Dynamic Algorithms for Multicast with Intra-session Network Coding

Tracey Ho*and Harish Viswanathan

Lucent Technologies, Bell Labs
Murray Hill, NJ
{traceyho, harishv}@lucent.com

Abstract

We establish, for multiple multicast sessions with intra-session network coding, the capacity region of input rates for which the network remains stable in ergodically time-varying networks. Building on the back-pressure approach introduced by Tassiulas et al., we present dynamic algorithms for multicast routing, network coding, rate control, power allocation, and scheduling that achieves stability for rates within the capacity region. Decisions on routing, network coding, and scheduling between different sessions at a node are made locally at each node based on virtual queues for different sinks. For correlated sources, the sinks locally determine and control transmission rates across the sources. The proposed approach yields a completely distributed algorithm for wired networks. In the wireless case, scheduling and power control among different transmitters are centralized while routing, network coding, and scheduling between different sessions at a given node are distributed.

1 Introduction

Network coding has been shown to improve capacity over routing for multicast in wired and wireless networks [2, 12, 19]. Most analytical work in network coding to date assumes fixed rate sources and fixed link capacities. However, in networks with bursty traffic, optimal multicasting of information involves not only routing and network coding but also link scheduling and scheduling of different flows on the active links, and may depend on the current network state such as link data rates and buffer occupancy.

Routing, scheduling, and rate and power control in networks with bursty traffic has received significant attention, with much work, e.g. [11, 14, 20], building on the ideas of [4, 17] for routing and scheduling in multicommodity flow problems using queue sizes to prioritize among flows. Such approaches are often called *back-pressure* based, as routing is done along queue size gradients. The back pressure approach has also been extended to multicast in wired networks, but with much higher complexity: the algorithm of [15] involves enumeration of all multicast trees used while that in [3] involves maintaining a virtual queue for every subset of sinks for every session. Using network coding within multicast sessions, we not only enlarge the capacity region, but also obtain a simpler,

*Tracey Ho is now with the California Institute of Technology

distributed back-pressure algorithm in which each node has just one virtual queue for each sink of each session (for independent sources) or for each source-sink pair of each session (for correlated sources).

More specifically, we consider multiple multicast sessions, where each session consists of a set of sources and sinks such that data from all the sources is intended for all the sinks. We establish, for multiple multicast sessions with intra-session network coding, the capacity region of input rates that can be stabilized. We present dynamic algorithms for routing, network coding, scheduling and rate control across correlated sources that achieve stability for rates within the capacity region. Decisions on multicast routing, network coding, rate control across correlated sources and scheduling among different sessions at a node are made locally based on virtual queues for each sink of each session. This gives a completely distributed algorithm for wired networks; in the wireless case, scheduling and power control among interfering transmitters is done centrally. Our algorithms exploit the wireless multicast advantage [18] where a single transmission can simultaneously be received by multiple nodes and the combining advantage [1] where nodes can soft-combine the same information signal transmitted by different nodes.

Proofs of our results are omitted owing to space constraints but are given in [10].

2 Preliminaries

2.1 Network model

We consider a network composed of a set $\mathcal{N}$ of $N = |\mathcal{N}|$ nodes with communication links between them that are fixed or time-varying according to some specified ergodic processes and transmission of a set of multicast sessions $\mathcal{C}$ through the network. Each session $c \in \mathcal{C}$ is associated with a set $\mathcal{S}_c \subset \mathcal{N}$ of sources, and an exogenous process of data arrivals at each of these sources which must be transmitted over the network to each of a set $\mathcal{T}_c \subset \mathcal{N}$ of sinks. Transmissions are assumed to occur in slotted time, with time slots of length T. Routing, scheduling and other such decisions are made at most once a slot. For simplicity, we assume fixed length packets and link transmission rates that are restricted to integer multiples of the packet-length/time-slot quotient. That is, an integer number of packets can be transmitted in each slot.

For the wired network case the network connectivity and the link rates are fixed. For the wireless case, these depend on the transmission powers. We model wireless transmissions by generalized links, denoted by (a, Z), where a is the originating node and Z is the set of receiving nodes. Each generalized link can correspond to

- a single transmission from node a to neighboring destination nodes at which the SINR of the transmission is greater than some threshold, or
- a multi-step transmission in which a first transmits to one or more destination nodes, some of which, possibly together with a, repeat the transmission in the next time slot, and so on. Combining receptions from multiple transmissions of the same information has been investigated for broadcast in [1, 13]. Partial receptions are combined at destination nodes; the transmission rates in a transmission scenario containing such links are averaged over the number of time slots over which such forwarding takes place.

Link rates $\underline{\mu}(\underline{P}, \underline{S}) = (\mu_{aZ}(\underline{P}, \underline{S}))$ are determined by the vector of transmit powers $\underline{P}(t) = (P_{aZ}(t))$ and a channel state vector $\underline{S}(t)$. $\underline{S}(t)$ is assumed to be constant over

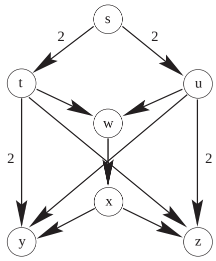

Figure 1: An example illustrating throughput loss caused by batch coding. Source node s is multicasting to sink nodes y and z. All the links have average capacity 1, except for the four labeled links which have average capacity 2. The optimal solution requires link (w, x) to carry coded information for both receivers. However, variability in the instantaneous capacities of links (u, w), (u, y), (t, w) and (t, z) may cause the number of sink y packets in a batch arriving at node w to differ from the number of sink z packets of that batch arriving at w.

each time slot, i.e., state transitions occur only on slot boundaries $t = kT$, k integer. We also assume that $\underline{S}(t)$ takes values from a finite set and is ergodic; we denote by $\pi_{\underline{S}}$ the time average probability of state $\underline{S}$. $\underline{P}(t)$ is also held constant over each time slot, and is chosen from a compact set Π of feasible apower allocations.

2.2 Network coding considerations

We use the approach of distributed randomized network coding [6, 7, 9], in which network nodes form output data by taking random linear combinations of input data. The contents of each packet, as a linear combination of the input packets, are specified by a *coefficient vector* in the packet header, updated by applying to the coefficient vectors the same linear transformations as to the data. A sink is able to decode when it receives a full set of packets with linearly independent coefficient vectors.

How to determine or achieve the optimal capacity for multiple multicast sessions with coding across sessions is an open question. We consider the simpler problem of achieving optimal capacity for the case where coding is done only across packets of the same session, and give policies that asymptotically achieve optimality in this case. For simplicity, we analyze the case where no restrictions are placed on coding among packets from the same multicast session. In practice, to reduce the code description overhead and decoding complexity, we can group packets formed within some time interval into a batch, and restrict network coding to occur only among packets of the same batch, as in [6], or among some number of consecutive batches. This may decrease the capacity region for some topologies [6]; restricting coding to only one batch decreases the capacity region further for variable sources/links since packets near the batch boundaries may have to be transmitted without coding. An illustration is given in Figure 2.2. Such capacity loss decreases with increasing batch size. Analysis of this case would require more detailed source and channel statistics.

2.3 Stability for multicast

Let $U_i^c(t)$ be the amount of session c data queued at node i at time t. Our notion of stability follows [14], which considers the "overflow" functions

$$\gamma_i^c(M) = \limsup_{t\to\infty} \frac{1}{t}\int_0^t 1_{[U_i^c(\tau)>M]} d\tau \tag{1}$$

$$\gamma_{sum}(M) = \limsup_{t\to\infty} \frac{1}{t}\int_0^t 1_{[\sum_{i,c} U_i^c(\tau)>M]} d\tau. \tag{2}$$

The session c queue at node i is considered stable if $\gamma_i^c(M) \to 0$ as $M \to \infty$. A network of queues is considered stable if each individual queue is stable.

Our approach for multicast considers virtual queues of data intended for different sinks; the same actual data may be present in more than one virtual queue. All data originating at the sources of a session c is intended for all sinks of that session, but data transmitted on different outgoing links of a node may be intended for different sinks.

Let $U_i^{c\beta}(t)$ be the amount of session c data intended for sink β queued at node i at time t, and define

$$\gamma_{virtualsum}(M) = \limsup_{t\to\infty} \frac{1}{t}\int_0^t 1_{[\sum_{i,c,\beta} U_i^{c\beta}(\tau)>M]} d\tau.$$

Lemma 1 *(a) As $M \to \infty$, $\gamma_i^c(M) \to 0\ \forall\, i, c$ if and only if $\gamma_{virtualsum}(M) \to 0$.*

(b) If the network is stable, there exists a finite value M such that arbitrarily large times $\tilde{t}$ can be found for which $U_i^{c\beta}(\tilde{t}) < M$ for all i, c, β simultaneously.

3 Independent sources

We present results for the wireless case; wired networks are a special case of wireless networks in which each link has a fixed capacity and only one destination node. Let Λ be the set of all input rate matrices (λ_i^c) such that there exist variables $\{f_{abZ}^{c\beta}, g_{aZ}^c\}$ satisfying:

$$f_{abZ}^{c\beta} \geq 0 \quad \forall\ a, b, c, \beta \in \mathcal{T}_c, Z \tag{3}$$

$$\lambda_i^c = \sum_{b,Z} f_{ibZ}^{c\beta} - \sum_{a,Z} f_{aiZ}^{c\beta} \quad \forall\ i, c, \beta \neq i \tag{4}$$

$$\sum_i \lambda_i^c = \sum_{a,Z} f_{a\beta Z}^{c\beta} \quad \forall\ c, \beta \tag{5}$$

$$\sum_{b\in Z} f_{abZ}^{c\beta} \leq g_{aZ}^c \quad \forall\ a, b, c, \beta \in \mathcal{T}_c, Z \tag{6}$$

$$\left(\sum_c g_{aZ}^c\right) \leq (R_{aZ}) \quad \text{for some} \quad (R_{aZ}) \in \Gamma \tag{7}$$

where we define, as in [14], the network *graph family*

$$\Gamma = \sum_{\underline{S}} \pi_{\underline{S}} ConvexHull\{\underline{\mu}(\underline{P}, \underline{S}) | \underline{P} \in \Pi\}, \tag{8}$$

which is shown subsequently to represent the set of all long-term transmission rates (R_{aZ}) supportable by the network. Variables $\{f_{abZ}^{c\beta}\}$ define a flow solution from the session c

sources to β; the total usage of link (a,Z) by (c,β) is the sum over all nodes $b \in Z$ of the variables $f_{abZ}^{c\beta}$.

Theorem 1 *(a) A necessary condition for stability of multiple multicast sessions with intra-session network coding is* $(\lambda_i^c) \in \Lambda$.

(b) A sufficient condition for stability is that (λ_i^c) *is strictly interior to* Λ.

Back-pressure policy for independent sources with network coding

In each time slot $[t, t+T)$, the following are carried out:

Scheduling: For each link (a,Z), one session

$$c_{aZ}^* = \arg\max_c \left\{ \sum_{\beta \in \mathcal{T}_c} \max\left(\max_{b \in Z} \left(U_a^{c\beta} - U_b^{c\beta} \right), 0 \right) \right\}$$

is chosen. Let

$$w_{aZ}^* = \sum_{\beta \in \mathcal{T}_{c_{aZ}^*}} \max\left(\max_{b \in Z} \left(U_a^{c_{aZ}^*\beta} - U_b^{c_{aZ}^*\beta} \right),\ 0 \right). \tag{9}$$

Power control: The state $\underline{S}(t)$ is observed, and a power allocation

$$\underline{P}(t) = \arg\max_{\underline{P} \in \Pi} \sum_{a,Z} \mu_{aZ}(\underline{P}, \underline{S}(t)) w_{aZ}^* \tag{10}$$

is chosen.

Network coding: For each link (a,Z), a random linear combination of data corresponding to each (session, sink) pair $(c_{aZ}^*, \beta \in \mathcal{T}_{c_{aZ}^*})$ for which $\max_{b \in Z} \left(U_a^{c_{aZ}^*\beta} - U_b^{c_{aZ}^*\beta} \right) > 0$ is sent at the rate offered by the power allocation. Each destination node $d \in Z$ associates the received information with the virtual buffers corresponding to sinks $\beta \in \mathcal{T}_{c_{aZ}^*}$ for which $d = \arg\max_{b \in Z} \left(U_a^{c_{aZ}^*\beta} - U_b^{c_{aZ}^*\beta} \right)$.

In a network where simultaneous transmissions interfere, optimizing (10) requires a centralized solution. If there are enough channels for independent transmissions, the optimization can be done independently for each transmitter.

Theorem 2 *If input rates* (λ_i^c) *satisfy* $(\lambda_i^c + \epsilon) \in \Lambda$, *the back-pressure policy stabilizes the system and guarantees an average total buffer occupancy upper bounded by* $\frac{TBN}{\epsilon}$, *where*

$$B = \frac{\tau_{max}}{2} \left(\frac{1}{N} \sum_{i,c} E\left\{ \left(\frac{A_i^c}{T} \right)^2 \right\} + (\mu_{max}^{out} + \mu_{max}^{in})^2 \right)$$

Theorem 3 *Suppose the source data rates are* $(\lambda_i^c - \epsilon')$ *for any* $\epsilon' > 0$, *and that the sources generate random linear combinations of their own data at rates* (λ_i^c), *which are the input to the network. If* (λ_i^c) *is strictly interior to* Λ, *for network coding with no batch restrictions, for sufficiently large time t, the probability that not all sinks are able to decode their respective information decreases exponentially in the length of the code.*

4 Correlated sources

In this section, we consider transmission problems where, for each session c, the exogenous data arriving at sources $\alpha \in \mathcal{S}_c$ in each unit time period are drawn i.i.d. from some joint distribution Q_c. We assume that the exogenous session c arrival rate λ_α^c of each source $\alpha \in \mathcal{S}_c$ is less than or equal to the maximum outflow rate μ_{max}^{out} of a node, and note that

$$\sum_{\alpha \in \mathcal{S}'} \lambda_\alpha^c \geq \sum_{\alpha \in \mathcal{S}'} H(\alpha) \geq H(\mathcal{S}') \geq H(\mathcal{S}' | (\mathcal{S}_c \backslash \mathcal{S}')) \ \forall \ c, \mathcal{S}' \subseteq \mathcal{S}_c.$$

A transmission problem with correlated sources is considered *achievable* with intra-session network coding if there exists a sequence of codes such that the probability of decoding any session c source symbol in error at any sink in $\mathcal{T}_c$ tends to zero. Decoding can be done by a variety of methods such as typical set decoding or minimum entropy decoding.

Let Λ be the set of all values of $(\{H(\mathcal{S}')|\mathcal{S}' \subset \mathcal{S}_c, c \in \mathcal{C}\}, \{\lambda_\alpha^c | \alpha \in \mathcal{S}_c, c \in \mathcal{C}\})$ such that there exist variables (R_{aZ}) and $\{f_{abZ}^{c\alpha\beta}, g_{aZ}^c, \lambda^{c\alpha\beta}\}$ satisfying:

$$f_{abZ}^{c\alpha\beta} \geq 0 \quad \forall \quad a, b, Z, c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{11}$$

$$\sum_{Z, a \in Z} f_{aiZ}^{c\alpha\beta} - \sum_{Z, b \in Z} f_{ibZ}^{c\alpha\beta} = \begin{cases} -\lambda^{c\alpha\beta} & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i = \alpha \\ \lambda^{c\alpha\beta} & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i = \beta \\ 0 & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i \notin \{\alpha, \beta\} \end{cases} \tag{12}$$

$$\sum_{\alpha \in \mathcal{S}_c, b \in Z} f_{abZ}^{c\alpha\beta} \leq g_{aZ}^c \quad \forall \quad a, Z, c, \beta \in \mathcal{T}_c \tag{13}$$

$$\left(\sum_c g_{aZ}^c \right) \leq (R_{aZ}) \quad \text{for some} \quad (R_{aZ}) \in \Gamma \tag{14}$$

$$\lambda^{c\alpha\beta} \leq \lambda_\alpha^c \ \forall \ c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{15}$$

$$\sum_{\alpha \in \mathcal{S}'} \lambda^{c\alpha\beta} \geq H(\mathcal{S}' | (\mathcal{S}_c \backslash \mathcal{S}')) \ \forall \ c, \mathcal{S}' \subseteq \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{16}$$

where Γ is defined as in (8).

Analogously to the independent sources case, variables $\{f_{abZ}^{c\alpha\beta}\}$ define a session c flow of size $\lambda^{c\alpha\beta}$ from source α to sink β. For each (c, β), (16) defines the Slepian-Wolf region [16].

Theorem 4 *A necessary condition for achievability of a transmission problem is that the source statistics satisfy* $(\{H(\mathcal{S}')|\mathcal{S}' \subset \mathcal{S}_c, c \in \mathcal{C}\}, \{\lambda_\alpha^c | \alpha \in \mathcal{S}_c, c \in \mathcal{C}\}) \in \Lambda$.

The back-pressure policy for correlated sources differs from that for independent sources primarily in the operation at the sinks and the sources. The rates at which packets are injected into the network by the different sources of a session may have to be traded off against each other if the total information rate from all the sources exceeds the joint entropy rate. We propose a mechanism in which the different sinks monitor the amount of information received from each of the sources and provide feedback implicity through back-pressure to throttle the source rates. This is accomplished by maintaining virtual queues on a per source basis at each of the sinks and emptying these queues at appropriate rates. The information in these virtual queues creates the necessary gradient in queue sizes that then propagates back to the sources. The sources compress the information stream and transmit packets into the network at rates limited by the gradients and thus each source in the set of correlated sources transmits at the appropriate rate.

Specifically, suppose we have a rate matrix (R_{aZ}) and flow variables $\{f_{abZ}^{c\alpha\beta}, g_{aZ}^c, \lambda^{c\alpha\beta}\}$ satisfying, for some $\epsilon > 0$:

$$f_{abZ}^{c\alpha\beta} \geq 0 \quad \forall \quad a, b, Z, c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{17}$$

$$\sum_{Z, a \in Z} f_{aiZ}^{c\alpha\beta} - \sum_{Z, b \in Z} f_{ibZ}^{c\alpha\beta} = \begin{cases} -\lambda^{c\alpha\beta} & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i = \alpha \\ \lambda^{c\alpha\beta} & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i = \beta \\ 0 & \forall \quad c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c, i \notin \{\alpha, \beta\} \end{cases} \tag{18}$$

$$\sum_{\alpha \in \mathcal{S}_c, b \in Z} f_{abZ}^{c\alpha\beta} \leq g_{aZ}^c \quad \forall \quad a, Z, c, \beta \in \mathcal{T}_c \tag{19}$$

$$\left(\sum_c g_{aZ}^c \right) \leq (R_{aZ}) \quad \text{for some} \quad (R_{aZ}) \in \Gamma \tag{20}$$

$$\lambda^{c\alpha\beta} \leq \lambda_\alpha^c \ \forall \ c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{21}$$

$$\sum_{\alpha \in \mathcal{S}'} \lambda^{c\alpha\beta} \geq H(\mathcal{S}' | (\mathcal{S}_c \backslash \mathcal{S}')) + \epsilon \ \forall \ c, \mathcal{S}' \subseteq \mathcal{S}_c, \beta \in \mathcal{T}_c \tag{22}$$

Each source has a *source buffer* which stores random linear combinations of exogenous data. Specifically, suppose that the link and source data rates are in bits per unit time, where the unit of time is chosen such that all rates are integers. A source $\alpha \in \mathcal{S}_c$ collects its exogenous data bits into length-n blocks, which are viewed as symbols in the finite field $\mathbb{F}_{2^n}$. These symbols are in turn collected into batches of $\tilde{t}\lambda_\alpha^c$ symbols. For each such batch, the source buffer stores an equal number of symbols that are random linear combinations in $\mathbb{F}_{2^n}$ of the exogenous data symbols in the batch. To reduce the code description overhead and complexity, we can collect a number of batches into a group, put one symbol from each batch of a group into a packet, and apply the same random combinations to all symbols in a packet. As in the independent sources case, there is some capacity loss from link/source variability and restrictions on coding at batch boundaries that depends on the link/source statistics, so for simplicity, our analysis does not consider batch restrictions.

Each node i maintains, for each (session, source, sink) triple $(c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c)$, a virtual queue $Q_i^{c\alpha\beta}$ whose length at time t is denoted $U_i^{c\alpha\beta}(t)$, and whose maximum length is M. We define $V_i^{c\alpha\beta}(t) = M - U_i^{c\alpha\beta}(t)$.

Each sink $\beta \in \mathcal{T}_c$ has a *token queue* of length $W^{c\alpha\beta}(t)$ for each $\alpha \in \mathcal{S}_c$. $W^{c\alpha\beta}(t)$ keeps track of the net amount of data "owed" when there is insufficient data to remove from $Q_\beta^{c\alpha\beta}$. This has parallels with the use of overflow buffers for positive flow in [5].

Back-pressure policy

In each time slot $[t, t+T)$, the following are carried out:

Initialization: For each $(c, \alpha \in \mathcal{S}_c, \beta \in \mathcal{T}_c)$, let

$$\hat{W}^{c\alpha\beta}(t) = \min\left(V - V_\beta^{c\alpha\beta}(t^-), W^{c\alpha\beta}(t^-) \right)$$

If this is positive, a corresponding amount of data is removed from $Q_\beta^{c\alpha\beta}$ and from the token queue at β; the queues at β are updated as:

$$\begin{aligned} V_\beta^{c\alpha\beta}(t^+) &= V_\beta^{c\alpha\beta}(t^-) + \hat{W}^{c\alpha\beta}(t) \\ W^{c\alpha\beta}(t^+) &= W^{c\alpha\beta}(t^-) - \hat{W}^{c\alpha\beta}(t) \end{aligned}$$

Outflow rate allocation: Let $\hat{\epsilon}$ be any positive constant less than ϵ. Each sink $\beta \in$

$\mathcal{T}_c$ $\forall$ c chooses outflow variables $\{A_{out}^{c\alpha\beta}(t)|\alpha \in \mathcal{S}_c\}$ to minimize

$$\sum_{\alpha} V_{\beta}^{c\alpha\beta}(t^+)A_{out}^{c\alpha\beta}(t)$$

subject to

$$\sum_{\alpha\in\mathcal{S}'} \frac{A_{out}^{c\alpha\beta}(t)}{T} \geq H(\mathcal{S}'|(\mathcal{S}_c\backslash\mathcal{S}')) + \epsilon - \hat{\epsilon} \ \forall \ c, \mathcal{S}' \subseteq \mathcal{S}_c \tag{23}$$

$$\frac{A_{out}^{c\alpha\beta}(t)}{T} \leq \lambda_{\alpha}^{c} \ \forall \ c, \alpha \in \mathcal{S}_c. \tag{24}$$

It then removes an amount

$$\hat{A}_{out}^{c\alpha\beta}(t) = \min\left[V - V_{\beta}^{c\alpha\beta}(t^+),\ A_{out}^{c\alpha\beta}(t)\right] \tag{25}$$

of data from queue $Q_{\beta}^{c\alpha\beta}$ and an amount $A_{out}^{c\alpha\beta}(t) - \hat{A}_{out}^{c\alpha\beta}(t)$ is added to the token queue at β.

Inflow rate control: Each source $\alpha \in \mathcal{S}_c$ $\forall$ c adds an amount $\min\left\{A_{in}^{c\alpha\beta}(t), V_{\alpha}^{c\alpha\beta}(t)\right\}$ of data from its source buffer to queue $Q_{\alpha}^{c\alpha\beta}$ for each sink $\beta \in \mathcal{T}_c$, where $A_{in}^{c\alpha\beta}(t) = T\lambda_{\alpha}^{c}$.

Scheduling: For each link (a, Z), one session

$$c_{aZ}^* = \arg\max_c \left\{ \sum_{\beta\in\mathcal{T}_c} \max\left(\max_{\alpha\in\mathcal{S}_c}\left(\max_{b\in Z}\left(V_b^{c\alpha\beta} - V_a^{c\alpha\beta}\right)\right),\ 0\right)\right\}$$

is chosen. For each $\beta \in \mathcal{T}_{c_{aZ}^*}$, let

$$\begin{aligned}
\alpha_{aZ}^{\beta*} &= \arg\max_{\alpha\in\mathcal{S}_{c_{aZ}^*}} \left(\max_{b\in Z}\left(V_b^{c_{aZ}^*\alpha\beta} - V_a^{c_{aZ}^*\alpha\beta}\right)\right) \\
b_{aZ}^{\beta*} &= \arg\max_{b\in Z}\left(V_b^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta} - V_a^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}\right) \\
w_{aZ}^* &= \sum_{\beta\in\mathcal{T}_{c_{aZ}^*}} \max\left(V_{b_{aZ}^{\beta*}}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta} - V_a^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta},\ 0\right)
\end{aligned}$$

Power control: The state $\underline{S}(t)$ is observed, and a power allocation

$$\underline{P}(t) = \arg\max_{\underline{P}\in\Pi} \sum_{a,Z} \mu_{aZ}(\underline{P}, \underline{S}(t))w_{aZ}^* \tag{26}$$

is chosen.

Network coding: For each link (a, Z), a random linear combination of data from queues $Q_a^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}$ for all sinks $\beta \in \mathcal{T}_{c_{aZ}^*}$ for which $V_{b_{aZ}^{\beta*}}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta} - V_a^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta} > 0$, up to an amount

$$\begin{cases} V_{b_{aZ}^{\beta*}}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}(t^+) & \text{for} \quad b_{aZ}^{\beta*} \neq \alpha_{aZ}^{\beta*} \\ \max\left\{V_{b_{aZ}^{\beta*}}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}(t^+) - A_{in}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}(t),\ 0\right\} & \text{for} \quad b_{aZ}^{\beta*} = \alpha_{aZ}^{\beta*}, \end{cases}$$

is sent on (a, Z) at the rate offered by the power allocation to the corresponding destination queues $Q_{b_{aZ}^{\beta*}}^{c_{aZ}^*\alpha_{aZ}^{\beta*}\beta}$.

Let σ_{max} and τ_{max} be the maximum number of sources and sinks respectively of a multicast session.

Theorem 5 *Suppose the source statistics satisfy* $(\{H(\mathcal{S}')+\epsilon|\mathcal{S}'\subset\mathcal{S}_c, c\in\mathcal{C}\}, \{\lambda_\alpha^c|\alpha\in\mathcal{S}_c, c\in\mathcal{C}\})\in\Lambda$. *The back-pressure algorithm with* $V=\frac{TBN}{\hat{\epsilon}}$ *and* $M=V+N\mu_{max}^{out}$, *where*

$$B=\frac{\tau_{max}}{2}\left(\frac{1}{N}\sum_{i,c}E\left\{\left(\frac{A_{in}^{c\alpha\beta}}{T}\right)^2\right\}+\frac{2}{N}\sigma_{max}\mu_{max}^{out}\mu_{max}^{in}+(\mu_{max}^{out})^2+(\mu_{max}^{in})^2\right)$$

is stable and asymptotically achieves the desired multicast rates.

5 Summary and Future Work

We have presented dynamic algorithms for multicast network coding, routing, power allocation and scheduling for bursty and time-varying networks, based on back-pressure. For the correlated sources case, source rate control is also achieved through back-pressure by control of per-source virtual queues at the sinks modulating the relative gradients between different sources. This in contrast to the Internet where source rate control is achieved through explicit feedback such as in the transmission control protocol (TCP). Combining network coding with the methods that are currently in widespread use for flow control and scheduling would be an important area for future research. Heuristics for efficient construction of good transmit scenarios and understanding the balance between network coding and interference reduction are also interesting topics for investigation.

References

[1] M. Agarwal, J. H. Cho, L. Gao, and J. Wu. Energy-efficient broadcast in wireless ad hoc networks with hitch-hiking. In *Proceedings of IEEE Infocom*, 2004.

[2] R. Ahlswede, N. Cai, S.-Y.R. Li, and R.W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46:1204–1216, 2000.

[3] B. Awerbuch, A. Brinkmann, and C. Scheideler. Anycasting and multicasting in adversarial systems: routing and admission control, preliminary technical report, 2002.

[4] B. Awerbuch and T. Leighton. A simple local control approximation algorithm for multicommodity flow. In *Proceedings of 34th IEEE Conference on Foundations of Computer Science*, 1993.

[5] B. Awerbuch and T. Leighton. Improved approximation algorithms for the multicommodity flow problem and local competitive routing in dynamic networks. In *Proceedings of 26th ACM Symposium on Theory of Computing*, 1994.

[6] P. A. Chou, Y. Wu, and K. Jain. Practical network coding. In *Proceedings of 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[7] T. Ho, R. Koetter, M. Médard, D. R. Karger, and M. Effros. The benefits of coding over routing in a randomized setting. In *Proceedings of 2003 IEEE International Symposium on Information Theory*, June 2003.

[8] T. Ho, M. Médard, M. Effros, R. Koetter, and D. R. Karger. Network coding for correlated sources. In *Proceedings of Conference on Information Sciences and Systems*, 2004.

[9] T. Ho, M. Médard, J. Shi, M. Effros, and D. R. Karger. On randomized network coding. In *Proceedings of 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[10] T. Ho and H. Viswanathan. Dynamic Algorithms for Multicast with Intra-session Network Coding. Bell Labs Technical Memorandum, August 2005.

[11] T. Klein and H. Viswanathan. Centralized power control in multihop wireless networks. In *Proceedings of IEEE International Symposium on Information Theory*, 2003.

[12] R. Koetter and M. Médard. An algebraic approach to network coding. *IEEE/ACM Transactions on Networking*, October 2003.

[13] I. Maric and R. Yates. Cooperative broadcast for maximum network lifetime.

[14] M. Neely, E. Modiano, and C. E. Rohrs. Dynamic power allocation and routing for time-varying wireless networks. In *Proceedings of IEEE Infocom*, 2003.

[15] S. Sarkar and L. Tassiulas. A framework for routing and congestion control for multicast information flows. *IEEE Transactions on Information Theory*, 2002.

[16] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 25:471 – 480, 1973.

[17] L. Tassiulas and A. F. Ephremides. Stability properties of constrained queuing systems and scheduling policies for maximum throughput in multihop networks. *IEEE Transactions on Information Theory*, 1992.

[18] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wirless networks. In *Proceedings of the IEEE INFOCOM*, 2000.

[19] Y. Wu, P. A. Chou, Q. Zhang, K. Jain, W. Zhu, and S.-Y. Kung. Network planning in wireless ad hoc networks: A cross-layer approach. *IEEE Journal on Selected Areas in Communications, Special Issue on Wireless Ad Hoc Networks*, 2005.

[20] E. Yeh and A. Cohen. Throughput optimal power and rate control in queued multiaccess and fading channels. In *Proceedings of IEEE International Symposium on Information Theory*, 2004.