

Scheduling for Network Coded Multicast: A Conflict Graph Formulation

Danail Traskov[†], Michael Heindlmaier[†], Muriel Médard[‡], Ralf Koetter[†], and Desmond S. Lun[§]

[†]Institute for Communications Engineering
Technical University Munich
Munich, Germany

[‡]LIDS
MIT
Cambridge, MA

[§]Broad Institute
MIT and Harvard
Cambridge, MA

Email: {danail.traskov, michael.heindlmaier, ralf.koetter}@tum.de Email: medard@mit.edu Email: dlun@broad.mit.edu

Abstract—Consider network coded multicast traffic over a wireless network in the bandwidth limited regime. We formulate the joint medium access and subgraph optimization problem by means of a graphical conflict model. The nature of network coded flows is not captured by classical link-based scheduling and therefore requires a novel approach based on conflicting hyperarcs. By means of simulations, we evaluate the performance of our algorithm and conclude that it significantly outperforms existing scheduling techniques.

I. INTRODUCTION

In a multi-hop wireless network an optimal strategy to multicast to a group of receivers is to use random linear network coding over an optimized subgraph. Random linear codes are capacity achieving for a given subgraph [1] and this subgraph can be computed by means of solving a linear or convex program [2]. The subgraph optimization is a fairly tractable problem and furthermore solvable in a distributed fashion making it relevant in practice. An essential assumption of [2] is that all nodes in the network transmit on orthogonal channels and thus conflicts due to interfering transmissions do not arise. This is a reasonable assumption in a power limited regime, where bandwidth is not scarce and one might simply avoid interference by orthogonalizing the entire network. However, in many if not most scenarios wireless networks are interference limited. To that end a high frequency reuse within the network is necessary which has to be achieved by carefully scheduling simultaneous transmissions.

The prevalent approach to this problem is to construct an interference-free transmission schedule by means of some heuristic and then to compute an optimal subgraph over this, now essentially orthogonal, network. As an example, in [3] the authors propose a suboptimal collision-free strategy where two nodes cannot transmit simultaneously if they are within two hops. While this is a practical solution, it is not clear how it affects the overall performance of the network. Furthermore, from a systems perspective it is unsatisfying to have one component (the network coding subgraph) carefully optimized, while the other essential part (the medium access) is done in a more or less ad-hoc fashion.

To that end, we suggest a framework, where subgraph optimization and channel access are treated jointly. We construct

a hypergraph that takes into account possible transmissions to every subset of neighbors of a node. Each such subset is represented by a hyperarc. The essential novelty of our approach is that we consider subsets of hyperarcs that can be activated simultaneously without interfering. These scheduling constraints are transformed into a conflict graph representation, where the hyperarcs are represented by nodes and the activation constraints are given by edges: A set of hyperarcs can be activated simultaneously only if they are not connected by any edge in the conflict graph. As a naïve approach, one might consider a sequence of valid configurations and search exhaustively over the network “unfolded” over time. Since the number of valid configurations scales exponentially in the size of the network such an approach is prohibitive from a computational point of view. We exploit the polymatroid representation of the rate regions associated with valid network configurations to compute a schedule succinctly as a convex combination of valid configurations.

The problem formulation of scheduling as finding stable sets is well known e.g. for routed traffic in [4], for scheduling in switches in [5], for network code construction in [6], and in the context of Banyan networks in [7]. For network coded traffic, however, it gives rise to a number of novel and interesting observations. It takes into account that nodes broadcast coded packets to all neighbors (the wireless broadcast advantage) and also that transmissions are subject to interference. By guaranteeing a node successful transmission to a subset of its neighbors and at the same time permitting conflicts on the remaining neighbors, we are not seeking to minimize the number of collisions per se. In fact, one can argue that we are scheduling conflicts for the nodes not contained in the activated hyperarc.

In summary, the novelty of our approach is that we

- formulate a precise scheduling model for network coded traffic,
- solve the joint multi-access and subgraph optimization problem,
- and report gains of up to 90% compared to widely used scheduling techniques.

In contrast to [8], where the authors focus on minimizing power consumption, we consider a wireless network where interference is the limiting factor. Furthermore, the scheduling of broadcast transmissions is introduced in [9] in a different context, namely in an attempt to analyze the opportunistic,

This work was supported by the United States Department of the Navy’s Space and Naval Warfare Systems Command (SPAWAR) under Contract No. N66001-06-C-2020 through BAE Systems (DARPA CBMANET) and by the European Union through NEWCOM++.

local combination of packets belonging to multiple unicast connections.

The remainder of the paper is organized as follows: In Section II, we introduce the network model and illustrate the construction of the conflict graph. In Section III, we derive the rate region for multicast traffic in terms of an optimization problem including explicitly the medium access constraints. Our simulation results are presented in Section IV, whereas the complexity in general and for certain special cases is the focus of Section V. Finally, in Section VI, we conclude the paper and briefly discuss opportunities for further work.

II. NETWORK MODEL

Consider a wireless network represented by a set of nodes \mathcal{N} and for each node $i \in \mathcal{N}$ a set of neighbors $N(i) \subset \mathcal{N}$. We assume that when i transmits all nodes in $N(i)$ are in radio range and can potentially receive or experience interference¹ from i . The network is assumed to operate in time slots. In one slot a node can either broadcast one constant-length packet or stay idle.

We will need the notion of a hypergraph, a generalization of a graph, which is defined as follows.

Definition 1: A hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{A})$ is a set of nodes \mathcal{N} and a collection of hyperarcs \mathcal{A} . A hyperarc $(i, J) \in \mathcal{A}$ is a generalization of an edge, where $i \in \mathcal{N}$ and $J \subset \mathcal{N}$.

From the neighborhood relation we construct a hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{A})$ with \mathcal{N} corresponding to the set of nodes in the network. For each node i we introduce $2^{|N(i)|} - 1$ hyperarcs, (i, J) where J ranges over all subsets of $N(i)$ excluding the empty set.

If node i injects a packet on hyperarc J it is received by some subset $K \subseteq J$, possibly K being the empty set \emptyset . Let $A_{iJ}(\tau)$ be the counting process describing packet injections on hyperarc J and $A_{iJK}(\tau)$ be the counting processes accounting for the packets received *precisely* by the subsets K . Obviously, we have $\sum_{K \subseteq J} A_{iJK}(\tau) = A_{iJ}(\tau)$. We assume that for the injection processes time averages exist, i.e. $\lim_{\tau \rightarrow \infty} \frac{A_{iJ}(\tau)}{\tau}$ exists with probability 1, is finite, and equals z_{iJ} . Similarly, we define $\lim_{\tau \rightarrow \infty} \frac{A_{iJK}(\tau)}{\tau} = z_{iJK}$. With these assumptions $z_{iJ} = \sum_{K \subseteq J} z_{iJK}$ is the average packet injection rate on hyperarc J . In the remainder of the paper we will assume that z_{iJK} is proportional to z_{iJ} and define

$$p_{iJK} = \frac{z_{iJK}}{z_{iJ}}, \quad (1)$$

to be the probability that a packet injected on J is received precisely by the subset K . We call the rate vector $(z_{iJ})_{(i,J) \in \mathcal{A}}$ the *network coding subgraph*.

Once the subgraph is computed our network coding technique of choice will be the well known random network coding [1], [2]. Roughly, that means that a node stores received packet in its memory and upon an opportunity transmits a linear combination of the stored packets with coefficients drawn

¹A popular and slightly more general model is to assume that a node can receive from i if it is contained in a set $N_1(i)$ but is subject to interference if it belongs to a superset $N_2(i) \supset N_1(i)$. Our framework can readily be extended to take into account such a setup, however, for the sake of a simpler notation we abide with the above model.

uniformly from a finite field. Coding is restricted to packets belonging to the same user (intra-session).

Which sets of hyperarcs can transmit simultaneously without a conflict depends on the system model of the network. In networks with primary interference, e.g. spread-spectrum systems, we restrict each node to at most communicate with one other node in any time slot. In networks with secondary interference we have the additional constraint that a node can only successfully receive if all other neighbors are silent.

We will call a set of conflict-free hyperarcs a transmission set or valid configuration, formally

Definition 2: We say that hyperarcs (i_1, J_1) and (i_2, J_2) , do *not* conflict if:

- 1) $i_1 \neq i_2$,
- 2) $i_1 \notin J_2, i_2 \notin J_1$, and

for networks with primary interference

- 3a) $J_1 \cap J_2 = \emptyset$, or

alternatively for networks with secondary interference

- 3b) $J_1 \cap N(i_2) = \emptyset$, and $J_2 \cap N(i_1) = \emptyset$.

For both, the primary and the secondary interference model, the definitions are symmetric in their arguments and therefore give rise to an undirected graph representing the scheduling conflicts between pairs of hyperarcs. We construct the conflict graph as follows.

Definition 3: The conflict graph \mathcal{G} of a hypergraph \mathcal{H} is an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with \mathcal{V} corresponding to the set of all hyperarcs. Two hyperarcs are adjacent if they conflict.

We can define a valid configuration of hyperarcs as a set of nodes in the conflict graph without any conflicting pair, i.e. a valid configuration is a stable set.

Definition 4: A stable set S of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a set of nodes any two of which are nonadjacent. Its incidence vector is a column vector of length $|\mathcal{V}|$, defined as

$$\chi_v^S = \begin{cases} 1 & \text{if } v \in S, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

A *maximal* stable set is one that is not contained in any other stable set. A *maximum* stable set is a stable set of largest cardinality. The stable set polytope $P_{STAB}(\mathcal{G})$ is the convex hull of the incidence vectors of all stable sets of \mathcal{G} .

Example To illustrate the notation, consider the hypergraph in Fig. 1 and its corresponding conflict graph (in this particular network both the primary and the secondary interference model give rise to the same conflict graph) in Fig. 2. The conflict graph has a node for each hyperarc, i.e. its set of vertices is $\mathcal{V} = \{(1, 2), (1, 3), (1, \{2, 3\}), (2, 4), (3, 4)\}$ and as it can be easily verified the stable set polytope is given by the convex hull of the incidence vectors of the following three stable sets $\{(1, 2), (3, 4)\}$, $\{(1, 3), (2, 4)\}$, and $(1, \{2, 3\})$.

III. THE MULTICAST RATE REGION WITH SCHEDULING CONSTRAINTS

Consider a hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{A})$ and a multicast connection of rate R with source $s \in \mathcal{N}$ and set of sinks $T \subset \mathcal{N}$. We can apply the flow formulation from [2] to compute the network coding subgraph. The rate region (in

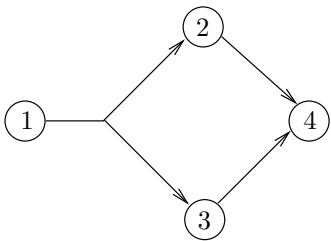


Fig. 1. An example of a hypergraph. Here, the node set is $\mathcal{N} = \{1, 2, 3, 4\}$ and the hyperarc set is $\mathcal{A} = \{(1, 2), (1, 3), (1, \{2, 3\}), (2, 4), (3, 4)\}$.

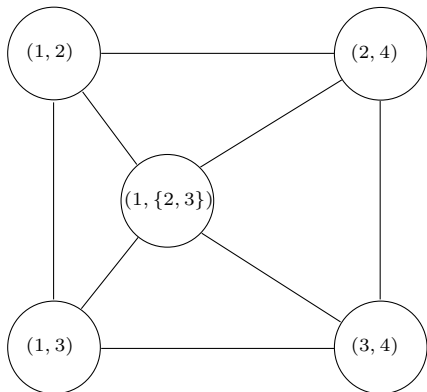


Fig. 2. The conflict graph corresponding to the hypergraph in Fig. 1.

terms of supportable end-to-end throughputs as opposed to the information theoretic rate region) for a multicast connection² is then the set of rates R that can be achieved subject to the following constraints

$$\sum_{j \in K} x_{iJj}^{(t)} \leq z_{iJ} b_{iJK}, \quad \forall (i, J) \in \mathcal{A}, K \subset J, t \in T, \quad (3)$$

$$\sum_{\{J|(i,J) \in \mathcal{A}\}} \sum_{j \in J} x_{iJj}^{(t)} - \sum_{\{j|(j,I) \in \mathcal{A}, i \in I\}} x_{jIi}^{(t)} = \begin{cases} R & i = s, \\ -R & i = t, \\ 0 & \text{else,} \end{cases} \quad (4)$$

$$\forall i \in \mathcal{N}, t \in T,$$

$$x_{iJj}^{(t)} \geq 0, \quad \forall (i, J) \in \mathcal{A}, j \in J, t \in T, \quad (5)$$

$$z = (z_{iJ}) \in P_{STAB}(\mathcal{G}), \quad (6)$$

where we define

$$b_{iJK} = \sum_{\{L \subset J | L \cap K \neq \emptyset\}} p_{iJL}. \quad (7)$$

We assume p_{iJL} , the probability that a packet injected on hyperarc (i, J) is received by precisely the set of nodes $L \subset J$, to be a constant.

The last constraint (6) explicitly accounts for interference by requiring the network coding subgraph z to lie in the

²The extension to multiple multicast connections with intra-session coding is straightforward and, again, only omitted to simplify exposition and notation.

stable set polytope of the conflict graph. Any vector in the stable set polytope can be written as a convex combination of schedules and therefore the demanded rate can be transmitted if we choose the granularity of time slots sufficiently fine. This is analogous to the Birkhoff-von Neumann decomposition of load matrices for scheduling in switches [5].

We can rewrite the linear program to yield a formulation with substantially fewer variables. A similar technique is used in [2], however, the following transformation and its proof are more general. Let for all $i \in \mathcal{N}$ and $j \in N(i)$

$$x_{ij}^{(t)} = \sum_{J \subset N(i)} x_{iJj}^{(t)}, \quad (8)$$

with the understanding that $x_{iJj}^{(t)} = 0$ if $j \notin J$, and consider the following formulation in terms of the new variables $x_{ij}^{(t)}$

$$\sum_{j \in K} x_{ij}^{(t)} \leq \sum_{J \subset N(i)} z_{iJ} b_{iJK}, \quad (9)$$

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

$$\sum_{j \in N(i)} x_{ij}^{(t)} - \sum_{\{j|i \in N(j)\}} x_{ji}^{(t)} = \begin{cases} R & i = s, \\ -R & i = t, \\ 0 & \text{else,} \end{cases} \quad (10)$$

$$\forall i \in \mathcal{N}, t \in T,$$

$$x_{ij}^{(t)} \geq 0, \quad \forall i \in \mathcal{N}, j \in N(i), t \in T, \quad (11)$$

$$z \in P_{STAB}(\mathcal{G}), \quad (12)$$

where consistently with definition (7), b_{iJK} is well defined even if K is not a subset of J .

Lemma 1: The rate region described by (3)-(6) is equivalent to the rate region given by the reduced formulation (9)-(12).

Proof: The new flow conservation constraint (10) is just a reformulation in terms of the new flow variables. What we have to show is the equivalence of the constraints (9) and (3). For a fixed node i , we have a number of hyperarcs, and for each hyperarc (i, J) constraint (3) gives us a rate region for the flows $x_{iJj}^{(t)}$. The claim is that the rate region for the sum of these flows as defined in (8) is given by the sum of the facets defining their rate regions. One direction is easy to show since, clearly, we have

$$\sum_{j \in K} x_{ij}^{(t)} = \sum_{j \in K} \sum_{J \subset N(i)} x_{iJj}^{(t)} \leq \sum_{J \subset N(i)} z_{iJ} b_{iJK}. \quad (13)$$

The other direction, that these bounds are also achievable is not trivial. In general, for convex polytopes defined by linear inequalities, the polytope generated by their Minkowski sum is not equal to the one defined by the sum of their individual constraints. In this case, however, we can exploit the special structure of the polytopes in (3). They are *polymatroids*, due to the fact that b_{iJK} , when viewed as a function of K , is a submodular function. Indeed, it is straightforward to verify from definition (7) that

$$b_{iJK} + b_{iJL} = b_{iJ(K \cap L)} + b_{iJ(K \cup L)}. \quad (14)$$

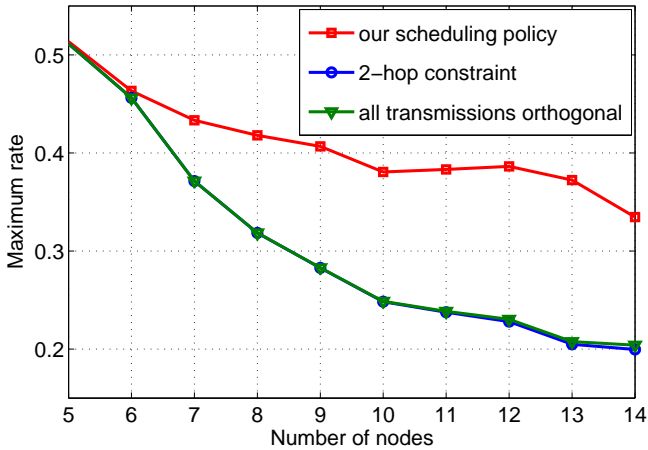


Fig. 3. Maximum rate of a multicast connection with one sender and two receivers as a function of the number of network nodes.

For two polymatroids, given by submodular set functions f_1 and f_2 respectively, i.e.

$$P_{f_i} := \left\{ x \in \mathbb{R}_+^{|J|} : \sum_{K \subset J} x_j \leq f_i(K) \quad \forall K \subset J \right\} \quad (15)$$

it is a well-know fact (see e.g. [11, Thm. 44.6, p. 781]) that the convex hull of their Minkowski sum is equivalent to the sum of the inequalities defining them, i.e.

$$P_{f_1+f_2} = P_{f_1} + P_{f_2}. \quad (16)$$

Since we consider a finite sum of polymatroids, our result follows readily. ■

IV. SIMULATION RESULTS

To get an idea of the performance gains of our approach we conduct simulations over random network topologies. For each random instance, we assume that a number of nodes are uniformly scattered over a square region with unit node density. Two nodes are in radio range if their distance is below some threshold, the radius of connectivity, which we take to be 1.8. The number of neighbors of a node is restricted to 5. We consider the leftmost node to be the sender, multicasting to two receivers, the two rightmost nodes. Transmissions are subject to erasures which may be due to distance attenuation and Rayleigh fading. When a node transmits a neighbor at distance d will receive the packet correctly if $\gamma d^{-2} \geq \beta$ where γ is a unit mean exponential variable and $\beta = \frac{1}{4}$ is our chosen SNR threshold, otherwise the packet is lost completely. We assume secondary interference constraints.

In Fig. 3, we compare the performance of our approach with two commonly used scheduling techniques. In the fully orthogonal model [2] all nodes in the network are assigned orthogonal channels, making the network interference-free. In the two-hop constraints model (see e.g. [3] for such a scheduling protocol) transmissions are scheduled such that if

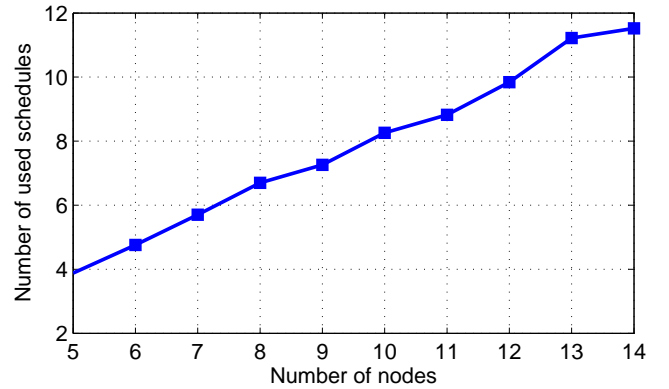


Fig. 4. The number of schedules which appear in the optimal solution of our scheduling algorithm. We consider schedules with a time sharing coefficient equal to or greater than 0.001.

node i transmits all nodes in a two-hop neighborhood are silent, eliminating the possibility of a node being in radio range of two simultaneous transmissions. Both, the orthogonal and the two-hop constraint model eliminate interference at the expense of suboptimal bandwidth reuse. We see that this is apparently a rather wasteful way of operating a interference limited wireless network. Furthermore, for the networks of moderate size that we consider the two-hop constraint is almost as restrictive as full orthogonalization. On the other hand, a significant increase in bandwidth efficiency is possible as we can see by the performance of our scheduling algorithm.

Fig. 4 shows the average number of network configurations comprising the optimal solution of our algorithm. Apparently, typically a rather small number of schedules is involved. This observation might be significant in the design of scheduling algorithms and particularly in seeking low-complexity solutions with a small optimality gap.

V. ON THE COMPLEXITY OF THE SCHEDULING PROBLEM

The stable set polytope membership of the network coding subgraph (12) can be a difficult constraint. In general, the question whether a point belongs to the stable set polytope cannot be answered in polynomial time, except for certain special classes of graphs. By the equivalence between optimization and separation³, these are the graphs, for which a maximum stable set can be computed in polynomial time. We conclude:

Theorem 1: The joint subgraph optimization and scheduling problem, i.e. maximizing a linear function over (9)-(12), can be solved in polynomial time if a maximum stable set in \mathcal{G} can be found in polynomial time.

Proof: If a stable set in \mathcal{G} can be found in polynomial time, then we can also decide if a vector belongs to $P_{STAB}(\mathcal{G})$ in polynomial time. Since we assumed that the number of constraints grows polynomially in the size of the network (by

³The optimization problem is to maximize $c^T x$ over a polytope P , the corresponding separation problem is to decide whether $x \in P$ and if not to display a violated constraint. As a consequence of the ellipsoid method the polynomial time solvability of one of the problems implies the polynomial time solvability of the other [10].

limiting the number of neighbors) we can also verify if a vector belongs to (9)-(11) in polynomial time. Since we are able to solve the separation problem in polynomial time we can also optimize over the polytope (9)-(12) in polynomial time. ■

There are a many classes of graphs with polynomial stable set algorithms, see e.g. [11] for an extensive survey. A graph family with particularly good algorithmic properties are the so called *perfect graphs*. They have a polynomial time maximum stable set algorithm, and furthermore their stable set polytope can be described by clique inequalities (for general graphs these are necessary but not sufficient).

Definition 5: An graph is perfect if and only if for every induced subgraph the clique number equals the chromatic number.

Proposition 1: The following characterizations of perfect graphs are equivalent [11]:

- The complement of a perfect graph is perfect.
- A graph is perfect if and only if it contains no odd holes (induced subgraphs that are cycles of odd length) and antiholes (their complements).
- $P_{STAB}(\mathcal{G}) = P_{QSTAB}(\mathcal{G})$,

where

$$P_{QSTAB}(\mathcal{G}) = \left\{ x \in \mathbb{R}_+^{|\mathcal{V}|} : \sum_{q \in Q} x_q \leq 1 \quad \forall \text{ cliques } Q \subset \mathcal{V} \right\} \quad (17)$$

is called the *fractional stable set polytope*.

Even if the graph is not perfect, yet not too far from being perfect, as measured by the *imperfection ratio* [12], there exist good polynomial-time approximation algorithms for the stable set problem. In general, the hardness of approximating the problem depends on the imperfection ratio. Another general approach is to replace the stable set polytope constraint with a smaller convex polytope $\hat{P} \subset P_{STAB}(\mathcal{G})$ that has a simpler structure and to optimize over the latter. This is, in fact, exactly what is being done in the two scenarios that we compare our algorithm with. In the fully orthogonal scheme, we effectively optimize over $\hat{P}_O = \text{CH} \{ \chi^{(i, N(i))} \}, i \in \mathcal{N}$, whereas the two-hop constraint polytope is simply given by $|\mathcal{N}|$ linear constraints. For every node i , the transmission rates of nodes within a two-hop neighborhood of i must sum to at most one.

More generally, consider a collection of maximal stable sets $S_i, i \in I$ and the convex hull of their incidence vectors $\hat{P} = \text{CH} \{ \chi^{S_i} \}, i \in I$. Ideally, the volume of \hat{P} should be not much smaller than the volume of $P_{STAB}(\mathcal{G})$ and \hat{P} should be the convex hull of a relatively small number of vectors. A possible strategy might be to choose an initial collection of stable sets and then iterate between computing the optimal network coding subgraph with respect to this set and updating the collection of stable sets which would yield a sequence of increasing rate regions (see also [8] for a related approach). For updating the set of valid configurations we could use the sensitivity information, that comes as a by-product of optimizing the network coding subgraph.

VI. CONCLUSION AND FURTHER WORK

We have suggested an algorithm for jointly solving the network coding subgraph and the scheduling problem in

wireless network coded multicast. The key construction is a conflict graph representing all possible valid configurations of the network. As simulation results show our approach outperforms widely used scheduling techniques significantly. We believe that our scheduling framework is an important first step in understanding scheduling for network coded multicast traffic and can be useful in algorithm design and performance analysis of both optimal and heuristical scheduling algorithms.

Concerning further work, we believe that our achievability result is a first step in designing off-line and online scheduling algorithms. In switching theory the achievability results have been the basis for optimal off-line algorithms (e.g. BVN-decomposition) and suboptimal, and yet provably good online algorithms (e.g. iSLIP [13]). The maximum stable set is a widely encountered practical problem and while its worst case complexity is exponential [14], there exist good approximation algorithms. Finally, it is desirable to investigate decentralized solutions and their performance. The network coding subgraph can be computed in a distributed way [2], so the challenge lies in finding efficiently stable sets in a decentralized fashion (see e.g. [15] for a simple algorithm) and combining this with the subgraph optimization.

REFERENCES

- [1] T. Ho, R. Koetter, M. Médard, M. Effros, J. Shi, and D. Karger, "A random linear network coding approach to multicast," *IEEE Trans. on Information Theory*, 52 (10), pp. 4413-4430, October 2006.
- [2] D.S. Lun, N. Ratnakar, M. Médard, R. Koetter, D.R. Karger, T. Ho, and E. Ahmed, "Minimum-cost multicast over coded packet networks," *IEEE Trans. on Information Theory*, 52(6):2608-2623, June 2006.
- [3] L.Bao, and J.J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," *Proc. IEEE MILCOM 2000*, Los Angeles, California, October 22-25, 2000.
- [4] L. Tassiulas and A. F. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Automat. Contr.*, vol. 37, no. 12, pp. 1936 - 1948, Dec. 1992.
- [5] J. Sundararajan, M. Médard, M. Kim, A. Eryilmaz, D. Shah, R. Koetter, "Network coding in a multicast switch," in *IEEE INFOCOM 2007*, Anchorage, AK, May 2007.
- [6] J. Sundararajan, M. Médard, R. Koetter, E. Erez, "A systematic approach to network coding problems using conflict graphs," invited paper, ITA workshop, UCSD, February 2006.
- [7] C. Caramanis, M. Rosenblum, M.X. Goemans, V. Tarokh, "Scheduling algorithms for providing flexible, rate-based, quality of service guarantees for packet-switching in Banyan networks," *CISS 2004*, Princeton, NJ, 2004.
- [8] 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*, vol. 23, no. 1, pp. 136-150, Jan. 2005.
- [9] S. Sengupta, S. Rayanchu, S. Banerjee "An analysis of wireless network coding for unicast sessions: The case for coding-aware routing," in *IEEE Infocom 2007*, Anchorage, AK, April 2007.
- [10] M. Grötschel, L. Lovász, A. Schrijver, *Geometric algorithms and combinatorial optimization*, Springer, Berlin-Heidelberg, 1988.
- [11] A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer-Verlag, 2004.
- [12] M. Kim, J. Sundararajan, M. Médard, "Network coding for speedup in switches," *ISIT 2007*, Nice, France, June 2007.
- [13] N. McKeown, "iSLIP: A scheduling algorithm for input-queued switches," *IEEE Transactions on Networking*, Vol 7, No.2, April 1999.
- [14] C. H. Papadimitriou, and K. Steiglitz, "Combinatorial Optimization: Algorithms and Complexity," Prentice Hall, Englewood Cliffs, NJ, 1982.
- [15] K. Jung, and D. Shah, "Low delay scheduling in wireless networks," *IEEE International Symposium on Information Theory*, June 2007, Nice, France.