On the Complexity of Wireless Multicast Optimization

Lihua Wan, *Student Member, IEEE*, and Jie Luo, *Member, IEEE*

***Abstract*—This letter considers a class of utility optimization problems in wireless networks. Under a set of physical layer assumptions, we show that the impact of physical and data-link layer configurations to the upper layers in a wireless network can be characterized using a configuration graph, which is similar to a wireline network topology graph. Network layer utility optimization can consequently be carried out via iterations that optimize network layer algorithms over the configuration graph and *incrementally* revise physical, data-link layer configurations to improve the configuration graph. For a class of wireless multicast networks with network coding, we show 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*—Complexity, utility optimization, 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. Extending such a graphic model to wireless networks, however, faces two key challenges. First, a wireless link can support a positive throughput so long as its channel gain is not strictly zero. This property often leads to a fully connected wireless network topology graph, which is not informative. Second, due to the open nature of wireless media, wireless node can communicate common information to multiple receivers simultaneously 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 [2]), the total number of feasible links in a wireless network can be exponential in the number of nodes.

The idea of using hyperarc links to model wireless broadcast transmissions was firstly introduced by Lun *et al.* in [2]. 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 [3] to optimize a generalized utility under the assumption that the topology graph-based network model with non-interfering links remains valid. In [4], 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 [5], 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.

II. System Modeling and Iterative Utility Optimization

Consider a wireless network with stationary memoryless channels and equal-length time slots. Let V be the set of nodes. 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 $\mu_{iJ}^{(C)}(t)$. 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 $S = \{C(0), C(1), \ldots, C(\mathcal{T}-1)\}$ as the periodic extension of a communication realization sequence $C(0), C(1), \ldots, C(\mathcal{T}-1)$, with $\mathcal{T}$ being the period and $C(i)$ being the communication realization of time slots $k\mathcal{T}+i$, where $k = \ldots, -1, 0, 1, \ldots$ is an integer. The communication rate vector and the power of the nodes specified by transmission schedule S are denoted by $\boldsymbol{\mu}^{(S)} = \boldsymbol{\mu}(S) = \frac{1}{\mathcal{T}}\sum_{t=0}^{\mathcal{T}-1}\boldsymbol{\mu}(C(t))$ and $\boldsymbol{p}^{(S)} = \frac{1}{\mathcal{T}}\sum_{t=0}^{\mathcal{T}-1}\boldsymbol{p}(C(t))$, respectively. An illustration of links, communication realizations and transmission schedule is given in Figure 1.

Note that the physical and data-link layer model described above implies two key assumptions. First, we assume decoding and forward as opposed to amplifying and forward for

Manuscript received June 22, 2012. The associate editor coordinating the review of this letter and approving it for publication was L. Le.

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 the National Science Foundation under Grants CCF-0728826, CCF-1016985, and CNS-1116134. The results have been presented in part at IEEE ISIT 2010 [1].

Digital Object Identifier 10.1109/WCL.2012.083012.120462

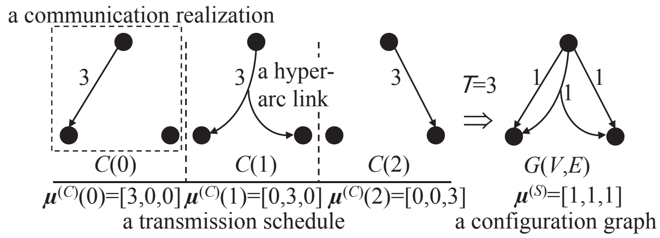

Fig. 1. An illustration of links, communication realizations, transmission schedule, and configuration graph construction.

information relay. Second, we assume that messages transmitted over different links should be encoded and decoded independently at the physical layer.

Given a transmission schedule $S = \{C(0), \ldots, C(\mathcal{T}-1)\}$. We construct a configuration graph $G^{(S)}(V,E) = G(S)$ with E being the union of links of all communication realizations. Let $\boldsymbol{\mu}^{(S)}$ 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 $G^{(S)}(V,E)$. An illustration of the configuration graph construction is given in Figure 1. 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 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 $\{\tilde{\boldsymbol{\mu}} | \tilde{\boldsymbol{\mu}} \leq \boldsymbol{\mu}^{(S)}\}$ 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

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

where, given a transmission schedule S, we assume that the utility $U(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)})$ and the set of constraints $\boldsymbol{H}(\boldsymbol{\mu}^{(S)}, \boldsymbol{p}^{(S)})$ are functions of link throughput $\boldsymbol{\mu}^{(S)}$ and transmission power $\boldsymbol{p}^{(S)}$, both determined by S.

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

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

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

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.

$$\min_{\boldsymbol{\lambda} \geq \mathbf{0}} \max_{S, G^{(S)}(V,E)=G(S)} L = U(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}), \tag{3}$$

where L is the Lagrangian function and $\boldsymbol{\lambda} \geq \mathbf{0}$ is the vector of Lagrangian relaxation parameters.

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, (3) can only be solved by iterative and constructive updates of the transmission schedule. Next, we show that, given a fixed Lagrangian parameter $\boldsymbol{\lambda} \geq \mathbf{0}$, the Lagrangian function L in (3) can indeed be maximized *monotonically* via iterative and *incremental* updates of S, where the meaning of "incremental" update is specified below.

Definition 1: Let $S_1 = \{C_1(0), C_1(1), \ldots, C_1(\mathcal{T}_1 - 1)\}$ and $S_2 = \{C_2(0), C_2(1), \ldots, C_2(\mathcal{T}_2 - 1)\}$ be two transmission schedules whose periods are $\mathcal{T}_1$ and $\mathcal{T}_2$, respectively. We say S_2 is incrementally different from S_1 if there exists a nonnegative integer k such that $\mathcal{T}_2 = k\mathcal{T}_1 + 1$, and for all nonnegative integers $\tilde{k} < k$ and $i \leq \mathcal{T}_1 - 1$, we have $C_2(\tilde{k}\mathcal{T}_1 + i) = C_1(i)$. 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 (3) for a given $\boldsymbol{\lambda}$. Define $L_{\boldsymbol{\lambda}}^*$ as $L_{\boldsymbol{\lambda}}^* = \max_{S, G^{(S)}(V,E)=G(S)} L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})$, where $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}, \boldsymbol{p}) = U(\boldsymbol{\mu}, \boldsymbol{p}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{\mu}, \boldsymbol{p})$. 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, and $L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_2, \boldsymbol{p}_2) > L_{\boldsymbol{\lambda}}(\boldsymbol{\mu}_1, \boldsymbol{p}_1)$.

The proof of Theorem 1 is skipped. Theorem 1 implies that the Lagrangian optimization problem (3) can be solved via iterative updates of $\boldsymbol{\lambda}$ and $(\boldsymbol{\mu}, \boldsymbol{p})$ pair, where each update of the $(\boldsymbol{\mu}, \boldsymbol{p})$ pair only involves an incremental revision, as opposed to a direct construction, of the transmission schedule.

III. Bounding The Number of Hyperarc Links in Multicast Utility Optimization

According to Theorem 1, the key task in incremental revision of the transmission schedule 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 an exhaustive search of the communication realizations, then its complexity is exponential in the number of feasible links. Because the number of feasible links is exponential in the number of nodes, the complexity is therefore doubly 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 a class of multicast utility optimization problems, optimal utility maximization can be carried out 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 $\tilde{J}$ can be achieved using any hyperarc link e_{iJ} so long as $\tilde{J} \subseteq J$, $\boldsymbol{\mu}$ 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 $\boldsymbol{\mu}$. ■

Now let us consider a wireless network where a source node s wants to multicast common information reliably to a set of destination nodes T. Let the throughput of the multicasting be denoted by R_{sT}. Note that, given a transmission schedule S, and its corresponding configuration graph $G^{(S)}(V, E)$, the exact expression of $R_{sT}(\boldsymbol{\mu}^{(G)})$ as a function of $\boldsymbol{\mu}^{(G)}$ is determined by the network layer protocol. If optimal network coding is applied to the configuration graph, then R_{sT} equals the max-flow of the minimum $s - T$ cut of $G^{(S)}(V, E)$ [6][2]. If only routing (i.e., copying and forwarding packets at intermediate nodes) is allowed at the network layer, expression of the $R_{sT}(\boldsymbol{\mu}^{(G)})$ function becomes not only complicated, but also less than or equal the multicast throughput with network coding [6].

The following theorem shows that, in terms of maximizing R_{sT}, one does not need to consider any link e_{iJ} with $|J| > |T|$.

Theorem 2: Consider a wireless network where source node s wants to multicast common information to a set of destination nodes T. Given a transmission schedule $\tilde{S}$ with rate $\boldsymbol{\mu}^{(\tilde{S})}$ and power $\boldsymbol{p}^{(\tilde{S})}$, let $R_{sT}^{(\tilde{S})}$ be the max-flow of the minimum $s-T$ cut of the corresponding configuration graph. There exist another transmission schedule S with rate $\boldsymbol{\mu}^{(S)}$ satisfying $\mu_{iJ}^{(S)} = 0$, $\forall i, J, |J| > |T|$, and with power $\boldsymbol{p}^{(S)}$ satisfying $\boldsymbol{p}^{(S)} = \boldsymbol{p}^{(\tilde{S})}$, such that $R_{sT}^{(S)} = R_{sT}^{(\tilde{S})}$.

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

$$\begin{aligned} &\tilde{\mu}_{iJ} \geq \sum_{j \in J} \tilde{x}_{iJj}^{(t)}, \quad \forall (i, J), e_{iJ} \in E, t \in T, \\ &\sum_{\{J | (i,J), e_{iJ} \in E\}} \sum_{j \in J} \tilde{x}_{iJj}^{(t)} - \sum_{\{j | (j,I), e_{jI} \in E, i \in I\}} \tilde{x}_{jIi}^{(t)} = \sigma_i^{(t)}, \\ &\hspace{20em} \forall i \in V, t \in T, \\ &\sigma_s^{(t)} = R_{sT}, \quad \sigma_t^{(t)} = -R_{sT}, \quad \text{and } \sigma_{i \neq s,t}^{(t)} = 0. \end{aligned} \tag{4}$$

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 $x_{iJj}^{(t)} \geq 0$ such that the degraded configuration rate vector $\boldsymbol{\mu}$ and the set of virtual link throughput $x_{iJj}^{(t)}$ satisfy the same inequality given in (4).

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 (4).

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 (4) remain satisfied by $\boldsymbol{\mu}$ and $\{x_{iJj}^{(t)}\}$. Go to Step 1.

Step 3: Assume that we can find $t \in T$ and $j \in J$, with $x_{iJj}^{(t)} > 0$. 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 (4) 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 $|J| > |T|$. 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 (4) is satisfied in every step, it remains satisfied when the algorithm stops. The validity of (4) implies that R_{sT} is no larger than the max-flow of the minimum $s-T$ cut of configuration graph $G(V, E)$. Therefore the desired transmission schedule $\tilde{S}$ can be constructed from S by appropriately discarding information at the nodes. ■

Consider a class of wireless multicast networks with single multicast session from node s to node set T. Assume that, given a transmission schedule, optimal network coding is applied to achieve a multicast throughput equaling the max-flow of the minimum $s - T$ cut of the corresponding configuration graph. Let the utility optimization problem be formulated as,

$$\max_{S, G^{(S)}(V,E) = G(S)} U(R_{sT}(\boldsymbol{\mu}^{(G)}), \boldsymbol{p}^{(G)}),$$
$$\text{s.t. } \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)}) \geq \boldsymbol{0}, \tag{5}$$

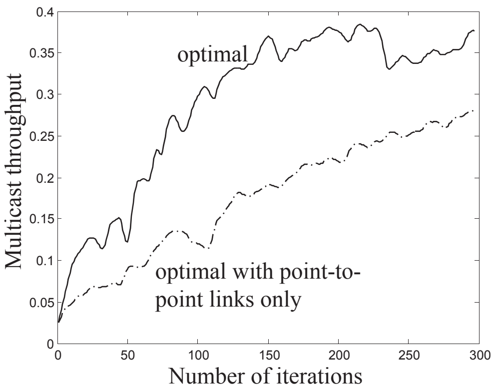

Fig. 2. Multicast throughputs of the optimal algorithms with and without hyperarc links.

where $-U(R_{sT}(\boldsymbol{\mu}^{(G)}), \boldsymbol{p}^{(G)})$ and $\boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})$ are both convex in $(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})$. Also assume that for any rate vector $\boldsymbol{\mu}_d^{(G)}$ that is a degraded version of $\boldsymbol{\mu}^{(G)}$, we have $\boldsymbol{H}(\boldsymbol{\mu}_d^{(G)}, \boldsymbol{p}^{(G)}) \geq \boldsymbol{H}(\boldsymbol{\mu}^{(G)}, \boldsymbol{p}^{(G)})$. According to Theorem 1, the multicast utility can be optimized using the iterative framework introduced in Section II with only incremental revision of the transmission schedule in each iteration. According to Theorem 2, the maximum number of feasible links involved in each incremental schedule revision is only polynomial in the number of nodes.

IV. Further Complexity Reduction and Computer Simulation

Note that with the complexity reduction introduced in Section III, the complexity of an optimal multicast utility maximization algorithm is still exponential (being reduced from doubly exponential) in the number of nodes. The complexity of an exhaustive search of link activation combinations can still be prohibitive for computer simulations even for a small scale network. Fortunately, we show in the following that the complexity of link combination search can be further reduced. Even though the further complexity reduction does not change the complexity scaling law, simulations of optimal utility maximization in some moderate-sized networks indeed become feasible.

Because a wireless node cannot simultaneously transmit and receive information in the same channel, link activations involving the transmission-reception conflict at any node should not be considered in the communication realization construction. Moreover, with optimal network coding applied to the configuration graph, throughput of a multicast session is bottlenecked by the max-flow of the minimum $s-T$ cut(s). Therefore, in the search of 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. The detailed implementation of these ideas is quite standard and is therefore skipped.

In the computer simulation, we consider a wireless network with $N = 16$ nodes located on a grid. Positions of the grid points are $(2x, 2y)$ where $x, y \in \{0, 1, 2, 3\}$. 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 $r = \frac{1}{2}\log(1 + \text{SNR})$ 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. Note that according to Theorem 2, we only need to consider wireless hyperarc links with at most 3 destinations.

Figure 2 illustrated the performances of the optimal algorithms with only point-to-point links and with point-to-multipoint hyperarc links. Complexity reduction is implemented for both algorithms. For the optimal algorithm with hyperarc links, the average and the maximum numbers of link combinations searched in one iteration are in the order of 10^7 and 10^{10}, respectively. It is clearly seen that excluding the hyperarc links in network optimization can lead to significant multicast throughput degradation.

References

[1] L. Wan and J. Luo, "Wireless multicasting via iterative optimization," *2010 IEEE ISIT*.

[2] 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. Inf. Theory*, vol. 52, pp. 2608–2623, June 2006.

[3] Y. Wu, M. Chiang, and S. Kung, "Distributed utility maximization for network coding based multicasting: a critical cut approach," *2006 IEEE WIOPT*.

[4] J. Yuan, Z. Li, W. Yu, and B. Li, "A cross-layer optimization framework for multihop multicast in wireless mesh networks," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 2092–2103, Nov. 2006.

[5] Y. Sagduyu and A. Ephremides, "On joint MAC and network coding in wireless ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 53, pp. 3697–3713, Oct. 2007.

[6] R. Ahlswede, N. Cai, S. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, pp. 1204–1216, July 2000.