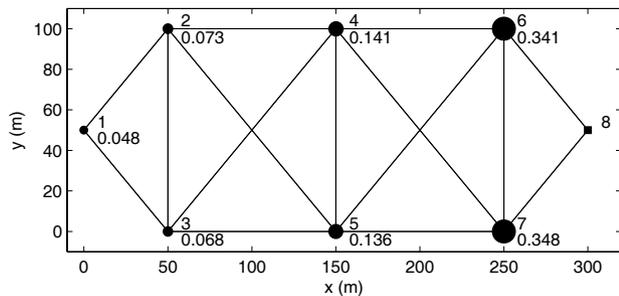


Fig. 3. A wireless network with its physical dimensions indicated; and incoming rates with node 8 as destination ( $a_i^8 + \sum_{j \in \mathcal{N}(i), j \neq 8} r_{ji}^8$ ), shown in bps/Hz below the node numbers. The area of each circle is proportional to these incoming rates.

TABLE VI

CERTAIN LINK CAPACITIES ACHIEVED WITH HIGH-SINR APPROXIMATION AS PERCENTAGE OF THOSE ACHIEVED WITH THE ALGORITHM BASED ON SCALE. THE FINAL VALUES OF  $\bar{c}_{ij}(t)$  (WEIGHTED RUNNING AVERAGES) ARE USED FOR THE COMPUTATION. VALUES ARE SIMILAR FOR LINKS SYMMETRIC TO THOSE SHOWN.

(1,2)	(4,2)	(4,3)	(4,5)
3.9	5.2	14.7	14.4
(2,1)	(2,3)	(2,4)	(2,5)
5.2	11.7	5.2	15.9

**B. Network simulation**

Simulation results from running the network with the strategy of Section III-C (online phase) are presented here. The network and its parameters are the same as those in Section V-A, where also quantities needed for the online phase—primal variables and Lagrange multipliers—have been obtained. In what follows, some conventions about the network operation used here are briefly described, and then, the simulation results are presented.

Each node maintains the queues indicated in Fig. 1, as

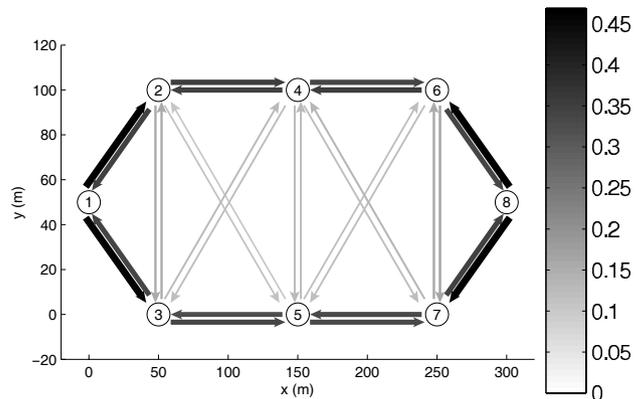


Fig. 4. Link capacities. The width of each arrow is proportional to the average capacity of the corresponding link. The color of each arrow is obtained by linearly mapping the capacities on a scale of gray (see bar on the right). Black corresponds to the largest capacity in the network. The numbers on the colorbar are in units of bps/Hz.

described in Section III-C. The numbers of bits entering or leaving the queues are conventionally assumed to take continuous values. This means equivalently that the packet sizes are very small as compared to the number of bits that the network control algorithm specifies to be moved at each slot.

The bits entering the physical layer queues in some slot, can be served in the same slot, if the queue becomes empty within that slot. This happens because these bits arrive from the network layer queues of the same node. The same convention is adopted for the bits exogenously generated at node  $i$  and entering the network layer queues. Now regarding the endogenous arrivals, the bits transmitted during the *previous* slot from the neighbors, can be served in the current time slot.

Hence, at every slot in each node, the exogenous arrivals of that slot and the endogenous arrivals of the previous slot (with

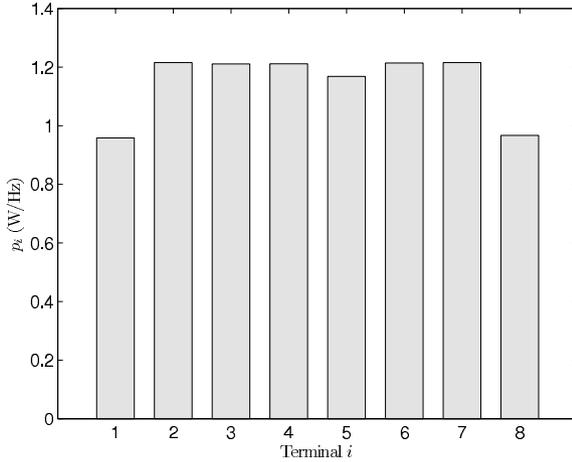


Fig. 5. Average power consumption at the network terminals.

destinations other than the present node) enter the network layer queues. Then, the bits leaving the queues are determined, according to  $\sum_{j \in \mathcal{N}(i)} \tilde{r}_{ij}^k(\ell)$ , where  $\tilde{r}_{ij}^k(\ell)$  are given by (19), and the final values of  $\tilde{r}_{ij}^k(t)$  (cf. Section V-A) are used in place of  $r_{ij}^k$ . If a network layer queue does not have so many bits, a random permutation of the neighbor indices is taken at the routing interface, in order to determine which physical layer queues will receive bits. Then, the number of bits leaving the physical layer queues and transmitted to the neighbors is determined by  $\tilde{c}_{ij}(\ell)$  (how  $\tilde{c}_{ij}(\ell)$  is computed will be explained shortly).

The exogenous arrivals are determined based on  $a_i^{k*}$ , obtained as final values of  $\tilde{a}_i^k(t)$ . In particular, the exogenous arrivals  $\tilde{a}_i^k(\ell)$  are i.i.d. across slots, nodes, and commodities, drawn from a uniform distribution. The mean of the distribution is a large fraction of the  $a_i^{k*}$ , as given in Table V, and the support is 20% of  $a_i^{k*}$ . The reason why the average arrival rates here are not exactly  $a_i^{k*}$ , is the following: the values of  $a_i^k$  and  $r_{ij}^k$  obtained in the previous subsection are not such that (8b) is exactly feasible; see Fig. 2b. More iterations of the subgradient method would suffice so that even higher arrival rates than those of Table V can be used.

The fading is slow, i.e., the channel vector keeps the same value,  $\mathbf{h}(\ell)$ , for the duration of the slot. The fading process is Rayleigh, i.i.d. across slots and independent across nodes, following the distribution given in Table V. The instantaneous power allocation for all links is determined based on  $\mathbf{h}(\ell)$ . The solution of (22) is used here, denoted as  $\mathbf{p}(\mathbf{h}(\ell); \ell)$ , where the final values of  $\bar{\Lambda}(s)$  are the Lagrange multipliers in (22), the same for all  $\ell$ . For the solution of (22), the algorithm based on SCALE is used, with parameters given in the last rows of Table V. Once  $\mathbf{p}(\mathbf{h}(\ell); \ell)$  is determined,  $\tilde{c}_{ij}(\ell)$  is determined by (20) using  $\mathbf{p}(\mathbf{h}(\ell); \ell)$  instead of  $\mathbf{p}^*(\mathbf{h}(\ell))$ .

Fig. 6 depicts the queue lengths for the queues maintained at node 1. Similar results hold for all other nodes. It is immediately observed that the queues are stable, i.e., the queue length does not grow without bound. In particular, note that the network layer queues get frequently empty.

Fig. 7a shows the long-term average power consumed at node 1, computed as follows. Let  $\tilde{p}_i(\ell) := \sum_{j,f} p_{ij}^f(\mathbf{h}(\ell); \ell)$  be the power consumed at node  $i$  at slot  $\ell$ . The long-term

average power consumed at node  $i$  is then  $\frac{1}{\ell} \sum_{l=1}^{\ell} \tilde{p}_i(l)$ . Fig. 7a shows that this is smaller than the average power computed by the subgradient method (final value of  $\bar{p}_i(t)$ , shown in Fig. 5). In other words, the network control strategy keeps the average power consumption approximately at its designed level. The discrepancy shown in Fig. 7a can be explained, if we recall that the Lagrange multipliers used for the computation of  $\mathbf{p}(\mathbf{h}(\ell); \ell)$  are not exactly optimal, but rather obtained from the subgradient method with *constant* stepsize, which ensures convergence within a ball of the optimal ones. Finally, similar comments hold for the long-term link capacities, shown in Fig. 7b.

## VI. CONCLUDING REMARKS

This paper dealt with algorithm development for cross-layer design of wireless networks in the presence of fading. The network terminals at the physical layer treat interference as noise. Based on the recently established optimality of dual decomposition, a subgradient descent algorithm was developed for the solution of the dual problem. Different power allocation options at the physical layer were developed, and seamlessly integrated into the subgradient method. Furthermore, optimal design variables of the network, that is, end-to-end rates, multicommodity flows, link capacities, and average powers, were obtained. A network control strategy utilizing the optimal solution of the cross-layer design problem was also outlined, along with simulation results.

The algorithmic framework developed in this paper attains the best operating point of the network offline, and then uses the optimal network variables for online network control. Interesting future directions include the development of fully online protocols for network control. Such protocols bypass the need for forming expectations offline, but estimate those on the fly during network operation using stochastic approximation tools.<sup>1</sup>

## APPENDIX

Here the proofs of Propositions 1 and 2 are given. Both rely on the following lemma, which collects intermediate results for the sequence  $\{\bar{\mathbf{x}}(s), \dot{\mathbf{p}}(\mathbf{h}; s)\}$  in (17) and (18).

**Lemma 1.** *Let ASI-AS4 hold and  $\Lambda^*$  denote an optimal Lagrange multiplier vector. Then for any  $s \geq 1$ , it holds that*

$$(i) \left\| \left[ \mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\dot{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \right]^+ \right\| \leq \frac{\|\Lambda(s)\|}{K_s} \quad (28)$$

$$(ii) f(\bar{\mathbf{x}}(s)) \geq D - \frac{\|\Lambda(0)\|^2}{2K_s} - \frac{G^2}{2K_s} \sum_{t=0}^{s-1} \epsilon_t^2 \quad (29)$$

$$(iii) f(\bar{\mathbf{x}}(s)) \leq D + \|\Lambda^*\| \left\| \left[ \mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\dot{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \right]^+ \right\|. \quad (30)$$

Part (i) of Lemma 1 provides an upper bound on the constraint violation caused by the sequence  $\{\bar{\mathbf{x}}(s), \dot{\mathbf{p}}(\mathbf{h}; s)\}$  as a function of the Lagrange multiplier iterates  $\Lambda(s)$ . Parts (ii) and (iii) give lower and upper bounds for the objective

<sup>1</sup>The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U. S. Government.

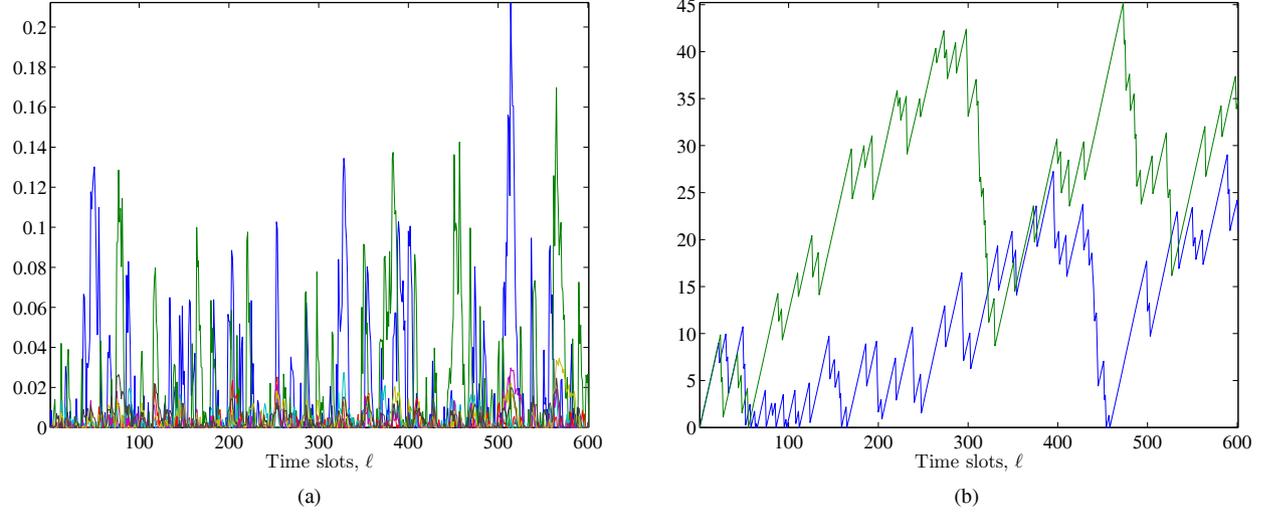


Fig. 6. Evolution of queue length for (a) network layer queues; and (b) physical layer queues, at node 1. The vertical axis is in bps/Hz, or equivalently, shows the queue lengths in bits, if the time slot duration is 1 sec and the channel bandwidth 1 Hz. In (a), there are seven trajectories, corresponding to the commodities entering node 1. In (b), there are two trajectories, corresponding to the links between node 1 and its neighbors.

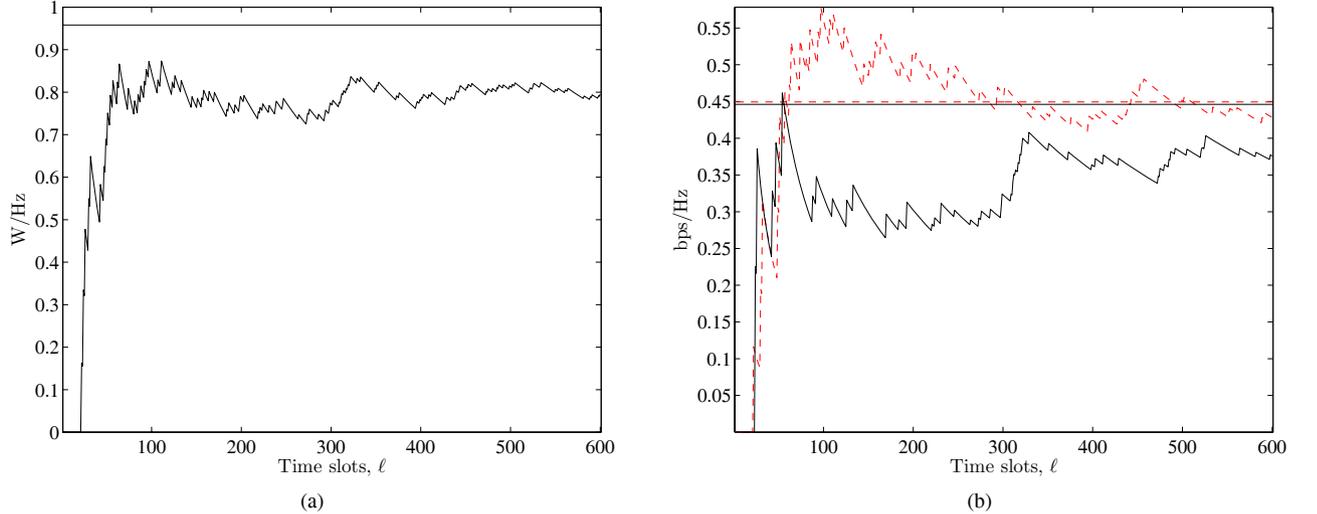


Fig. 7. (a) Long-term average power consumed at node 1. The horizontal line shows the optimal average power, given also in Fig. 5. The other trajectory corresponds to  $\frac{1}{\ell} \sum_{i=1}^{\ell} \tilde{p}_i(i)$ . (b) Long-term average capacity of links between node 1 and its neighbors. The horizontal lines show respectively the optimal link capacities of link (1,2) (dotted line) and link (1,3) (solid line); those are also depicted in Fig. 4. The other trajectories correspond to  $\frac{1}{\ell} \sum_{i=1}^{\ell} \tilde{c}_{12}(i)$  and  $\frac{1}{\ell} \sum_{i=1}^{\ell} \tilde{c}_{13}(i)$ .

function value at  $\bar{\mathbf{x}}(s)$ . The proof of Lemma 1 follows. Then the proofs of Propositions 1 and 2 are obtained by taking the limit  $s \rightarrow \infty$  in (28)–(30).

*Proof of Lemma 1:* (i) It holds for  $t \geq 0$  from the subgradient iteration [cf. (14b)] that

$$\begin{aligned} \Lambda(t+1) &= \left[ \Lambda(t) + \epsilon_t (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]) \right]^+ \\ &\geq \Lambda(t) + \epsilon_t (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]). \end{aligned} \quad (31)$$

Summing up (31) for  $t = 0, 1, \dots, s-1$ , and dividing by  $K_s$  yields

$$\begin{aligned} \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{g}_1(\mathbf{x}(t)) + \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})] \\ \leq \frac{\Lambda(s) - \Lambda(0)}{K_s} \stackrel{\Lambda(0) \geq 0}{\leq} \frac{\Lambda(s)}{K_s}. \end{aligned} \quad (32)$$

Due to the convexity of function  $\mathbf{g}_1(\cdot)$  and because  $\sum_{t=0}^{s-1} \epsilon_t / K_s = 1$ , it holds that  $\mathbf{g}_1(\bar{\mathbf{x}}(s)) \leq \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{g}_1(\mathbf{x}(t))$ . Using the latter and (18) into (32), it follows that  $\mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\bar{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \leq \Lambda(s) / K_s$ . Because  $\Lambda(s) \geq \mathbf{0}$  and  $[\cdot]^+ \geq \mathbf{0}$ , it is easy to deduce that

$$\left[ \mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\bar{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})] \right]^+ \leq \frac{\Lambda(s)}{K_s} \quad (33)$$

and then, (28) follows by taking the norm of both sides of (33).

(ii) Due to the concavity of  $f(\cdot)$ , it holds that  $f(\bar{\mathbf{x}}(s)) \geq \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t f(\mathbf{x}(t))$ . Adding and subtracting the same terms to the right-hand side of the latter, one arrives at

$$f(\bar{\mathbf{x}}(s)) \geq \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t f(\mathbf{x}(t))$$

$$\begin{aligned}
& -\frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]) \\
& + \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]). \quad (34)
\end{aligned}$$

Using  $f(\mathbf{x}(t)) - \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]) \equiv q(\mathbf{\Lambda}(t)) \geq D$  [cf. (14b), (10), (11)] into (34),

$$f(\bar{\mathbf{x}}(s)) \geq D + \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]). \quad (35)$$

Moreover, it follows from (14b) and the nonexpansive property of the projection [17, Prop. 2.2.1] that

$$\|\mathbf{\Lambda}(t+1)\|^2 \leq \|\mathbf{\Lambda}(t) + \epsilon_t (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})])\|^2 \quad (36)$$

and thus,

$$\|\mathbf{\Lambda}(t+1)\|^2 \leq \|\mathbf{\Lambda}(t)\|^2 + \epsilon_t^2 \|\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]\|^2 + 2\epsilon_t \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]). \quad (37)$$

Summing (37) for  $t = 0, 1, \dots, s-1$ , and dividing by  $2K_s$  yields

$$\begin{aligned}
& \frac{1}{K_s} \sum_{t=0}^{s-1} \epsilon_t \mathbf{\Lambda}(t)^T (\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]) \\
& \geq -\frac{1}{2K_s} \sum_{t=0}^{s-1} \epsilon_t^2 \|\mathbf{g}_1(\mathbf{x}(t)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\mathbf{p}(\mathbf{h}; t), \mathbf{h})]\|^2 \\
& \quad + \frac{\|\mathbf{\Lambda}(s)\|^2 - \|\mathbf{\Lambda}(0)\|^2}{2K_s}. \quad (38)
\end{aligned}$$

Combining (35) with (38) and introducing the bound on the subgradient norm, (29) follows.

(iii) Let  $\mathbf{\Lambda}^*$  be an optimal dual solution. (Such exists due to AS1–AS4.) It holds that

$$f(\bar{\mathbf{x}}(s)) = f(\bar{\mathbf{x}}(s)) - \mathbf{\Lambda}^{*T} (\mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\hat{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})]) + \mathbf{\Lambda}^{*T} (\mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\hat{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})]). \quad (39)$$

By the definitions of  $D$ ,  $\mathbf{\Lambda}^*$  [cf. (11)] and the dual function [cf. (10)], it holds that  $D = q(\mathbf{\Lambda}^*) = \max_{(\mathbf{x}, \mathbf{p}(\mathbf{h})) \in \mathcal{B}} L(\mathbf{x}, \mathbf{p}(\mathbf{h}), \mathbf{\Lambda}^*) \geq L(\bar{\mathbf{x}}, \hat{\mathbf{p}}(\mathbf{h}), \mathbf{\Lambda}^*)$ . Using the latter into (39), it follows that

$$f(\bar{\mathbf{x}}(s)) \leq D + \mathbf{\Lambda}^{*T} (\mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\hat{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})]). \quad (40)$$

Because  $\mathbf{\Lambda}^* \geq \mathbf{0}$ , and  $\mathbf{z} \leq [\mathbf{z}]^+$  for any vector  $\mathbf{z}$ , it holds from (40) that

$$f(\bar{\mathbf{x}}(s)) \leq D + \mathbf{\Lambda}^{*T} [\mathbf{g}_1(\bar{\mathbf{x}}(s)) + \mathbb{E}_{\mathbf{h}}[\mathbf{g}_2(\hat{\mathbf{p}}(\mathbf{h}; s), \mathbf{h})]]^+. \quad (41)$$

Applying the Cauchy-Schwartz inequality to the latter, (30) follows.  $\square$

*Proof of Proposition 1:* (i) It is shown in [10, Lemma 3] that the sequence of Lagrange multipliers obtained from the subgradient method with constant stepsize for convex problems is bounded under the Slater constraint qualification. It can be easily verified that the proof of this result is valid even for nonconvex problems, as long as there is zero duality gap and a dual optimal solution exists. But those are guaranteed under AS1–AS4. Hence, because  $\|\mathbf{\Lambda}(s)\|$  is bounded, part (i) of the proposition follows from (28) by taking the lim sup as  $s \rightarrow \infty$ .

(ii) Using  $\epsilon_t = \epsilon$  and  $D = P$  in (29), we infer that  $\liminf_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \geq P - \epsilon G^2/2$ . Moreover, the set of optimal dual solutions is bounded (see, e.g., [10] and references therein). It then follows from (30) with  $D = P$  and part (i) of the proposition that  $\limsup_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \leq P$ .  $\square$

*Proof of Proposition 2:* (i) Under the considered stepsize rule, sequence  $\{\mathbf{\Lambda}(s)\}$  converges to an optimal solution [17, Prop. 8.2.6]. Hence, it is bounded, and the result follows from (28).

(ii) Using  $D = P$  and  $\lim_{s \rightarrow \infty} (1/K_s) \sum_{t=0}^{s-1} \epsilon_t^2 = 0$  in (29), it follows that  $\liminf_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \geq P$ . The proof of  $\limsup_{s \rightarrow \infty} f(\bar{\mathbf{x}}(s)) \leq P$  is identical to that in Proposition 1.  $\square$

## REFERENCES

- [1] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.
- [2] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1–144, 2006.
- [3] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: a mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.
- [4] M. Chiang, "Geometric programming for communication systems," *Foundations and Trends in Commun. and Inf. Theory*, vol. 2, no. 1/2, pp. 1–156, 2005.
- [5] Y. Xi and E. M. Yeh, "Node-based optimal power control, routing, and congestion control in wireless networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4081–4106, Sep. 2008.
- [6] M. J. Neely, E. Modiano, and C. E. Rohrs, "Dynamic power allocation and routing for time-varying wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.
- [7] B. Radunovic and J.-Y. L. Boudec, "Power control is not required for wireless networks in the linear regime," in *Proc. 6th IEEE Int. Symp. World of Wireless Mobile and Multimedia Networks (WoWMoM)*, Taormina, Italy, June 2005, pp. 417–427.
- [8] A. Ribeiro and G. B. Giannakis, "Layer separability of wireless networks," in *Proc. 42nd Annu. Conf. Information Sciences and Systems*, Princeton, NJ, Mar. 2008, pp. 821–826.
- [9] —, "Optimal layered architectures of wireless networks," in *Proc. 42nd Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 2008, pp. 2147–2151.
- [10] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM J. Optimization*, vol. 19, no. 4, pp. 1757–1780, 2009.
- [11] T. Larrson, M. Patriksson, and A.-B. Strömberg, "Ergodic, primal convergence in dual subgradient schemes for convex programming," *Math. Programming*, vol. 86, no. 2, pp. 283–312, 1999.
- [12] S. Hayashi and Z.-Q. Luo, "Spectrum management for interference-limited multiuser communication systems," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 1153–1175, Mar. 2009.
- [13] J. Papandriopoulos and J. S. Evans, "SCALE: a low complexity distributed protocol for spectrum balancing in multiuser DSL networks," *IEEE Trans. Inf. Theory*, vol. 55, no. 8, pp. 3711–3724, Aug. 2009.
- [14] P. Tsiaflakis, M. Diehl, and M. Moonen, "Distributed spectrum management algorithms for multiuser DSL networks," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4825–4843, Oct. 2008.
- [15] A. Ribeiro and G. B. Giannakis, "Optimal FDMA over wireless mobile ad-hoc networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Las Vegas, NV, Mar. 2008, pp. 2765–2768.
- [16] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge University Press, 2004.
- [17] D. P. Bertsekas, A. Nedić, and A. Ozdaglar, *Convex Analysis and Optimization*. Belmont, MA: Athena Scientific, 2003.
- [18] P. Tsiaflakis, J. Vangorp, M. Moonen, and J. Verlinden, "A low complexity optimal spectrum balancing algorithm digital subscriber lines," *Signal Process.*, vol. 87, no. 7, pp. 1735–1753, July 2007.
- [19] N. Gatsis, A. Ribeiro, and G. B. Giannakis, "Cross-layer optimization of wireless fading ad-hoc networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, Apr. 2009, pp. 2353–2356.

- [20] Constrained nonlinear optimization. MATLAB Documentation. The MathWorks, Inc. [Online]. Available: <http://www.mathworks.com/access/helpdesk/help/toolbox/optim/ug/brnoxzl.html>



**Nikolaos Gatsis** received the Diploma degree in electrical and computer engineering from the University of Patras, Patras, Greece in 2005 with honors. Since September 2005, he has been working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN. His research interests include cross-layer designs, resource allocation, and signal processing for wireless networks.



**Alejandro Ribeiro** is an assistant professor at the Department of Electrical and Systems Engineering at the University of Pennsylvania, Philadelphia, where he started in 2008. He received the B.Sc. in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, in 1998. From 2003 to 2008 he was at the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis, where he received the M.Sc. and Ph.D. in electrical engineering. From 1998 to 2003 he was a member of the technical staff at Bellsouth Montevideo. His research interests lie in the areas of communication, signal processing, and networking. His current research focuses on wireless communications and networking, distributed signal processing, and wireless sensor networks. He is a Fulbright scholar and received best student paper awards at ICASSP 2005 and ICASSP 2006.



**Georgios B. Giannakis (F'97)** received his Diploma in Electrical Engr. from the Natl. 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 Univ. of Minnesota, where he now holds an ADC Chair in Wireless Telecommunications in the ECE Department, and serves as director of the Digital Technology Center.

His general interests span the areas of communications, networking and statistical signal processing - subjects on which he has published more than 300 journal papers, 500 conference papers, two edited books and two research monographs. Current research focuses on compressive sensing, cognitive radios, network coding, cross-layer designs, mobile ad hoc networks, wireless sensor and social networks. He is the (co-) inventor of 16 patents issued, and the (co-) recipient of seven 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 is a Fellow of EURASIP, has served the IEEE in a number of posts, and also as a Distinguished Lecturer for the IEEE-SP Society.