On The Modeling of Wireless Multicast Networks

Lihua Wan, Jie Luo

Abstract

This paper considers a class of utility optimization problems in wireless networks. Assume network utility is a function of link throughput and nodes' transmission power. It is shown that, under a set of physical layer assumptions, the impact of physical, data-link layer configurations in a wireless network can be characterized using a configuration graph, which is similar to a wireline network topology graph. Network utility optimization can consequently be carried out via iterations that optimize network layer algorithms over the configuration graph and revise physical, data-link layer configurations to improve the configuration graph. For a class of wireless multicast networks with intra-session network coding, it is shown that the number of point-to-multipoint links involved in network utility optimization is only polynomial, as opposed to exponential, in the number of nodes.

Index Terms

configuration graph, hyperarc link, network coding, wireless network

I. Introduction

It is well known that the topology of a wireline network can be modeled by a graph, in which, each vertex represents a network node and each edge represents a point-to-point (cable) link between two nodes [3]. Extending such a graphic model to wireless networks, however, faces two key challenges. First, in principle, a wireless link can support a positive throughput so long as its channel gain is not strictly zero [4]. Achievable throughput over a wireless link strongly depends on the communication activities over other neighboring links. Without detailed physical layer and data-link layer specifications, using a fixed topology graph to model a wireless network is often not informative. Second, since signal transmitted over a wireless medium can reach more than one receivers, it is possible for a wireless node to communicate common information to multiple receivers simultaneously [5] using the same power and bandwidth of a point-to-point transmission. When such transmissions are modeled by point-to-multipoint links (termed hyperarc links [6]), the total number of feasible links in a wireless

The authors are with the Electrical and Computer Engineering Department, Colorado State University, Fort Collins, CO 80523. E-mail: {carlawan, rockey}@engr.colostate.edu.

This work was supported by National Science Foundation grants CCF 07-28826 and CCF 10-16985. Part of the results have been presented in International Symposium on Information Theory [1][2].

network can be exponential in the number of nodes. Consequently, network optimization requiring an exhaustive search over all feasible links can be overly complex even for moderate-sized networks.

Utility optimization in wireless networks has been extensively investigated in the literature. The idea of using hyperarc links to model wireless broadcast transmissions was firstly introduced by Lun et al in [6]. It was shown that when energy efficiency is of primary concern, a wireless network can be modeled using a topology graph with non-interfering links. This enabled the formulation of a class of multicast utility maximization problems as convex optimization problems, which can be solved efficiently. The approach was extended by Wu et al in [7] to optimize a generalized utility under the assumption that the topology graph-based network model with non-interfering links remains valid. In [8], Yuan et al modeled a mesh network using a topology graph with coupled link capacities, and proposed an iterative framework to maximize the multicast network utility. In [9], Sagduyu and Ephremides investigated the modeling of wireless networks. Given a pre-determined set of "communication realizations" (see definition in Section II), a wireless network is modeled by a topology graph whose link rates are the time-shared rates of the corresponding links in the communication realizations. Link rates in the topology graph are therefore coupled through the scheduling of the communication realizations. This consequently enabled the optimization of a multicast network utility over the time-sharing coefficients under the assumption of optimal intra-session network coding.

In this paper, we consider a class of constrained utility optimization problems in wireless networks. Assume both network utility and constraints are functions of link throughput and nodes' transmission power. Under a set of physical layer assumptions, we first show that, the impact of physical and data-link layer configurations in a wireless network can be characterized by a configuration graph (defined in Section II), which is similar to a wireline network topology graph. If both the utility function and the constraint functions are convex in link throughput and transmission power, optimal network utility can be obtained using iterative algorithms that optimize network layer algorithms over the configuration graph and incrementally revise physical, data-link layer configurations to improve the configuration graph. Next, we consider a class of utility optimization problems for wireless multicast networks with optimal intra-session network coding. We show that, the total number of links involved in the iterative optimization is only polynomial, as opposed to exponential, in the number of nodes.

II. System Modeling and Problem Formulation

Consider a wireless network with stationary memoryless channels. Assume that time is partitioned into slots of equal length. Let V be the node set of the network. We define a hyperarc link e_{iJ} from node i to a node set J, if i can deliver common information simultaneously and directly to all nodes in J. We say e_{iJ} achieves a throughput of μ_{iJ} in a time slot if i communicates common information directly to all nodes in J at rate μ_{iJ} with a negligible error probability. We define a communication realization $C(t)$ as the simultaneous activation of a set of links, together with their physical layer configurations, in time slot t where t is an integer. We assume that the physical layer configuration includes a set of communication rates that can be achieved simultaneously for the active links. By setting the rates of inactive links at zero, we denote the rate vector of the communication realization $C(t)$ by $\boldsymbol{\mu}^{(C)}(t)$ or $\boldsymbol{\mu}(C(t))$. The element of $\boldsymbol{\mu}^{(C)}(t)$ corresponding to link e_{iJ} is denoted by $\mu_{iJ}^{(C)}(t)$. It is helpful to assume that the physical layer configuration should not involve any time-sharing operation within one time slot, although this assumption is not necessary. Because a communication realization $C(t)$ completely specifies the physical layer configuration in time slot t, it also determines the average transmission power of the nodes, denoted by vector $\boldsymbol{p}^{(C)}(t) = \boldsymbol{p}(C(t))$, in time slot t.

We define a transmission schedule $S = \{C(0), C(1), \dots, C(T-1)\}$ as the periodic extension of a communication realization sequence $C(0), C(1), \dots, C(T-1)$, with T being the period and $C(i)$ being the communication realization of time slots $kT+i$, $k = -\infty, \dots, -1, 0, 1, \dots, \infty$. Communication rate vector (or throughput vector) supported by a transmission schedule S equals the average rate vector of its communication realizations, denoted by $\boldsymbol{\mu}^{(S)} = \boldsymbol{\mu}(S) = \frac{1}{T}\sum_{t=0}^{T-1}\boldsymbol{\mu}(C(t))$. Similarly, the average transmission power of the nodes specified by transmission schedule S equals $\boldsymbol{p}^{(S)} = \frac{1}{T}\sum_{t=0}^{T-1}\boldsymbol{p}(t) = \frac{1}{T}\sum_{t=0}^{T-1}\boldsymbol{p}(C(t))$. An illustration of links, communication realizations and transmission schedule is given in Figure 1.

We make two key assumptions in the physical and data-link layer model described above. First, we only assume decoding and forward as opposed to amplifying and forward for information relay. We only distinguish successful and non-successful transmissions over each link, and a successful transmission requires all receivers of the link should obtain the transmitted messages reliably, as opposed to obtaining a corrupted or incomplete version. Second, messages transmitted over different links are encoded and decoded independently at the physical layer. There is no joint information encoding among multiple transmitters or joint information decoding over multiple links at the receiver.

Given a transmission schedule $S = \{C(0), \dots, C(T-1)\}$. We construct a configuration graph $G^{(S)}(V, E) = G(S)$ with E being the union of links of all communication realizations. Let $\boldsymbol{\mu}^{(S)} = \boldsymbol{\mu}(S) = \frac{1}{T}\sum_{t=0}^{T-1}\boldsymbol{\mu}(C(t))$ be the throughput vector of S. We define $\mu_{iJ}^{(S)}$ as the configuration rate of link e_{iJ}, and $\boldsymbol{\mu}^{(S)}$ as the configuration rate vector, in graph $G^{(S)}(V, E) = G(S)$. An illustration of the relation between a transmission schedule and a configuration graph is given in Figure 2. Note that given a fixed transmission schedule S, the constructed configuration graph is similar to a wireline network topology graph in the following senses. E is similar to the set of feasible links in a topology graph since other links are not activated in S. Because nodes can always discard information, the configuration rate vector $\boldsymbol{\mu}^{(S)}$ is similar to the link capacity vector in a topology graph since all link throughput vectors $\tilde{\boldsymbol{\mu}}$ in the hyper-cubic region $\{\tilde{\boldsymbol{\mu}}|\tilde{\boldsymbol{\mu}} \le \boldsymbol{\mu}^{(S)}\}$ are supported by the same set of physical and data-link layer configurations specified by S. When we present $\boldsymbol{\mu}^{(S)}$ and $\boldsymbol{p}^{(S)}$ as parameters associated with configuration graph $G^{(S)}(V, E)$, we also denote them as $\boldsymbol{\mu}^{(G)}$ and $\boldsymbol{p}^{(G)}$.

Now consider a class of network utility maximization problems formulated as

$$\max_{S} U(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)}), \qquad \text{s.t. } \boldsymbol{H}(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)}) \ge \boldsymbol{0}, \tag{1}$$

where, given a transmission schedule S, we assume that the utility function $U(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)})$ and the set of constraint functions $\boldsymbol{H}(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)})$ are functions of link throughput $\boldsymbol{\mu}^{(S)}$ and transmission power $\boldsymbol{p}^{(S)}$, both determined by S. The objective is to maximize the network utility over all feasible transmission schedules S, under the constraints that $\boldsymbol{H}(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)}) \ge \boldsymbol{0}$.

Network optimization problem (1) can be written in the following equivalent form

$$\max_{S, G^{(S)}(V,E)=G(S)} U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}), \qquad \text{s.t. } \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}) \ge \boldsymbol{0}, \tag{2}$$

where $\boldsymbol{\mu}^{(G)}$ is now the configuration rate vector of the configuration graph $G^{(S)}(V, E)$ constructed from S.

Note that in a practical system, formulation of the network optimization problem (2), particularly the expression of the utility function $U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})$, often depends on network layer algorithms, as illustrated in the following example.

Example 1: Consider a wireless multicast network where a source node s wants to multicast common information reliably through the network to a set of destination nodes T. Let the throughput of the multicasting be denoted by R_{sT}. Assume the objective is to maximize a multicast utility $U(R_{sT}, \boldsymbol{p})$ subject to transmission power constraints $\boldsymbol{p} \le \boldsymbol{P}$. A network utility optimization problem can be formulated as follows.

$$\max_{S} U(R_{sT}, \boldsymbol{p}), \qquad \text{s.t. } \boldsymbol{P} - \boldsymbol{p} \ge \boldsymbol{0}. \tag{3}$$

Given S, $R_{sT} = R_{sT}(\boldsymbol{\mu}^{(S)})$ can be written as a function of a link throughput vector $\boldsymbol{\mu}^{(S)} = \boldsymbol{\mu}(S)$. However, the exact expression of $R_{sT}(\boldsymbol{\mu}^{(S)})$ depends on the network layer protocol. If we assume optimal network coding, and let $\boldsymbol{\mu}^{(S)}$ be the configuration rate vector of a configuration graph $G^{(S)}(V, E) = G(S)$, then R_{sT} equals the max-flow of the minimum $s - T$ cut of $G^{(S)}(V, E)$, which is the sum configuration rates of links crossing the minimum cut that separates s from at least one destination node in T [10]. Expression of the $R_{sT}(\boldsymbol{\mu}^{(S)})$ function becomes complicated if only routing is allowed at the network layer [10].

III. Iterative Network Utility Optimization

Consider the network utility optimization problem given in (2). When both $-U(\boldsymbol{\mu}, \boldsymbol{p})$ and $\boldsymbol{H}(\boldsymbol{\mu}, \boldsymbol{p})$ are convex in $\boldsymbol{\mu}$ and $\boldsymbol{p}$, (2) is a convex optimization problem. The unique optimal solution $(\boldsymbol{\mu}, \boldsymbol{p})^*$ of (2) can therefore be obtained by solving the following Lagrangian problem.

$$\max_{S, G^{(S)}(V,E)=G(S)} \min_{\boldsymbol{\lambda} \geq \boldsymbol{0}} L = U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}), \tag{4}$$

where L is the Lagrangian function and $\boldsymbol{\lambda} \geq \boldsymbol{0}$ is the Lagrangian relaxation parameter.

In a practical wireless network, it is often infeasible to derive the achievable region of the $(\boldsymbol{\mu}, \boldsymbol{p})$ vector pair. It is equally challenging to determine whether a $(\boldsymbol{\mu}, \boldsymbol{p})$ pair is supported by any transmission schedule. Consequently, (4) can only be solved by iterative and constructive updates of the transmission schedule. Next, we show that, given a fixed Lagrangian parameter $\boldsymbol{\lambda} \geq \boldsymbol{0}$, the Lagrangian function L in (4) can indeed be maximized *monotonically* via iterative and *incremental* updates of S, where the meaning of "incremental" update is specified below.

Definition 1: Let $S_1 = \{C_1(0), C_1(1), \ldots, C_1(T_1 - 1)\}$ and $S_2 = \{C_2(0), C_2(1), \ldots, C_2(T_2 - 1)\}$ be two transmission schedules whose periods are T_1 and T_2, respectively. We say S_2 is incrementally different from S_1 if there exists a nonnegative integer k such that $T_2 = kT_1 + 1$, and for all nonnegative integers $\tilde{k} < k$ and $i \leq T_1 - 1$, we have $C_2(\tilde{k}T_1 + i) = C_1(i)$. When S_2 is incrementally different from S_1, we also say S_1 is incrementally different from S_2.

Theorem 1: Let $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}, \boldsymbol{p})$ be the Lagrangian function in (4) for a given $\boldsymbol{\lambda}$. Define $L^*_{\boldsymbol{\lambda}}$ as

$$\begin{aligned} L^*_{\boldsymbol{\lambda}} &= \max_{S, G^{(S)}(V,E)=G(S)} L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}) \\ L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}, \boldsymbol{p}) &= U(\boldsymbol{\mu}, \boldsymbol{p}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{\mu}, \boldsymbol{p}). \end{aligned} \tag{5}$$

Let S_1 be a transmission schedule whose transmission power vector is $\boldsymbol{p}_1$. Let $\boldsymbol{\mu}_1$ be the configuration rate vector of a configuration graph $G_1(V, E)$ constructed from S_1. If $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1) < L^*_{\boldsymbol{\lambda}}$, then there exists another transmission schedule S_2 with transmission power vector $\boldsymbol{p}_2$ and a configuration rate vector $\boldsymbol{\mu}_2$ of $G_2(V, E)$ constructed from S_2, such that,

- S_2 is incrementally different from S_1.
- $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_2, \boldsymbol{p}_2) < L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)$.

Proof: By assumption, $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}, \boldsymbol{p})$ is concave in $(\boldsymbol{\mu}, \boldsymbol{p})$. Since $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1) < L^*_{\boldsymbol{\lambda}}$, we can find a transmission schedule $\tilde{S} = \{\tilde{C}(0), \ldots, \tilde{C}(\tilde{T} - 1)\}$ with $\boldsymbol{p}(\tilde{S}) = \tilde{\boldsymbol{p}}$, and a rate vector $\tilde{\boldsymbol{\mu}} = \boldsymbol{\mu}(\tilde{S})$, such that

$$\left(\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{\mu}}\right)^T \tilde{\boldsymbol{\mu}} + \left(\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{p}}\right)^T \tilde{\boldsymbol{p}} > 0, \tag{6}$$

where $\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{\mu}}$ and $\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{p}}$ are partial sub-derivatives of $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}, \boldsymbol{p})$ at $(\boldsymbol{\mu}_1, \boldsymbol{p}_1)$.

Since $\boldsymbol{p}(\tilde{S}) = \tilde{\boldsymbol{p}}$, $\boldsymbol{\mu}(S) = \tilde{\boldsymbol{\mu}}$, there exists a sequence of $(\tilde{\boldsymbol{\mu}}(i), \tilde{\boldsymbol{p}}(i))$ pairs, $i \in \{0, \ldots, \tilde{T}-1\}$, to satisfy $\tilde{\boldsymbol{p}}(i) = \boldsymbol{p}(\tilde{C}(i))$, $\tilde{\boldsymbol{\mu}}(i) \in \tilde{C}(i)$, $\forall i$, and

$$\tilde{\boldsymbol{\mu}} = \frac{1}{\tilde{T}} \sum_{i=0}^{\tilde{T}-1} \tilde{\boldsymbol{\mu}}(i), \qquad \tilde{\boldsymbol{p}} = \frac{1}{\tilde{T}} \sum_{i=0}^{\tilde{T}-1} \tilde{\boldsymbol{p}}(i). \tag{7}$$

From (6) and (7), we can find $j \in \{0, \ldots, \tilde{T}-1\}$, such that the $(\tilde{\boldsymbol{\mu}}(j), \tilde{\boldsymbol{p}}(j))$ pair corresponding to communication realization $\tilde{C}(j)$ satisfies

$$\left(\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{\mu}}\right)^T \tilde{\boldsymbol{\mu}}(j) + \left(\frac{\partial L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)}{\partial \boldsymbol{p}}\right)^T \tilde{\boldsymbol{p}}(j) > 0. \tag{8}$$

Consequently, there exists a positive integer k to satisfy

$$L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1) < L_{\boldsymbol{\lambda}}\left(\frac{kT_1\boldsymbol{\mu}_1 + \tilde{\boldsymbol{\mu}}(j)}{kT_1+1}, \frac{kT_1\boldsymbol{p}_1 + \tilde{\boldsymbol{p}}(j)}{kT_1+1}\right). \tag{9}$$

Now construct a transmission schedule $S_2 = \{C_2(0), C_2(1), \ldots, C_2(T_2-1)\}$ with $T_2 = kT_1 + 1$. For all nonnegative integers $\tilde{k} < k$ and $i \leq T_1 - 1$, we choose $C_2(\tilde{k}T_1 + i) = C_1(i)$, where $C_1(i)$ is the ith communication realization of S_1. We also let $C_2(T_2-1) = \tilde{C}(j)$. With this construction, S_2 is only incrementally different from S_1. Since $\boldsymbol{p}_2 = \boldsymbol{p}(S_2) = \frac{kT_1}{kT_1+1}\boldsymbol{p}_1 + \frac{1}{kT_1+1}\tilde{\boldsymbol{p}}(j)$, $\boldsymbol{\mu}_2 = \frac{kT_1}{kT_1+1}\boldsymbol{\mu}_1 + \frac{1}{kT_1+1}\tilde{\boldsymbol{\mu}}(j) = \boldsymbol{\mu}(S_2)$, according to (9), we have $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_2, \boldsymbol{p}_2) > L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)$. ■

Theorem 1 implies that the Lagrangian optimization problem (4) can be solved via iterative updates of $\boldsymbol{\lambda}$ and $(\boldsymbol{\mu}, \boldsymbol{p})$ pair, where each update of $(\boldsymbol{\mu}, \boldsymbol{p})$ pair only involves an incremental revision, as opposed to a direct construction, of the transmission schedule.

IV. Constraint The Number of Hyperarc Links

From the proof of Theorem 1, we can see that, in the incremental revision of the transmission schedule, the key task is to find a communication realization $\tilde{C}(j)$ with $\tilde{\boldsymbol{p}}(j) = \boldsymbol{p}(\tilde{C}(j))$ and $\tilde{\boldsymbol{\mu}}(j) = \boldsymbol{\mu}(\tilde{C}(j))$, such that updating $(\boldsymbol{\mu}, \boldsymbol{p})$ in the direction of $(\tilde{\boldsymbol{\mu}}(j), \tilde{\boldsymbol{p}}(j))$ improves the Lagrangian utility. If such a task needs to be accomplished by exhaustive search of the communication realizations, then its complexity is exponential in the number of feasible links, and in principle, the number of feasible links is exponential in the number of nodes.

In this section, we show that the link throughput region supported by an arbitrary communication realization possesses a *hyper-coordinate-convexity* property. With the help of this property, for the class of multicast utility optimization problems given in Example 1, optimal utility can be achieved by searching a number of links that is only polynomial in the number of nodes.

Definition 2: Let $\boldsymbol{\nu}$ be a vector whose elements correspond to all feasible links. Denote the element of $\boldsymbol{\nu}$ corresponding to hyperarc link e_{iJ} by ν_{iJ}. We say $\boldsymbol{\nu}$ is a *throughput degrading vector* if there exists a user i, two user subsets J, $\tilde{J} \subset J$, and a constant $\delta \geq 0$, such that $\nu_{iJ} = -\delta$, $\nu_{i\tilde{J}} = \delta$, and all other elements of $\boldsymbol{\nu}$ equal 0.

Let $\boldsymbol{\mu}_d$, $\boldsymbol{\mu}$ be two link throughput vectors. We say $\boldsymbol{\mu}_d$ is *a degraded version* of $\boldsymbol{\mu}$ if $\boldsymbol{\mu}_d - \boldsymbol{\mu}$ equals the summation of some throughput degrading vectors.

Definition 3: A link throughput vector region Γ is *hyper-coordinate convex* if for all thoughput vectors $\tilde{\boldsymbol{\mu}}$, we have $\tilde{\boldsymbol{\mu}} \in \Gamma$ so long as there exists two throughput vectors $\boldsymbol{\mu}$, $\boldsymbol{\mu}_d$, such that $\boldsymbol{\mu} \in \Gamma$, $\boldsymbol{\mu}_d$ is a degraded version of $\boldsymbol{\mu}$, and $\tilde{\boldsymbol{\mu}} \leq \boldsymbol{\mu}_d$.

Lemma 1: The union of link throughput vectors supported by an arbitrary communication realization forms a hyper-coordinate convex throughput vector region.

Proof: Because multicasting common information from node i to a node set J can be achieved using any hyperarc link $e_{i\tilde{J}}$ so long as $\tilde{J} \supseteq J$, $\boldsymbol{\mu}$ being supported by communication realization C implies that $\boldsymbol{\mu}_d$ is also supported by communication realization C if $\boldsymbol{\mu}_d$ is a degraded version of $\boldsymbol{\mu}$. ■

Theorem 2: Consider the class of multicast networks given in Example 1. Assume that optimal network coding is used at the network layer once a transmission schedule and a configuration graph are given. If there exists a transmission schedule to support multicast throughput of R_{sT} with a transmission power vector $\boldsymbol{p}$, then there must also exist a transmission schedule S to support the same multicast throughput R_{sT} with the same transmission power vector $\boldsymbol{p}(S) = \boldsymbol{p}$, together with a throughput rate vector $\boldsymbol{\mu} = \boldsymbol{\mu}(S)$ satisfying $\mu_{iJ} = 0$, $\forall i, J, |J| > |T|$.

Proof: According to the assumption, there exists a transmission schedule $\tilde{S}$ with transmission power $\boldsymbol{p} = \boldsymbol{p}(\tilde{S})$, and a configuration graph $G(V,E)$ constructed from $\tilde{S}$, such that the max-flow of the minimum $s-T$ cut of $G(V,E)$ equals R_{sT}. Let the configuration rate vector be $\tilde{\boldsymbol{\mu}} = \boldsymbol{\mu}(\tilde{S})$. From the proof of [6, Thoerem 2], we know that there exists a set of link rates $\tilde{x}^{(t)}_{iJj}$ to satisfy

$$\begin{aligned}
&\tilde{\mu}_{iJ} \geq \sum_{j\in J} \tilde{x}^{(t)}_{iJj}, \quad \forall (i,J), e_{iJ} \in E, t \in T, \\
&\sum_{\{J|(i,J),e_{iJ}\in E\}} \sum_{j\in J} \tilde{x}^{(t)}_{iJj} - \sum_{\{j|(j,I),e_{jI}\in E, i\in I\}} \tilde{x}^{(t)}_{jIi} = \sigma^{(t)}_i, \forall i \in V, t \in T, \\
&\sigma^{(t)}_s = R_{sT}, \quad \sigma^{(t)}_t = -R_{sT}, \quad \text{and } \sigma^{(t)}_{i\neq s,t} = 0.
\end{aligned} \tag{10}$$

Next, we will show that, by applying a recursive throughput degrading algorithm, throughput vector $\tilde{\boldsymbol{\mu}}$ can be degraded to another vector $\boldsymbol{\mu}$ which possesses the following properties.

1. $\mu_{iJ} = 0$, $\forall i, J, |J| > |T|$.
2. There exists a set of virtual link throughput $x^{(t)}_{iJj} \geq 0$ such that the degraded configuration rate vector $\boldsymbol{\mu}$ and the set of virtual link throughput $x^{(t)}_{iJj}$ satisfy the same inequality given in (10).

The throughput degrading algorithm is described below.

Initialization: Initialize the algorithm by $\boldsymbol{\mu} = \tilde{\boldsymbol{\mu}}$ and $x_{iJj}^{(t)} = \tilde{x}_{iJj}^{(t)}$, $\forall i, J, j, t$. Note that the initial condition satisfies (10).

Step 1: Find a link e_{iJ} with $|J| > |T|$ and $\mu_{iJ} > 0$. The algorithm stops if such a link does not exist.

Step 2: If for all $t \in T$ and $j \in J$, $x_{iJj}^{(t)} = 0$, we choose an arbitrary $j \in J$. Define a throughput degrading vector $\boldsymbol{\nu}$ by $\nu_{iJ} = -\mu_{iJ}$, $\nu_{ij} = \mu_{iJ}$, and set all other elements of $\boldsymbol{\nu}$ at zero. We degrade $\boldsymbol{\mu}$ by $\boldsymbol{\mu} = \boldsymbol{\mu} + \boldsymbol{\nu}$. Note that (10) remain satisfied by $\boldsymbol{\mu}$ and $\{x_{iJj}^{(t)}\}$. Go to Step 1.

Step 3: Assume we can find $t \in T$ and $j \in J$, such that $x_{iJj}^{(t)} > 0$. Define a positive constant δ by $\delta = \min_{j \in J, t \in T, x_{iJj}^{(t)} > 0} x_{iJj}^{(t)}$. For every $t \in T$, we define a node index j_t as follows. If $x_{iJj}^{(t)} = 0$ for all $j \in J$, we let $j_t = \text{NULL}$. Otherwise, $j_t = \text{argmin}_{j \in J, x_{iJj}^{(t)} > 0} x_{iJj}^{(t)}$.

Define node subset $\tilde{J}$ as the collection of the j_t indices, $\tilde{J} = \{j_t | t \in T, j_t \neq \text{NULL}\}$. Define a throughput degrading vector $\boldsymbol{\nu}$ by $\nu_{iJ} = -\delta$, $\nu_{i\tilde{J}} = \delta$, and set all other elements of $\boldsymbol{\nu}$ at zero. We degrade $\boldsymbol{\mu}$ by $\boldsymbol{\mu} = \boldsymbol{\mu} + \boldsymbol{\nu}$. For each $t \in T$, we also revise $x_{iJj}^{(t)}$ and $x_{i\tilde{J}j}^{(t)}$ as follows. If $j_t \neq \text{NULL}$, we let $x_{i\tilde{J}j_t}^{(t)} = x_{i\tilde{J}j_t}^{(t)} + \delta$ and $x_{iJj_t}^{(t)} = x_{iJj_t}^{(t)} - \delta$.

It can be verified that, after degrading $\boldsymbol{\mu}$ and revising $x_{iJj}^{(t)}$, we have $x_{iJj}^{(t)} \geq 0$, $x_{i\tilde{J}\tilde{j}}^{(t)} \geq 0$ for all $t \in T$, $j \in J$, and (10) still holds for $\boldsymbol{\mu}$ and $\{x_{iJj}^{(t)}\}$.

Go to Step 1.

The above algorithm reduces μ_{iJ} monotonically for all links e_{iJ} with $|J| > |D|$. The degraded throughput is allocated to link $e_{i\tilde{J}}$ in Steps 2 and 3 with $|\tilde{J}| \leq |T|$. Therefore, when the algorithm stops, we must have $\mu_{iJ} = 0$ for all $|J| > |T|$, and $\boldsymbol{\mu}$ is a degraded version of $\tilde{\boldsymbol{\mu}}$. Because (10) is satisfied in every step, it remains satisfied when the algorithm stops. The validity of (10) implies that R_{sT} is no larger than the max-flow of the minimum $s - T$ cut of configuration graph $G(V, E)$. In other words, R_{sT} is achievable on $G(V, E)$ with optimal network coding. ■

Theorem 2 implies that, for the class of multicast utility optimization problems considered in Example 1, in terms of achieving optimal utility, one only needs to consider links e_{iJ} with $|J| \leq |T|$. In other words, (3) can be *equivalently* written as

$$\max_{S, R_{sT}(\boldsymbol{\mu}^{(S)}), \mu_{iJ}^{(S)} = 0 \ \forall e_{iJ} \text{with } |J| > |T|} U(R_{sT}, \boldsymbol{p}^{(S)}), \quad \text{s.t. } \boldsymbol{p}^{(S)} - \boldsymbol{P} \leq \boldsymbol{0}. \tag{11}$$

The maximum number of links involved in an exhaustive search algorithm to solve (11) is only polynomial in the number of nodes $|V|$ (but is still exponential in $|T|$). It is straightforward to extend this conclusion to utility optimization problems with a more general set of constraints. It is also straightforward to extend the conclusion to networks with multiple multicast sessions and optimal intra-session network coding.

V. Further Complexity Reduction

Consider the class of multicast utility optimization problems in Example 1. Assume the network contains $|V|$ nodes with one multicast session from 1 source to $|T|$ destinations. The number of feasible links in this case equals $|V|\sum_{i=1}^{|T|}\begin{pmatrix} |V|-1 \\ i \end{pmatrix} \propto O(|V|^{|T|+1})$. For instance, for a network with $|V|=5$ and $|T|=2$ in an open wireless environment, the total number of feasible links is $5(4+6)=50$. Since the total number of link activation combinations is exponential in the total number of links, the complexity of an exhaustive search of link activation combinations is in the oder of 2^{50}, which is still prohibitive for computer simulations. In this section, we show that by exploiting fundamental properties of wireless communication and network coding, the practical complexity of link combination search can be significantly reduced, and this consequently enables the simulations of optimal utility maximization in some moderate-sized networks.

A well known property of wireless communication is that a wireless node cannot simultaneously transmit and receive information in the same channel. Consequently, link activations involving the transmission-reception conflict at any node should not be considered in the communication realization construction. Moreover, a key property of optimal network coding is that, given a configuration graph, throughput of a multicast session is bottlenecked by the maximum flow of the minimum $s-T$ cut, shown in (10). Therefore, when search for a communication realization to incrementally update the transmission schedule, the communication realization should not involve links that do not contribute to any of the minimum $s-T$ cuts. To introduce the principle of exploiting this property for complexity reduction, we will first present the detailed iterative utility optimization algorithm in the following.

Given a configuration graph $G^{(S)}(V,E)$ constructed base upon transmission schedule S. For any destination $t \in T$, we define $\gamma_t(G)$ as a cut that separates source s from t. Without causing any confusion, the cut value is also denoted by $\gamma_t(G)$. According to the network coding property, utility optimization problem (3) can be equivalently written as

$$\max_{S,G^{(S)}(V,E)=G(S)} U\left(\min_{t\in T}\min_{\gamma_t}\gamma_t(G),\boldsymbol{p}^{(G)}\right), \quad \text{s.t. } \boldsymbol{p}^{(G)}-\boldsymbol{P}\leq\boldsymbol{0}. \tag{12}$$

Since network utility is nondecreasing with multicast throughput, this problem can be further transformed to

$$\min_{\boldsymbol{\xi},\boldsymbol{\xi}\geq\boldsymbol{0}}\ \min_{\lambda_d\geq 0,\sum_{t\in T}\lambda_t=1}\ \min_{\lambda_{\gamma_t}\geq 0,\sum_{\gamma_t}\lambda_{\gamma_t}=1}\ \max_{S,G^{(S)}(V,E)=G(S)} U\left(\sum_{t\in T}\lambda_t\sum_{\gamma_t}\lambda_{\gamma_t}\gamma_t(G),\boldsymbol{p}^{(G)}\right)-\boldsymbol{\xi}^T(\boldsymbol{p}^{(G)}-\boldsymbol{P}), \tag{13}$$

where we have introduced several sets of auxiliary variables, $\boldsymbol{\xi}$, $\{\lambda_t,\forall t\in T\}$, and $\{\lambda_{\gamma_t},\forall\gamma_t\}$ for each $t\in T$.

Consequently, we can obtain the optimal solution of the problem using the following iterative algorithm.

Initialization: Construct an arbitrary transmission schedule and the corresponding configuration graph. Initialize λ_t and λ_{γ_t} for all γ_t and $t \in T$, and $\boldsymbol{\xi} = \mathbf{0}$.

Step 1: Update λ_t as

$$\lambda_t = \lambda_t - \delta_1 \frac{\partial U(R_{sT}, \boldsymbol{p})}{\partial R_{sT}} \sum_{\gamma_t} \lambda_{\gamma_t} \gamma_t(G), \quad \forall t \in T, \tag{14}$$

where $\delta_1 > 0$ is a small step size parameter. Then normalize λ_t to satisfy $\lambda_t \geq 0$ for all $t \in T$ and $\sum_{t \in T} \lambda_t = 1$.

Step 2: For each $t \in T$, update λ_{γ_t} as

$$\lambda_{\gamma_t} = \lambda_{\gamma_t} - \delta_2 \frac{\partial U(R_{sT}, \boldsymbol{p})}{\partial R_{sT}} \lambda_t \gamma_t(G), \quad \forall t \in T, \tag{15}$$

where $\delta_2 > 0$ is a small step size parameter. Then normalize λ_{γ_t} to satisfy $\lambda_{\gamma_t} \geq 0$ and $\sum_{\gamma_t} \lambda_{\gamma_t} = 1$ for each $t \in T$.

Step 3: Update $\boldsymbol{\xi}$ as

$$\boldsymbol{\xi} = \boldsymbol{\xi} + \delta_3(\boldsymbol{p} - \boldsymbol{P}), \tag{16}$$

where $\delta_3 > 0$ is a small step size parameter. Then set all negative-valued elements of $\boldsymbol{\xi}$ at zero.

Step 4: Construct a communication realization C with throughput vector $\boldsymbol{\mu}(C)$ and power $\boldsymbol{p}(C)$. Let $\gamma_t(C)$ be the sum throughput of links in C that cross cut γ_t. Communication realization C should be constructed to maximize

$$\frac{\partial U(R_{sT}, \boldsymbol{p})}{\partial R_{sT}} \sum_{t \in T} \lambda_t \sum_{\gamma_t} \lambda_{\gamma_t} \gamma_t(C) + \left[\frac{\partial U(R_{sT}, \boldsymbol{p})}{\partial \boldsymbol{p}} - \boldsymbol{\xi} \right]^T \boldsymbol{p}(C). \tag{17}$$

Carry out an incremental update of the transmission schedule S using communication realization C.

Go to Step 1 till convergence.

We say a cut γ_t is "effective" if $\lambda_{\gamma_t} \neq 0$. The key idea of complexity reduction is to maintain a small list of effective cuts in the iterative algorithm, such that, when constructing the communication realization C in Step 3, hyperarc link e_{iJ} with at least one arm e_{iJj} not crossing any effective cut should not be considered.

Assume for each $t \in T$, the system maintains a list of effective $s-t$ cuts, denoted by Γ_t. The list is initially empty. Let $\epsilon > 0$ be a small threshold parameter (e.g. $\epsilon = 0.3$). We add an extra step, Step 1.5, into the iterative algorithm.

Step 1.5: For each $t \in T$, find a minimum $s-t$ cut γ_t^* in the configuration graph $G(V, E)$ and add γ_t^* into the cut list Γ_t (with a small λ_{γ_t} value). For all $\gamma_t \in \Gamma_t$, let the cut value be denoted by $\gamma_t(G)$. If $\gamma_t(G) \geq (1+\epsilon)\gamma_t^*(G)$, remove the cut γ_t from the list. After that, normalize λ_{γ_t} to satisfy $\lambda_{\gamma_t} \geq 0$ and $\sum_{\gamma_t} \lambda_{\gamma_t} = 1$ for each $t \in T$.

Note that Step 1.5 corresponds to an aggressive update of the λ_{γ_t} variables. The revised iterative algorithm is still optimal.

VI. Suboptimal Network Utility Optimization

With the complexity reduction techniques proposed in Section V, simulations of the optimal network utility optimization algorithm for some moderate-sized networks, such as a network with $|V| = 16$ nodes and a single multicast session with $|T| = 3$, become feasible. Unfortunately, none of the proposed techniques can change the fundamental scaling law that the total number of link activation combinations is exponential in the number of nodes[1]. Due to this exponential scaling law, most optimal utility maximization algorithms for networks of size $|V| \geq 30$ are still overly complex for simulation. In this section, we introduce a class of suboptimal algorithms that guarantee a polynomial complexity in the number of nodes. Even though the performance of these algorithms can be significantly suboptimal, they represent a typical idea on avoiding combinatorial search in network utility optimization. In order to simplify the presentation, we will make several assumptions to the system model. Note that extending the basic idea of the suboptimal algorithms to a general system model is straightforward.

We consider the utility optimization problem in a wireless network presented in (4). Assume that the wireless channels are memoryless. Assume that each node can transmit over at most one active link in a time slot. For each communication realization C, we associate a probability parameter $P_{r_{iJ}}$ with each link e_{iJ}. We assume that a node i will transmit over a link e_{iJ} with probability $P_{r_{iJ}}$. Node i is receiving signal with probability $1 - \sum_{e_{iJ}} P_{r_{iJ}}$, where we have imposed the constraint $\sum_{e_{iJ}} P_{r_{iJ}} \leq 1$ for any node i. Suppose that each node makes transmission decisions independently. Let p_i denote the transmission power of node i. Let h_{ij} denote the channel gain from node i to node j. Given that link e_{iJ} is active and a node $j \in J$ is receiving, we compute the signal to noise ratio (SNR) at node j for link e_{iJ} by

$$\mathrm{SNR}_{iJj} = \frac{p_i |h_{ij}|^2}{N_0 + \sum_{e_{\tilde{i}\tilde{J}}, \tilde{i} \neq i,j} P_{r_{\tilde{i}\tilde{J}}} p_{\tilde{i}} |h_{\tilde{i}j}|^2}, \tag{18}$$

The achieved link rate of e_{iJ} is then computed by

$$\begin{aligned} \mu_{iJ}^{(C)} &= P_{r_{iJ}} \prod_{j \in J} \left(1 - \sum_{e_{j\tilde{J}}} P_{r_{j\tilde{J}}}\right) R\left(\min_{j \in J} \mathrm{SNR}_{iJj}\right) \\ &= \min_{\xi_{iJj}, \xi_{iJj} \geq 0, \sum_{j \in J} \xi_{iJj} = 1} P_{r_{iJ}} \prod_{j \in J} \left(1 - \sum_{e_{j\tilde{J}}} P_{r_{j\tilde{J}}}\right) R\left(\sum_{j \in J} \xi_{iJj} \mathrm{SNR}_{iJj}\right) \end{aligned} \tag{19}$$

[1]Note that, without the help of Theorem 2, the total number of link activation combinations is double exponential in the number of nodes.

where $R(\text{SNR})$ is a rate function of SNR, e.g., the Shannon capacity function $R(\text{SNR}) = \frac{1}{2}\log(1+\text{SNR})$. In (19), $P_{r_{iJ}} \prod_{j\in J}\left(1 - \sum_{e_{j\tilde{J}}} P_{r_{j\tilde{J}}}\right)$ represents the probability that node i is transmitting over link e_{iJ} and all nodes in J are receiving.

We use expected transmission power to represent the power vector of the communication realization, i.e., for each node i, $p_i^{(C)}$ is computed by

$$p_i^{(C)} = p_i \sum_{e_{iJ}} P_{r_{iJ}}. \tag{20}$$

The detailed suboptimal iterative optimization algorithm is given below.

Initialization: Construct an arbitrary transmission schedule and the corresponding configuration graph. Initialize $\boldsymbol{\lambda} \geq \mathbf{0}$, where $\boldsymbol{\lambda}$ is the Lagrangian vector introduced in (4).

Step 1: Update $\boldsymbol{\lambda}$ by

$$\boldsymbol{\lambda} = \boldsymbol{\lambda} - \delta_1 \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}), \tag{21}$$

where $\delta_1 > 0$ is a small step size parameter. Set all negative-valued elements of $\boldsymbol{\lambda}$ at zero.

Step 2: Initialize a communication realization C with all the probability parameters $P_{r_{iJ}}$ set at zero. Initialize all $\xi_{iJj} > 0$ where $\{\xi_{iJj}\}$ is the set of variables introduced in (19). Define the instantaneous objective function as

$$\begin{aligned} U^{(C)}(\boldsymbol{\mu}^{(C)}, \boldsymbol{p}^{(C)}) = &\sum_{e_{iJ}} \left[\frac{\partial U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})}{\partial \mu_{iJ}^{(G)}} - \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})}{\partial \mu_{iJ}^{(G)}} \right] \mu_{iJ}^{(C)} \\ &+ \sum_{i} \left[\frac{\partial U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})}{\partial p_i^{(G)}} - \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})}{\partial p_i^{(G)}} \right] p_i^{(C)}, \end{aligned} \tag{22}$$

where $\mu_{iJ}^{(C)}$ and $p_i^{(C)}$ are defined in (19) and (20), respectively.

Step 3: Update the set of $\{\xi_{iJj}\}$ variables in their gradient directions of $U^{(C)}(\boldsymbol{\mu}^{(C)}, \boldsymbol{p}^{(C)})$. Normalize the $\{\xi_{iJj}\}$ variables to satisfy $\xi_{iJj} \geq 0$ and $\sum_{j\in J} \xi_{iJj} = 1$ for all e_{iJ}.

Step 4: For every node i, update p_i in the gradient direction of $U^{(C)}(\boldsymbol{\mu}^{(C)}, \boldsymbol{p}^{(C)})$. Set $p_i = 0$ if the updated p_i has a negative value.

Step 5: For every link e_{iJ}, update its associated probability parameter $P_{r_{iJ}}$ in the gradient direction of $U^{(C)}(\boldsymbol{\mu}^{(C)}, \boldsymbol{p}^{(C)})$. Normalize the probability parameters to satisfy $0 \leq P_{r_{iJ}} \leq 1$ for all e_{iJ}, and for all i, $\sum_{e_{iJ}} P_{r_{iJ}} \leq 1$.

Step 6: Find a link e_{iJ} whose associated probability parameter $P_{r_{iJ}}$ is close to one, i.e., $P_{r_{iJ}} \geq 1-\epsilon$ where $\epsilon > 0$ is a pre-determined small constant. If such a link exists, set e_{iJ} to be active with probability one, and set other links that have transmission-reception conflicts with e_{iJ} to be inactive with probability one. Fix these probability parameters for the communication realization permanently.

Step 7: Go to Step 3 till all associated probability parameters are either set at one or zero. The communication realization C is then successfully constructed.

Step 8: Carry out an incremental update of the transmission schedule S using communication realization C.

Go to Step 1 till convergence.

The key idea of the suboptimal algorithm is that, in the communication realization construction, instead of searching for link activation combinations, the suboptimal algorithm assumes that all links are active with certain probabilities. By associating with each node a link activation probability vector, SNR experienced at each node for a particular link is replaced by the expected value, and the link rate is replaced by the expected link throughput. Objective functions such as (22) can therefore be written as functions of the node transmission powers and the link activation probabilities. Consequently, instantaneous utility optimization can be carried out via iteratively updating the power and probability parameters. Upon convergence, if the link activation probabilities are close to zero or one, then the corresponding link activation combination can be identified naturally. Note that the idea of using probabilistic data association to avoid combinatorial search was originally proposed in [11].

VII. Simulation Results

In this section, we use computer simulations to illustrate the performances of the optimal and suboptimal algorithms proposed in the previous sections. We consider a wireless network with $N = 16$ nodes located on a grid. Positions of the grid points are $(2x, 2y)$ where $x, y \in \{0, 1, 2, 3\}$, as illustrated in Figure 3. We consider a single multicast session where the source node located at $(0, 0)$ wants to send common information to three destination nodes located at $(0, 6)$, $(6, 0)$, $(6, 6)$. The wireless channels are assumed to be memoryless with additive Gaussian noise of zero mean and unit variance. The channel gain between node i and node j is set at $h_{ij} = d_{ij}^{-3/2}$ where d_{ij} is the distance between the two nodes. In order to avoid the complication of power optimization, we assume that, in any time slot, each node can transmit over at most one active link, with a fixed transmission power of 100. We use the capacity function $r = \frac{1}{2}\log(1 + \mathrm{SNR})$ to calculate the rate of a link whose signal to noise ratio is SNR. The objective is to maximize the multicast throughput from the source node to the three destination nodes.

In this example, the iterative algorithm proposed in Section V is strictly optimal. We adopts an adaptive step size adjustment approach whose detail is skipped in the paper. We also set $\epsilon = 2$ in Step 1.5 in the algorithm introduced in Section V. In other words, any cut with a value larger than 3 times the minimum $s - T$ cut is removed from the cut list in the algorithm. Note that according to Theorem 2, we only need to consider wireless hyper-arc links with at most 3 destinations. Figure 4 illustrates the performances of the optimal and the suboptimal algorithms, as well as the performance of the constrained optimal algorithm that only considers point-to-point links. Because the iterative optimization algorithms involves many step-size parameters, obtaining a fast and smooth utility convergence is generally difficult. Nevertheless, it can be clearly seen from the results that either excluding the hyper-arc links or avoiding the search of link activation combinations can lead to significant multicast throughput degradation. The low complexity optimal algorithm is therefore important in the sense of providing performance reference for the development of efficient suboptimal algorithms.

VIII. Conclusions

We showed under certain conditions that the impact of physical, data-link layer configurations of a wireless network can be characterized using a configuration graph. Network utility optimization can consequently be carried out via iterations that optimize network layer algorithms over the configuration graph and revise physical, data-link layer configurations to improve the configuration graph. For a class of wireless multicast networks with intra-session network coding, we showed that the number of links involved in network utility optimization is only polynomial, as opposed to exponential, in the number of nodes. Low complexity optimal algorithms are developed to enable computer simulations of multicast utility optimization in moderate-sized networks.

References

[1] L. Wan, J. Luo, and A. Ephremides, *An Iterative Framework for Optimizing Multicast Throughput in Wireless Networks,* IEEE ISIT, Toronto, Canada, Jul. 2008.

[2] L. Wan and J. Luo, *Wireless Multicasting via Iterative Optimization,* IEEE ISIT, Austin, TX, Jun. 2010.

[3] D. Bertsekas and R. Gallager, *Data Network,* 2nd Ed., Englewood Cliffs, NJ: Prentice Hall, 1992.

[4] T. Cover and J. Thomas, *Elements of Informaiton Theory,* 2nd Ed., John Wiley & Sons, Inc., 2005.

[5] J. Wieselthier, G. Nguyen, and A. Ephremides, *Energy-efficient Broadcast and Multicast Trees in Wireless Networks*, Mobile Networks and Applications, Vol. 7, pp. 481-492, 2002.

[6] D. Lun, N. Ratnakar, M. Medard, R. Koetter, D. Karger, T. Ho, E. Ahmed, and F. Zhao, *Minimum-cost Multicast over Coded Packet Networks*, IEEE Trans. Inform. Theory, Vol. 52, pp. 2608-2623, Jun. 2006.

[7] Y. Wu, M. Chiang, and S.Kung, *Distributed Utility Maximization for Network Coding Based Multicasting: A Critical Cut Approach,* IEEE WIOPT, Boston, MA, Apr. 2006.

[8] J. Yuan, Z. Li, W. Yu, and B. Li, *A Cross-Layer Optimization Framework for Multihop Multicast in Wireless Mesh Networks,* IEEE J. Sel. A. Commun., Vol. 24, pp. 2092-2103, Nov. 2006.

[9] Y. Sagduyu and A. Ephremides, *On Joint MAC and Network Coding in Wireless Ad Hoc Networks,* IEEE Trans. Inform. Theory, Vol. 53, pp. 3697-3713, Oct. 2007.

[10] R. Ahlswede, N. Cai, S. Li, and R. Yeung, *Network Information Flow*, IEEE Trans. Inform. Theory, Vol. 46, pp. 1204-1216, Jul. 2000.

[11] Y. Bar-Shalom and E. Tse, *Tracking in A Clustered Enviroment with Probabilistic Data Association,* Automatica, Vol. 11, pp. 451460, Sep. 1975.

[12] R. Etkin, D. Tse, and H. Wang, *Gaussian Interference Channel Capacity to Within One Bit*, submitted to IEEE Transactions on Information Theory.

[13] J. Luo and A. Ephremides, *Wireless Multicasting with Channel Uncertainty*, CISS, Baltimore, MD, Mar. 2007.

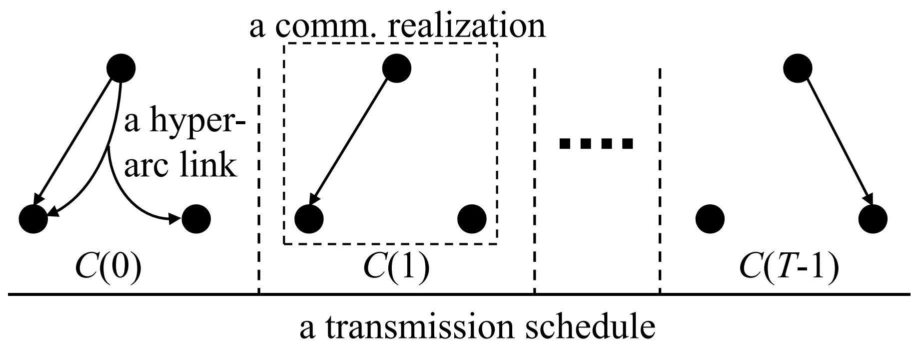

Fig. 1. An illustration of links, communication realizations and transmission schedule.

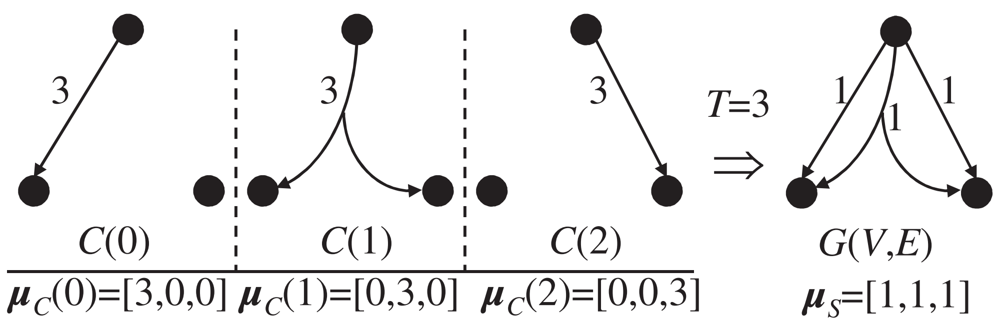

Fig. 2. An illustration of configuration graph construction.

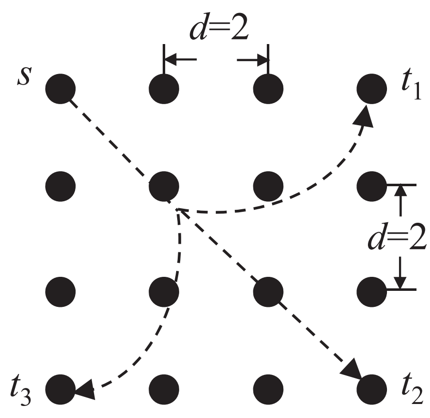

Fig. 3. A grid network with 16 nodes and one multicast session.

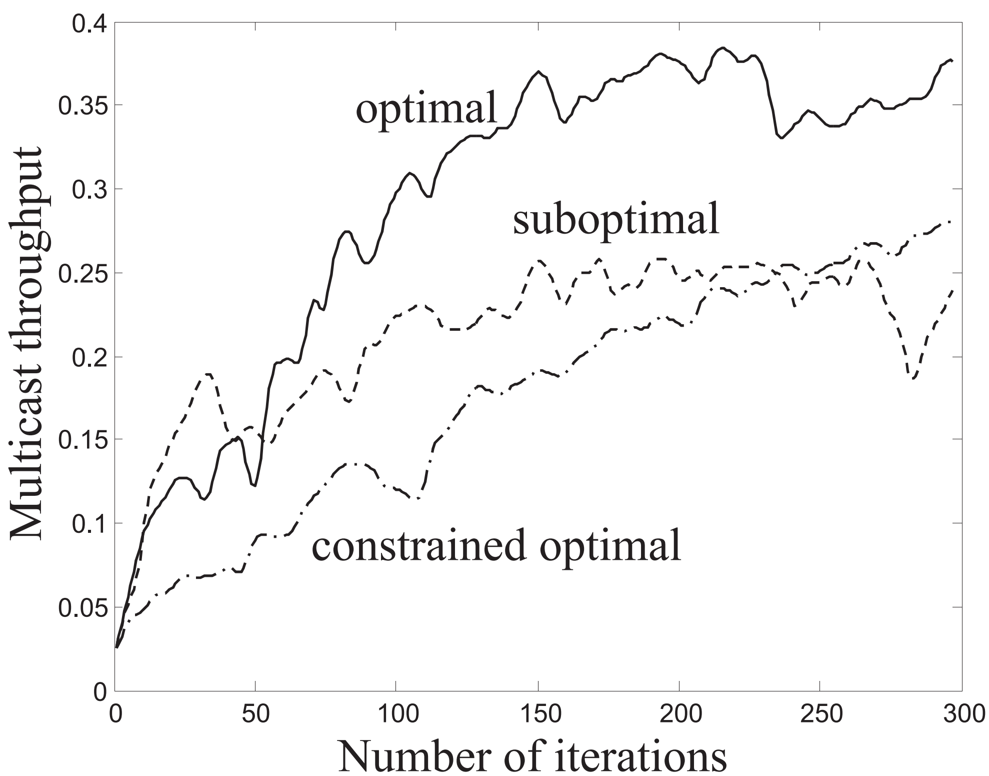

Fig. 4. Multicast throughputs of the optimal algorithm, the suboptimal algorithm, and the constrained optimal algorithm with only point-to-point links.