Network Planning in Wireless Ad Hoc Networks: A Cross-Layer Approach

Yunnan Wu, *Student Member, IEEE*, Philip A. Chou, *Fellow, IEEE*, Qian Zhang, *Senior Member, IEEE*, Kamal Jain, Wenwu Zhu, *Senior Member, IEEE*, and Sun-Yuan Kung, *Fellow, IEEE*

***Abstract*—In this paper, the network planning problem in wireless ad hoc networks is formulated as the problem of allocating physical and medium access layer resources or *supplies* to minimize a cost function, while fulfilling certain end-to-end communication *demands*, which are given as a collection of multicast sessions with desired transmission rates. We propose an iterative cross-layer optimization, which alternates between: 1) jointly optimizing the timesharing in the medium access layer and the sum of max of flows assignment in the network layer and 2) updating the operational states in the physical layer. We consider two objectives, minimizing aggregate congestion and minimizing power consumption, respectively, corresponding to operating in a bandwidth-limited regime and in an energy-limited regime. The end result is a set of achievable tradeoffs between throughput and energy efficiency, in a given wireless network with a given traffic pattern. We evaluate our approach quantitatively by simulations of community wireless networks and compare with designs that decouple the layers. We demonstrate that significant performance advantages can be achieved by adopting a full-fledged cross-layer optimization. Furthermore, we observe that optimized solutions generally profit from network coding, physical-layer broadcasting, and traffic-dependent physical states.**

***Index Terms*—Capacity, cross-layer design, interference, multicast, network coding, wireless ad hoc networks.**

I. Introduction

NETWORK planning is concerned with the cost-effective deployment of a communication infrastructure to provide adequate coverage, throughput, and quality for end user services. In this paper, we investigate dynamic service provisioning in wireless ad hoc networks. Abstractly, the network planning problem considered in this paper is the problem of allocating physical and medium access layer resources or *supplies* to minimize a cost function, while fulfilling certain transport layer communication *demands*.

We model the demands in a set V_0 of network nodes as a collection of multicast sessions in the form $\mathcal{S}_m \equiv \langle s^m, T^m, r^m \rangle, s^m \in V_0, T^m \subseteq V_0, r^m > 0, m = 1, \ldots, M$, where in each multicast session $\mathcal{S}_m$, a source s^m transmits common information to a set of destinations T^m at rate r^m (bits per second). We model the allocation of supplies as a timesharing within a collection of all possible physical layer states.

This formulation necessitates an interaction across the network protocol stack. However, the unique physical and link layer characteristics of wireless communication render cross-layer optimization especially challenging. The key to practical cross-layer optimization is to find an appropriate abstraction of each layer. To form these abstractions, we introduce the notion of a *capacity graph*. A *capacity graph* can be regarded as a network of lossless channels, each capable of carrying information between neighboring nodes at a certain rate. We denote a capacity graph by a tuple $G = (V, E, c)$, where V and E are a set of vertices and edges respectively and c is function assigning to each edge $vw \in E$ a nonnegative edge capacity $c(vw)$. A capacity graph is used as the interface between the lower and upper layers: the physical and link layers provide bit-rate resources in the form of a *supported capacity graph*, taking into account interference and other wireless channel characteristics, and the network layer consumes these bit-rate resources to deliver information from the source to the destinations, without being concerned with how the capacity graph is realized in the physical and link layers.

The physical layer can be abstracted as a set of elementary capacity graphs. An *elementary capacity graph* is a capacity graph that represents a physical layer state, correponding to an arrangement of concurrently active links between neighbors. The set of elementary capacity graphs depend on the wireless channel characteristics, most importantly, the interference among concurrent links, as well as the communication schemes, e.g., power control, modulation, channel encoding.

The link layer (or more precisely, the medium access layer) can be abstracted by a set of all possible timesharings between the elementary capacity graphs. A timesharing or convex combination, of elementary capacity graphs is itself a supported capacity graph, representing a specific allocation of bit-rate resources.

The network layer transforms the end-to-end traffic demand into a link-by-link traffic demand compatible with a supported capacity graph G. For this purpose, we assume that network coding can be used for the multicast sessions. Network coding generalizes the traditional routing paradigm in which interior nodes in the network can only replicate and forward information received, by allowing nodes to perform arbitrary operations on the information received to generate the output. In their pioneering theoretical work on network coding, Ahlswede *et al.* [1]

Manuscript received November 6, 2003; revised August 8, 2004. The work of Y. Wu was done while he was a visiting student with Microsoft Research, Asia, Beijing, China, and during his summer internship at Microsoft Research, Redmond, WA, in 2003.

Y. Wu and S.-Y. Kung are with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: yunnanwu@princeton.edu; kung@princeton.edu).

P. A. Chou and K. Jain are with the Microsoft Corporation, Redmond, WA 98052-6399 USA (e-mail: pachou@microsoft.com; kamalj@microsoft.com).

Q. Zhang and W. Zhu are with Microsoft Research, Beijing 100080 China (e-mail: qianz@microsoft.com; wwzhu@microsoft.com.

Digital Object Identifier 10.1109/JSAC.2004.837362

0733-8716/$20.00 © 2005 IEEE

have demonstrated that it is in general suboptimal to restrict the intermediate nodes to perform only routing. They show that the *multicast capacity*, which is defined as the maximum rate that a source can communicate common information to a set of destinations, can be achieved with network coding. Using network coding, the capacity subgraph U of G given by a *sum of max of flows* is sufficient to accommodate multiple multicast sessions, corresponding to a "separation" solution where the sessions are first allocated disjoint shares of bit-rate resources, and then each session communicates using its share. We show that with this separation approach, the set of all sum of max of flows that can accommodate a given load can be characterized by a linear system of equalities and inequalities.

Degrees of freedom exist in the lower layers when choosing a convex combination of elementary capacity graphs, and in the upper layers when choosing the sum of max of flows. As we will show, these can be jointly optimized using a linear program. However, computational difficulties can arise since the number of elementary capacity graphs is generally exponential in the size of the network. To alleviate these difficulties, we propose an iterative optimization procedure, which alternates between: 1) heuristically selecting a manageable collection of elementary capacity graphs and 2) jointly optimizing (using the linear program) the allocation of supplies and the flow assignment. In essence, feedback is now introduced, potentially allowing the network supplies to more efficiently match the traffic demand.

The rest of this paper is organized as follows. From the perspective of layering, Section II summarizes the physical layer, Section III summarizes the network layer, and Section IV is devoted to the iterative cross-layer joint optimization, including determination of timesharing in the link layer. Simulation results are presented in Section V. A review of related work is given in Section VI.

II. Capacity Graphs for Wireless Networks

We use the name *physical state* to represent a "snapshot" of all the nodes in the physical layer, such as which nodes are transmitting and what transmitting powers are being used. A physical state can support a set of concurrent *links* in the physical layer. In a wireless network, a single transmission by a certain transmitter may result in multiple nodes recovering the transmitted signals. This physical-layer broadcast property was called the *wireless multicast advantage* in [2]. To model this effect, we consider the general case of point-to-multipoint links. Let V_0 denote the set of nodes in the network. A *link* can be described as $u \xrightarrow{c} Y_u$, where $u \in V_0$ is the transmitter, $Y_u \subseteq V_0$ is the set of associated receivers, and c is the associated bit rate in a reliable communication.

Each set of concurrent links supported by a certain physical state corresponds to an *elementary capacity graph* (ECG). By timesharing among different physical states, it is possible to achieve any convex combination of the ECGs. That is, if λ_k is the relative share of time for the ECG $G_k = (V_k, E_k, c_k)$, then it is possible to achieve on average the capacity graph $G = (\cup_k V_k, \cup_k E_k, \sum_k \lambda_k c_k)$, where the edge capacities c_k are each extended to $\cup_k E_k$ in the obvious way. We denote such combinations $G = \sum_k \lambda_k G_k$ in this paper. The capacity graphs resulting from timesharing the ECGs will be referred to as *supported (composite) capacity graphs*. Hence, a distinguishing feature of wireless networks is the characterization of supported capacity graphs as convex combinations of ECGs. This can be stated mathematically as

$$\mathcal{G}_0 = \left\{ G \,\middle|\, G = \sum_k \lambda_k G_k, \sum_k \lambda_k \leq 1, \lambda_k \geq 0 \forall k, G_k \in \mathcal{B}_0 \right\}$$

where $\mathcal{G}_0$ is the entire set of capacity graphs supported/spanned by the set of all feasible ECGs $\mathcal{B}_0$. It may take an enormous number of ECGs to completely cover the entire set of capacity graphs. From a practical standpoint, we are interested in identifying a few "promising" ECGs that provide a reasonably good span for a specific application. With a finite set of ECG's $\mathcal{B} \subseteq \mathcal{B}_0$, the convex set of supported capacity graphs becomes

$$\mathcal{G}(\mathcal{B}) = \left\{ G \,\middle|\, G = \sum_k \lambda_k G_k, \sum_k \lambda_k \leq 1, \lambda_k \geq 0 \forall k, G_k \in \mathcal{B} \right\}$$

where the dependence on $\mathcal{B}$ is explicitly shown.

A. Elementary Capacity Graphs

We now state some modeling assumptions and discuss the elementary capacity graphs under these assumptions.

Consider a set of concurrent links (abbreviated as a *link set*) $\mathcal{E} = \{x \xrightarrow{c} Y_x, Y_x \subseteq V_0, x \in X\}$, where X is the set of transmitters and Y_x is the set of nodes receiving common information from x. A given link set $\mathcal{E}$ is said to be *feasible* if there exists a physical state supporting it. We now show how to determine the feasibility of a given link set $\mathcal{E}$ and if it is feasible, how to find a physical state supporting it. As a modeling simplification, we assume that if a physical state supports a link $x \xrightarrow{c} Y_x$, then the communication rate c is a unit rate. Thus, we may drop c from the notation of a link and write $x \to Y_x$ instead.

For each transmitter $x \in X$, let p_x denote its transmitting power. Denote the path loss factor from node i to node j by a_{ij} and the noise variance by σ^2. Then, the signal-to-interference and noise ratio (SINR) at a node $y \in Y_x$ receiving information from $x \in X$ can be expressed as

$$\mathrm{SINR}_{xy} = \frac{a_{xy} p_x}{\sum_{x' \in X, x' \neq x} a_{x'y} p_{x'} + \sigma^2}. \tag{1}$$

Link set $\mathcal{E}$ is said to be supported by the physical state with transmitting powers $\{p_x\}$ if and only if

$$\mathrm{SINR}_{xy} \geq \gamma, \quad \forall y \in Y_x, \quad \forall x \in X \tag{2}$$

where γ is a fixed threshold.

We further assume that the transmission power of each node can be flexibly adjusted in the system. The following linear program minimizes the total transmission power over variables $\{p_x\}$ subject to the SINR requirements

$$\min \sum_{x \in X} p_x$$
$$\text{subject to:} \quad \mathrm{SINR}_{xy} \geq \gamma, \quad \forall y \in Y_x, \quad \forall x \in X.$$

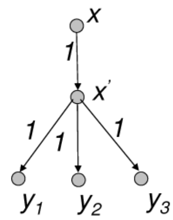


Fig. 1. Modeling a physical-layer broadcast link by introducing a virtual vertex that has an incoming edge from the transmitter and outgoing edges with the receivers. Please note the terminology used: a *link* is represented in the capacity graph by several *edges*.

With power control, a link set $\mathcal{E}$ is said to be feasible if the corresponding linear program for power control has a feasible solution. If a link set is feasible, a physical state supporting it can be constructed from the solution of the linear program.

A feasible link set $\mathcal{E} = \{x \to Y_x, Y_x \subseteq V_0, x \in X\}$ corresponds to an ECG in a natural way: for each link $x \to Y_x$, the ECG contains an edge with capacity 1 from x to a new virtual vertex x', and an edge with capacity 1 from x' to each node y in Y_x, as illustrated in Fig. 1. More examples can be found in Fig. 7. The virtual vertex plays the role of an artificial bottleneck which constrains the rate of new information going out of the transmitter. Since these virtual vertices do not physically exist, they can only perform routing, instead of arbitrary network coding. In other words, for a virtual vertex x', information flowing on its outgoing edges are not allowed to be general functions of information flowing on the incoming edges. Fortunately, the following Theorem 1 shows that there is no need to perform network coding on these virtual vertices, as far as throughput is concerned.

Theorem 1: Suppose in a capacity graph $G = (V, E, c)$, the vertex set V is composed of two disjoint sets, V_0 and V_1, and $s \in V_0, T \subseteq V_0$. Each vertex $u' \in V_1$ has only one incoming edge $uu' \in E, u \in V_0$ and multiple outgoing edges $u'v_j, v_j \in V_0, j = 1, \ldots, n$, and $c(uu') = c(u'v_j), j = 1, \ldots, n$. The maximum rate for multicasting from source s to destinations T can still be achieved even if each vertex $u' \in V_1$ can only perform routing instead of arbitrary network coding.

Proof: See [3]. ■

B. Initialization of Elementary Capacity Graphs

Suppose K ECG's $G_k, k = 1, \ldots, K$ are to be generated. For each G_k, a simple random packing strategy is adopted here. Fix a radius R which is a typical value for the communication range and a larger radius R_i as the interference range. In each step, we randomly select a node as the transmitter, draw two circles originated from it with radius R and R_i, respectively. If some existing receivers associated with other transmitters fall into the larger interference circle, then this node does not qualify as a transmitter. If the node passes this test, then its associated receivers are chosen to be those nodes that are inside the communication circle with radius R and outside any other existing interference circles. Next, we include this link into the linear program for power control. If a feasible solution exists, then the link is packed. Otherwise, the steps above are repeated until no links can be further packed.

III. Multisession Multicast

Let us temporarily assume that one capacity graph $G = (V, E, c)$ has been given exogenously as the provision of bit-rate resources. This section addresses the following question: Given the end-to-end demands as multiple multicast sessions, how should one assign the traffic on each edge and, hence, coordinate the information flow in the network?

A. Max of Flows

Given a capacity graph $G = (V, E, c)$ and two vertices $s, t \in V$, an *s-t-flow* is a nonnegative-valued function f on edges satisfying the following constraints:

$$0 \le f(vw) \le c(vw), \quad \forall vw \in E,$$
$$\sum_{w \in V: vw \in E} f(vw) - \sum_{w \in V: wv \in E} f(wv) = 0, \quad \forall v \in V \backslash \{s, t\}$$

$$0 \le f(vw) \le c(vw), \quad \forall vw \in E, \sum_{w \in V: vw \in E} f(vw)$$
$$- \sum_{w \in V: wv \in E} f(wv) = 0, \quad \forall v \in V \backslash \{s, t\}$$

and the *flow value* is

$$|f| \equiv \sum_{w \in V: sw \in E} f(sw) - \sum_{w \in V: ws \in E} f(ws).$$

We use the notation $G' = (V', E', c') \preceq G = (V, E, c)$ to indicate that (V', E') is a subgraph of (V, E), and $c'(vw) \le c(vw), \forall vw \in E' \subseteq E$. Given a set of flows $\{f_t, t \in T\}$ on G with f_t being an s-t-flow, the *max of flows* is a subgraph $G_g = (V, E, g) \preceq G$, where the function g is

$$g(vw) \equiv \max_{t \in T} f_t(vw), \quad \forall vw \in E. \tag{3}$$

The subgraph G_g has a multicast capacity greater than or equal to $\min_{t \in T} |f_t|$ because the multicast capacity is equal to the minimum of the maximum flow values from s to each $t \in T$ [1]. Thus, *any* max of flows (assuming $|f_t| \ge r, \forall t \in T$) is sufficient to provide a rate r with network coding. Conversely, if a subgraph $G' = (V, E, g') \preceq G$ can provide rate r, then it is easy to see that $\exists G'' = (V, E, g'') \preceq G'$ where G'' is a max of flows providing the same rate r. This shows the necessity of a max of flows.

B. Sum of Max of Flows

The problem of multisession multicast, where multiple multicast sessions need to be communicated over a network, remains open. Yeung [4] has shown that it is generally suboptimal to simply superimpose the different streams of information, even if the streams are independent. In other words, it is suboptimal to separate the communications of different sessions. Conversely, cross-session network coding is in general needed to achieve optimality.

Albeit suboptimal, superimposing communications for different streams is still a viable approach to this problem, especially from a practical standpoint. The following Theorem 2 shows that different ways of superimposing the multiple sessions can be conveniently characterized by a linear system of equalities and inequalities.

Theorem 2 (Sum of Max of Flows): Consider a capacity graph $G = (V, E, c)$ and a collection of multicast sessions $\mathcal{S}_m = \langle s^m, T^m, r^m \rangle, m = 1, \ldots, M$ as the end-to-end traffic demands. The sessions can be accommodated in G by allocating disjoint shares of bit-rate resources to the sessions and letting each session communicate using its share, if and only if the following system of linear equalities and inequalities has a feasible solution

$$\sum_{m=1}^{M} g^m(vw) \le c(vw), \quad \forall vw \in E \tag{4}$$

$$0 \le f_t^m(vw) \le g^m(vw), \quad \forall vw \in E, \quad \forall t \in T^m, \quad \forall m \tag{5}$$

$$\sum_{w \in V: vw \in E} f_t^m(vw) - \sum_{w \in V: wv \in E} f_t^m(wv) = 0, \quad \forall v \in V \backslash \{s^m, t\}, \quad \forall t \in T^m, \quad \forall m \tag{6}$$

$$\sum_{w \in V: s^m w \in E} f_t^m(s^m w) - \sum_{w \in V: ws^m \in E} f_t^m(ws^m) = r^m, \quad \forall t \in T^m, \quad \forall m. \tag{7}$$

Constraints (5)–(7) state that a capacity subgraph (V, E, g^m) with $g^m \ge \max_{t \in T^m} f_t^m(vw)$ is used by the m-th session to achieve multicast rate r^m. Constraint (4) states that for each edge, the sum of the rates consumed by the sessions has to be less than or equal to the total available rate.

Theorem 2 is essentially a result of the sufficiency and necessity properties of max of flows discussed in the previous subsection, based on which a proof can be established.

For a feasible flow assignment, we use the name *sum of max of flows* (or, *multicommodity max of flows*) to denote a capacity graph, where the capacity on edge vw is specified as $\sum_{m=1}^{M} \max_{t \in T^m} f_t^m(vw)$. Accordingly, we call the solution space of (4)–(7) *the sum of max of flows polyhedron*. When a cost function is specified, we call a solution of (4)–(7) that minimizes the cost function a *minimum-cost sum of max of flows*. When referring to a sum of max of flows, the symbol U will often be used in the sequel.

IV. Cross-Layer Iterative Optimization

As discussed in Sections I and II, the physical layer provides to the upper layers a convex set of capacity graphs supported by a finite set or basis of elementary capacity graphs $\mathcal{B} = \{G_k\}$. The set of *all* possible ECG's $\mathcal{B}_0$ grows exponentially in the number of network nodes. Indeed, plenty of degrees of freedom exist in these lower layers, and the choice of $\mathcal{B}$ is typically far from unique. Similarly, as discussed in Section III, for any fixed G, the traffic demands in the transport layer may be accommodated by a sum of max of flows $U \preceq G$. Since flows are generally far from unique, any sum of max of flows is generally far from unique. The objective is to pick a basis $\mathcal{B} = \{G_k\}$, corresponding timesharing coefficients $\{\lambda_k\}$ and a sum of max of flows $U \preceq \sum_k \lambda_k G_k$, such that U satisfies the traffic demands $\{\langle s^m, T^m, r^m \rangle\}$ and the capacity graph $G = \sum_k \lambda_k G_k$, which supplies U, minimizes a desired cost function.

In this section, we show how to select $\mathcal{B}$, $\{\lambda_k\}$ and U jointly, using an iterative descent optimization algorithm that minimizes a desired cost function at each step. With $\mathcal{B}$ fixed, the optimization over $\{\lambda_k\}$ and U turns out to be a linear program. Given feedback on the traffic loads U that are to be supported, a heuristic scheme is proposed that updates the basis of ECG's $\mathcal{B}$. The proposed optimization alternates between these two steps until convergence.

A. Cost Functions

1) Bandwidth-Limited Regime: When system bandwidth is scarce, it is of interest to find the maximum supportable traffic load given the bandwidth constraints. Specifically, given the demands $\langle s^m, T^m, r^m \rangle, m = 1, \ldots, M$, it is of interest to find the largest possible rate-scaling factor α that when multiplied by the rates $r^m, m = 1, \ldots, M$, still leads to a feasible sum of max of flows solution.

Given a set of ECG's $\mathcal{B}$, we use V to denote the enlarged vertex set including the physical nodes in V_0 and the introduced virtual vertices, and E to denote the union of the edges. With $\mathcal{B}$ fixed, maximizing the rate scaling factor results in the following linear program, where the timesharing coefficients $\{\lambda_k\}$, as well as the flows $\{f_t^m(vw)\}$ are treated as variables

$$\max \alpha, \quad \text{subject to:} \tag{8}$$

$$\sum_{m=1}^{M} g^m(vw) \le \sum_k \lambda_k c_k(vw), \quad \forall vw \in E \tag{9}$$

$$\sum_k \lambda_k \le 1 \tag{10}$$

$$\lambda_k \ge 0, \quad \forall k \tag{11}$$

$$0 \le f_t^m(vw) \le g^m(vw), \quad \forall vw \in E, \quad \forall t \in T^m, \quad \forall m \tag{12}$$

$$\sum_{w \in V: vw \in E} f_t^m(vw) - \sum_{w \in V: wv \in E} f_t^m(wv) = 0, \quad \forall v \in V \backslash \{s^m, t\}, \quad \forall t \in T^m, \quad \forall m \tag{13}$$

$$\sum_{w \in V: s^m w \in E} f_t^m(s^m w) - \sum_{w \in V: ws^m \in E} f_t^m(ws^m) = \alpha r^m, \quad \forall t \in T^m, \quad \forall m. \tag{14}$$

The main difference between this linear program and (4)–(7) is that the bit-rate supply $c(vw)$ in (5) is now replaced by $\sum_k \lambda_k c_k(vw)$ in (9), which can be adjusted by changing $\{\lambda_k\}$. Note that scaling the demands can be converted into scaling the supplies. Consequently, after some mathematical manipulations, it can be shown that the reciprocal of the maximum rate scaling factor admits a more explicit formulation:

$$1/\alpha_{\max} = \min \sum_k \lambda_k, \quad \text{subject to:} \tag{15}$$

$$\sum_{m=1}^{M} g^m(vw) \le \sum_k \lambda_k c_k(vw), \quad \forall vw \in E \tag{16}$$

$$\lambda_k \ge 0, \quad \forall k \tag{17}$$

$$0 \le f_t^m(vw) \le g^m(vw), \quad \forall vw \in E, \quad \forall t \in T^m, \quad \forall m \tag{18}$$

$$\sum_{w\in V:vw\in E} f_t^m(vw) - \sum_{w\in V:wv\in E} f_t^m(wv) = 0, \quad \forall v \in V\backslash\{s^m,t\}, \quad \forall t\in T^m, \quad \forall m \tag{19}$$

$$\sum_{w\in V:s^m w\in E} f_t^m(s^m w) - \sum_{w\in V:ws^m\in E} f_t^m(ws^m) = r^m, \quad \forall t\in T^m, \quad \forall m. \tag{20}$$

Consequently, for a feasible solution to the system of inequality constraints (16)–(20), we can regard the quantity $\sum_k \lambda_k$ as the "cost" of the *generalized* allocation of supplies $G = \sum_k \lambda_k G_k$. It is termed a generalized allocation since the constraint $\sum_k \lambda_k \leq 1$ is not enforced. If $\sum_k \lambda_k$ is indeed less than one, then G corresponds to a feasible arrangement in the physical and link layers. In this case, if the traffic demands $r^m, m = 1,\dots,M$ are gradually increased at a common rate, G can be proportionally scaled to accommodate the traffic, until it becomes infeasible. When the traffic is so heavy that the network is operating at a point near infeasibility, "congestion" is said to occur. Thus, this quantity $\sum_k \lambda_k$ bears an interpretation as a congestion measure of the network.

Definition 1: Aggregate Congestion Measure: In a wireless network, consider the end-to-end traffic demands $\langle s^m, T^m, r^m\rangle, m = 1,\dots,M$. The *aggregate congestion measure*, which quantifies the congestion induced to the network by the given traffic demands, is defined as $\sum_k \lambda_k$ for a generalized allocation of supplies $\sum_k \lambda_k G_k$ dominating a sum of max of flows and, hence, satisfying the traffic demands.

2) Energy-Limited Regime: To minimize the required energy subject to traffic contraints, we can choose total power consumption as the cost function for the linear optimization. Specifically, we substitute for (15) the objective

$$\min \sum_k \lambda_k P(G_k) \tag{21}$$

where $P(G_k)$ is the power consumption for G_k and add the constraint $\sum_k \lambda_k \leq 1$.

3) Refinement With a Secondary Linear Program: After the above optimization is done, we can further improve the quality of the solution by removing redundancies (e.g., cycles) in the flow definitions. A secondary linear program can be used for this purpose, where the objective (15) is replaced by

$$\min \sum_{m=1}^{M} \sum_{vw\in E} g^m(vw) \tag{22}$$

where the timesharing coefficients $\{\lambda_k\}$ are now treated as known constants with values substituted from the solution of the primary linear program (minimizing congestion or power). This secondary linear program also has the effect of ensuring that $g^m(vw) = \max_{t\in T^m} f_t^m(vw)$. This refinement will not further improve the objective of the primary linear program directly. However, improvement may be achieved in conjunction with the heuristic refinement of the basis of ECGs.

4) Summary: In the above, we have described two example definitions of optimization objectives. Many other possibilities exist in defining an appropriate cost function. For example, we can minimize the maximum power consumption over all nodes so as to reflect some load balancing/fairness considerations. It can be shown that the revised formulation incorporating min–max type performance criteria would remain a linear program.

B. Optimization by Iterative Descent

Up to this point, we have assumed that the basis of elementary capacity graphs $\mathcal{B} = \{G_k\}$ is fixed and given. For any $\mathcal{B}$, we have shown how to solve for the timesharing coefficients $\{\lambda_k\}$, as well as the edge loads $\{f_t^m(vw)\}$ such that the sum of max of flows $U = (V, E, \sum_m \max_{t\in T^m} f_t^m) \preceq G = \sum_k \lambda_k G_k$ satisfies the given demands while minimizing a desired cost function. However, the number of physical states and, hence, the number of possible ECG's $\{G_k\}$ are exponential in the number of nodes in the network. A computational difficulty would arise if $\mathcal{B}$ contained all such possible ECGs. Ideally, it is desirable to have a small but "efficient" basis of ECGs. However, the ultimate efficiency measure is the cost of fulfilling the given demands or dominating the sum of max of flows U, which introduces loops in reasoning. Rather than preparing a large basis without any knowledge of demands, our idea is to introduce feedback into the optimizations to gradually improve the efficiency of the basis. To start the computations, one can initialize $\mathcal{B}$ without any knowledge of demand and run the linear program to find U. However, it cannot be expected that such a choice of $\mathcal{B}$ will provide an efficient match between U and $\sum_k \lambda_k G_k$. Some of the bit-rate supplies may be unbalanced and redundant. Removal of these redundancies can reduce the cost function, e.g., power consumption, by shutting off unnecessary transmitters and reducing interference. Thus, it is possible to refine $\mathcal{B}$ to reduce the cost function for a given U. This leads to an iterative descent algorithm that alternates between optimizing the λ_ks and $f_t^m(vw)$s and updating $\mathcal{B}$. This procedure is described as follows.

- Step 1) Initialize the set of ECG's $\mathcal{B} = \{G_k\}$.
- Step 2) Given $\mathcal{B}$, solve the linear programs discussed in Section IV-A to get the coefficients $\{\lambda_k\}$ and the minimum-cost sum of max of flows $U = (V, E, \sum_m \max_{t\in T^m} f_t^m)$.
- Step 3) Given $\{\lambda_k\}$ and U, heuristically update $\mathcal{B}$ to reduce the cost function.
- Step 4) Repeat Steps 2) and 3).

Since this is an iterative descent algorithm, convergence is guaranteed if the cost function is nonnegative. The complexity of the algorithm depends on the size of the linear program (and, thus, the size of $\mathcal{B}$), the complexity of the heuristic operations, and the convergence speed of the iterations. Because of the heuristic operations involved, it is difficult to obtain a theoretical analysis regarding the convergence speed. The algorithm has been empirically observed to converge after a small number of iterations, on the example networks in the simulations section.

C. Heuristic Refinement of Elementary Capacity Graphs

In this section, we investigate heuristic refinement of the set of elementary capacity graphs $\mathcal{B}$, with the objective of lowering the cost of fulfilling the link-layer demands U. In the simulations section, we will walk through the procedures to be described below, in a small example network consisting of nine nodes. For ease of understanding, it might be suggested that the rest of this section be read together with Section V-A.

From the linear programs discussed in Section IV-A, a decomposition of G is readily available

$$U \preceq G = \sum_k \lambda_k G_k. \tag{23}$$

First of all, we can distribute the traffic load of U to the edges in G_k. This step essentially removes redundancies in the set of ECGs. More specifically, first choose a large enough integer Q (e.g., 200) and round $g^m(vw)Q, vw \in E$, and $\lambda_k Q, k = 1, \ldots, K$ into integers $\tilde{g}^m(vw), vw \in E$, and $\tilde{\lambda}_k, k = 1, \ldots, K$. Next, we initialize the new set of ECGs with $\tilde{\lambda}_k$ copies of G_k for $k = 1, \ldots, K$. Recall that one virtual vertex is introduced for each point-to-multipoint link. For a link $u \to \{v_j, j = 1, \ldots, n\}$ in ECG G_k, suppose the corresponding virtual vertex is u'. Then, we distribute the traffic load $\tilde{g}^m(uu'), m = 1, \ldots, M$ to those elementary capacity graphs. For each session m, these branching edges $u'v_j$ with $\tilde{g}^m(u'v_j) = 0$ can be pruned from the link and $\tilde{g}^m(uu')$ copies of the link (after pruning) are needed, out of the $\tilde{\lambda}_k$ copies. Denote the resulting set of ECGs by $\mathcal{B}'$.

1) Deflation: The *deflation* operation is invoked when the aggregate congestion measure is adopted as the minimization objective. The basic observation is that the "space" previously occupied by the redundant links (and/or branches of links) may be used to "deflate" $\mathcal{B}'$ and, hence, leave more room for new traffic. Since each ECG in $\mathcal{B}'$ contributes approximately $1/Q$ to $\sum_k \lambda_k$, if we can pack all its constituent links into other ECGs, the aggregate congestion measure will be reduced by that amount. Note that this opportunity of reducing the aggregate congestion measure could not have been discovered through the linear program alone, since an ECG is treated as a basic operating unit in the optimization.

Now, we describe a deflation procedure, which heuristically reduces the number of required ECGs. Define a primitive action as moving one link ϵ from a source ECG src to a destination ECG dst, which we shall denote with $\mathrm{MOVE}(\epsilon, src, dst)$. Let us temporarily assume we have a function $\rho_1(\cdot)$ which evaluates the crowdedness factor for a given ECG. This crowdedness factor is a physical layer measure of congestion in each ECG. It is introduced as an immediate metric used to guide the heuristic operations toward minimizing the aggregate congestion measure. Then, accompanying the primitive operation $\mathrm{MOVE}(\epsilon, src, dst)$, the crowdedness factor of the source will decrease from $\rho_1(src)$ to $\rho_1(src - \epsilon)$, while the crowdedness factor of the destination will increase from $\rho_1(dst)$ to $\rho_1(dst + \epsilon)$. If the sum of the crowdedness factors is used as the criterion, the net reduction in the sum crowdedness factor is $\rho_1(src) + \rho_1(dst) - \rho_1(dst + \epsilon) - \rho_1(src - \epsilon)$. We set the maximum number of trials to be performed, where in each trial we select a source ECG src and try to move all the links in it to other ECGs. The source ECG can be heuristically chosen to be the one with the smallest crowdedness factor ρ_1 from those ECGs that have not been tried as a source. With src fixed, we loop over all the links $\epsilon \in src$. Each link ϵ can be moved into the destination ECG with the least increase in the crowdedness factor.

2) Inflation: The *inflation* operation is invoked when the power consumption is adopted as the minimization objective. The basic observation is that links may be moved among ECGs such that overall the set of ECGs become less crowded and, hence, require less transmission power. In particular, if a certain ECG G_k has more than one link and $\sum_k \tilde{\lambda}_k < Q$, that is, there are some empty ECGs, then the total power consumption will be reduced by "moving" a link from G_k into an empty ECG. This result can be established with the following Theorem 3. Intuitively, Theorem 3 says that power reduction can be achieved by spreading out links in ECGs.

Theorem 3 (Triangle Inequality of Power Consumption): Consider a feasible link set $\mathcal{E} = \{x \to Y_x, x \in X, Y_x \subseteq V_0\}$. Suppose $\mathcal{E}$ can be partitioned into two disjoint sets of links $\mathcal{E}_1$ and $\mathcal{E}_2$. Then

$$P(\mathcal{E}_1) + P(\mathcal{E}_2) \leq P(\mathcal{E}_1 \cup \mathcal{E}_2). \tag{24}$$

Proof: Let $\{p_i^*, i \in X\}$ denote an optimal solution to the linear program for power control, with link set $\mathcal{E} = \mathcal{E}_1 \cup \mathcal{E}_2$. Let X_1 and X_2 denote the set of transmitters for $\mathcal{E}_1$ and $\mathcal{E}_2$, respectively. Then, $p_i^*, i \in X_1$ (with respect to X_2) give a feasible solution to the linear program for power control with link set $\mathcal{E}_1$ (with respect to $\mathcal{E}_2$), since a reduction in the interference level can never decrease the SINR. Hence, the result. ∎

Similar to the deflation procedure above, we can construct a heuristic inflation procedure. Assume for now there is a crowdedness factor $\rho_2(\cdot)$, possibly different from $\rho_1(\cdot)$. We set the maximum number of trials to be performed, where in each trial we try to move a link ϵ in a selected source ECG src into other ECGs. The source ECG can be heuristically chosen to be the one with the largest crowdedness factor ρ_2. With src fixed, we can search for the action $\mathrm{MOVE}(\epsilon, src, dst)$ with the largest reduction in the sum crowdedness factor.

3) Crowdedness Factor: We propose to use the power consumption of an ECG as the crowdedness factors ρ_1 and ρ_2 for two reasons. First, it is a linear metric, i.e., the power consumption of a linear combination of ECGs is the corresponding linear combination of the power consumptions of the ECGs. Second, a crowded ECG will typically incur a large power consumption, because the heavy interference condition necessitates high transmission power in order to maintain the SINR above the threshold γ.

V. Simulation Results

We conduct simulations on two test networks, shown in Fig. 2(a) and (b), respectively. We consider these two networks as example community wireless networks, formed by wireless devices in a neighborhood. In Fig. 2, the locations of the houses are marked with dots.

Fig. 2(a) is a toy example, consisting of nine houses spaced uniformly on a 3×3 grid. It is included mainly for ease in understanding the proposed optimizations. The end-to-end demands for this smaller network are set to be $\mathcal{S}_1 = \langle 1, \{9\}, 1 \rangle$ and

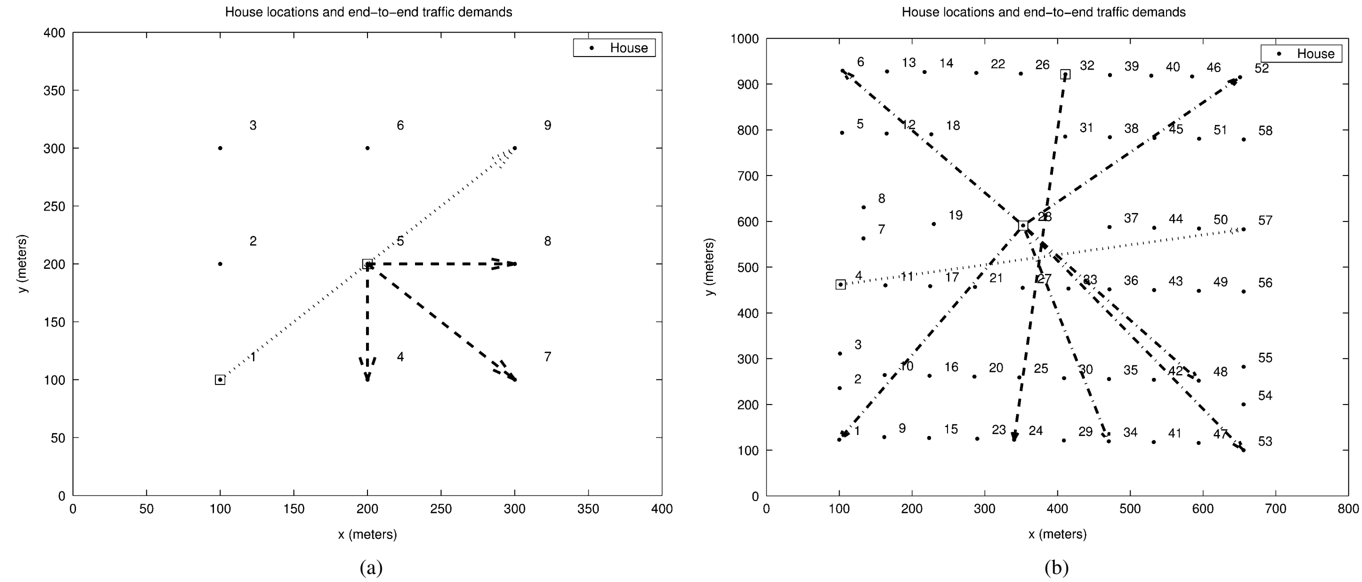


Fig. 2. House locations and end-to-end traffic demands for the (a) smaller network and (b) the larger network.

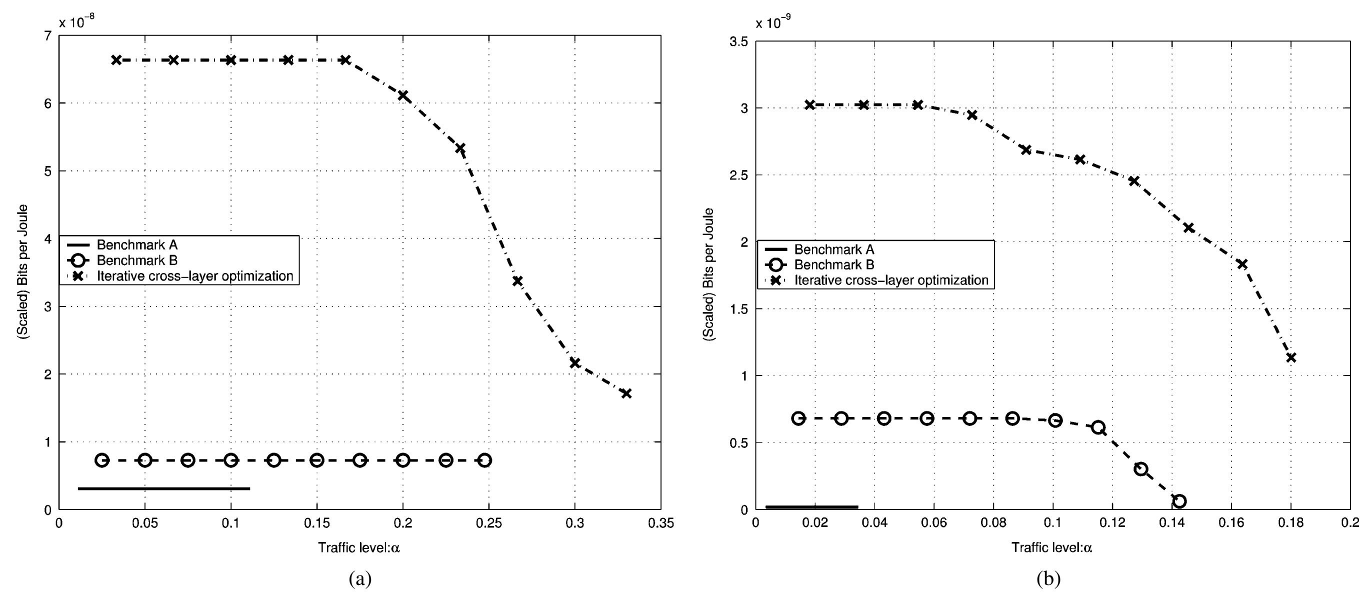


Fig. 3. Main performance results: Bits-per-Joule versus rate scaling factor (traffic level). (a) Results for the smaller network. (b) Results for the larger network.

$\mathcal{S}_2 = \langle 5, \{4, 7, 8\}, 1\rangle$, as illustrated in Fig. 2(a) by a dotted line and three dashed lines.

Fig. 2(b) is a more realistic example, consisting of 58 houses spaced roughly in three rows. The locations of the houses are obtained from measurement. The end-to-end demands for this larger network are set to be $\mathcal{S}_1 = \langle 4, \{57\}, 1\rangle, \mathcal{S}_2 = \langle 32, \{24\}, 1\rangle$, and $\mathcal{S}_3 = \langle 28, \{1, 6, 34, 48, 52, 53\}, 1\rangle$.

We focus our experimental investigations on two performance criteria: the aggregate congestion and the power efficiency. For the former, we evaluate $\alpha_{\max}$, which is the largest rate scaling factor that can be supported. For the latter, we evaluate the (scaled) bits-per-Joule. Combining the two criteria, we choose the primary performance measure to be a curve showing the (scaled) bits-per-Joule β as a function of the traffic level α. As a convention, we define the value of β to be zero if the traffic level cannot be carried. Thus, this curve will cut off to zero at $\alpha_{\max}$. The corresponding curves for the two test networks are shown in Fig. 3(a) and (b), respectively. These amount to the main simulation results. We discuss these in detail shortly.

Let $\underline{\lambda}$ denote the vector of $\lambda_k, k = 1, \ldots, K$. Define

$$\beta_o(\alpha, \underline{\lambda}, \mathcal{B}) \equiv \frac{\alpha}{\sum_{k:G_k \in \mathcal{B}} \lambda_k P(G_k)} \tag{25}$$

if rate scaling factor α can be supported by $\sum_k \lambda_k G_k$, otherwise, let

$$\beta_o(\alpha, \underline{\lambda}, \mathcal{B}) \equiv 0. \tag{26}$$

The condition that α can be supported by $\sum_k \lambda_k G_k$ refers to the existence of a feasible flow assignment on a given supported capacity graph, which is obtained by timesharing ECGs in $\mathcal{B}$ with parameter $\underline{\lambda}$. Specifically, the condition can be checked by examining the feasibility of the system of constraints in (9)–(14), with variables $\{f_t^m(vw)\}$ and $\{g^m(vw)\}$.

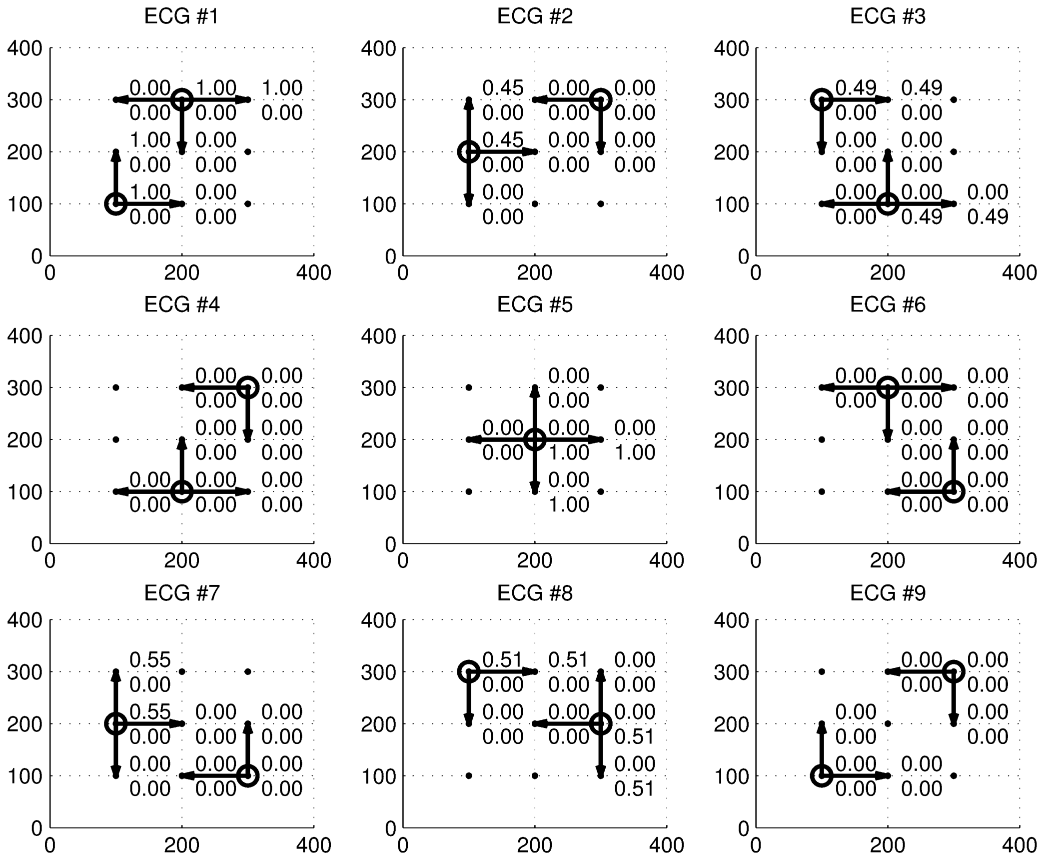


Fig. 4. Initial basis of ECGs obtained by the random packing procedure.

We implement two benchmark algorithms and compare them with the proposed iterative cross-layer optimization. The objective is to quantitatively evaluate the advantages of cross-layer coupling. Benchmark A models the case where the physical and link layers are decoupled from the network layer. The composite capacity graph G, which is used to represent the physical and link layers, is fixed and given by equally timesharing each ECG in the initial basis of ECGs. That is $G = 1/K \sum_k G_k, G_k \in \mathcal{B}_i$, where $\mathcal{B}_i$ denotes the initial basis of ECGs, constructed by the random packing procedure. The maximum achievable rate-scaling factor, $\alpha_{\max}^A$, can be found by solving the linear program in (8)–(14), with $\lambda_k = 1/K$. Since the composition of the composite capacity graph is fixed, the bits-per-Joule index will be constant. In essence, for benchmark A, the bits-per-Joule/traffic curve will be

$$\beta_A(\alpha) = \beta_o\left(\alpha_{\max}^A, \frac{1}{K}\mathbf{1}, \mathcal{B}_i\right), \quad \alpha \leq \alpha_{\max}^A \tag{27}$$

where $\mathbf{1}$ is the vector of all 1's.

Benchmark B decouples the physical layer from the link layer and above by using a fixed basis $\mathcal{B}_i$, but allowing the timesharing proportions $\lambda_k, k = 1, \ldots, K$ to be jointly optimized, together with the flow assignment. The maximum achievable rate-scaling factor $\alpha_{\max}^B$, can be found by solving the linear program in (8)–(14), with $\lambda_k, k = 1, \ldots, K$ treated as variables. Then, for each traffic level $\alpha \leq \alpha_{\max}^B$, the maximum bits-per-Joule can be obtained through the linear program minimizing the power consumption. Hence, the bits-per-Joule/traffic curve will be

$$\beta_B(\alpha) = \beta_o(\alpha, \underline{\lambda}(\alpha), \mathcal{B}_i), \quad \alpha \leq \alpha_{\max}^B. \tag{28}$$

In comparison, given a traffic level α, the full-fledged cross-layer optimization adjusts both the timesharing proportions $\lambda_k, k = 1, \ldots, K$ and the basis of ECG's $\mathcal{B}$. Hence, the bits-per-Joule/traffic curve will be

$$\beta_C(\alpha) = \beta_o(\alpha, \underline{\lambda}(\alpha), \mathcal{B}(\alpha)), \quad \alpha \leq \alpha_{\max}^C. \tag{29}$$

The basis of ECGs starts with an initial set and gets iteratively refined with feedback information about the link traffic load.

Some simulation parameters are set up as follows. The SINR threshold γ is set to be 4 dB. The path loss coefficients are set as $a_{ij} = d(i,j)^{-3}$, where $d(i,j)$ is the distance between node i and j. The noise level is set to be 1. Since the normalization renders the absolute value of power consumption less meaningful, we examine the relative power efficiencies for different schemes in the subsequent discussions. The number of ECG's Q in the intermediate steps of deflation/inflation is around 200.

A. "Toy" Network

We first consider the toy example in Fig. 2(a). Fig. 4 shows the initial basis of ECG's $\mathcal{B}_i$, constructed by the random packing procedure in Section II-B. Since session related information is unavailable at the beginning, every node could potentially serve as a source, a relay, or a destination. To ensure that each node

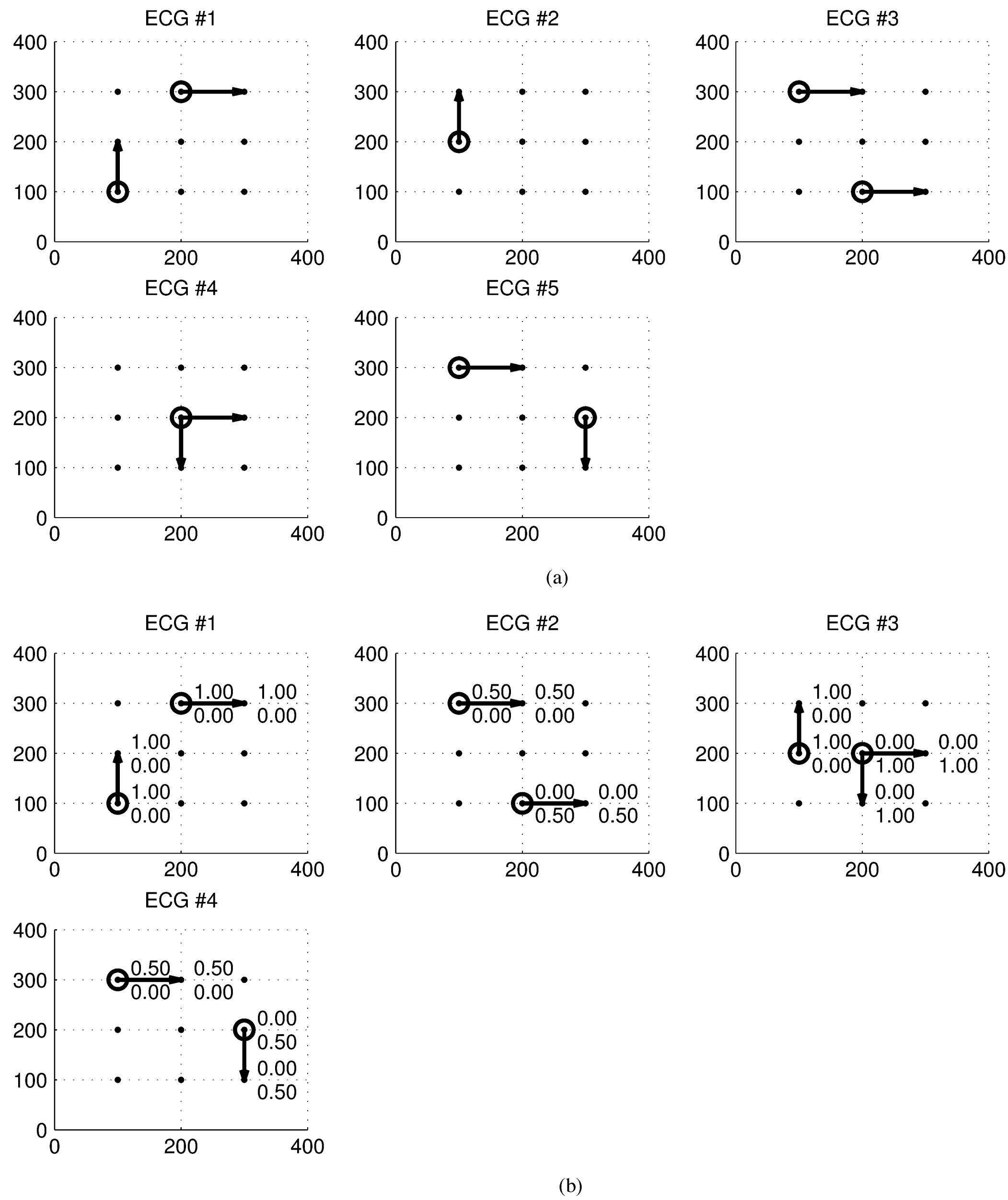


Fig. 5. (a) Set of ECGs after pruning redundancies. (b) Resulting basis of ECGs after iteratively minimizing the aggregate congestion measure.

serves as a transmitter at least once and as a receiver at least once, when packing each ECG, the first transmitter loops over the set of all nodes.

The communication range R and the interference range R_i are both set to be 120 m. Each link is denoted by a circle at the transmitter and arrows pointing from the transmitter to the receivers. Circles are used to highlight the fact that a point-to-multipoint link cannot be simply treated as a collection of independent point-to-point links, each pointing directly from the transmitter to a receiver. This is because distinct information may be loaded on these point-to-point links in the latter scenario. In the underlying graph, a virtual vertex is introduced for each transmitter in each ECG.

First, assume the operating regime is bandwidth-limited and the goal is to economically allocate resources in order to leave more room to accommodate new traffic. Equivalently, the optimization objective is to minimize the aggregate congestion measure. In Fig. 4, a column vector is shown on the right of each vertex. This vector gives the assigned link traffic load for each session, i.e., the vector of max of flows $\underline{g}(vw)$ whose entries are $g^m(vw), m = 1, \ldots, M$. In each link of an ECG, e.g., $u \to \{v_j, j = 1, \ldots, n\}$ with u' being the virtual vertex, the vector $\underline{g}(uu')$ is shown on the right of the transmitter u, and the vector $\underline{g}(u'v_j)$ is shown on the right of the receiver v_j. The maximum of $\sum g^m(vw)$ over all edges vw in the ECG G_k is λ_k, in the solution of the linear program (15)–(20). Thus, it can be checked that the aggregate congestion measure $\sum_k \lambda_k$ is 4.0 with $\mathcal{B}_i$.

The heuristic refinement of ECGs starts by pruning redundancies in $\mathcal{B}_i$, according to the assigned link loads. First, divide the vector of max of flows $\underline{g}(vw)$ and λ_k by $\sum_k \lambda_k$, such that the aggregate congestion measure becomes 1 after this nor-

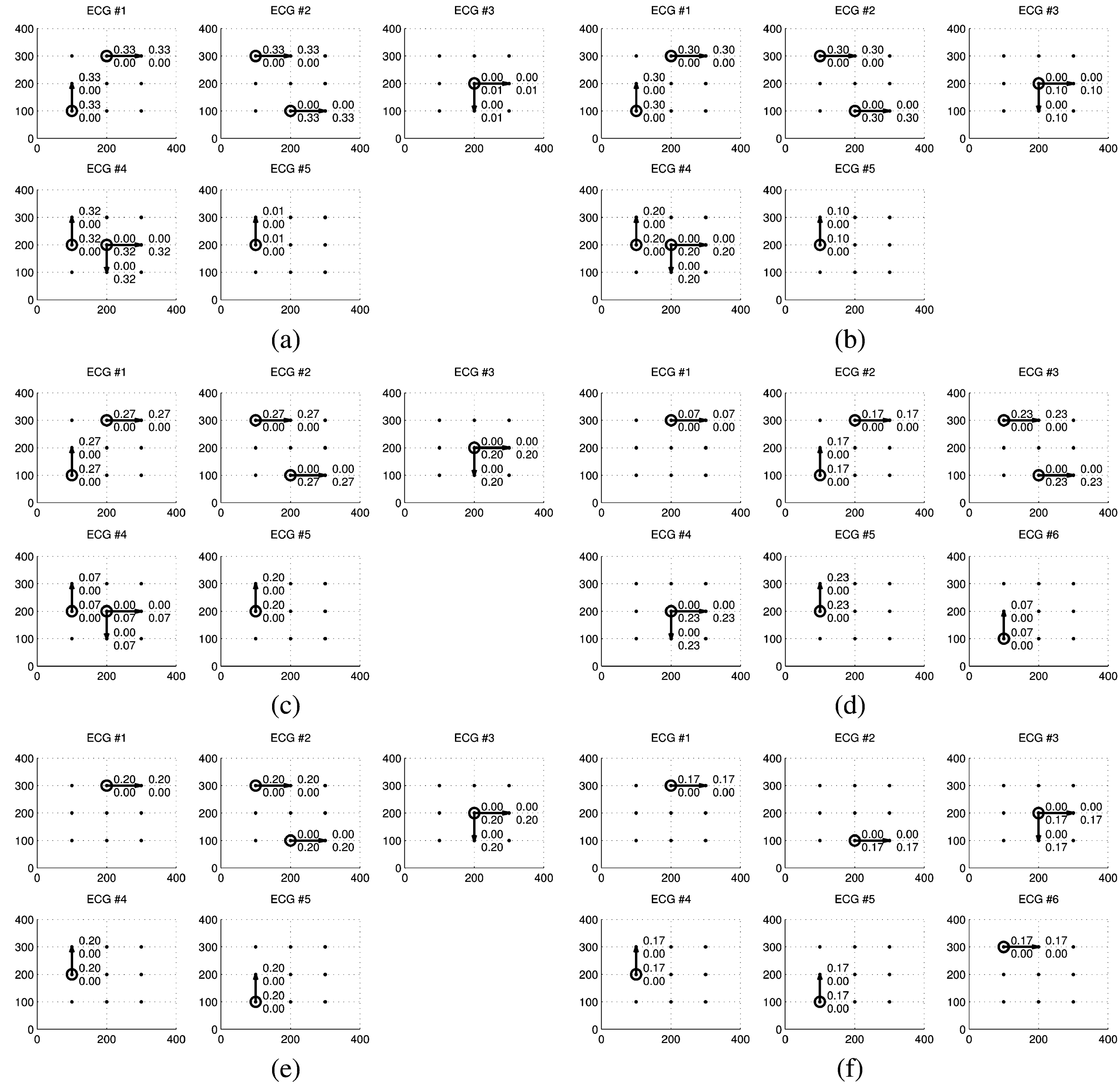
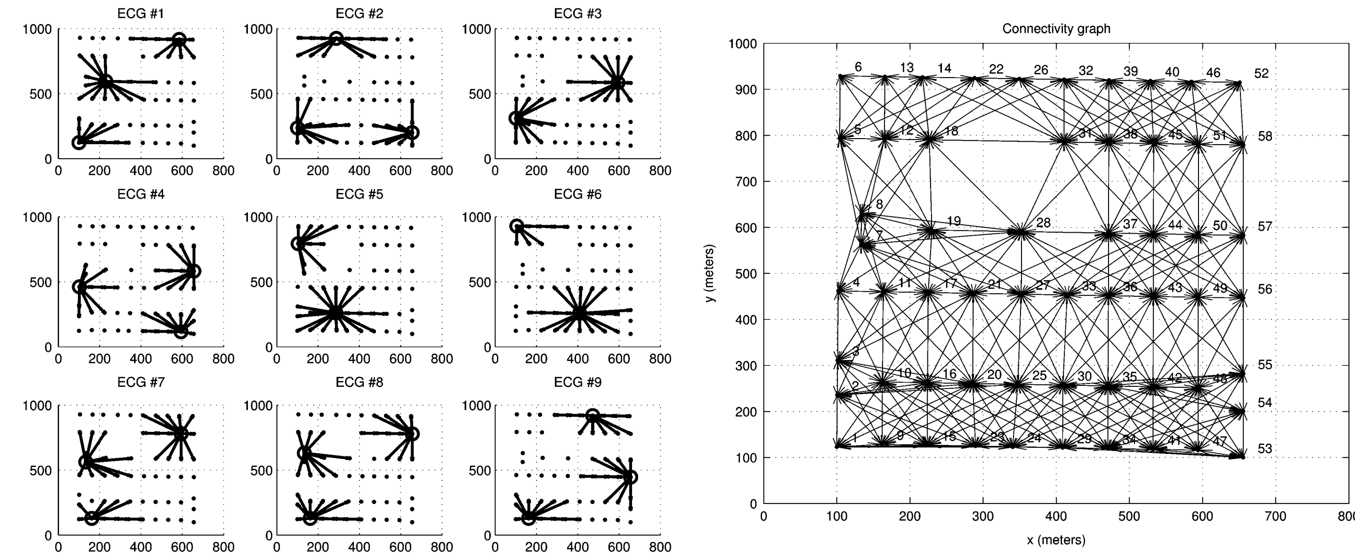


Fig. 6. Resulting bases obtained by iteratively minimizing the power consumption. (a) $\alpha = 99\%\alpha_{\max}$. (b) $\alpha = 90\%\alpha_{\max}$. (c) $\alpha = 80\%\alpha_{\max}$. (d) $\alpha = 70\%\alpha_{\max}$. (e) $\alpha = 60\%\alpha_{\max}$. (f) $\alpha = 50\%\alpha_{\max}$.

malization. Next, multiply the normalized vectors $\underline{g}(vw)$ by $Q = 200$ and compute the ceiling. Denote the nine ECGs in $\mathcal{B}_i$ by $G_1, \ldots, G_9$. Retain 50 copies of G_1, 23 copies of G_2, 25 copies of G_3, 50 copies of G_5, 27 copies of G_7, and 26 copies of G_8. Then, process one by one these six types of ECGs $G_1, G_2, G_3, G_5, G_7, G_8$, the links for each type, and the M sessions. Take the link $1 \rightarrow \{2, 4\}$ in G_1 as an example. Session 2 has zero load on this link. For session 1, since the branch to node 4 has zero load, this edge is pruned from the broadcast link. Then, 50 copies of the link appear in the first 50 ECGs, corresponding to ECG #1 in Fig. 5(a).

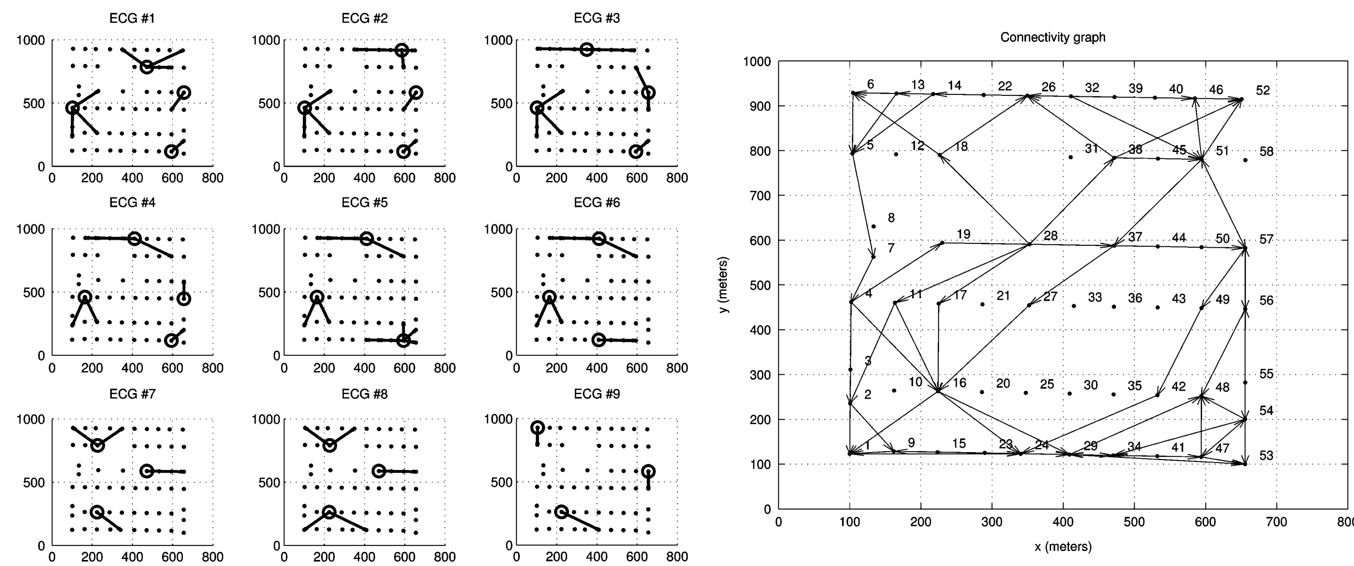
The set of ECGs after pruning is given in Fig. 5(a). The deflation operation is then applied to this set. After this, the next iteration is started. Fig. 5(b) shows the resulting set of ECGs after iteratively minimizing the aggregate congestion measure. In fact, it has converged after only two iterations. It can be seen that ECG #2 and #4 in Fig. 5(a) have been successfully merged to form ECG #3 in Fig. 5(b). We denote this converged set in Fig. 5(b) with $\mathcal{B}_m$. It can be seen that the aggregate congestion measure $\sum_k \lambda_k$ has been reduced to 3.0. Hence, $\alpha_{\max} = 1/3$. Comparing Fig. 4 with Fig. 5(b), it can be observed that the links in the initial set tend to be omnidirectional, whereas iterative and heuristic refinement has made the resulting ECGs match better with the end-to-end traffic demands.

Let us now switch attention from the bandwidth-limited regime to the energy-limited regime and discuss the entire



(a) (b)

Fig. 7. (a) First nine ECGs in the initial basis $\mathcal{B}_i$ obtained by the random packing procedure. (b) Initial connectivity.

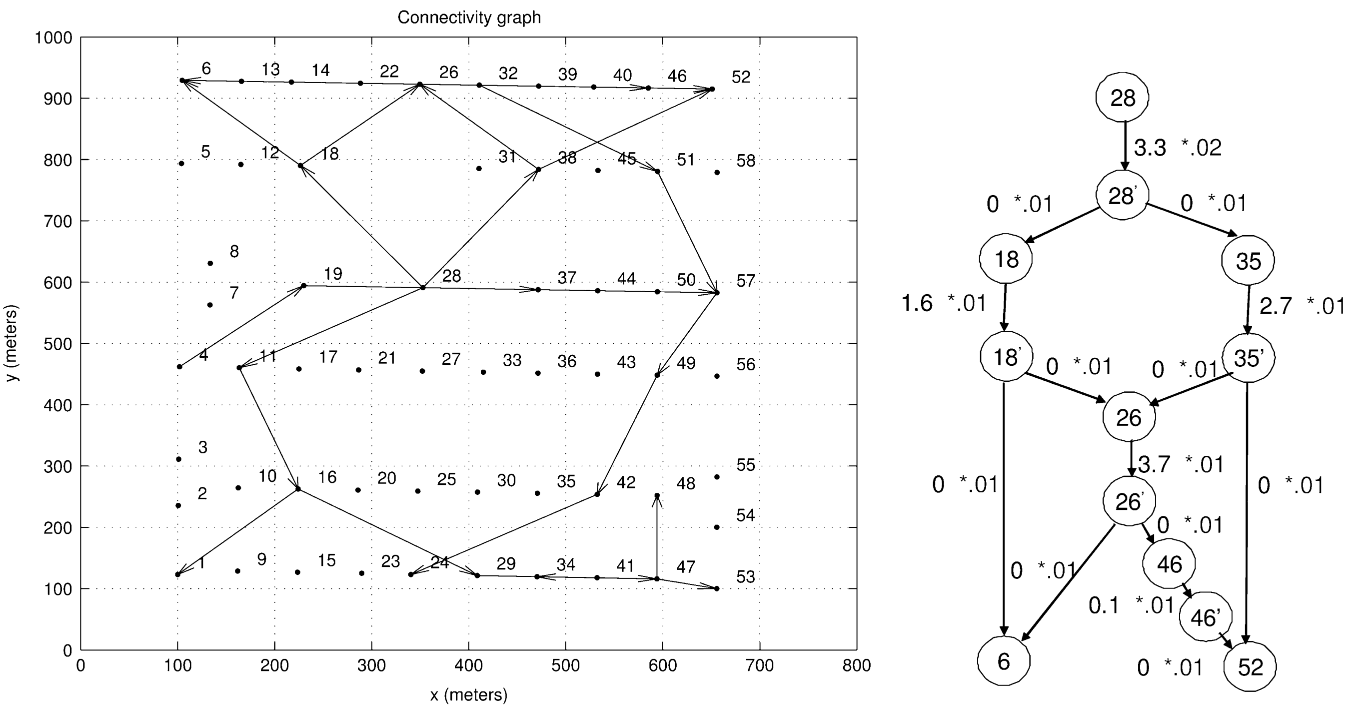


(a) (b)

Fig. 8. (a) First nine ECGs in the resulting basis $\mathcal{B}_m$ obtained by iteratively minimizing the aggregate congestion measure. (b) Connectivity after iteratively minimizing the aggregate congestion measure.

bits-per-Joule/traffic curve $\beta(\alpha)$. Fig. 6(a)–(f) shows the resulting sets of ECGs after iteratively minimizing the power consumption, for traffic levels ranging from $\alpha = 99\%\alpha_{\max}$ to $\alpha = 50\%\alpha_{\max}$. The result for $\alpha = 99\%\alpha_{\max}$ is obtained with $\mathcal{B}_m$ [Fig. 5(b)] being the initial basis of ECGs. The resulting basis of ECGs is used as the initialization for $\alpha = 90\%\alpha_{\max}$, and so on. The inflation operation is called in each iteration, trying to arrange interfering links "spaciously." Moving from Fig. 6(a) to Fig. 6(f), as the traffic becomes lighter and lighter, links become more and more spread out in the ECGs and routes become more and more energy efficient. Note the congestion measure is 1 with all these six resulting sets, although the offered traffic levels are only fractions of $\alpha_{\max}$. This confirms that available "space" should be used as much as possible for power reduction, as dictated in Theorem 3. Fig. 6(f) shows one extreme case where the traffic demand is so light that every link can afford to occupy an ECG alone.

Now, let us come back to the main simulation results in Fig. 3(a), which shows the (scaled) bits-per-Joule performance as a function of the traffic level. The top curve corresponds to the full-fledged iterative cross-layer optimization. The sets of ECGs for the rightmost six data points (from right to left) in the top curve are those in Fig. 6(a)–(f), respectively. The curve in the middle is for Benchmark B, where a fixed set of ECG's $\mathcal{B}_i$ is used. The cut-off point of this curve is its maximum rate scaling factor $\alpha_{\max}^B = 1/4$. The curve is flat and much lower than the top one, showing that in this case the limited degrees of freedom in adjusting the timesharing parameters are not powerful enough to effectively achieve power reduction. The bottom curve is for Benchmark A, where the composite



(a) (b)

Fig. 9. (a) Connectivity after iteratively minimizing the power consumption, at a light traffic level $\alpha = 10\%\alpha_{\max}$. (b) Graph for information multicast from node 28 to node 6 and node 52. The traffic demand is $\alpha = 0.0182$, which has been rounded to 0.02. All the edges have unit capacity. Each link has an associated "price," which is the energy-per-bit, i.e., power per unit rate. In the figure, the cost for using a link $u \rightarrow \{v_j\}$ is charged to edge uu'. The multicast scheme achieving the required (low) multicast rate while consuming the least amount of resources is given by performing network coding on the minimum-cost max of flows. Two numbers have been marked on each edge: the left one is the price (scaled power) and the right one is the assigned max of flows $g(vw)$ returned by the linear program minimizing the total power consumption.

capacity graph is fixed. It has the worst performance in both aspects: $\alpha^A_{\max} = 1/9$ and the bits-per-Joule stays at a low level.

B. Larger Network

Next, we consider the more realistic example in Fig. 2(b). Fig. 7(a) shows the first nine ECGs out of the initial set $\mathcal{B}_i$ with 58 ECGs, constructed by the random packing procedure with the communication range $R = 250$ m and the interference range $R_i = 300$ m. Taking the union of the ECGs in this set, we can obtain the connectivity graph of the network, shown in Fig. 7(b).

Fig. 8(a) shows part of the resulting set $\mathcal{B}_m$ after iteratively minimizing the aggregate congestion measure, which has 34 distinct ECGs. Similar to the smaller network, effects of the pruning and deflation operations can be observed. Comparing Fig. 7(b) with Fig. 8(b), it can be clearly seen that with omnidirectional and randomly packed physical-layer broadcasts, the network is more connected. Given the traffic information, unnecessary connectivity has been removed in Fig. 8(b). The remaining sparser connectivity graph can promise a larger throughput for the demands.

At the other extreme sits energy-efficient communication under very light traffic load. The offered traffic level $\alpha = 10\%\alpha_{\max}$ is so light that each ECG has only one link. Fig. 9(a) gives the corresponding connectivity graph. It can be seen that a distribution tree is used for the information multicast from 28 to $\{1, 34, 48, 53\}$. The multicast from 28 to $\{6, 52\}$ turns out to be very illustrative of the unique advantages of network coding. We have extracted this part of the network as Fig. 9(b). In Fig. 9(b), all the edges have unit capacity. Each link has an associated "price," which is the energy-per-bit, i.e., power per unit rate. The cost for using a link $u \rightarrow \{v_j\}$ is charged to edge uu'. The multicast scheme achieving the required (low) multicast rate while consuming the least amount of resources is given by performing network coding on the minimum-cost max of flows. Coincidentally, the structure of this graph and its solution closely resemble the (classical) example in [1], which demonstrates that network coding can achieve a higher throughput than routing. The current example, in contrast, shows that network coding can be more economic in using network resources than routing. Generally, the characterization of sum of max of flows as linear constraints facilitates finding network coding solutions that are efficient in using resources.

The main performance results are given in Fig. 3(b). Among the three schemes, Benchmark A has the smallest achievable rate scaling factor $\alpha^A_{\max} = 1/29 = 0.035$. For Benchmark B $\alpha^B_{\max} = 0.144$, which is 3.1 times larger than $\alpha^A_{\max}$. The iterative cross-layer optimization has the largest rate, $\alpha_{\max} = 0.182, 26.4\%$ larger than $\alpha^B_{\max}$. In terms of energy-efficiency, Benchmark A has the worst performance since a significant amount of energy is not contributing to the traffic demands. The curve for Benchmark B is monotonically decreasing. A proof is as follows. If $\alpha_1 < \alpha_2 \leq \alpha^B_{\max}$, we can always accommodate the lower traffic level α_1 by proportionally scaling the time-sharing coefficients $\lambda_k, k = 1, \ldots, K$ used in accommodating α_2. The curve for Benchmark B is not flat, showing that the degree of freedom for adjusting $\underline{\lambda}$ can be useful in improving the power efficiency. This effect can also be seen from Fig. 6(a)–(c) for the toy network, which exhibit different power efficiency with the same basis of ECGs. The top curve is for the iterative cross-layer optimization. The data points were collected from right (heaviest traffic) to left (lightest traffic) and in collecting each data point, the resulting basis of ECGs associated with the previous data point is adopted as an initialization. For a traffic level α around 0.02, the bits-per-Joule value of the iterative optimization is 3.44 times larger than that for Benchmark B.

C. Summary

Overall, the major performance results have been presented in Fig. 3 as the bits-per-Joule versus traffic level curves. In Fig. 3, the cut-off points characterize the maximum supportable traffic level and the part of the curves before cut-off gives the energy efficiency. The iterative cross-layer optimization has been compared with Benchmark A, which decouples the physical and link layers from the network layer, and Benchmark B, which decouples the physical layer from the link layer and above. The performance gaps in these curves provide evidence for the benefits of cross-layer coupling, and for the need for the physical states to be traffic-dependent.

The usefulness of physical-layer broadcast in supporting end-to-end multicast energy-efficiently is empirically confirmed. From the simulation results, network coding has been observed to offer unique advantages over routing in economically using network resources.

VI. Related Work

In recent years, many works have focused on the performance characterization of multihop wireless networks. These works can be roughly classified into two categories. The first category focuses primarily on how the system scales with space and the number of users. In their seminal paper [5], Gupta and Kumar show that in a network with N nodes, the throughput in bit-meters per second is $\Theta(\sqrt{N}/\sqrt{\log N})$ assuming random node distribution and random traffic pattern. This throughput limit arises fundamentally as a result of mutual interference among concurrent transmissions. Since the primary interest of the works in this category is on growth rate, these works model the cross-layer coupling from a higher abstraction level than that pursued in the current paper.

The second category aims at a detailed performance characterization in a deterministic setting. Previous papers [7] and [8] fall into this category. Both papers [7] and [8] considered bounding the maximum throughput for multiple unicast sessions. Both papers combine the linear constraints that characterize the allocated bit-rate resources and the constraints that characterize feasible multicommodity flow (sum of flows) assignment to arrive at joint linear programs. Using the "free of secondary interference" model [6], Kodialam and Nandagopal [7] presented a linear program that can be used to find an achievable rate for a unicast session, which can be guaranteed to be within 67% of the optimal solution under the model. They also considered multiple unicast sessions. Jain *et al.* [8] introduced the notion of a *conflict graph* to model interference and established a framework for the joint optimization of routing, medium access control, and physical layer transmissions, for interference-limited wireless networks. They also showed that finding the maximum throughput is NP-hard under the interference model.

In this paper, we have pursued this line further, starting with a converse problem: given the traffic demands, how to economically allocate the network resources to minimize certain cost functions. More specifically, we considered minimizing the congestion measure (equivalently, maximizing the throughput) and minimizing the total power consumed in providing different achievable rates for multiple multicast sessions. We presented models for power control and physical-layer broadcast in the physical layer, and network coding in the network layer. We proposed an iterative optimization as a new computational method. The result is a set of achievable tradeoffs between throughput and energy efficiency, in a given wireless network with a given traffic pattern.

In general, finding the optimal tradeoffs between throughput and energy efficiency is difficult because of the combinatorial nature of the problem that arises due to the effects of interference. The proposed method in this paper yields achievable but possibly suboptimal performance because of the heuristic initialization and the greedy updates of the ECGs, etc. In another paper [9], [10], we focus on the ultimate energy-limited regime and consider finding the minimum energy-per-bit for multicasting information from a source to multiple destinations, under the layered model. It was observed that if the required rate is sufficiently small, then the effect of interference becomes immaterial, enabling one to find the minimum energy-per-bit in polynomial time. Consequently, the entire space of pertinent supported capacity graphs becomes a convex combination of a polynomial number of ECGs. Given this observation, the minimum energy-per-bit can be found by a linear program that identifies a supported capacity graph and a max of flows assignment, such that the ratio of the power consumption achieved to the multicast rate achieved, that is, the energy-per-bit, is minimized. A solution to the minimum energy-per-bit in a mobile ad hoc network, where the effect of mobility was modeled by introducing a time dimension into the capacity graph was also proposed in [9] and [10].

VII. Conclusion and Future Works

In this paper, we have formulated the network planning problem in wireless ad hoc networks as economically allocating network resources (information carrier *supplies*) such that certain end-to-end communication *demands* are fulfilled. We model the demands as a collection of multicast sessions in the transport layer. We represent an allocation of supplies as the choice of one capacity graph, out of a vast collection of supported capacity graphs. This network planning formulation necessitates an interaction across the network stack. In this paper, a cross-layer approach is undertaken, which jointly optimizes the flow assignment, timesharing in the link layer, and coordination of concurrent links.

At the physical layer, the network operates in many different physical states, each corresponding to an elementary capacity graph. At the link layer, by timesharing among different physical states, convex combinations of the elementary capacity graphs can be achieved, hence presenting to the upper layers a set of supported composite capacity graphs. At the network layer, given a capacity graph, the end-to-end traffic demands are transformed into link-by-link traffic loads, with which network coding can be applied. Integrating these components, an iterative cross-layer optimization is proposed. Two optimization objectives, minimizing an aggregate congestion measure and minimizing the power consumption, are considered, depending on whether the operating regime is bandwidth-limited or energy-limited.

Simulations have been conducted on community wireless networks. Benefits of a cross-layer approach have been quantitatively evaluated by comparing with designs that decouple the layers. These results demonstrate that significant performance advantages, in terms of congestion reduction and energy efficiency, can be achieved by adopting a full-fledged cross-layer optimization.

The proposed method outputs a set of achievable tradeoffs between throughput and energy efficiency. These results are potentially suboptimal because: 1) the layered model is suboptimal from an information theoretic perspective; 2) the sum of max of flows leads to a suboptimal solution to the multisession multicast problem; and 3) the selection of ECGs is potentially suboptimal. Better performance may be obtained by improving along these directions. It would also be interesting to investigate (upper) bounds to the optimal tradeoffs between throughput and energy efficiency.

The proposed method has been presented as a centralized algorithm. In the future, we are interested in devising and analyzing practical and distributed schemes with practical cross-layer coupling.

Acknowledgment

The authors are very grateful to Guest Editor Dr. J. Wieselthier and the anonymous reviewers for their helpful suggestions. The authors would like to thank Dr. J. Padhye for providing the data for the community wireless networks.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[2] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On construction of energy-efficient broadcast and multicast trees in wireless networks," in *Proc. IEEE Conf. Comput. Commun.*, Tel-Aviv, Israel, 2000, pp. 586–594.

[3] Y. Wu, P. A. Chou, Q. Zhang, K. Jain, W. Zhu, and S.-Y. Kung, "Network planning in wireless ad hoc networks: A cross-layer approach," Microsoft Research, Tech. Rep. MSR-TR-2004-74, Aug. 2004.

[4] R. W. Yeung, "Multilevel diversity coding with distortion," *IEEE Trans. Inf. Theory*, vol. 41, pp. 412–422, Mar. 1995.

[5] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

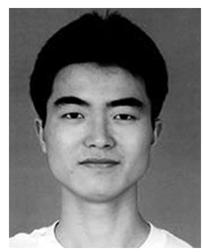
[6] D. J. Baker, J. E. Wieselthier, and A. Ephremides, "A distributed algorithm for scheduling the activation links in a self-organizing mobile radio networks," in *Proc. IEEE Int. Conf. Commun.*, 1982, pp. 2F6.1–2F6.5.

[7] M. Kodialam and T. Nandagopal, "Charactering achievable rates in multi-hop wireless networks: The joint routing and scheduling problem," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, Sep. 2003, pp. 42–54.

[8] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, Sep. 2003, pp. 66–80.

[9] Y. Wu, P. A. Chou, and S.-Y. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Trans. Commun.*, Mar. 2004, submitted for publication.

[10] Y. Wu, P. A. Chou, and S.-K. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding," in *Proc. IEEE Inf. Theory Workshop*, San Antonio, TX, Oct. 2004.



Yunnan Wu (S'02) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2000 and the M.A. degree in electrical engineering from Princeton University, Princeton, NJ, in 2002. He is currently working towards the Ph.D. degree electrical engineering at Princeton University.

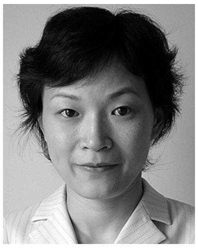
From 1999 to 2001, he was with Microsoft Research, Asia, Beijing, China, as a Research Assistant. In 2002, he was with Bell Laboratories, Lucent Technologies, as a Summer Intern and with Microsoft Research, Redmond, WA, as a Summer Intern in 2003. His research interests include networking, signal processing, information theory, multimedia, and wireless communications.

Mr. Wu received the Best Student Paper Award at the 2000 SPIE and IS&T Visual Communication and Image Processing Conference. He is a recipient of the Microsoft Research Graduate Fellowship from 2003 to 2005.

Philip A. Chou (S'82–M'87–SM'00–F'04) received the B.S.E. degree in electrical engineering and computer science from Princeton University, Princeton, NJ, in 1980, the M.S. degree in electrical engineering and computer science from the University of California, Berkeley, in 1983, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1988.

From 1988 to 1990, he was Member of Technical Staff at AT&T Bell Laboratories, Murray Hill, NJ. From 1990 to 1996, he was a Member of Research Staff at the Xerox Palo Alto Research Center, Palo Alto, CA. In 1997, he was Manager of the Compression Group, VXtreme, Mountain View, CA, before it was acquired by Microsoft in 1997. From 1998 to the present, he has been a Senior Researcher with Microsoft Research, Redmond, WA. He also served as a Consulting Associate Professor at Stanford University from 1994 to 1995, and has been an Affiliate Professor at the University of Washington since 1998. His research interests include data compression, information theory, communications, and pattern recognition, with applications to video, images, audio, speech, and documents.

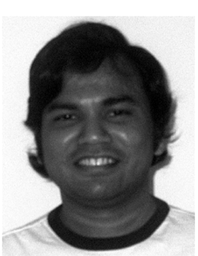
Dr. Chou is the recipient of the 2002 ICME Best Paper Award with A. Seghal and the 1993 Signal Processing Society Paper Award with T. Lookabaugh. He served as an Associate Editor in source coding for the IEEE Transactions on Information Theory from 1998 to 2001, and served as a Guest Associate Editor for special issues in the IEEE Transactions on Image Processing and the IEEE Transactions on Multimedia in 1996 and 2004, respectively. From 1998 to 2004, he was a member of the IEEE Signal Processing Society's Image and Multidimensional Signal Processing Technical Committee (IMDSP TC). He is a member of Phi Beta Kappa, Tau Beta Pi, Sigma Xi, and the IEEE Computer, Information Theory, Signal Processing, and Communications societies, and was an active member of the MPEG Committee.



Qian Zhang (M'00–SM'04) received the B.S., M.S., and Ph.D. degrees from Wuhan University, Wuhan, China, in 1994, 1996, and 1999, respectively, all in computer science.

She joined Microsoft Research, Asia, Beijing, China, in 1999, and now is Research Manager of the Wireless and Networking Group. She participated in many activities in the Internet Engineering Task Force (IETF) Robust Header Compression (ROHC) WG Group for TCP/IP header compression. She has published about 80 refereed papers in international leading journals and key conferences. She is the inventor of about 20 pending patents. Her current research interest includes seamless roaming across different wireless networks, multimedia delivery over wireless, Internet, next-generation wireless networks, and P2P network/ad hoc network.

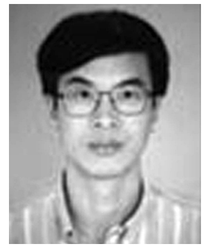
Dr. Zhang received the Asia-Pacific Best Young Researcher Award by IEEE ComSoc. She is a member of the Visual Signal Processing and Communication Technical Committee and the Multimedia System and Application Technical Committee of the IEEE Circuits and Systems Society. She is also a member and Chair of QoSIG of the Multimedia Communication Technical Committee of the IEEE Communications Society. She is now serving as an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and Guest Editor for the Special Issue on Wireless Video of the *IEEE Wireless Communication Magazine*. In 2004, she was named one of the top 100 young creative individuals by MIT Technology Review (TR100).



Kamal Jain received the Ph.D. degree from Georgia Institute of Technology, Atlanta, in 2000.

Since 2000, he has been working with Microsoft Research, Redmond, WA. His main interest is applying combinatorics in various branches of information and computational sciences including discrete optimization, networking, information theory, cryptography, and computational economics.

Dr. Jain has won various awards for his work on discrete optimization, the latest being the Optimization Prize from Informs.



Wenwu Zhu (S'92–M'97–SM'01) received the B.E. and M.E. degrees from the National University of Science and Technology, Hefei, China, in 1985 and 1988, respectively, the M.S. degree from Illinois Institute of Technology, Chicago, and the Ph.D. degree from Polytechnic University, Brooklyn, NY, in 1993 and 1996, respectively, all in electrical engineering.

He has been working at Intel Communication Technology Lab China as Director since September 2004. From 1999 to September 2004, he was with Microsoft Research, Asia, Beijing, China, as Research Manager of the Wireless and Networking Group. From 1996 to 1999, he was with Bell Laboratories, Lucent Technologies, NJ, as a Member of Technical Staff. He has published over 160 refereed papers in various key journals and conferences, and has contributed to the Internet Engineering Task Force (IETF) Robust Header Compression (ROHC) WG draft on robust TCP/IP header compression over wireless links. He is the inventor of more than 12 pending patents. His current research interest is in the area of wireless/Internet multimedia communication and networking, and wireless communication and networking.

Dr. Zhu received the Best Paper Award in the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2001. He is a Guest Editor for the PROCEEDINGS OF THE IEEE, an Associate Editor for the IEEE TRANSACTIONS ON MOBILE COMPUTING, the IEEE TRANSACTIONS ON MULTIMEDIA, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS for VIDEO TECHNOLOGY. Currently, he is the Chairman of the IEEE Circuits and System Society Beijing Chapter and the Secretary of Visual Signal Processing and Communication Technical Committee. He is a member of Eta Kappa Nu, Multimedia System and Application Technical Committee and Life Science Committee in the IEEE Circuits and Systems Society, and Multimedia Communication Technical Committee in the IEEE Communications Society.

Sun-Yuan Kung (S'74–M'78–SM'84–F'88) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1977.

In 1974, he was an Associate Engineer at Amdahl Corporation, Sunnyvale, CA. From 1977 to 1987, he was a Professor of Electrical Engineering-Systems at the University of Southern California, Los Angeles. Since 1987, he has been a Professor of Electrical Engineering at Princeton University, Princeton, NJ. He held a Visiting Professorship at Stanford University and the Delft University of Technology, Delft, The Netherlands, in 1984, a Toshiba Chair Professorship at Waseda University, Japan, in 1984, an Honorary Professorship at Central China University of Science and Technology, in 1994, and a Distinguished Chair Professorship at the Hong Kong Polytechnic University since 2001. He has coauthored more than 400 technical publications and numerous textbooks including VLSI Array Processors (Englewood Cliffs, NJ: Prentice-Hall, 1988), Digital Neural Networks (Englewood Cliffs, NJ: Prentice-Hall, 1993), Principal Component Neural Networks (New York: Wiley, 1996), and Biometric Authentication: A Machine Learning and Neural Network Approach (Englewood Cliffs, NJ: Prentice-Hall, 2004). His research interests include VLSI array processors, system identification, neural networks, wireless communication, multimedia signal processing, bioinformatic data mining, and biometric authentication.

Prof. Kung was a recipient of the IEEE Signal Processing Society's Technical Achievement Award for his contributions on parallel processing and neural network algorithms for signal processing in 1992, the IEEE Signal Processing Society's Best Paper Award for his publication on principal component neural networks in 1996, and the IEEE Third Millennium Medal in 2000. He was a Distinguished Lecturer of the IEEE Signal Processing Society in 1994. Since 1990, he has been the Editor-in-Chief of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS (VLSI).