

Optimal Distributed Multicast Routing using Network Coding

Yi Cui, Yuan Xue and Klara Nahrstedt

Abstract—Multicast is an important communication paradigm, also a problem well known for its difficulty (NP-completeness) to achieve certain optimization goals, such as minimum network delay. Recent advances in network coding has shed a new light onto this problem. In network coding, forwarding nodes can perform arbitrary operations on data received, other than forwarding or replicating, to enhance throughput of a multicast session. In this paper, we show that with the aid of network coding, the once intractable optimal multicast routing problem becomes tractable. In this problem, given a set of multicast sessions and their traffic demands, one tries to route the multicast traffic regarding various objectives, such as to minimize overall delay, or to maximize the battery life of each node. We further show that his problem can be solved in a distributed fashion: each node makes its own routing decisions based on periodic updating information from neighboring nodes. We prove that starting from any initial routing assignment, the proposed distributed routing algorithm is able to converge to the point where the value of the objective function is optimized. Our solution can be fit into a variety of networks to achieve different optimization goals. The example in this paper is maximum lifetime routing in multi-hop wireless network.

I. INTRODUCTION

Optimal data routing in a network can be often understood as a multicommodity flow problem. Given a network and a set of commodities, i.e., a set of source-destination pairs, one tries to achieve certain optimization goal, such as minimum delay, maximum throughput, while maintaining certain fairness among all commodities. The constraints of such optimization problems are usually network link capacity and traffic demand of each commodity. Multicommodity flow problem has been well studied as a typical linear programming problem. Its distributed solutions have also been proposed[1][2].

However, when each commodity becomes a multicast session consisting of a source and several destination nodes, the same problem becomes intractable even in a centralized fashion. If the goal is to minimize network delay, it becomes the Steiner tree problem, which is NP-hard[3]. If the goal is to maximize achievable throughput, its difficulty is equivalent to packing Steiner trees, a problem even harder[4], [5], [6].

Recent advances on network coding has shed a new light onto this problem. Network coding generalizes traditional routing paradigm in which relaying nodes can only forward or replicate, by allowing them to perform arbitrary operation on information received to generate output. It is proved [7][8] that with network coding, the achievable throughput of a multicast session is the minimum of the maximum flow from the sender to any receiver.

Yi Cui and Yuan Xue are with the Department of Electrical Engineering and Computer Science, Vanderbilt University. Their email addresses are {yi.cui, yuan.xue}@vanderbilt.edu. Klara Nahrstedt is with the Department of Computer Science, University of Illinois at Urbana-Champaign. Her email address is klara@cs.uiuc.edu.

What we consider the biggest advantage of network coding is the discovery that it makes the once intractable *optimal multicast routing* problem tractable. Furthermore, we show that this problem can be solved in an entirely distributed fashion. The problem is roughly defined as follows. Given a network, a set of multicast sessions, each with their own traffic demands, we try to route the multicast traffic regarding various objectives, such as to avoid congestion, minimize overall delay, or to maximize the battery life of each node in a wireless network. The rigorous definitions of these objective functions are given in Sec. III.

With the aid of network coding, we are able to formulate the optimal multicast routing problem in the fashion of multicommodity flow (details in Sec. II). The major contribution of this work is an optimal distributed solution to the same problem. In this solution, each node makes its own routing decisions based on periodic updating information from neighboring nodes. More importantly, starting from any initial routing assignment, it should finally converge to the optimal point, such as minimum network delay. Our solution inherits the same design philosophy of Gallager’s algorithm[1], but is significantly different from it, since we try to achieve optimal routing in the setting of multicast communication with network coding.

Although our solution is general enough to fit into a variety of networks, we consider it more suitable to a new generation of application-level networks, such as overlay multicast[9], P2P content distribution, wireless sensor network, where multicast plays an essential role. These networks often has a system goal such as lifetime maximization in sensor network, i.e., to maximize the duration that all sensor nodes are up until one of them has its battery drained. Such a goal of “system optimization” is radically different to “user optimization”, which is the goal of most current networking algorithms. In Internet, each node attempts to send each packet over a route that minimizes that packet’s delay with no regard to other packet’s delays. If the same analogy is applied to sensor network routing, each node would attempt to minimize the amount of energy it spends on each packet transmitted, which deviates from the system optimization goal of maximum lifetime.

We believe our solution is well suited to achieve the goal of “system optimization” in the above mentioned network and application settings. This view is shared by existing works [10][11], which differ from this work in that our solution is suitable to address the case of multiple multicast sessions, while previous works focus on the scenario of single multicast session. Due to space constraint, we delay all proofs in this paper to our technical report, which can be found online[12].

The rest of this paper is organized as follows. We briefly go over the concept of network coding and present our network

model in Sec. II. In Sec. III, we first discuss necessary and sufficient conditions to achieve optimal routing in the general network model, then illustrate the sample scenario: maximum-lifetime routing in multi-hop wireless network. Sec. IV presents our distributed routing algorithm and proves that it is able to converge to the point where the value of the objective function is optimized. Sec. V discusses some practical issues. Finally, Sec. VI concludes the paper.

II. PRELIMINARIES

A. Network Coding: The Concept

An example to illustrate the concept of network coding is shown in Fig. 1. Consider a directed network in which each link has identical capacity. We have a multicast session where S is the sender, R_1 and R_2 are receivers. a and b represent two independent information flows originating from the sender S . As shown in Fig. 1 (b), node 3 transmits the coded flow $a \oplus b$ along the ‘‘bottleneck’’ link (3, 4) to node 4, which in turn forwards the coded flow to receivers r_1 and r_2 , which can recover $\{a, b\}$ from $\{a, a \oplus b\}$ and $\{b, a \oplus b\}$. On the other hand, without network coding (Fig. 1 (a)), receivers r_1 and r_2 can only receive one of the two flows.

It is proved by Ahlswede et al.[7] that with network coding, the achievable throughput of a multicast session can be acquired by running max-flow algorithm from the source to each individual receiver, then choosing the minimal result. Koetter et al.[8] prove the same result using algebraic approach. Li et al.[13] further shows that the above result can be obtained by running linear coding. Chou et al.[14] are the first to propose a practical network coding solution.

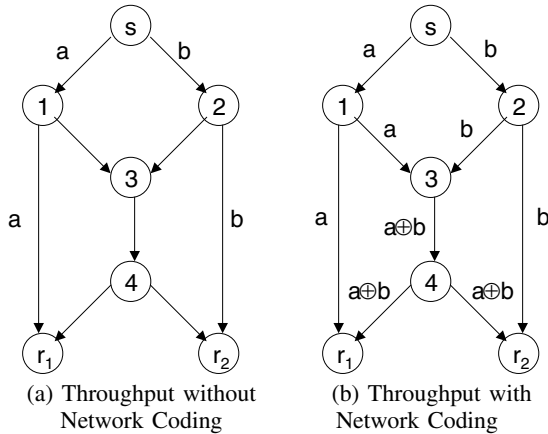


Fig. 1. The Effects of Network Coding

B. Network Model

We consider a n -node network, where the nodes are represented as $\mathcal{N} = \{1, 2, \dots, n\}$. Let \mathcal{L} be the set of links, denoted as $\mathcal{L} = \{(i, k) \mid \text{a link goes from } i \text{ to } k\}$. Each link (i, k) is associated with a capacity C_{ik} . There are a set of multicast sessions \mathcal{M} . For each session $m \in \mathcal{M}$, it has a sender $S(m)$, and a set of receivers $R(m)$.

Let $r_i^m(j) \geq 0$ be the traffic of session m , in bits/s, generating at node i and destined for node j (data sink).

$r_i^m(j) > 0$ only if node $i = S(m)$, and $j \in R(m)$, i.e., if node i is the sender of session m , and node j is one of the receivers of m . We also define node flow $t_i^m(j)$ to be the total traffic of session m at node i destined for node j . $t_i^m(j)$ includes both $r_i^m(j)$ and the traffic from other nodes that is routed through i to destination j . Finally, $\phi_{ik}^m(j)$ is the fraction of the node flow $t_i^m(j)$ routed over link (i, k) . It is always true that $\phi_{ik}^m(j) = 0$ if $(i, k) \notin \mathcal{L}$ (no traffic can be routed through non-existent link), or $i = j$ (traffic that has reached its destination is not sent back into the network). Also, node i must route its entire node flow $t_i^m(j)$ through all links, i.e.,

$$\sum_{k \in \mathcal{N}} \phi_{ik}^m(j) = 1, \forall i, j \in \mathcal{N}, \forall m \in \mathcal{M} \quad (1)$$

Now we express the relation of above notations as follows:

$$t_i^m(j) = r_i^m(j) + \sum_{l \in \mathcal{N}} t_l^m(j) \phi_{li}^m(j), \forall i, j \in \mathcal{N}, \forall m \in \mathcal{M} \quad (2)$$

Eq. (2) expresses flow conservation: for a given multicast session s , the traffic into a node for a given destination is equal to the traffic out of it for the same destination.

Lemma 1 Given the input set \mathbf{r} and routing variable set ϕ , the set of equations (2) has a unique solution for \mathbf{t} . Each element $t_i(j)$ is nonnegative and continuously differentiable as a function of \mathbf{r} and ϕ .

For each session s , we define the amount of traffic on link (i, k) as the union of all flows through it.

$$f_{ik}^m = \max_j t_i^m(j) \phi_{ik}^m(j), \forall (i, k) \in \mathcal{L} \quad (3)$$

According to Ahlswede et al.[7], for a given input set $\mathbf{r} = \{r_i^m(j) \mid i, j \in \mathcal{N}, m \in \mathcal{M}\}$, if there exists a routing solution $\phi = \{\phi_{ik}^m(j) \mid i, j, k \in \mathcal{N}, m \in \mathcal{M}\}$ that is feasible, i.e.,

$$f_{ik} = \sum_{m \in \mathcal{M}} f_{ik}^m \leq C_{ik}, \forall (i, k) \in \mathcal{L} \quad (4)$$

then the achievable throughput by network coding in each multicast session m is $\min_{j \in R(m)} r_{S(m)}^m(j)$. Furthermore, any feasible solution is schedulable by a network coding assignment.

Now we can formalize our ‘‘system optimization’’ goal according to the following format. For example, if the delay on each link (i, k) is a function of traffic on it, $D_{ik}(f_{ik})$, and our goal is to minimize the overall network delay, it can be formalized into the following optimization problem.

$$\begin{aligned} \mathbf{D:} \quad & \text{minimize} \quad D = \sum_{(i,k) \in \mathcal{L}} D_{ik}(f_{ik}) \\ & \text{subject to} \quad (1), (2) \text{ (flow constraint)} \\ & \quad \quad \quad (3) \text{ (union of flow constraint)} \\ & \quad \quad \quad (4) \text{ (capacity constraint)} \end{aligned}$$

III. OPTIMALITY CONDITIONS FOR DISTRIBUTED MULTICAST ROUTING

In this section, we analyze the optimality conditions for distributed multicast routing. We show that when the system

delay D is minimized, within each session m , each node i , for a given receiver j , the partial derivative of D to the routing variable $\phi_{ik}^m(j)$ (marginal delay on link (i, k)) is the same for all links (i, k) originating from node i .

An analogy is that within an electrical network where each wire has different resistance, certain currents flow from the sender node to the receiver node. By Dirichlet principle, the potentials taken within the electrical network minimize the total energy dissipation. And when it happens, the potentials (partial derivative of energy dissipation to currents) of all wires sharing the same positive end are the same.

Sec. III-A goes through the formal analysis to reach the above result. Note that although we use delay as an example objective, the same conclusion holds for any type of objective function. In Sec. III-B, we show that with necessary adjustment to the network model and objective function, we can derive the same optimality conditions for multicast routing in a wide spectrum of problem settings. Our example is maximum lifetime routing in multi-hop wireless network.

A. General Model

We calculate the partial derivatives of the delay D with respect to the inputs \mathbf{r} and the routing variables ϕ . We first consider $\partial D/\partial r_i^m(j)$. Assume a small increment ϵ in the input $r_i^m(j)$. For each adjacent node k , an increment $\epsilon\phi_{ik}^m(j)$ of this new incoming traffic will flow over (i, k) , and to first order, this will cause an increment delay on that link of

$$\epsilon\phi_{ik}^m(j)D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$$

where

$$D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) = \frac{dD_{ik}(f_{ik})}{df_{ik}} \cdot \frac{df_{ik}}{d(t_i^m(j)\phi_{ik}^m(j))}$$

$D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$ can be calculated as follows. A commonly used link delay function is defined by Kleinrock[15] as follows.

$$D_{ik}(f_{ik}) = \frac{f_{ik}}{(C_{ik} - f_{ik})} \quad (5)$$

This function assumes that queueing delays are the only noneligible source of delay in a network, and each link traffic can be modelled as Poisson message arrivals with independent exponentially distributed lengths. In fact, we do not need to know what $D_{ik}(f_{ik})$ is, as long as this function is increasing and convex in f_{ik} . In practice, we can also choose to directly measure D_{ik} and its derivative, which we will discuss in Sec. V.

According to Eq. (4), $df_{ik}/d(t_i^m(j)\phi_{ik}^m(j)) = 1$. According to Eq. (3),

$$\frac{df_{ik}^m}{d(t_i^m(j)\phi_{ik}^m(j))} = \begin{cases} 1/n & \text{if } t_i^m(j)\phi_{ik}^m(j) \text{ and } n-1 \\ & \text{other flows on link}(i, k) \\ & \text{are the maximum} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

If node k is not the destination node, then the increment $\epsilon\phi_{ik}^m(j)$ of extra traffic at node k will cause the same incremental delay onward as an increment $\epsilon\phi_{ik}^m(j)$ of new input traffic at node k . To first order this incremental delay will be

$\epsilon\phi_{ik}^m(j)\partial D/\partial r_k^m(j)$. Summing over all adjacent nodes k , then, we find that,

$$\begin{aligned} \frac{\partial D}{\partial r_i^m(j)} &= \sum_{k \in \mathcal{N}} \phi_{ik}^m(j) \left[D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) + \frac{\partial D}{\partial r_k^m(j)} \right] \\ &= \sum_{k \in \mathcal{N}} \phi_{ik}^m(j) \delta_{ik}^m(j) \end{aligned} \quad (7)$$

Here, $\delta_{ik}^m(j) = D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) + \frac{\partial D}{\partial r_k^m(j)}$ is called the marginal delay of link (i, k) with respect to receiver j . (7) asserts that the marginal delay of a node is the convex sum of the marginal delays of its outgoing links with respect to the same destination. By the definition of ϕ , we can see that $\partial D/\partial r_j^m(j) = 0$, since $\phi_{jk}^m(j) = 0$, i.e., no traffic of receiver j needs to be routed anymore once it arrives to the destination.

Next consider $\partial D/\partial \phi_{ik}^m(j)$. An increment ϵ in $\phi_{ik}^m(j)$ causes an increment $\epsilon t_i^m(j)$ in the portion of $t_i^m(j)$ flowing on link (i, k) . If $k \neq j$, this causes an addition $\epsilon t_i^m(j)$ to the traffic at k destined for j . Thus for $(i, k) \in \mathcal{L}$, $i \neq j$,

$$\begin{aligned} \frac{\partial D}{\partial \phi_{ik}^m(j)} &= t_i^m(j) \left[D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) + \frac{\partial D}{\partial r_k^m(j)} \right] \\ &= t_i^m(j) \delta_{ik}^m(j) \end{aligned} \quad (8)$$

To summarize above discussions, we have the following theorems.

Theorem 1: Let a network have inputs \mathbf{r} and routing variables ϕ , and let each marginal delay $dD_{ik}(f_{ik})/df_{ik}$ be continuous in f_{ik} , $(i, k) \in \mathcal{L}$. Then the set of equations (7), $i \neq j$, has a unique (and correct) set of solutions for $\partial D/\partial r_i^m(j)$. Furthermore, (8) is valid and both $\partial D/\partial r_i^m(j)$ and $\partial D/\partial \phi_{ik}^m(j)$ for $i \neq j$, $(i, k) \in \mathcal{L}$ are continuous in \mathbf{r} and ϕ .

Theorem 2: Assume that D_{ik} is convex and continuously differentiable for f_{ik} . let ψ be the set of ϕ , the necessary condition for ϕ to minimize D over ψ is

$$\frac{\partial D}{\partial \phi_{ik}^m(j)} \begin{cases} = \min_l \partial D/\partial \phi_{il}^m(j) & \text{if } \phi_{ik}^m > 0 \\ \geq \min_l \partial D/\partial \phi_{il}^m(j) & \text{if } \phi_{ik}^m = 0 \end{cases} \quad (9)$$

and the sufficient condition for ϕ to minimize D over ψ is

$$\delta_{ik}^m(j) \geq \frac{\partial D}{\partial r_i^m(j)}, \forall i \neq j, (i, k) \in \mathcal{L}, \forall m \in \mathcal{M} \quad (10)$$

The necessary condition (9) in Theorem 2 states that within session m , at node i , for a given receiver j , all links (i, k) that have any portion of flow $t_i^m(j)$ routed through ($\phi_{ik}^m(j) > 0$) must achieve the same minimum marginal delay with respect to j , and that this minimum marginal delay must be less than or equal to the same marginal delays of the links with no flow routed ($\phi_{ik}^m(j) = 0$).

The sufficient condition (10) states that within session m , at node i , for a given receiver j , the marginal delay of all links (i, k) with respect to j must be greater than or equal to the marginal delay of node i .

B. Maximum lifetime Routing in Multihop Wireless Network

In multi-hop wireless network such as sensor network, data is sent through wireless link, which consumes limited battery energy of both sender node and receiver node. Energy-efficient routing thus becomes an important issue. Our goal here is to maximize the lifetime of the network, i.e., the duration in which all nodes are up until one of them is drained of energy.

We define E_i the energy reserve at node i . Let p_i^r (J/bit) be the power consumption at node i , when it receives one unit of data, and p_{ik}^t (J/bit) be the power consumption when one unit of data is sent from i over link (i, k) . Based on the first order radio model, we have the following.

$$p_i^r = a \quad (11)$$

$$p_{ik}^t = a + b \cdot (d_{ik})^\theta \quad (12)$$

Here, a is a distance-independent constant that represents the energy consumption to run the transmitter or receiver circuitry, and b is the coefficient of the distance-dependent term that represents the transmit amplifier. d_{ik} is the distance from node i to k . The exponent θ is determined from field measurements, which is typically a constant between 2 and 4. The power consumption ratio (J/s) of node i is

$$p_i = \sum_{k \in \mathcal{N}} [f_{ik} \cdot p_{ik}^t + f_{ki} \cdot p^r], \forall i \in \mathcal{N} \quad (13)$$

Now it is clear that the lifetime of node i is

$$T_i = \frac{E_i}{p_i} \quad (14)$$

Our target is to maximize the minimum lifetime of all nodes, i.e., the duration that all nodes within the network are up. Associating T_i with a utility U_i , this goal can be formalized as to maximize the aggregate utility of all nodes as follows.

$$\begin{aligned} \mathbf{U:} \quad & \text{maximize} \quad U = \sum_{i \in \mathcal{N}} U_i = \sum_{i \in \mathcal{N}} \frac{T_i^{1-\gamma}}{1-\gamma}, \gamma \rightarrow \infty \\ & \text{subject to} \quad (1), (2) \text{ (flow constraint)} \\ & \quad (3) \text{ (union of flow constraint)} \\ & \quad (4) \text{ (capacity constraint)} \\ & \quad (13), (14) \text{ (power constraint)} \end{aligned}$$

Here γ can be made an arbitrarily large number to infinitely approximate the optimal value.

We first consider $\partial U / \partial r_i^m(j)$, the *marginal utility* on node i with respect to receiver j . Assume that there is a small increment ϵ on the input traffic $r_i^m(j)$. Then $\epsilon \phi_{ik}^m(j)$ from this new incoming traffic will flow over wireless link (i, k) . This will cause an increment power consumption on node i ,

$$\epsilon \phi_{ik}^m(j) p_{ik}^t \frac{df_{ik}^m}{d(t_i^m(j) \phi_{ik}^m(j))}$$

in order to send out the incremented traffic. The definition of $df_{ik}^m / d(t_i^m(j) \phi_{ik}^m(j))$ can be found at Eq. (6). And the consequent utility change of node i is

$$\epsilon \phi_{ik}^m(j) U_i'(p_i) p_{ik}^t \frac{df_{ik}^m}{d(t_i^m(j) \phi_{ik}^m(j))}$$

Similarly, on the receiver side, the utility change of node k is

$$\epsilon \phi_{ik}^m(j) U_k'(p_k) p_k^r \frac{df_{ik}^m}{d(t_i^m(j) \phi_{ik}^m(j))}$$

If node k is not the destination node, then the increment $\epsilon \phi_{ik}^m(j)$ of extra traffic at node k will cause the same utility change onward as a result of the increment $\epsilon \phi_{ik}^m(j)$ of input traffic at node k . To first order this utility change will be $\epsilon \phi_{ik}^m(j) \partial U / \partial r_k(j)$. Summing over all adjacent nodes k , then, we find that,

$$\begin{aligned} \frac{\partial U}{\partial r_i^m(j)} &= \sum_{k \in \mathcal{N}} \phi_{ik}^m(j) \left[\frac{\partial U}{\partial r_k^m(j)} + \right. \\ & \quad \left. (p_{ik}^t U_i'(p_i) + p_k^r U_k'(p_k)) \frac{df_{ik}^m}{d(t_i^m(j) \phi_{ik}^m(j))} \right] \\ &= \sum_{k \in \mathcal{N}} \phi_{ik}^m(j) \left[U'_{ik}(t_i^m(j) \phi_{ik}^m(j)) + \frac{\partial U}{\partial r_k^m(j)} \right] \end{aligned} \quad (15)$$

where $U'_{ik}(t_i^m(j) \phi_{ik}^m(j)) = (p_{ik}^t U_i'(p_i) + p_k^r U_k'(p_k)) \cdot \frac{df_{ik}^m}{d(t_i^m(j) \phi_{ik}^m(j))}$ is called the marginal utility on link (i, k) , and $U'_{ik}(t_i^m(j) \phi_{ik}^m(j)) + \frac{\partial U}{\partial r_k^m(j)}$ is called the marginal utility of link (i, k) with respect to receiver j .

(15) asserts that the marginal utility of a node is the convex sum of the marginal utilities of its outgoing links with respect to the same receiver. By the definition of ϕ , we can see that $\partial U / \partial r_j^m(j) = 0$, since $\phi_{jk}^m(j) = 0$, i.e., no traffic of receiver j needs to be routed anymore once it arrives to the destination.

Next we consider $\partial U / \partial \phi_{ik}^m(j)$. An increment ϵ in $\phi_{ik}^m(j)$ causes an increment $\epsilon t_i^m(j)$ in the portion of $t_i^m(j)$ flowing on link (i, k) . If $k \neq j$, this causes an addition $\epsilon t_i^m(j)$ to the traffic at k destined for j . Thus for $(i, k) \in \mathcal{L}$, $i \neq j$,

$$\frac{\partial U}{\partial \phi_{ik}^m(j)} = t_i^m(j) \left[U'_{ik}(t_i^m(j) \phi_{ik}^m(j)) + \frac{\partial U}{\partial r_k^m(j)} \right] \quad (16)$$

Similar to Theorem 1 and 2, we are able to prove corollaries about maximum lifetime routing in wireless network, which can be found in our technical report[12].

IV. DISTRIBUTED ROUTING ALGORITHM

By understanding the optimality conditions (general model discussed in Sec. III-A) to multicast routing, the design philosophy of our routing scheme should now be clear. The algorithm works in an iterative fashion. In each iteration, for each session m , each node i and a given receiver j , i must incrementally increase the fraction of traffic on link (i, k) (by increasing $\phi_{ik}^m(j)$) whose marginal delay $\delta_{ik}^m(j)$ is small, and do the reverse for those links whose marginal delay is big, until the marginal delays of all links carrying traffic are equal. When this condition is met for all nodes regarding all receivers within all sessions, the entire system reaches the optimal point.

Therefore, for each session m , each node i , each iteration involves two steps: (1) the calculation of marginal delay $D'_{ik}(t_i^m(j) \phi_{ik}^m(j))$ for each outgoing link (i, k) , and each of its downstream neighbors k 's marginal delay $\partial D / \partial r_k^m(j)$; (2) the adjustment of routing variables $\phi_{ik}^m(j)$ based on the values of $D'_{ik}(t_i^m(j) \phi_{ik}^m(j))$ and $\partial D / \partial r_k^m(j)$. We will elaborate them in details as follows.

Sec. IV-A introduces how the calculation and update of marginal delays $D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$ and $\partial D/\partial r_k^m(j)$ are executed. Sec. IV-B discusses how to maintain loop-free routing. Sec. IV-C formally presents the algorithm, whose optimal property is analyzed in Sec. IV-D.

A. Calculation of Marginal Delays

We first see how each node i calculates its marginal delay $\partial D/\partial r_i^m(j)$, with respect to receiver j . In order to do so, based on Eq. (7), i needs to know $\delta_{ik}(j) = D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) + \partial D/\partial r_k^m(j)$, the marginal delays of all its outgoing links regarding receiver j . In Sec. III-A, we have discussed how to calculate $D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$, and $\partial D/\partial r_k^m(j)$ is the marginal delay of i 's downstream neighbor k . Now it is clear that $\partial D/\partial r_i(j)$ should be calculated in a recursive way. Starting from receiver, $\partial D/\partial r_j^m(j) = 0$ based on definition. j then sends the values of $\partial D/\partial r_j^m(j)$ to its upstream neighbor, say k . Upon receiving the updates, node k can calculate $D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$ as described above, then acquire $\partial D/\partial r_k^m(j)$. Then, k repeats the same procedure to its upstream neighbor, until node i is reached.

B. Loop-free Routing

From the above calculation, we can see that among all nodes carrying traffic of session m , their marginal delays follow a partial ordering. Each receiver j has the lowest marginal delay, which is 0. Its upstream neighbors have higher marginal delays, whose own upstream neighbors have even higher marginal delays. Therefore, the recursive procedure of node marginal delay calculation is free of deadlock if and only if such a partial ordering is maintained, i.e., the routing variable set ϕ is loop free.

In order to achieve loop-free routing, for each node i , with respect to receiver j , we introduce a set $B_{i,\phi}^m(j)$ of blocked nodes k for which $\phi_{ik}^m(j) = 0$ and the algorithm is not permitted to increase $\phi_{ik}^m(j)$ from 0. $k \in B_{i,\phi}^m(j)$ if one of the following conditions is met.

- 1) $(i, k) \notin \mathcal{L}$, i.e., k is not the neighbor of i .
- 2) $\phi_{ik}^m(j) = 0$ and $\partial D/\partial r_i^m(j) \leq \partial D/\partial r_k^m(j)$, i.e., the marginal delay of k is already greater than or equal to the marginal delay of i .
- 3) $\phi_{ik}^m(j) = 0$ and $\exists (l, p) \in \mathcal{L}$ such that (a) $l = k$ or l is downstream to k with respect to receiver j ; (b) $\phi_{lp}^m(j) > 0$, and $\partial D/\partial r_l^m(j) \leq \partial D/\partial r_p^m(j)$, i.e., (l, p) is an improper link.

C. Algorithm

Now we are ready to formalize our algorithm. We use $\phi^{(k)}$ to represent the routing variable set at the iteration k . $\Delta\phi^{(k)}$ is the changes made to $\phi^{(k)}$ during the iteration k . Apparently, $\phi^{(k+1)} = \phi^{(k)} + \Delta\phi^{(k)}$. Also for node i ,

- $\delta_i^m(j) = (\delta_{i1}^m(j), \dots, \delta_{in}^m(j))^T$ is the vector of its routing variable regarding receiver j and session m .
- $\Delta\phi_i^m(j) = (\Delta\phi_{i1}^m(j), \dots, \Delta\phi_{in}^m(j))^T$ is the vector of changes to $\phi_i^m(j)$.

- $\delta_i^m(j) = (\delta_{i1}^m(j), \dots, \delta_{in}^m(j))^T$ is the vector of marginal delays of all i 's neighbors.

At iteration k , node i operates according to the following steps.

- 1) For each session m , calculate link marginal delay $D'_{ik}(t_i^m(j)\phi_{ik}^m(j))$ for each of its outgoing links (i, k) , get updates of marginal delays $\partial D/\partial r_k^m(j)$ from each of its downstream neighbors k , then calculate $\delta_{ik}^m(j) = D'_{ik}(t_i^m(j)\phi_{ik}^m(j)) + \partial D/\partial r_k^m(j)$.
- 2) Calculate its own marginal delay $\partial D/\partial r_i^m(j)$ according to Eq. (7), and send it to all its upstream neighbors.
- 3) Calculate $\phi_i^m(j)^{(k)}$ by solving the problem

$$\begin{aligned} & \text{minimize} && \delta_i^m(j)^T \Delta\phi_i^m(j) + \frac{t_i^m(j)}{2\alpha} \cdot && (17) \\ & && (\Delta\phi_i^m(j)^{(k)})^T \mathbf{M}_i^m(j)^{(k)} \Delta\phi_i^m(j)^{(k)} \\ & \text{subject to} && \phi_i^m(j)^{(k)} + \Delta\phi_i^m(j)^{(k)} \geq 0, \\ & && \sum_{l \in \mathcal{N}} \Delta\phi_{il}^m(j)^{(k)} = 0, \Delta\phi_{il}^m(j)^{(k)} = 0, \\ & && \forall l \in B_{i,\phi^{(k)}}^m(j) \end{aligned}$$

where $\alpha > 0$ is some positive stepsize, and matrix $\mathbf{M}_i^m(j)^{(k)}$ is some symmetric matrix which is positive definite on the subspace $\{\Delta\phi_i^m(j) \mid \sum_{l \in \mathcal{N}} \Delta\phi_{il}^m(j) = 0\}$.

- 4) Adjust routing variables

$$\begin{aligned} \phi_i^m(j)^{(k+1)} &= \phi_i^m(j)^{(k)} + \Delta\phi_i^m(j)^{(k)} \\ \forall i \in \mathcal{N} - \{j\}, \forall m \in \mathcal{M} \end{aligned}$$

Note that in problem (17), $\mathbf{M}_i^m(j)^{(k)}$ can be any positive definite matrix, and any solution $\Delta\phi_i^m(j)$ to this problem will allocate more traffic on the link with the minimum marginal delay, and decrease traffic on other links. If we implement $\mathbf{M}_i^m(j)^{(k)}$ as the identity matrix, the solution to $\Delta\phi_i^m(j)^{(k)}$ boils down to

$$\begin{aligned} & \Delta\phi_{il}^m(j)^{(k)} \\ &= \begin{cases} 0 & \text{if } l \in B_{i,\phi^{(k)}}^m(j) \\ -\min\{\phi_{il}^m(j)^{(k)}, \frac{\alpha(\delta_{il}^m(j) - \delta_{\min}^m(j))}{t_i^m(j)}\} & \text{if } \delta_{il}^m(j) \neq \delta_{\min}^m(j) \\ \sum_{\delta_{ip}^m(j) \neq \delta_{\min}^m(j)} \Delta\phi_{ip}^m(j)^{(k)} & \text{if } \delta_{il}^m(j) = \delta_{\min}^m(j) \end{cases} \end{aligned}$$

where $\delta_{\min}^m(j) = \min_{p \notin B_{i,\phi^{(k)}}^m(j)} \delta_{ip}^m(j)$.

This algorithm increase the fraction of traffic on the link with the minimum marginal delay, and reduces the fraction of other links. The amount of reduction on link (i, l) , given by $\Delta\phi_{il}^m(j)^{(k)}$, is proportional to $\delta_{il}^m(j) - \delta_{\min}^m(j)$, the difference of marginal delays between (i, l) itself and the link with the minimum marginal delay. It is further restricted that $\Delta\phi_{il}^m(j)^{(k)} \leq \phi_{il}^m(j)^{(k)}$, i.e., $\Delta\phi_{il}^m(j)^{(k)}$ should not turn $\phi_{il}^m(j)^{(k)}$ to negative. The amount of reduction is also inversely proportional to $t_i^m(j)$, since the change in link traffic is related to $\Delta\phi_{il}^m(j)^{(k)} t_i^m(j)$. When $t_i^m(j)$ is small, $\Delta\phi_{il}^m(j)^{(k)}$ can be changed by a large amount without greatly affecting the marginal delays. Finally, the change depends on the stepsize α . As shown later in Theorem 3, convergence can be guaranteed

if α is small enough. As α increases, the speed of convergence increases but the danger of no convergence also increases.

We can implement $M_i^m(j)^{(k)}$ differently to further improve convergence speed. For example, Bertsekas et al.[2] choose to set $M_i^m(j)^{(k)}$ as a diagonal matrix where the element at the l th row and l th column is the second derivative¹ of delay D to routing variable $\phi_{il}^m(j)$, i.e., $\partial^2 D / (\partial \phi_{il}^m(j))^2$. We investigate the performances of both implementation choices in our technical report[12].

D. Analysis

The following theorem shows the main convergence result.

Theorem 3: Let the initial routing $\phi^{(0)}$ be loop-free and satisfy $D(\phi^{(0)}) \leq D_0$ where D_0 is some scalar. Assume also that there exist two positive scalars λ, Λ such that for each session m , each node i , and each receiver j , the sequences of matrices $\{M_i^m(j)^{(k)}\}$ satisfy the following two conditions.

(a) The absolute value of each element of $M_i^m(j)^{(k)}$ is bounded above by Λ .

(b) There holds

$$\lambda |v_i|^2 \leq v_i^T M_i^m(j)^{(k)} v_i$$

for all v_i such that $\sum_{l \notin B_{i, \phi^{(k)}}^m(j)} v_{il} = 0$.

Then there exists a positive scalar $\bar{\alpha}$ (depending on D_0, λ , and Λ) such that for all $\alpha \in (0, \bar{\alpha}]$ and $k = 0, 1, \dots$, the sequence $\{\phi^{(k)}\}$ generated by the algorithm satisfies

$$D(\phi^{(k+1)}) \leq D(\phi^{(k)})$$

$$\lim_{k \rightarrow \infty} D(\phi^{(k+1)}) = \min_{\phi \in \psi} D(\phi)$$

Furthermore, every limit point of $\{\phi^{(k)}\}$ is an optimal solution to problem defined in step (3) of the algorithm.

In the similar fashion, we derive the algorithm for maximum-lifetime routing in wireless network, which can be found in our technical report[12].

V. PRACTICAL ISSUES

In the real minimum-delay (maximum-lifetime) routing environment, we cannot assume the delay (energy consumption) function of a link to be exactly the same as what is defined in Eq. (5) (or Eq. (11) and (12)). In [16], a procedure is presented for estimating online marginal packet delays through links with respect to link flows without making the standard assumptions (exponentially distributed packet lengths, Poisson arrival processes). This procedure is based on a technique known as perturbation analysis. No knowledge of network parameters (arrival rates, link capacities) is required. Similarly, in maximum-lifetime wireless routing environment, we can adopt the same approach. During the calculation of marginal utility $U'_{ik}(t_i^m(j)\phi_{ik}^m(j))$, node i or k can estimate its power consumption ratio by directly measuring the amount of data sent and the corresponding energy dissipation during the most recent period, then derive the marginal utility based on

¹In fact, since $\partial^2 D / (\partial \phi_{il}^m(j))^2$ is difficult to compute, this element is usually set to be its upper bound.

Eq. (14) and the definition of U , both of which are predefined independent of power consumption models of wireless nodes.

In each iteration of our algorithm, the destination node of each link needs to update the marginal delay or marginal utility of this link to the source node. Therefore, a total of $|\mathcal{L}|$ messages need to be sent, $|\mathcal{L}|$ being the number of links inside the network. In case there are more than one multicast sessions, the number of messages required can stay unchanged if each node aggregates its marginal delays or marginal utilities regarding all sessions into a single message. Such messaging overhead can be further saved if we piggyback these messages into data/acknowledgement packets.

VI. CONCLUSION

This paper presents a general solution for optimal multicast routing. We show that with the aid of network coding, the once intractable optimal multicast routing problem becomes tractable. We further show that this problem can be solved in an entirely distributed fashion by presenting a distributed routing algorithm, which is proved to converge to the point where the value of the objective function is optimized. Our solution can be fit into a variety of networks to achieve different optimization goals, such as maximum lifetime routing in multi-hop wireless network.

REFERENCES

- [1] R. Gallager, "A minimum delay routing algorithm using distributed computation," *IEEE Tran. Commun.*, vol. 25, 1977.
- [2] D. Bertsekas, E. Gafni, and R. Gallager, "Second derivative algorithms for minimum delay distributed routing in networks," *IEEE Tran. Commun.*, vol. 32, 1984.
- [3] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, 1979.
- [4] M. Thimm, "On the approximability of the steiner tree problem," in *Mathematical Foundations of Computer Science*. 2001, Springer LNCS 2136.
- [5] G. Robins and A. Zelikovsky, "Improved steiner tree approximation in graphs," in *Proc. of 7th ACM-SIAM Symp. on Discrete Algorithms*, 2000.
- [6] K. Jain, M. Mahdian, and M. Salavatipour, "Packing steiner trees," in *Proc. of 10th ACM-SIAM Symp. on Discrete Algorithms*, 2003.
- [7] R. Ahlswede, N. Cai, S.R. Li, and R.W. Yeung, "Network information flow," *IEEE Tran. Information Theory*, vol. 46, 2000.
- [8] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE Tran. Networking*, vol. 11, 2003.
- [9] Y. Chu, R. Rao, and H. Zhang, "A case for end system multicast," in *ACM SIGMETRICS*, 2000.
- [10] D. S. Lun, N. Ratnakar, R. Koetter, M. Medard, E. Ahmed, and H. Lee, "Achieving minimum-cost multicast: A decentralized approach based on network coding," in *IEEE INFOCOM*, 2005.
- [11] P. S. Lun, M. Medard, T. Ho, and R. Koetter, "Network coding with a cost criterion," in *International Symposium on Information Theory and Its Applications (ISITA)*, 2004.
- [12] Y. Cui, Y. Xue and K. Nahrstedt, "Optimal distributed multicast routing using network coding: Theory and applications," <http://cairo.cs.uiuc.edu/~yicui/report.pdf>, 2004.
- [13] S.R. Li, R.W. Yeung, and N. Cai, "Linear network coding," *IEEE Tran. Information Theory*, vol. 49, 2003.
- [14] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," *Allerton Conference on Communication, Control, and Computing*, 2003.
- [15] L. Kleinrock, *Communication Nets: Stochastic Message Flow and Delay*, McGraw-Hill, 1964.
- [16] C. Cassandras, M. Abidi, and D. Towsley, "Distributed routing with on-line marginal delay estimation," *IEEE Tran. Commun.*, vol. 38, 1990.