Cross-Layer Design of Coded Multicast for Wireless Random Access Networks

Ketan Rajawat, Nikolaos Gatsis, Seung-Jun Kim, and Georgios B. Giannakis

***Abstract*—Joint optimization of network coding and Aloha-based medium access control (MAC) for multi-hop wireless networks is considered. The multicast throughput with a power consumption-related penalty is maximized subject to flow conservation and MAC achievable rate constraints to obtain the optimal transmission probabilities. The relevant optimization problem is inherently non-convex and hence difficult to solve even in a centralized manner. A successive convex approximation technique is employed to obtain a Karush-Kuhn-Tucker solution. A separable problem structure is obtained and the dual decomposition technique is adopted to develop a distributed solution. The algorithm is thus applicable to large networks, and amenable to online implementation. Numerical tests verify performance and complexity advantages of the proposed approach over existing designs. A network simulation with implementation of random linear network coding shows performance very close to the one theoretically designed.**

***Index Terms*—Network coding, random access, cross-layer optimization.**

I. Introduction

TACTICAL wireless ad hoc networks play a crucial role when it comes to communication dominance in the battlefield. Important requirements for such networks include resilience and efficiency. In order to accommodate units such as soldiers, military vehicles, or a field hospital, tactical networks are typically multi-hop; see e.g., Fig. 1. Thus, it becomes important to deploy decentralized protocols, so that no single node exposes vulnerability of the network. Aloha is a simple, yet widely deployed medium access control (MAC) protocol, whose operation is distributed and resilient to both random, and jamming-induced link failures.

This work focuses on multicasting applications for tactical networks, where information needs to be distributed from a source to multiple target nodes. Efficient multicasting is realized using network coding whereby nodes perform encoding functions on packets traveling in the network. Network coding for wireline networks can achieve the multicast capacity using random or deterministic linear strategies [1], [2]. Although the multicast capacity region of wireless networks is not known, the rate region achievable with linear network coding has been characterized [3], [4]. Both wireline and wireless rate regions can be practically achieved by fully distributed random linear network coding strategies [2], [5], [6]. Random network coding also results in each packet getting distributed spatially, thus providing some inherent protection against eavesdropping. Moreover, wedding of network coding with Aloha is particularly attractive for military networks because the network operation becomes extremely simple. Specifically, given the access probabilities, each node simply transmits random linear combinations of the packets in its buffer at a pre-specified rate [2], [5], [6]. The protocol does not require ACKs (nor retransmissions) in the MAC or network layers.

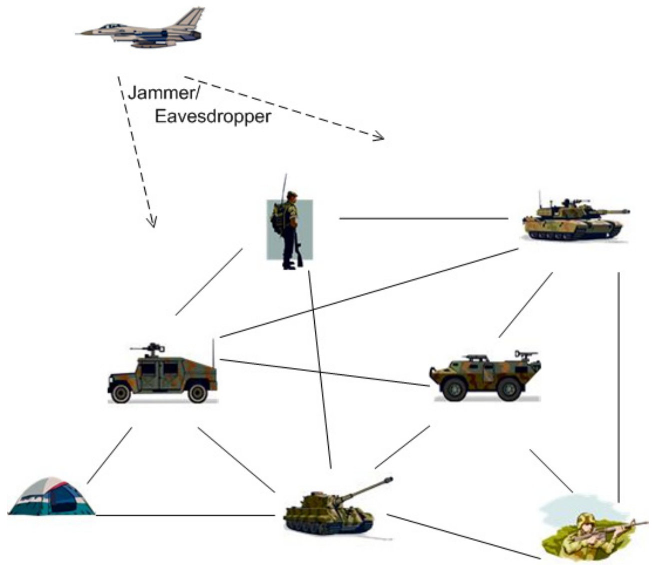

Fig. 1. A Tactical Network

The present paper considers joint design of wireless *multi-hop* networks employing random network coding and slotted Aloha. A cross-layer optimization problem is formulated, where network coding rates (also called subgraphs) and transmission probabilities are jointly determined to maximize a network-wide objective. In contrast to simple protocol operation, the joint design itself is notoriously difficult. This is because the Aloha capacity region, even for three nodes with fully backlogged queues is described by non-convex signomial constraints [7].

Joint design of network coding and Aloha MAC has been undertaken for *single-hop* network topologies. The performance of slotted Aloha for star networks was analyzed in [8]. A game theoretic approach for throughput maximization in a single-hop setting was proposed in [9]. The performance for two-hop (relay) networks with bi-directional traffic was reported in [10]. These works underline the significance of the cross-layer approach for coded Aloha networks.

Manuscript received 15 November 2010; revised 29 April 2011. This work was supported by NSF grants CCF-0830480, 1016605; and ECCS-0824007, 1002180; Part of this paper has been submitted to the 45th Annual Conference on Information Sciences and Systems, Princeton, NJ.

The authors are with Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, USA (e-mails: {ketan,gatsisn,seungjun,georgios}@umn.edu).

Digital Object Identifier 10.1109/JSAC.2011.111207.

Joint design of coded Aloha multi-hop networks has been attempted. A branch-and-bound method was employed in [11] to obtain globally optimal transmission probabilities and subgraphs. While offering a benchmark for comparison, the resultant protocol may be too complex for use in large networks. A heuristic algorithm was proposed in [12], where the access probabilities and network coding rates were optimized separately. Albeit practical, the approach in [12] is suboptimal, and does not provide performance guarantees.

In this work, a successive convex approximation approach is adopted to obtain solutions that are guaranteed to be locally optimal. Convex surrogate problems are constructed in such a way to guarantee convergence of the overall algorithm to a Karush-Kuhn-Tucker (KKT) point of the original non-convex problem, and thus enable a tractable, locally optimal solution, even for large networks. This requires an efficient re-formulation of the MAC constraints, which constitutes our first contribution.

The constructed surrogate problems are not amenable to distributed solution. To this end, a separable structure is created by further approximating the problem, while still preserving KKT optimality of the overall algorithm. This forms our second contribution, which involves approximating even convex terms in order to make the overall problem separable. The dual subgradient method is employed for the resulting convex problems, where the primal and dual updates can be performed in a parallel and distributed fashion. A primal solution yielding near-optimal access probabilities and network coding rates, is recovered by primal averaging. An online network control protocol is also introduced to perform the optimization task.

The coded Aloha scheme enjoys features attractive for military networks. Specifically, the resulting protocol is simple and decentralized, whereby every node transmits random linear combinations with its access probability. Moreover, the optimal designs of this work take into account the packet loss probability due to the wireless medium (erasures); this feature can be leveraged to ensure jamming-resilience by preemptively setting higher erasure probabilities for any part of the network that is likely to be jammed. Furthermore, the subgradient-based online optimization and control uses a constant stepsize, which enables adaptation to slowly time-varying environments, for instance, due to mobility of branch units, or non-stationary jamming.

The rest of the paper is organized as follows. The system model and the problem statement are given in Sec. II. The successive convex approximation algorithm is described in Sec. III. A distributed solution and its online implementation are provided in Sec. IV. Numerical tests as well as a network simulation with suitable implementation of random linear network coding are presented in Sec. V, followed by the conclusions in Sec. VI.

II. System Model and Problem Statement

A. System Model

Consider a wireless network represented by a hypergraph $(\mathcal{N}, \mathcal{A})$ with the set of nodes $\mathcal{N}$ and the set of hyperarcs $\mathcal{A}$. A hyperarc $(i, J) \in \mathcal{A}$ models the broadcast channel between node i, and the set of receivers $J \subset \mathcal{N}$. The super-set $\mathcal{J}_i$ collects all such sets of receivers $\{J|(i, J) \in \mathcal{A}\}$ for node $i \in \mathcal{N}$. The one-hop neighborhood of node i is denoted by $N(i)$ and includes all nodes belonging to at least one set $J \in \mathcal{J}_i$. The hyperarc model is very general and allows nodes to transmit at different rates and powers on each hyperarc; see e.g., [3], [4], [6]. It also subsumes point-to-point, and broadcast-only scenarios, as detailed later in Section II-C.

Consider further a multicast session involving a source node $s \in \mathcal{N}$, and a set of sink nodes $\mathcal{T} \subset \mathcal{N}$. The aim is to maximize the multicast rate R at which node s can transmit the same information to all the sink nodes $t \in \mathcal{T}$. The network operates in a time-slotted fashion.

Leveraging the characterization of the multicast rate region in [3], [4], the present section formulates a cross-layer optimization problem to maximize the multicast rates supported by a slotted Aloha network. To this end, a set of auxiliary variables $\{r_{ij}^{(t)}\}$ is introduced, with $r_{ij}^{(t)} \geq 0$ representing the virtual transmission rate (also called virtual flow) from node i to a neighboring node $j \in N(i)$ for sink $t \in \mathcal{T}$. Virtual flows abide by the flow conservation constraints [3]

$$\sum_{j \in N(i)} r_{ij}^{(t)} - \sum_{j: i \in N(j)} r_{ji}^{(t)} = R\mathbf{1}_{\{i=s\}} - R\mathbf{1}_{\{i=t\}}, \quad i \in \mathcal{N}, t \in \mathcal{T} \quad (1)$$

where $\mathbf{1}_{\{.\}}$ is the indicator function that takes the value one when the expression inside the curly brackets is true, and zero otherwise.

Optimization solvers usually require all constraints to be expressed as inequalities. Therefore, the following relaxed version of virtual flow constraints (1) is used here

$$\sum_{j \in N(i)} r_{ij}^{(t)} - \sum_{j: i \in N(j)} r_{ji}^{(t)} \geq R\mathbf{1}_{\{i=s\}}, \quad t \in \mathcal{T}, i \in \mathcal{N} \setminus \{t\}. \quad (2)$$

To obtain (2), note that in (1), the set of equations for $i = t$ can be omitted since they are implied by the other equations. Relaxation of the flow constraints for $i \neq t$ is then equivalent to allowing each node i to transmit at higher rate than received, which amounts to adding virtual sources at all nodes. Note, however, that sending nonzero flow from these virtual sources to the sinks can never increase R, which is the flow from s to $t \in \mathcal{T}$. Thus, even if the optimal solution has some nodes injecting extra flows, they can all be set to zero without impeding R.

B. Characterization of MAC Constraints

The MAC layer employs the slotted Aloha protocol. At every time slot, each node $i \in \mathcal{N}$ transmits on hyperarc (i, J) with probability p_{iJ} and (instantaneous) physical layer (PHY) rate c_{iJ}. The transmissions of different nodes are independent. Not all nodes in J can decode the packets received from i, because of collisions or erasures. Let $I(m)$ denote the set of nodes whose transmissions interfere with the reception at node m. Reception at node m may fail (a) due to collisions—when a node $j \in I(m)$ or m itself (half-duplex contraint) is transmitting at the same time slot—or (b) due to erasures caused by impairments of the wireless medium or jamming. Erasure means that although the link may be collision-free, the receiving end cannot decode the transmitted packets with some probability due to, e.g., fading. Occurrence of erasures is independent of collisions. To summarize, a transmission from i to m is successful when (a) no node $j \in (I(m) \cup \{m\}) \backslash \{i\}$ transmits, and (b) there is no erasure on link (i, m).

Let S_{iJ}^m denote the event that a packet transmitted on hyperarc (i, J) is correctly decoded by node $m \in J$, and define $q_i := 1 - \sum_{J \in \mathcal{J}_i} p_{iJ}$, the probability that i remains silent. Assume that erasures happen independently across links and time slots, and let $1 - s_{iJm}$ be the probability of erasure on the link (i, m) of a packet transmitted at PHY rate c_{iJ}. Assuming fully backlogged queues at the link layer, so that all nodes have packets to transmit at every time slot, one can write the probability of S_{iJ}^m as

$$\Pr\left(S_{iJ}^m\right) = s_{iJm} \prod_{j \in (I(m) \cup \{m\}) \backslash \{i\}} q_j \quad m \in J, (i, J) \in \mathcal{A}. \tag{3}$$

Next, introduce for each $K \subset N(i)$ the probability b_{iJK} that at least one node in K receives the packets injected on the hyperarc (i, J) correctly; i.e,

$$b_{iJK} := \Pr\left(\bigcup_{m \in K} S_{iJ}^m\right), \qquad K \subset N(i), (i, J) \in \mathcal{A}. \tag{4}$$

It is clear from this definition that $b_{iJK} = 0$ if $J \cap K = \varnothing$. From the inclusion-exclusion principle [13, p. 6], the probability of the union of events in (4) can be expanded as

$$\Pr\left(\bigcup_{m \in K} S_{iJ}^m\right) = \sum_{k=1}^{|J \cap K|} \sum_{\substack{M \subset J \cap K \\ |M| = k}} (-1)^{k-1} \Pr\left(\bigcap_{m \in M} S_{iJ}^m\right),$$
$$K \subset N(i), (i, J) \in \mathcal{A}. \tag{5}$$

Define $I(M)$, for a set of nodes $M \subset \mathcal{N}$, as the set of nodes whose transmissions interfere with at least one node in M; i.e., $I(M) = \bigcup_{m \in M} I(m)$. The probability that all nodes in M decode the packet is

$$\Pr\left(\bigcap_{m \in M} S_{iJ}^m\right) = \left(\prod_{m \in M} s_{iJm}\right)\left(\prod_{j \in (I(M) \cup M) \backslash \{i\}} q_j\right),$$
$$M \subset J, i \in \mathcal{N}. \tag{6}$$

The average rate at which packets are injected in the hyperarc (i, J) is given by $z_{iJ} := c_{iJ} p_{iJ}$. The virtual flow rate for each sink $t \in \mathcal{T}$ can be related to $\{z_{iJ}\}_{(i,J) \in \mathcal{A}}$ through the following set of inequalities [4]

$$\sum_{j \in K} r_{ij}^{(t)} \le \sum_{J \in \mathcal{J}_i} z_{iJ} b_{iJK}, \qquad K \subset N(i), i \in \mathcal{N}, t \in \mathcal{T}. \tag{7}$$

The right-hand side represents the rate at which packets transmitted by node i reach at least one node in K, through various hyperarcs (i, J). Combining (4)–(6), the virtual flow constraints (7) become

$$\sum_{j \in K} r_{ij}^{(t)} \le \sum_{J \in \mathcal{J}_i} c_{iJ} p_{iJ} \sum_{k=1}^{|J \cap K|} \sum_{\substack{M \subset J \cap K \\ |M| = k}} (-1)^{k-1} \prod_{m \in M} s_{iJm},$$
$$K \subset N(i),\ i \in \mathcal{N},\ t \in \mathcal{T}. \tag{8}$$

C. Problem Formulation

The problem of interest is to maximize the multicast throughput R while minimizing energy consumption, subject to network coding and random access constraints. Since higher values of q_i should translate to lower energy consumption at node $i \in \mathcal{N}$, a convex, decreasing function $v_i(q_i)$ is used as a cost to penalize the energy consumption.

First, the following definition is introduced, in order to streamline the notations in (8)

$$\mathcal{C}_{iK} := \{(J, M, k) | J \in \mathcal{J}_i, M \subset J \cap K, k = |M|\}. \tag{9}$$

Also define $\mathcal{I}_{iM} := (I(M) \cup M) \setminus \{i\}$ and $s_{iJM} := \prod_{m \in M} s_{iJm}$. The overall optimization problem is formulated as follows:

$$(\mathbf{P}^0) \qquad \min_{R \ge 0, \{r_{ij}^{(t)} \ge 0\}} \quad \sum_{i \in \mathcal{N}} v_i(q_i) - R \tag{10a}$$

$$\text{s.t.} \quad \sum_{j : i \in N(j)} r_{ji}^{(t)} + R\mathbf{1}_{\{i=s\}} - \sum_{j \in N(i)} r_{ij}^{(t)} \le 0,$$
$$t \in \mathcal{T}, i \in \mathcal{N} \setminus \{t\} \tag{10b}$$

$$\sum_{j \in K} r_{ij}^{(t)} + \sum_{(J,M,k) \in \mathcal{C}_{iK}} (-1)^k c_{iJ} p_{iJ} s_{iJM} \prod_{j \in \mathcal{I}_{iM}} q_j \le 0,$$
$$K \subset N(i), i \in \mathcal{N}, t \in \mathcal{T} \tag{10c}$$

$$\sum_{J \in \mathcal{J}_i} p_{iJ} + q_i - 1 \le 0, \qquad i \in \mathcal{N}. \tag{10d}$$

Note that (10d) is a relaxed version of the original equality constraint $\sum_{J \in \mathcal{J}_i} p_{iJ} + q_i = 1$. If the optimal solution is such that strict inequality holds in (10d) for a node $i \in \mathcal{N}$, then the value of q_i can be increased without changing any p_{iJ}. This will likely decrease the probability of collisions due to node i for other nodes, thus allowing at least as much throughput as before.

Problem $(\mathbf{P}^0)$ is non-convex, because constraint (10c) is non-convex. A logarithmic change of variables as in [14, Sec. 2] does not convexify the problem either, as (10b) and (10c) both become signomial constraints. For this reason, a successive convex approximation approach is pursued in the next section to obtain a KKT optimal solution efficiently.

Remark 1. The problem formulation (10) can also be used when there are no erasures—also referred to as lossless network—by setting $s_{iJm} = 1$ for all links. This is the case when, e.g., sufficiently strong error correction codes are employed at the link layer, possibly combined with appropriately reduced rates c_{iJ}. Erasures correlated over space, e.g., due to jamming, can also be incorporated in the formulation by directly plugging in the appropriate values of s_{iJM} for each set M in (10c).

Before concluding this section, it is worth mentioning that the proposed model subsumes wireless networks with point-to-point, and broadcast-only transmissions. The problem formulation also becomes simpler under these special cases and is briefly outlined next.

1) Point-to-point Transmissions: When only point-to-point transmissions are allowed, the network can be modeled by a regular graph with edges $\mathcal{E}$ instead of hyperarcs $\mathcal{A}$. Using the set $\mathcal{J}_i = \{j|(i,j) \in \mathcal{E}\}$ in (10c), the new constraints become

$$r_{ik}^{(t)} - c_{ik} p_{ik} s_{ik} \prod_{j \in I(k)\cup\{k\}\setminus\{i\}} q_j \leq 0, \qquad k \in N(i), i \in \mathcal{N}, t \in \mathcal{T}.$$

where $1 - s_{ik}$ is the erasure probability on link $(i,k) \in \mathcal{E}$. The sum-of-probabilities constraint (10d) also simplifies to $\sum_{j\in N(i)} p_{ij} + q_i \leq 1$.

2) Broadcast-only Transmissions: In networks with broadcast-only transmissions, node i transmits all its packets on the hyperarc $(i, N(i))$. Such a scenario arises when each node $i \in \mathcal{N}$ can only transmit at the same PHY rate c_i to all its neighbors. In this case, transmitting on a hyperarc (i, J) such that $J \subsetneq N(i)$ does not yield any rate advantage. Under this assumption, the Aloha protocol is also simplified and at each time slot, node i only transmits on $(i, N(i))$ with probability $p_{iN(i)} = 1 - q_i$. The set $\mathcal{C}_{iK}$ is replaced here by the set

$$\bar{\mathcal{C}}_{iK} = \{(M,k)|M \subset K, k = |M|\}. \tag{11}$$

Defining $\mathcal{I}_{iM}^1 := (I(M) \cup M) \setminus \{i\}$ and $\mathcal{I}_{iM}^2 := I(M) \cup M$, constraint (10c) becomes

$$\sum_{j\in K} r_{ij}^{(t)} \leq c_i \sum_{(M,k)\in\bar{\mathcal{C}}_{iK}} (-1)^{k+1}(1-q_i) s_{iM} \prod_{j\in\mathcal{I}_{iM}} q_j$$
$$= c_i \sum_{p=1}^{2} \sum_{(M,k)\in\bar{\mathcal{C}}_{iK}} (-1)^{k+p} s_{iM} \prod_{j\in\mathcal{I}_{iM}^p} q_j,$$
$$K \subset N(i), i \in \mathcal{N}, t \in \mathcal{T} \tag{12}$$

where $s_{iM} := s_{iN(i)M}$. Note that problem (10) remains non-convex even under both special cases.

The broadcast-only case was also considered in [11], where a centralized algorithm was developed for small-size networks. In addition to focusing here on distributed optimization that is scalable for larger networks, characterization of the MAC constraints in (12) (as well as in (10c)) is more efficient. Specifically, the MAC constraints in [11] are captured through the variables z_{iJ}, which, in turn, are described using a sum with the number of terms growing exponentially in $|\mathcal{N}|$, whereas in (12), the number of terms is exponential only in $|N(i)|$.

III. Successive Convex Approximation

Optimization over general non-convex constraints is well known to be difficult. However, depending on the problem structure, several approximation methods are available. An option is offered by successive convex approximation, which, under certain regularity conditions, guarantees first order KKT optimality [15]. In this section, the successive convex approximation approach is applied to $(\mathbf{P}^0)$. First, the general method is reviewed.

A. Successive Convex Approximation Procedure

Suppose that the objective function to be minimized is convex, and the constraint set is the intersection of a set $\mathcal{H} := \{\mathbf{y}|h_i(\mathbf{y}) \leq 0, i = 1,2,\ldots,I\}$ with a convex set $\mathcal{C}$. Functions $\{h_i(\mathbf{y})\}$ are differentiable but may be non-convex in general. The set $\mathcal{C}$ captures convex constraints, if any. The idea is to solve a sequence of surrogate problems, indexed by $\ell \in \{1,2,\ldots\}$, where $\mathcal{H}$ is substituted per iteration ℓ by a convex set $\mathcal{H}^\ell$. Since the intersection of convex sets is a convex set [16, Sec. 2.3.1], the resulting optimization problems are convex. Set $\mathcal{H}^{\ell+1}$ is constructed as $\mathcal{H}^{\ell+1} := \{\mathbf{y}|\tilde{h}_i(\mathbf{y};\mathbf{y}^\ell) \leq 0, i = 1,2,\ldots,I\}$, where $\mathbf{y}^\ell$ is the solution of the convex approximation at the ℓ-th iteration, and $\tilde{h}_i(\mathbf{y};\mathbf{y}^\ell)$ for each i is a differentiable convex function satisfying the following three conditions:

(c1) $h_i(\mathbf{y}) \leq \tilde{h}_i(\mathbf{y};\mathbf{y}^\ell)$ for all $\mathbf{y} \in \mathcal{H}^{\ell+1} \cap \mathcal{C}$
(c2) $h_i(\mathbf{y}^\ell) = \tilde{h}_i(\mathbf{y}^\ell;\mathbf{y}^\ell)$; and
(c3) $\nabla h_i(\mathbf{y}^\ell) = \nabla \tilde{h}_i(\mathbf{y}^\ell;\mathbf{y}^\ell)$.

The procedure is initialized at an arbitrary feasible point $\mathbf{y}^0 \in \mathcal{H} \cap \mathcal{C}$. As shown in [15], the limit of the sequence $\{\mathbf{y}^\ell\}$ is precisely a KKT point of the original (non-convex) problem.

B. Centralized Solution

In order to apply the successive convex approximation method to $(\mathbf{P}^0)$, consider first the change of variables $\tilde{p}_{iJ} := \log p_{iJ}$ and $\tilde{q}_i := \log q_i$. The objective in (10a) remains convex provided that the cost function $v_i(q_i) = v_i(e^{\tilde{q}_i})$ for each i is chosen to be convex in $\tilde{q}_i$. Such a requirement is not too restrictive, as it is satisfied by a large class of useful cost functions including, e.g., $v_i(q_i) = -\ln q_i$ and $v_i(q_i) = q_i^{-\alpha}$, $\alpha > 0$. Such cost functions do not allow $q_i = 0$, or, equivalently each node remains silent with nonzero probability, which has desirable effects on fairness as well as power savings. Constraints (10b) are not affected by the change of variables, and hence remain convex (linear). Constraints (10d) become

$$\sum_{J\in\mathcal{J}_i} \exp(\tilde{p}_{iJ}) + \exp(\tilde{q}_i) - 1 \leq 0, \qquad i \in \mathcal{N} \tag{13}$$

which are convex.

Constraints (10c) become

$$\sum_{j\in K} r_{ij}^{(t)} - \sum_{(J,M,k)\in\mathcal{C}_{iK}^1} c_{iJ} s_{iJM} \exp\left(\tilde{p}_{iJ} + \sum_{j\in\mathcal{I}_{iM}} \tilde{q}_j\right)$$
$$+ \sum_{(J,M,k)\in\mathcal{C}_{iK}^2} c_{iJ} s_{iJM} \exp\left(\tilde{p}_{iJ} + \sum_{j\in\mathcal{I}_{iM}} \tilde{q}_j\right) \leq 0,$$
$$K \subset N(i), i \in \mathcal{N}, t \in \mathcal{T} \tag{14}$$

where the odd-k and even-k subsets of $\mathcal{C}_{iK}$ are defined as

$$\mathcal{C}_{iK}^1 := \{(J,M,k)|(J,M,k) \in \mathcal{C}_{iK}, k \text{ odd}\} \tag{15}$$
$$\mathcal{C}_{iK}^2 := \{(J,M,k)|(J,M,k) \in \mathcal{C}_{iK}, k \text{ even}\}. \tag{16}$$

It is noted that the second summand (with its sign) in (14) is concave in the optimization variables, while the rest are convex. However, it is possible to upper-bound the concave terms by an affine function [16, p. 69]. Specifically, given the solution $\tilde{p}_{iJ}^{(\ell)}$ and $\tilde{q}_j^{(\ell)}$ of the ℓ-th convex approximation, (14) can be replaced with the following convex constraint at the

$(\ell+1)$-th approximation:

$$\sum_{j\in K} r_{ij}^{(t)} - \sum_{(J,M,k)\in\mathcal{C}_{iK}^1} c_{iJ}s_{iJM}\alpha_{iJM}^{(\ell)}\left(1+\tilde{p}_{iJ}-\tilde{p}_{iJ}^{(\ell)}+\sum_{j\in\mathcal{I}_{iM}}(\tilde{q}_j-\tilde{q}_j^{(\ell)})\right) + \sum_{(J,M,k)\in\mathcal{C}_{iK}^2} c_{iJ}s_{iJM}\exp\left(\tilde{p}_{iJ}+\sum_{j\in\mathcal{I}_{iM}}\tilde{q}_j\right)\le 0,$$
$$K\subset N(i), i\in\mathcal{N}, t\in\mathcal{T} \quad (17)$$

where for $(J,M,k)\in\mathcal{C}_{iK}^1$, $K\subset N(i)$, and $i\in\mathcal{N}$, it holds that $\alpha_{iJM}^{(\ell)} := \exp\left(\tilde{p}_{iJ}^{(\ell)}+\sum_{j\in\mathcal{I}_{iM}}\tilde{q}_j^{(\ell)}\right)$. It is easily verified that the approximation introduced in (17) satisfies conditions (c1)–(c3).

The resulting convex optimization problem for the $(\ell+1)$-th iteration is given by

$$(\mathbf{P}_\ell^1) \quad \min_{R\ge 0,\{r_{ij}^{(t)}\ge 0\},\{\tilde{p}_{iJ}\le 0\},\{\tilde{q}_i\le 0\}} \sum_{i\in\mathcal{N}} v_i(e^{\tilde{q}_i}) - R \quad (18a)$$
$$\text{subject to} \quad (10b),\ (13),\ \text{and}\ (17) \quad (18b)$$

and can be solved by generic algorithms for convex programs such as interior-point methods; see e.g., [17], [16]. Note that in the first iteration, $\{\tilde{p}_{iJ}^{(0)},\tilde{q}_i^{(0)}\}$ must be initialized to a feasible point of the original non-convex problem $(\mathbf{P}^0)$. This can be done by selecting arbitrary values for p_{iJ} such that $\sum_{J\in\mathcal{J}_i} p_{iJ} < 1$, setting $q_i = 1-\sum_J p_{iJ}$, and R as well as $\{r_{ij}^{(t)}\}$ to zero.

C. Implementation

The successive convex approximation procedure outlined in Section III-B can be used to solve $(\mathbf{P}^0)$ to KKT-optimality. The algorithm must be executed offline in a centralized fashion to obtain the transmission probabilities $\{p_{iJ}, q_i\}$. Using the scheme of [6], at each time slot, node i simply transmits random linear combinations of packets in its buffer on hyperarc (i,J) with rate c_{iJ} and probability p_{iJ}. However, a centralized solver may require a long time to solve each surrogate problem $(\mathbf{P}_\ell^1)$ and need several successive approximations to converge.

Fig. 2 describes an online variation of previously described algorithm, which uses the probabilities $\{\exp(\tilde{p}_{iJ}^{(\ell)}), \exp(\tilde{q}_i^{(\ell)})\}$ for transmission, as and when they become available. This is allowed since in the limit, the variables $\{\tilde{p}_{iJ}^{(\ell)},\tilde{q}_i^{(\ell)}\}$ become KKT-optimal. The random network coding scheme, adopted from [5], ensures that the asymptotic throughput achieved is also KKT-optimal. Interestingly, the scheme does not require MAC/network-layer acknowledgments or retransmissions; only the sinks need to signal the end of each generation.

As the size of the network scales, it is of prime interest to solve $(\mathbf{P}^0)$ in a distributed manner. Moreover, it is desired that the iterative optimization is performed online so that (slow) variations in the network topology and parameters can be tracked. Towards these ends, a distributed algorithm is developed next, which also lends itself to an online implementation.

initialize
  Convex approximation index $\ell = 0$
  Current generation index $g = 1$
  $\{\tilde{p}_{iJ}^{(0)},\tilde{q}_i^{(0)}\}$ to arbitrary values satisfying (13)
foreach *time slot* **do**
  `// Protocol operation`
  foreach *node* i **do**
    if *node* i *has packets of generation* g **then**
      Transmit a random linear combination of packets from generation g on the hyperarc (i,J) with probability $\exp(\tilde{p}_{iJ}^{(\ell)})$
    end
    if *packet is received at node* i **then**
      Store packet if it is linearly independent of the packets already stored at node i.
    end
  end
  if *each sink* $t\in\mathcal{T}$ *can decode all packets of generation* g **then**
    Flush all packets of generation g from all nodes in the network
    update $g\leftarrow g+1$
  end
  `// Update the transmission probabilities`
  if *solution* $\{\tilde{p}_{iJ}^*,\tilde{q}_i^*\}$ *to* $(\mathbf{P}_\ell^1)$ *available* **then**
    update
      $\tilde{p}_{iJ}^{(\ell+1)}\leftarrow\tilde{p}_{iJ}^*$, $\tilde{q}_i^{(\ell+1)}\leftarrow\tilde{q}_i^*$ for $i\in\mathcal{N}$, $(i,J)\in\mathcal{A}$
      $\ell\leftarrow\ell+1$
  end
end

Fig. 2. Online Implementation of the Centralized Algorithm

IV. Distributed Algorithm

Solving convex network optimization problems in a distributed fashion usually involves application of problem-specific decomposition techniques. The aim is to decompose the original problem into smaller sub-problems, which can be solved by distributed processors coordinated through local message passing. A popular method is the dual decomposition technique based on Lagrangian duality, which is well-motivated when the primal problem has a separable structure [17, Sec. 5.1.6], [18].

Unfortunately, the convex approximation $(\mathbf{P}_\ell^1)$ is not separable. In particular, the summands in (17) with even k involve exponentials of the sum of the transmission probabilities of neighboring nodes. Therefore, they do not take the form of a sum of terms that depend on individual node variables. To cope with this hurdle, additional approximation is introduced first to effect a separable structure. Moreover, a set of auxiliary variables $\{R^{(t)}\}_{t\in\mathcal{T}}$ is introduced to allow decomposition of the problem to the individual sinks in $\mathcal{T}$. For simplicity, the algorithm development hereafter specializes to the broadcast-only case. However, the methodology extends straightforwardly.

A. Creating Separable Structure

As noted earlier, the distributed solution is developed here for networks with broadcast-only transmissions. Changing the variables $\tilde{q}_i := \log q_i$, and defining $\bar{\mathcal{C}}^1_{iK}$ and $\bar{\mathcal{C}}^2_{iK}$ as the odd-k and even-k subsets of $\bar{\mathcal{C}}_{iK}$ [cf. (11)], (12) becomes

$$\sum_{j\in K} r_{ij}^{(t)} - c_i \sum_{(M,k)\in\bar{\mathcal{C}}^1_{iK}} s_{iM}\exp\left(\sum_{j\in\mathcal{I}^1_{iM}}\tilde{q}_j\right) + c_i \sum_{(M,k)\in\bar{\mathcal{C}}^2_{iK}} s_{iM}\exp\left(\sum_{j\in\mathcal{I}^1_{iM}}\tilde{q}_j\right) + c_i \sum_{(M,k)\in\bar{\mathcal{C}}^1_{iK}} s_{iM}\exp\left(\sum_{j\in\mathcal{I}^2_{iM}}\tilde{q}_j\right) - c_i \sum_{(M,k)\in\bar{\mathcal{C}}^2_{iK}} s_{iM}\exp\left(\sum_{j\in\mathcal{I}^2_{iM}}\tilde{q}_j\right) \le 0 \quad (19)$$

which can be expressed compactly as

$$\sum_{j\in K} r_{ij}^{(t)} - c_i \sum_{x=1}^{2}\sum_{p=1}^{2}\sum_{(M,k)\in\bar{\mathcal{C}}^x_{iK}} (-1)^{x+p} s_{iM}\exp\left(\sum_{j\in\mathcal{I}^p_{iM}}\tilde{q}_j\right) \le 0. \quad (20)$$

Of the five terms in (19), the second and fifth terms (those corresponding to even $x+p$ in (20)) are non-convex and can be upper-bounded using affine functions as in the centralized solution. Thus, given $\tilde{q}_j^{(\ell)}$ at the ℓ-th iteration, the following approximations are used for $x = p \in \{1,2\}$:

$$\exp\left(\sum_{j\in\mathcal{I}^p_{iM}}\tilde{q}_j\right) \ge \alpha^{(\ell)}_{iMp}\left(1 + \sum_{j\in\mathcal{I}^p_{iM}}(\tilde{q}_j - \tilde{q}_j^{(\ell)})\right) \quad (21)$$

where, similar to before, $\alpha^{(\ell)}_{iMp} := \exp\left(\sum_{j\in\mathcal{I}^p_{iM}}\tilde{q}_j^{(\ell)}\right)$.

Note that the resultant affine terms are already separable. To make the remaining terms separable, another layer of approximation is applied to (19). The idea is to use the arithmetic-geometric inequality to upper-bound each term in the third and the fourth summations in (19). Specifically, it is noted that [14, p. 32]

$$\prod_{j\in\mathcal{I}^p_{iM}}\exp(\tilde{q}_j) \le \sum_{j\in\mathcal{I}^p_{iM}}\beta^{(\ell)}_{iMjp}\exp\left(\frac{\tilde{q}_j}{\beta^{(\ell)}_{iMjp}}\right) \quad (22)$$

is satisfied for terms corresponding to $x = 3-p \in \{1,2\}$, provided that $\beta^{(\ell)}_{iMjp} > 0$ and $\sum_{j\in\mathcal{I}_{iM}}\beta^{(\ell)}_{iMjp} = 1$ hold. Moreover, it can be verified that conditions (c1)–(c3) are satisfied at $\tilde{q}_j = \tilde{q}_j^{(\ell)}$, $j \in \mathcal{I}^p_{iM}$, if the approximation parameters $\{\beta^{(\ell)}_{iMjp}\}$ are chosen for $(M,k) \in \bar{\mathcal{C}}_{iK}$, $K \subset N(i)$, $j \in \mathcal{I}^p_{iM}$, and $i \in \mathcal{N}$ as $\beta^{(\ell)}_{iMjp} = \frac{\tilde{q}_j^{(\ell)}}{\sum_{j'\in\mathcal{I}^p_{iM}}\tilde{q}_{j'}^{(\ell)}}$. Thus, (19) can be surrogated by

$$\sum_{j\in K} r_{ij}^{(t)} - c_i \sum_{\substack{x=p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}^x_{iK}}} s_{iM}\alpha^{(\ell)}_{iMp}\left(1+\sum_{j\in\mathcal{I}^p_{iM}}(\tilde{q}_j-\tilde{q}_j^{(\ell)})\right) + c_i \sum_{\substack{x=3-p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}^x_{iK}}} s_{iM}\sum_{j\in\mathcal{I}^p_{iM}}\beta^{(\ell)}_{iMjp}\exp\left(\frac{\tilde{q}_j}{\beta^{(\ell)}_{iMjp}}\right) \le 0,$$

$$K \subset N(i), i\in\mathcal{N}, t\in\mathcal{T} \quad (23)$$

which is now separable in the per-node optimization variables $\{r_{ij}^{(t)}\}$ and $\tilde{q}_i$ for each $i\in\mathcal{N}$.

To induce per-sink decomposability of constraint (10b), a set of auxiliary variables $\{R^{(t)}\}_{t\in\mathcal{T}}$ is introduced, which represents the multicast rates for the individual sinks $t\in\mathcal{T}$, and additional constraints are imposed to ensure that the sinks can support the optimal R. Specifically, (10b) is substituted with

$$\sum_{j:i\in N(j)} r_{ji}^{(t)} + R^{(t)}\mathbf{1}_{\{i=s\}} - \sum_{j\in N(i)} r_{ij}^{(t)} \le 0, \quad t\in\mathcal{T}, i\in\mathcal{N}\setminus\{t\} \quad (24)$$

$$R - R^{(t)} \le 0, \quad t\in\mathcal{T}. \quad (25)$$

The resulting problem

$$(\mathbf{P}^2_\ell) \quad \min_{R\ge 0,\{R^{(t)}\ge 0\},\{r_{ij}^{(t)}\ge 0\},\{\tilde{q}_i\le 0\}} \sum_{i\in\mathcal{N}} v_i(e^{\tilde{q}_i}) - R \quad (26a)$$

$$\text{subject to } (23), (24), \text{ and } (25) \quad (26b)$$

is amenable to a distributed solution, as detailed next.

B. Distributed Solution via Dual Subgradient Method

The convex optimization problem (26) is solved here in a distributed fashion via the dual decomposition technique. Since the objective function in (26a) is not *strictly* convex with respect to all primal variables, the dual function may not be differentiable. Thus, the subgradient method is employed to solve the dual problem [19, Ch. 8]. The subgradient method is widely used in cross-layer optimization; see e.g., [3], [18], [20], [21] and references therein. Also, to ensure feasibility of the primal solution recovered from the dual optimal variables, primal averaging is employed [22].

Upon introducing the Lagrange multipliers $\{\lambda_{iKt}\ge 0\}$ and $\{\mu_t \ge 0\}$ to relax constraints (23) and (25), respectively, the partial Lagrangian for (26) is written as

$$\mathcal{L}(R,\{R^{(t)}\},\{r_{ij}^{(t)}\},\{\tilde{q}_i\}) = \sum_{i\in\mathcal{N}} v_i\left(e^{\tilde{q}_i}\right) - R + \sum_{t\in\mathcal{T}}\mu_t\left(R-R^{(t)}\right) + \sum_{\substack{K\subset N(i),\\ i\in\mathcal{N},t\in\mathcal{T}}}\lambda_{iKt} \left\{\sum_{j\in K} r_{ij}^{(t)} - c_i \sum_{\substack{x=p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}^x_{iK}}} s_{iM}\alpha^{(\ell)}_{iMp}\left(1+\sum_{j\in\mathcal{I}^p_{iM}}(\tilde{q}_j-\tilde{q}_j^{(\ell)})\right) + c_i \sum_{\substack{x=3-p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}^x_{iK}}} s_{iM}\sum_{j\in\mathcal{I}^p_{iM}}\beta^{(\ell)}_{iMjp}\exp\left(\frac{\tilde{q}_j}{\beta^{(\ell)}_{iMjp}}\right)\right\}. \quad (27)$$

Thus, the dual function is given by

$$D(\{\lambda_{iKt}\},\{\mu_t\}) = \min_{\substack{R\ge 0,\{R^{(t)}\ge 0\},\\ \{r_{ij}^{(t)}\ge 0\},\{\tilde{q}_i\le 0\}}} \mathcal{L}(R,\{R^{(t)}\},\{r_{ij}^{(t)}\},\{\tilde{q}_i\}), \quad \text{subject to } (24) \quad (28)$$

and the dual problem by

$$\max_{\{\lambda_{iKt}\ge 0\},\{\mu_t\ge 0\}} D(\{\lambda_{iKt}\},\{\mu_t\}). \quad (29)$$

Remark 2. The choice of which constraints to explicitly relax via dual variables and which to keep implicit may affect complexity of the minimization step in (IV-B), as well as the convergence speed of the algorithm. Specifically, if only few constraints are kept implicit, the primal solution step (IV-B) may be simple, but the subgradient method to solve (29) may take long time to converge. On the other hand, keeping many constraints implicit may hinder distributed implementation of (IV-B), as it becomes hard to exploit the separable structure. Here, inspired by [3] and [23], the virtual flow constraints (24) are kept implicit, which leads to a favorable trade-off between decomposability and convergence speed.

The separable structure of (26) allows terms in the Lagrangian function to be re-grouped according to the corresponding layers in the networking protocol. Thus, minimization of Lagrangian decomposes to per-layer sub-problems in the link layer (involving the log probabilities $\{\tilde{q}_i\}$); the network layer (involving the network coding parameters $\{R^{(t)}\}$ and $\{r_{ij}^{(t)}\}$); and the transport layer (involving the multicast rate R), each of which can be solved individually given the Lagrange multipliers. In the sequel, distributed solutions to the sub-problems are developed.

1) Link layer sub-problem: The link layer sub-problem can be further decomposed to the node level. Upon defining the set $\mathcal{I}_i^{-p}$ of nodes that are interfered by node i's transmission as

$$\mathcal{I}_i^{-p} := \{m \in \mathcal{N} | i \in \cup_{M \subset N(m)} \mathcal{I}_{mM}^p\} \tag{30}$$

the link layer sub-problem for node $i \in \mathcal{N}$ is obtained by collecting in $\mathcal{L}(\cdot)$ of (IV-B) the terms containing $\tilde{q}_i$ (henceforth, τ denotes the iteration index of the subgradient updates to be discussed later):

$$\begin{aligned}\tilde{q}_i(\tau) \in \arg\min_{\tilde{q}_i \le 0} & \, v_i\left(e^{\tilde{q}_i}\right) \\ & - c_i\tilde{q}_i\left(\sum_{\substack{x=1\\p=x}}^{2} \sum_{m\in\mathcal{I}_i^{-p}} \sum_{\substack{K\subset N(m)\\t\in\mathcal{T}}} \sum_{(M,k)\in\mathcal{C}_{mK}^x} s_{mM}\alpha_{mMp}^{(\ell)}\lambda_{mKt}(\tau)\right) \\ & + c_i\left(\sum_{\substack{x=1\\p=3-x}}^{2} \sum_{m\in\mathcal{I}_i^{-p}} \sum_{\substack{K\subset N(m)\\t\in\mathcal{T}}} \sum_{(M,k)\in\mathcal{C}_{mK}^x} s_{mM}\beta_{mMip}^{(\ell)}\lambda_{mKt}(\tau) \exp\left(\frac{\tilde{q}_i}{\beta_{mMip}^{(\ell)}}\right)\right).\end{aligned} \tag{31}$$

2) Network layer sub-problem: The network layer sub-problem can be further decomposed to the sink level. Thus, $R^{(t)}$ and $\{r_{ij}^{(t)}\}_{j\in N(i), i\in\mathcal{N}}$ can be updated by solving the per-sink problem for each $t \in \mathcal{T}$ given by

$$(R^t(\tau), \{r_{ij}^{(t)}(\tau)\}) \in \arg\min_{R^{(t)}\ge 0, \{r_{ij}^{(t)}\ge 0\}} \sum_{j\in N(i), i\in\mathcal{N}} r_{ij}^{(t)} \sum_{\substack{K\subset N(i)\\K\ni j}} \lambda_{iKt}(\tau) - \mu_t(\tau)R^{(t)}$$

$$\text{s.t.} \sum_{j\in N(i)} r_{ji}^{(t)} + R^{(t)}\mathbf{1}_{\{i=s\}} \le \sum_{i\in N(j)} r_{ij}^{(t)}, \quad i \in \mathcal{N}\backslash\{t\} \tag{32a}$$

$$r_{ij}^{(t)} \le c_i, \qquad j \in N(i), i \in \mathcal{N}. \tag{32b}$$

Problem (32) can be reduced to the standard minimum-cost flow problem by adding a virtual link from node t to node s with infinite capacity and cost $-\mu_t$ [24]. The minimum-cost flow on this graph then yields the solution to the original problem with $R^{(t)}$ given by the flow on the virtual t-s link.

The minimum-cost flow problem is a well-studied problem; see e.g., [24] for a detailed survey. Many of the algorithms available are amenable to distributed implementation, and terminate in a number of steps polynomially bounded by the number of nodes. In our case, the iterative primal updates only involve changes in the link costs. Therefore, it would be useful to choose a method that can soft-start from an available feasible solution from the previous iteration. One such method is the ϵ-relaxation method; see e.g., [24, Ch. 7], [25, Ch. 6], [26].

3) Transport layer sub-problem: In order to obtain the update equation for R, note first that the optimal R is necessarily upper-bounded because the per node maximum transmission rates c_i are bounded. In particular, from (10b) with $i = s$ and (10c) with $K = N(i)$, it holds that

$$R \le \sum_{j\in N(s)} r_{sj}^{(t)} \le c_s|\mathcal{C}_{sN(s)}| =: R_{\max}. \tag{33}$$

Using (33) as an additional constraint, the multicast rate R is updated as

$$R(\tau) \in \arg\min_{0\le R\le R_{\max}} \Big(\sum_t \mu_t(\tau) - 1\Big)R \tag{34}$$

which can be solved straightforwardly.

4) Dual update and primal recovery: Once the primal iterates $\mathbf{y}(\tau) := [\{\tilde{q}_i(\tau)\}, \{R^{(t)}(\tau)\}, \{r_{ij}^{(t)}(\tau)\}, R(\tau)]$ have been obtained, the dual variables are updated to solve (29). The subgradient projection method is employed, which amounts to updating the dual iterates through

$$\begin{aligned}\lambda_{iKt}(\tau+1) = \Bigg[\lambda_{iKt}(\tau) + \sigma\Bigg\{ & \sum_{j\in K} r_{ij}^{(t)}(\tau) \\ & - c_i \sum_{\substack{x=p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}_{iK}^x}} s_{iM}\alpha_{iMp}^{(\ell)}\left(1 + \sum_{j\in\mathcal{I}_{iM}^p}\left(\tilde{q}_j(\tau) - \tilde{q}_j^{(\ell)}\right)\right) \\ & + c_i \sum_{\substack{x=3-p\in\{1,2\}\\(M,k)\in\bar{\mathcal{C}}_{iK}^x}} s_{iM} \sum_{j\in\mathcal{I}_{iM}^p} \beta_{iMjp}^{(\ell)} \exp\left(\frac{\tilde{q}_j(\tau)}{\beta_{iMjp}^{(\ell)}}\right)\Bigg\}\Bigg]^+,\end{aligned}$$

$$K \subset N(i), i \in \mathcal{N}, t \in \mathcal{T} \tag{35}$$

$$\mu_t(\tau+1) = \Big[\mu_t(\tau) + \sigma\left(R(\tau) - R^{(t)}(\tau)\right)\Big]^+, \; t \in \mathcal{T} \tag{36}$$

where $[\cdot]^+ :\equiv \max\{0, \cdot\}$, and $\sigma > 0$ is the step size. The dual iterates can be initialized to arbitrary non-negative values. The subgradient method with a constant step size converges into a ball of the optimal dual variables, whose radius is proportional to the step size; see e.g., [19, Prop. 8.2.2] for the exact claim and the convergence rates.

Due to the lack of strict convexity, the primal iterates $\mathbf{y}(\tau)$ recovered from the dual iterates may not converge in general. Nevertheless, their running average $\bar{\mathbf{y}}(\tau) := \frac{1}{\tau}\sum_{\rho=0}^{\tau-1}\mathbf{y}(\rho)$ is asymptotically feasible, and converges to the optimum solution of $(\mathbf{P}_\ell^2)$ [22]. The running averages are then used to update $\{\tilde{q}_i^{(\ell+1)}\}$ (to evaluate $\{\alpha_{iMp}^{(\ell+1)}, \beta_{iMjp}^{(\ell+1)}\}$) for the next approximation $(\mathbf{P}_{\ell+1}^2)$ and the subgradient iterations restarted.

It is also possible to combine the subgradient and convex approximation iterations by *not* reinitializing the dual iterates when updating the values of $\{\tilde{q}_j^{(\ell)}\}$. If the surrogate problems $(\mathbf{P}_\ell^2)$ and $(\mathbf{P}_{\ell+1}^2)$ are not too different, the final dual iterates of $(\mathbf{P}_\ell^2)$ will also be near-optimal for $(\mathbf{P}_{\ell+1}^2)$. Retaining the dual iterates is therefore equivalent to "soft-starting" the dual subgradient method with near-optimal initial values. The next section builds upon this combined algorithm, and describes its distributed and online implementation.

C. Distributed and Online Protocol

The present section describes a distributed, parallel, and online implementation of the successive convex approximation algorithm. Recall from the centralized algorithm in Fig. 2, that it is possible to operate the network using a sequence of transmission probabilities $\{1-\exp(\tilde{q}_i^{(\ell)})\}$, converging to KKT-optimal values. In the present case, these values are provided by the combined subgradient and convex approximation algorithm outlined in Sec. IV-B. Fig. 3 describes the message passing and variable updates required by the algorithm at each node $i \in \mathcal{N}$. Each subgradient iteration in Fig. 3 takes up several time slots; cf. Fig. 2.

Observe that the message passing required at each iteration is moderate. Specifically, node i collects primal variables $\tilde{q}_j(\tau)$ from all nodes $j \in (\cup_{M,p}\mathcal{I}_{iM}^p) \setminus \{i\} = \mathcal{I}_{iN(i)}^1$, and dual variables $\{\lambda_{mKt}\}$ from all nodes $m \in \mathcal{I}_i^{-p}$ for $p = 1, 2$. Roughly speaking, these quantities pertain to the two-hop neighborhood of node i. Further, the source needs to solve (32) at each iteration and for each sink (in parallel), using an asynchronous, distributed method such as ϵ-relaxation. Finally, the convex approximation parameters $\{\alpha_{iJM}^{(\ell)}, \beta_{iM}^{(\ell)}\}$ at node i depend on $\tilde{q}_j^{(\ell)}$ for $j \in \mathcal{I}_{iN(i)}^1$. These variables are anyway made known to node i for the purpose of dual updates. Overall, if each node has at most d neighbors, it exchanges $O(2^d)$ variables with each of its $O(d^2)$ two-hop neighbors, per subgradient iteration. Each node also exchanges $O(d)$ variables per ϵ-relaxation iterations. Finally, the storage requirement for each node is $O(2^d d^2)$ variables.

In general, the subgradient method does not specify a stopping criterion, and it is customary to use a fixed number of iterations. Alternatively, the subgradient algorithm can stop when the primal averages converge and remain unchanged for several iterations. The use of subgradient algorithm offers some flexibility in the choice of the time-scale of iterations. It is not necessary to wait for convergence of the subgradient method for updating the transmission probabilities. Indeed, the running averages $\{1-\exp(\bar{\tilde{q}}_i(\tau))\}$ can also be used as transmission probabilities at intermediate iterations, since these converge to $\{1-\exp(\tilde{q}_j^{(\ell)})\}$, which in turn converge to the KKT-optimal probabilities.

maintain variables
  $\alpha_{iMp}^{(\ell)}$ and $\beta_{iMjp}^{(\ell)}$ for $M \subset N(i)$, $j \in \mathcal{I}_{iM}^p$, $p = 1, 2$
  $\tilde{q}_i(\tau)$, $r_{ij}^{(t)}(\tau)$ for $j \in N(i)$, and $\lambda_{iKt}(\tau)$ for $K \subset N(i)$, $t \in \mathcal{T}$, $i \neq t$
  if *node i is source* **then** $R(\tau)$, $R^{(t)}(\tau)$, $\mu_t(\tau)$ for $t \in \mathcal{T}$
initialize
  probabilities $\tilde{q}_i^{(0)}$ and evaluate $\alpha_{iMp}^{(0)}$, $\beta_{iMjp}^{(0)}$
  $\lambda_{iKt}(1) = 0$ for $K \subset N(i)$, $t \in \mathcal{T}$, $i \neq t$
  if *node i is source* **then** $\mu_t(1) = 0$, for $t \in \mathcal{T}$
  successive convex approximation index $\ell = 0$
  running average $\bar{\tilde{q}}_i(0) = 0$, and $\tau_0 = 0$
foreach $\tau = 1, 2, \dots$ **do**
  collect
    $\{\lambda_{mKt}(\tau)\}$ from nodes $m \in \mathcal{I}_i^{-p}$, $p = 1, 2$
    $\{\tilde{q}_j(\tau)\}$ from nodes $j \in \mathcal{I}_{iN(i)}^1$
  update
    primal iterates $\tilde{q}_i(\tau)$ and $r_{ij}^{(t)}(\tau)$ [cf. (31) and (32)]
    dual iterates $\lambda_{iKt}(\tau+1)$ [cf. (35)]
    running average
    $\bar{\tilde{q}}_i(\tau) \leftarrow \frac{\tau-\tau_\ell-1}{\tau-\tau_\ell}\bar{\tilde{q}}_i(\tau-1) + \frac{1}{\tau-\tau_\ell}\tilde{q}_i(\tau)$
  if *node i is the source* **then**
    primal iterates $R(\tau)$ and $R^{(t)}(\tau)$ [cf. (34) and (32)]
    dual iterates $\mu_t(\tau+1)$ [cf. (36)]
  end
  if *subgradient iterations have converged or maximum iterations reached* **then**
    update $\tilde{q}_i^{(\ell+1)} \leftarrow \bar{\tilde{q}}_i(\tau)$ *and evaluate* $\alpha_{iMp}^{(\ell+1)}$ *and* $\beta_{iMjp}^{(\ell+1)}$
    update $\ell \leftarrow \ell + 1$
    reinitialize running average $\bar{\tilde{q}}_i(\tau) \leftarrow 0$, and set $\tau_\ell = \tau$
  end
end

Fig. 3. Distributed and Online Algorithm for Node i

V. Numerical Results

Numerical tests are performed for the centralized and distributed algorithms proposed in Sec. III and IV. Related algorithms from [11] and [12] are compared as benchmarks.

A. Simulation Set-up

Random networks are generated using the MAX-DPA algorithm [27], which generates graphs by placing nodes one by one, while respecting certain maximum-degree and node proximity constraints so as to simulate a realistic ad hoc network. The algorithm parameters are chosen to be $d = 3$, $d_{\max} = 6$, and $d_0 = 0.2$ (see [27]), and the nodes are placed in a square area with the average node density 1. The erasure probability for a pair of neighboring nodes i and j separated by distance d_{ij} is given by $1 - \exp(-d_{ij}^2/4)$, assuming Rayleigh fading. The transmission rate c_i is assumed to be unity for all nodes $i \in \mathcal{N}$. The multicast session is chosen so that the leftmost node is the source and the two rightmost nodes are the sink nodes. To compare with the existing algorithms on an equal footing, the broadcast-only scenario is considered and only the throughput is maximized while $\{v_i(q_i)\}$ are set to zero.

B. Centralized Algorithm

Table I gives the maximum throughput achieved with different schemes, averaged over 100 random network realizations. Four different methods are compared: the heuristic

TABLE I
AVERAGE OPTIMIZED THROUGHPUT.

# of nodes	Heurist.	Central.	B. & B.	Orth. Schedul.
7	0.1883	0.3103	0.3138	0.3144
8	0.1762	0.2736	0.2782	0.2667
9	0.1664	0.2584	0.2647	0.2498
10	0.1632	0.2354	0.2426	0.2214
20	0.1242	0.1890	–	0.1263
40	0.1110	0.1615	–	0.0776

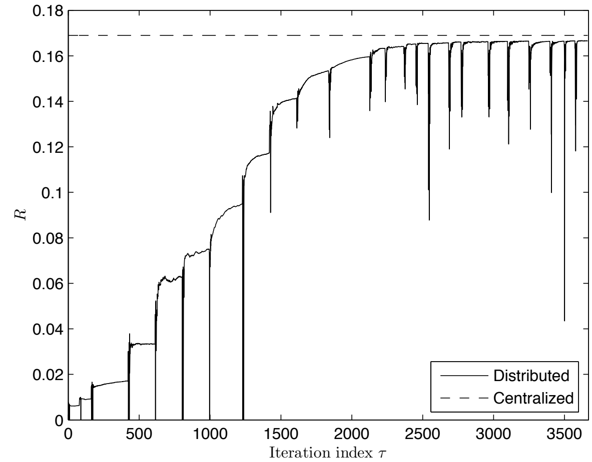

Fig. 4. Evolution of the end-to-end throughput R in the subgradient method with step size $\sigma = 0.5$ for the first surrogate problem ($\ell = 0$) and $\sigma = 0.1$ thereafter. The vertical lines result from the fact that the primal averages are refreshed whenever the value of ℓ is advanced. Therefore the solution obtained from the next few subgradient iterations is of poor quality and gives low values of R. However, the network throughput depends only on the access probabilities at the instants when the subgradient iterations converge.

method from [12], the proposed method (centralized version), the branch-and-bound method from [11], and the orthogonal scheduling from [3] with only one transmitting node per time slot. The proposed centralized algorithm is initialized by considering a set of 20 randomly chosen probabilities q_i, and picking the one that yields the maximum value of R (which can be easily obtained by solving a linear program). It can be seen that for small-size networks, where the branch-and-bound algorithm runs in a reasonable time, the average throughput of the proposed centralized algorithm is close to the global optimum. The suboptimality is due to possible convergence of the algorithm to a KKT point.

C. Distributed Algorithm

1) Evolution of the Subgradient Method: The algorithm of Fig. 3 is simulated on a randomly generated network with 40 nodes. The initial point was chosen again as in Sec. V-B. Fig. 4 shows the evolution of the throughput achieved with the running average $\{\bar{\bar{q}}_i\}$ (which is close but not exactly equal to the running average $\bar{R}$) across the subgradient iterations and successive convex approximations. Recall that the running averages are refreshed when the convex approximation is updated. Interestingly, the throughput converges to a near-optimal value in very few convex approximation iterations.

2) Online Implementation: The algorithm of Fig. 3 is implemented with the random network coding scheme of [5]

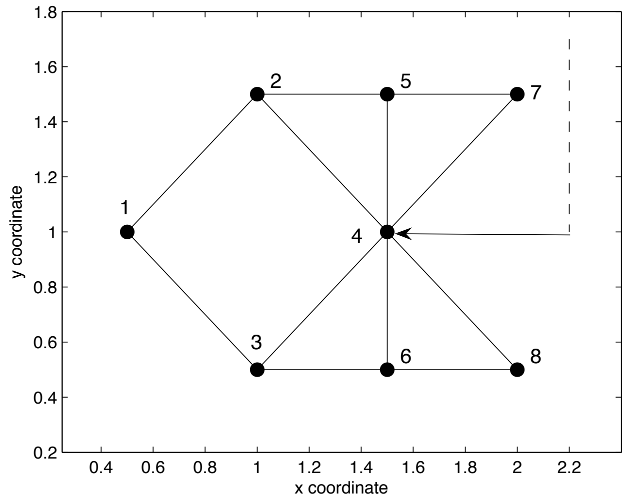

Fig. 5. Dynamic network used for simulation. Node 4 joins the network at time slot 4×10^4.

on a simple dynamic network. The network is shown in Fig. 5 and initially consists of all nodes except node 4. The aim is to multicast packets from source node 1 to sink nodes 7 and 8. The network is simulated for 7×10^4 time slots, and node 4 joins the network at time slot 3×10^4 and starts transmitting with arbitrary probability.

The network coding scheme is implemented using a generation size of 100 packets, and field size 2^8. The source is infinitely backlogged, i.e., there are generations waiting to be transmitted at all time slots. It is assumed that an end-to-end network error-correction code is employed; see e.g., [28] and references therein. Consequently, the sinks are required to collect only 90 linearly independent packets for each generation. This implies that the uncoded throughput is 90% of the value obtained from the centralized solution. The subgradient algorithm runs in parallel with the network protocol, and updates the transmission probabilities every 10^3 time slots.

Fig. 6 shows the evolution of the per-generation throughput of the system, represented by dots. The per-generation throughput is $R_g := 90/T_g$, where T_g is the time-difference (measured in time slots) between transmission of the first packet of generation g and the reception of 90 linearly independent packets of generation g at all sinks. The solid curve R_{avg} represents the moving average of R_g for 10 previously received generations. Finally, the dotted line (R_{opt}) shows the 90% of the KKT-optimal value of R obtained by running the centralized algorithm.

It can be observed that the per-generation throughput is low in the beginning as all nodes start transmitting at sub-optimal access probabilities. The throughput improves as the subgradient iterations evolve, but decreases again when node 4 joins the network. This is because when node 4 enters, it also starts transmitting at an arbitrary probability, and interferes with reception at other nodes. Eventually though, the subgradient iterations evolve to a new optimum, and the throughput increases again. Intuitively, node 4 helps by providing more paths for packets being multicast to nodes

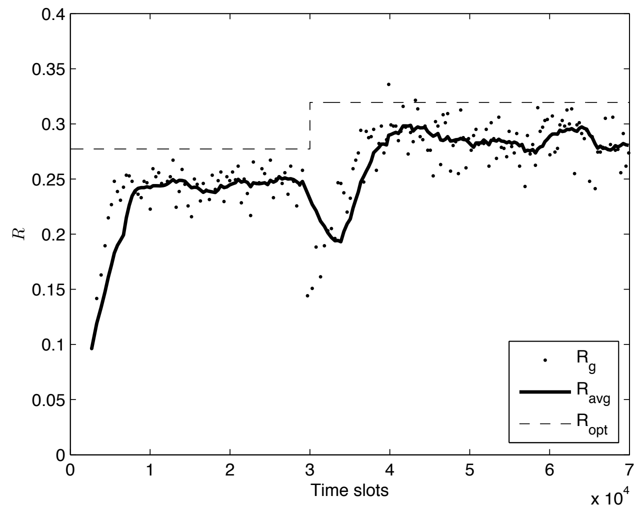

Fig. 6. Evolution of the R values. A dot at a given time slot represents the throughput of the generation that is received at that time slot. Since generations are transmitted serially, the moving average of the per-generation throughput represents the throughput achieved over several generations.

7 and 8, and therefore the overall throughput is higher than before. The remaining gap between the centralized solution and achieved throughput is because of the overhead inherent to the network coding scheme. This gap can be reduced by using a larger generation size or more sophisticated schemes (such as generation-interleaving [5]) at the expense of increased end-to-end packet delay.

VI. Conclusion

The problem of joint optimization of network coding and Aloha-based medium access control (MAC) for multi-hop wireless networks was considered. The multicast throughput with a power consumption-related penalty was maximized subject to flow conservation and MAC achievable rate constraints to obtain the optimal transmission probabilities. The relevant optimization problem turns out to be non-convex and hence difficult to solve even in a centralized manner. A successive convex approximation technique was employed to obtain a Karush-Kuhn-Tucker (KKT) solution. The idea was also extended to create a separable structure in the problem, and the dual decomposition technique is applied to derive a distributed solution. The algorithm is thus applicable for large networks, and amenable to online implementation. The numerical tests verify performance and complexity advantages of the proposed approach over existing designs. A network simulation with implementation of random linear network coding shows performance very close to the theoretical.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
[2] T. Ho, M. Medard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.
[3] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-cost multicast over coded packet networks," *IEEE Trans. Info. Theory*, vol. 52, no. 6, pp. 2608–2623, Jun. 2006.
[4] D. Traskov, M. Heindlmaier, M. Médard, R. Koetter, and D. Lun, "Scheduling for network coded multicast: A conflict graph formulation," in *Proc. of the IEEE GLOBECOM Workshops*, New Orleans, LA, Nov.-Dec. 2008, pp. 1–5.
[5] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. of the 41st Annual Allerton Conf. on Commun., Control, and Computing*, Monticello, IL, Oct. 2003, pp. 40–49.
[6] D. S. Lun, M. Medard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, Mar. 2008.
[7] R. Rao and A. Ephremides, "On the stability of interacting queues in a multiple-access system," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 918–930, Sep. 1988.
[8] H.-K. Lee and S.-L. Kim, "Network coded ALOHA for wireless multihop networks," in *Proc. IEEE Wireless Commun. Net. Conf. (WCNC)*, Budapest, Hungary, Apr. 2009, pp. 1–5.
[9] C. Ibars, L. Giupponi, and S. Addepalli, "Distributed multiple access and flow control for wireless network coding," in *Proc. IEEE 71st Veh. Technol. Conf. (VTC-Spring)*, Taipei, Taiwan, May 2010, pp. 1–6.
[10] D. Umehara, T. Hirano, S. Denno, M. Morikura, and T. Sugiyama, "Wireless network coding in slotted ALOHA with two-hop unbalanced traffic," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 5, pp. 647–661, Jun. 2009.
[11] M. Riemensberger, M. Heindlmaier, A. Dotzler, D. Traskov, and W. Utschick, "Optimal slotted random access in coded wireless packet networks," in *Proc. of the 6th Workshop Resource Allocation in Wireless Networks (RAWNET)*, Avignon, France, Jun. 2010, pp. 374–379.
[12] D. Traskov, D. S. Lun, R. Koetter, and M. Médard, "Network coding in wireless networks with random access," in *Proc. of the IEEE Int'l. Symp. Info. Theory (ISIT)*, Nice, France, Jun. 2007, pp. 2726–2730.
[13] S. M. Ross, *Introduction to Probability Models*, 8th ed. San Diego, CA: Academic Press, 2003.
[14] M. Chiang, "Geometric programming for communication systems," *Communications and Information Theory*, vol. 2, no. 1/2, pp. 1–154, Jul. 2005.
[15] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, no. 4, pp. 681–683, July-August 1978.
[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY: Cambridge University Press, 2004.
[17] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.
[18] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.
[19] D. P. Bertsekas, A. Nedić, and A. Ozdaglar, *Convex Analysis and Optimization*. Belmont, MA: Athena Scientific, 2003.
[20] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.
[21] L. Chen, T. Ho, S. H. Low, M. Chiang, and J. C. Doyle, "Optimization based rate control for multicast with network coding," in *Proc. IEEE INFOCOM Conf.*, Anchorage, AK, May 2007, pp. 1163–1171.
[22] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1757–1780, 2009.
[23] Y. Wu and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: A shortest path approach," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1475–1488, Aug. 2006.
[24] D. P. Bertsekas, *Network Optimization: Continuous and Discrete Models*. Belmont, MA: Athena Scientific, 1998.
[25] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.
[26] A. Frangioni and A. Manca, "A computational study of cost reoptimization for min-cost flow problems," *INFORMS J. Computing*, vol. 18, no. 1, pp. 61–70, Winter 2006.
[27] F. A. Onat, I. Stojmenovic, and H. Yanikomeroglu, "Generating random graphs for simulation of wireless ad-hoc, actuator, and internet networks," *Pervasive and Mobile Computing (Elsevier)*, vol. 4, no. 5, pp. 597–615, 2008.
[28] H. Bartz, T. Lutz, C. Hausl, and J. Barros, "Practical network coding with resilient subspace codes," in *Proc. of the 19th Intl. Conf. on Commun. and Networks*, Zurich, Switzerland, Aug. 2010.

Ketan Rajawat (S'06) received his B.Tech.and M.Tech.degrees in Electrical Engineering from Indian Institute of Technology Kanpur in 2007. Since August 2007, he has been working towards his Ph.D. degree at the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis.

His current research focuses on network coding and optimization in wireless networks.

Nikolaos Gatsis (S'04) received the Diploma degree in electrical and computer engineering from the University of Patras, Patras, Greece in 2005 with honors. Since September 2005, he has been working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN.

His research interests include cross-layer designs and resource allocation for wireless networks.

Seung-Jun Kim (M'07) received his B.S. and M.S. degrees from Seoul National University in Seoul, Korea in 1996 and 1998, respectively, and his Ph.D. from the University of California at Santa Barbara in 2005, all in electrical engineering.

From 2005 to 2008, he worked for NEC Laboratories America in Princeton, New Jersey, as a research staff member. Since 2008, he has been with the Department of Electrical and Computer Engineering at the University of Minnesota, where he is currently a research assistant professor. His research interests lie in applying signal processing and optimization techniques to various domains including wireless communication and networking, smart power grids, bio and social networks.

G. B. Giannakis (M'86-SM'91-F'97) received his Diploma in Electrical Engr. from the Ntl. Tech. Univ. of Athens, Greece, 1981. From 1982 to 1986 he was with the Univ. of Southern California (USC), where he received his MSc. in Electrical Engineering, 1983, MSc. in Mathematics, 1986, and Ph.D. in Electrical Engr., 1986.

Since 1999 he has been a professor with the Univ. of Minnesota, where he now holds an ADC Chair in Wireless Telecommunications in the ECE Department, and serves as director of the Digital Technology Center. His general interests span the areas of communications, networking and statistical signal processing - subjects on which he has published more than 300 journal papers, 500 conference papers, 20 book chapters, two edited books and two research monographs. Current research focuses on compressive sensing, cognitive radios, network coding, cross-layer designs, wireless sensors, social and power grid networks.

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