

Heuristics for Network Coding in Wireless Networks

Song Yean Cho
Hipercom Team
LIX, Ecole Polytechnique,
Palaiseau, France
Cho@lix.polytechnique.fr

Cédric Adjih
Hipercom Team, INRIA
Rocquencourt, France
Cedric.Adjih@inria.fr

Philippe Jacquet
Hipercom Team,
INRIA Rocq. and LIX
Palaiseau, France
Philippe.Jacquet@inria.fr

ABSTRACT

Multicast is a central challenge for emerging multi-hop wireless architectures such as wireless mesh networks, because of its substantial cost in terms of bandwidth.

In this article, we study one specific case of multicast: broadcasting, sending data from one source to all nodes, in a multi-hop wireless network. The broadcast we focus on is based on *network coding*, a promising avenue for reducing cost; previous work of ours showed that the performance of network coding with simple heuristics is asymptotically optimal: each transmission is beneficial to nearly every receiver. This is for homogeneous and large networks of the plan. But for small, sparse or for inhomogeneous networks, some additional heuristics are required. This article proposes such additional new heuristics (for selecting *rates*) for broadcasting with network coding. Our heuristics are intended to use only simple local topology information. We detail the logic of the heuristics, and with experimental results, we illustrate the behavior of the heuristics, and demonstrate their excellent performance.

Categories and Subject Descriptors

C.2.1 [Network Architecture and Design]: Wireless communication

Keywords

wireless, broadcast, network coding, hypergraph, min-cut

1. INTRODUCTION

The confirmed success of wireless networks has made wireless communication ubiquitous. One of the predicted uses of wireless networks is multicast, which could be used for multimedia content diffusion, video conference, software distribution, and a number of other applications. However, in multi-hop networks such as wireless mesh networks, multicast transmissions incur a substantial cost by simple virtue of requiring relaying over several forwarders, in order to

cover every destination. For this reason, efficient techniques for multicasting are of prime interest. One of them is the recently proposed method, network coding. Network coding was introduced by the seminal work of [1] as a new paradigm where intermediate nodes mix information from different flows (different bits or different packets). We use network coding specifically for broadcasting rather than general multicasting in wireless multi-hop networks.

The problem that we are addressing is *efficient broadcast*:

- Broadcast packets from one source to all nodes, with the minimum number of transmissions.

Without network coding, finding the optimal broadcasting is an NP complete problem [2], but a number of heuristics exist for efficient broadcasting such as MPR-flooding [3] or techniques based on connected dominating sets [4, 5]. But with network coding, the optimal broadcasting can be found in polynomial time. Finding an optimal solution (*rate selection*) consists in finding the coding nodes and their optimal rates [6–8]. This problem can be formulated as a linear program, which can be solved in polynomial time [9, 10], and possibly in a distributed fashion [9].

However, we adopt a different, even simpler, approach: previous work [11, 12] has shown that a simple heuristic for selecting rates could achieve asymptotically the optimal efficiency for homogeneous large and dense wireless networks of the plane — and also that, noticeably, it would outperform methods without network coding. This is asymptotically true for homogeneous networks, but the heuristic needs adjustments for less homogeneous, smaller or sparser networks. The adjustment is the topic of this article. Our key contributions are the following:

- We propose an improved heuristic for rate selection, inspired by [13], but with a two- to threefold improvement. It requires only local topology information: knowledge of two-hop neighbors.
- We empirically study its performance on representative graphs with different densities and different sizes. We investigate and explain the variation of the performance, and also compare it to other techniques (including without using network coding).

The rest of the paper is organized as follows: section 2 details the network model and definitions, section 3 describes the heuristics, section 4 analyzes performance with experimental results and section 5 concludes.

2. DEFINITIONS

2.1 Network Model

In this article, we study the problem of broadcasting from

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WICON 2007, October 22–24, Austin, USA

Copyright © 2007 978-963-9799-12-7

DOI 10.4108/wicon.2007.2097

one source to all nodes. We will focus on getting a preliminary idea of the performance of our heuristics for wireless networks. Hence, we will assume an ideal wireless model in this article (and realistic models would be subject of future work): lossless wireless transmissions without collisions or interferences. We also assume that each node of the network is operating below its maximum transmission capacity.

In an idealized model, multi-hop wireless networks can be modeled as *unit disk graphs* of the plane, where two nodes are neighbors if their distance is lower than a fixed radio range as seen in Fig. 1. In addition, in wireless networks, the *wireless broadcast advantage* is used: each transmission is overheard by several nodes. As a result the graph is in reality a (*unit disk*) *hypergraph*. Precisely, we consider the following networks: unit disk [hyper]graphs where nodes are either distributed randomly (Fig. 1.1) or more regularly organized in a lattice (Fig. 1.2). In addition, in both cases, we also consider their variants where the network is a torus, with wrap-around connections in both the x and y directions as on Fig. 1.3.

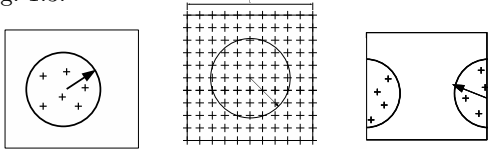


Fig. 1.1: random unit disk graph Fig. 1.2: lattice graph Fig. 1.3: on a torus

Figure 1: Network Models

2.2 Notation

We will consider the following items in each instance of the several types of graphs in section 2, and use the following notation in the rest of the article:

- **Nodes:** \mathcal{V} , set of vertices (nodes) of the graph
- **Number of nodes:** N
- **Expected/average number of neighbors:** M
- **Hyperarc:** $H_v, H_v \subset \mathcal{V}$ is the subset of nodes reached by one transmission of node v (the neighbors of v)
- **Rate:** Each node v retransmits coded packets with a fixed rate C_v
- **Cost per broadcast:** E_{cost} ; defined in section 2.3
- **Optimal cost per broadcast :** E_{optimal} (section 2.3)
- **Min-cut of the source s , for broadcast to the entire network:** $C_{\min}(s)$; see section 2.4

2.3 Performance

Because we focus on broadcasting, our approach uses a simplified rate selection, where every node is a coding node. The rate selection consists of defining the rate of each node – by the heuristic.

To evaluate the performance of the heuristics, we use the number of transmissions per broadcast as the metric for the cost. We consider:

- the number of retransmissions from every node per unit time (directly given by selected rates).
- the number of packets successfully broadcast from the source to the entire network per unit time; it is the achievable broadcast rate.

By dividing the number of retransmissions by the number of packets successfully broadcast, we obtain our metric for the cost per broadcast and denote it E_{cost} . For reference, we will also use the cost of the optimal solution, E_{optimal} , obtained by solving the linear program as presented in [9].

The number of packets successfully broadcast per unit time, i.e. the maximum achievable broadcast rate is computed as the *min-cut* from the source to every destination in the network considered as a hypergraph as defined in section 2.4. In practice, it can be achieved asymptotically by random linear coding for packet networks as detailed in [8].

2.4 Achievable Broadcast Rate: Min-cut

A central result of the performance of network coding in wireless networks gives the maximum broadcast (more generally: multicast) rate for a source. It is given by the *min-cut* from the source to each individual destination of the networks, viewed as a hypergraph [14].

Let us consider the source s , and one of the broadcast destinations $t \in \mathcal{V}$. The definition of a *s-t cut* is: a partition of the set of vertices V in two sets S, T such as $s \in S$ and $t \in T$. Let $Q(s, t)$ be the set of such *s-t cuts*: $(S, T) \in Q(s, t)$.

We denote ΔS , the set of nodes of S which are neighbors of at least one node of T ; the *capacity of the cut* $C(S)$ is defined as the maximum rate between the nodes in S and the nodes in T :

$$C(S) \triangleq \sum_{v \in \Delta S} C_v \quad (1)$$

The *min-cut* between s and t is the cut of $Q(s, t)$ with the minimum capacity. Let us denote $C_{\min}(s, t)$ its capacity. From [14, 15], the maximum broadcast capacity is given by the minimum of capacity of the min-cut of every destination $C_{\min}(s)$, with:

$$C_{\min}(s, t) \triangleq \min_{(S, T) \in Q(s, t)} C(S) \quad \text{and} \quad C_{\min}(s) \triangleq \min_{t \in \mathcal{V} \setminus \{s\}} C_{\min}(s, t)$$

Once computed, the performance given by the global min-cut $C_{\min}(s)$ can be achieved by different coding methods (precisely or asymptotically).

3. HEURISTICS FOR RATE SELECTION

3.1 Previous Heuristic

In [11, 12] we proposed a simple heuristic, where most nodes have the same rate except the source and some nodes near the edge of the network (*exceptional nodes*): IREN/IRON (*Increased Rate for Exceptional Nodes, Identical Rate for Other Nodes*).

That simple heuristic achieved asymptotically near-optimal efficiency for unit disk graphs in large dense homogeneous networks. In other words, the simple heuristic enables for every transmission to bring *innovative* information to almost every receiver.

Let us reproduce the logic of the heuristic:

1. Assume that the every node has an identical retransmission rate. Assume it is 1, arbitrarily, e.g. one packet per second.
2. Then every node with M neighbors can receive M coded packets per second. Assume that nearly all of them innovative (non-redundant).
3. Then the source should inject at least M packets per a second.
4. An issue is the nodes of near the border, because they have less neighbors - so in order to be safe, their rate is set to M as well (IREN).

Within this framework, it was proven that the achievable broadcast capacity, the min-cut, would be M for lattice graphs, and asymptotically M for dense random unit-disk graphs. For large lattice or random unit-disk graphs,

the cost per broadcast would also converge to the optimal. However the result is asymptotic; for a practical instance of a graph, the cost of the “increased rate” of the border nodes could be considerable, and unnecessary high.

An immediate way to alleviate the cost is to omit step 4 of the reasoning entirely, i.e., ignoring the issue of border, thus using the following heuristics:

- IRON only (Identical Rate for Other Nodes): every node retransmits with rate 1, except for the source, which transmits with rate M .

For illustrative purposes, we show the maximum achievable broadcast rate, for two instances of the networks with the rates selected by “IRON only” on Fig. 2.

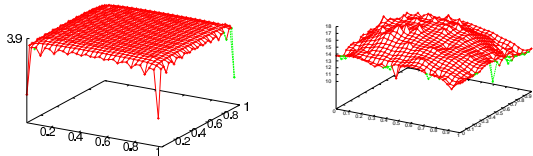


Fig. 2.1: min-cut with IRON only, on a lattice Fig. 2.2: IRON only, on a random unit disk

Figure 2: Min-cut with IRON only

The Fig. 2 represents the min-cut $C_{\min}(s, t)$ of each node t with respect to its position when the total number of nodes is $N = 400$. Both x and y axis represent the position of each node on the plane; the value on the vertical z axis represents min-cut¹. On the lattice of Fig. 2.1, every node has $M = 4$ neighbors, and the source is in the middle of the network. In the random unit disk graph in Fig. 2.2 we select the node with the most number of neighbors as the source, and generate graphs with density (the average number of nodes in a range) equal to $M = 20$.

The min-cut was computed with the software library implementing the max-flow computation algorithm from [16] (and one additional layer to model a directed hypergraph as a directed graph). The optimizations for reusing search trees from [17] were also used.

As seen in Fig. 2, the nodes near the border have a min-cut much lower than nodes in the middle of the network (4 compared to 2 for Fig. 2.1). From section 2.4, recall that maximum achievable broadcast rate from the source, $C_{\min}(s)$, is the minimum of the min-cuts $C_{\min}(s, t)$ to each destination t .

Hence, because of these nodes near the border, the source rate has to be much lower than it could otherwise be. Also in random unit-disk graph, Fig. 2.2, not the same phenomenon occurs on the borders, but also now, the irregularity of number of neighbors results in irregularity of the min-cut (and hence potential lower minimum min-cut).

These are reasons of inefficiency of IRON.

3.2 The Proposed Heuristic: IR-MS

As illustrated with the simple heuristics in section 3.1, some nodes receive less than M packets per unit time and these nodes cause a decrease of the global broadcast performance. We name these nodes *starving nodes*.

To alleviate the bottleneck from the starving nodes, their neighbors compensate their starving by increasing the transmission rates with the following heuristic, inspired by [13]:

- IR-MS (Increased Rate for the Most Starving node): the rate of a node v is set to C_v , with:

¹for the randomly generated unit disk graph, there is interpolation

$$C_v = \frac{M}{\min_{u \in H_v} (|H_u|)}, \text{ where } H_w \text{ is the set of neighbors of } w, \text{ and } M \text{ is the source rate.}$$

This rate selection adjusts the step 2 of the reasoning presented in section 3.1, to the fact that some nodes have less than M neighbors. With IR-MS, if a node v has less than $|H_v| < M$ neighbors, it would still receive a packet rate $\geq \frac{M}{|H_v|}$ from each of its $|H_v|$ neighbors, and hence an overall rate at least equal to M . Notice that this does not necessarily result in a min-cut $C_{\min}(s, v) \geq M$.

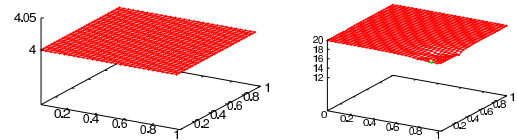


Fig. 3.1: min-cut with IR-MS, on a lattice Fig. 3.2: IR-MS, on a random unit disk

Figure 3: Min-cut with IR-MS

The Fig. 3 represents the value of the min-cut with the rate selection of IR-MS, with the same topologies and same parameters as for IRON only, in previous Fig. 2. It appears that for these topologies, the min-cut has a value which is much closer to the targeted value M . The increased rates of IR-MS at neighbors of bottleneck may bring additional costs. However this additional cost is relatively much lower than the increase of min-cut, thus the performance is overall improved. We will present more systematic experiments on the performance of the heuristic IR-MS in the next section.

Notice that [13] proposed the same expression of C_v , except that it was multiplied by a coefficient k . It was in a slightly different context, but their theoretical argument implied that $k \geq 3$ is necessary, whereas our theoretical argument of section 3.1 and of [11] implies $k = 1$: ensuring that every node can decode at the end, can be done by a termination phase, of bounded length, negligible for large generations. Hence a threefold gain. In the next sections, the actual min-cut is sometimes less than the target min-cut = M , hence, roughly the gain may be only around 2.

4. EXPERIMENTAL RESULTS

4.1 Efficiency on Different Types of Networks

We first evaluate the efficiency of the heuristic IR-MS, with the relative cost w.r.t. the optimal: $E_{\text{rel-eff}} \triangleq \frac{E_{\text{optimal}}}{E_{\text{cost}}}$

One reference point is the approximative upper bound in [13], the achievable performance without network coding, which translates into: $E_{\text{bound-rel-eff}}^{(\text{no-coding})} \approx 0.609 \dots$

For comparison purposes, we evaluated the IR-MS heuristic on instances of lattice unit disk graphs and random unit disk graphs, both with and without torus effect — four variants in total. Our parameters are the following: number of nodes $N = 196$, the avg. number of neighbors is successively 4, 12, 28, 48, 80.

Fig. 5 represents the efficiency obtained for different cases (average of 10 results). The central result is the appreciable performance of IR-MS compared to the achievable performance without network coding: it mostly outperforms the upper bound without network coding (0.609).

Now consider the four different types of networks:

- First type, the most regular graphs: lattice unit disk

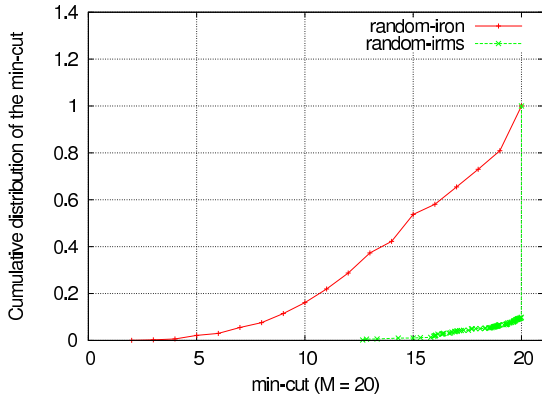


Fig. 4.1: mincut: cumulative distribution

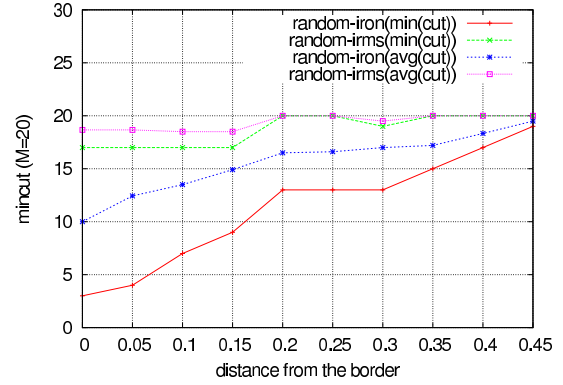


Fig. 4.2: mincut vs distance from the border

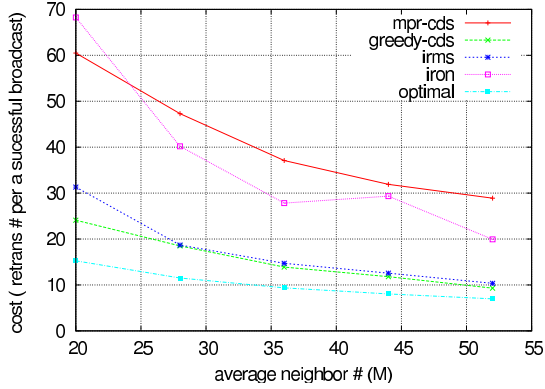


Fig. 4.3: cost vs avg neighbor num on square $N=200$

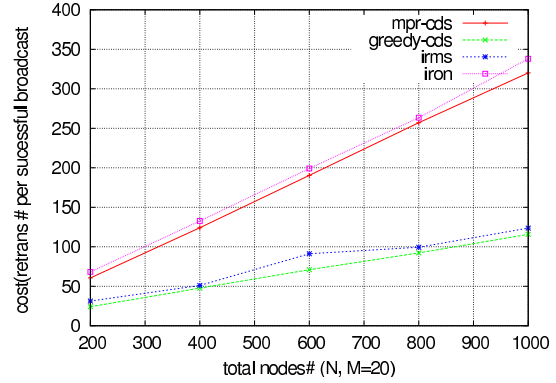


Fig. 4.4: cost vs total node num on square $M=20$

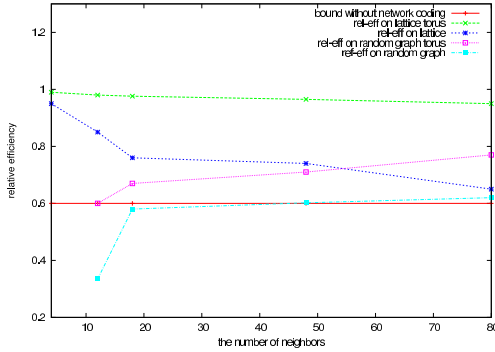


Figure 5: Relative cost (efficiency compared to optimal)

graphs on torus. IR-MS closely approaches optimality (> 0.95). This is because every node has the same number of neighbors and the rate adjustment is not needed.

- Second type, lattice unit disk graphs without torus: the nodes near the border have less neighbors than others are the issue. IR-MS successfully solves the border issue, shown in next section, approaching the targeted maximum broadcast rate, $\frac{\text{min-cut}}{M} = 1$. However efficiency decreases as density increases, because of an increase of the cost of nodes near the border.

- Third type, random unit disk graph on torus. No border effect here, but IR-MS has to overcome the effects of non-homogeneity, which is done convincingly.

- Fourth type, genuine random unit disk graph. The performance is acceptable, but on low density, performance becomes lower, because IR-MS does not fully achieve the max-

imum broadcast rate, M .

4.2 Distribution of the Min-cut

In this section, further results for case of the random unit disk graphs are provided: these graphs were the ones with the most problematic performance.

Again, the minimum of the min-cuts $C_{\min}(s, t)$ (for all $t \in \mathcal{V}$) decides the overall maximum broadcast rate of the source. Hence good performance is achieved when the distribution of these min-cuts is tighter. Deeper insight is gained by analyzing the cumulative distribution of the min-cut of each node, for the random graph $N = 400$ previously studied: it is displayed on Fig. 4.1. As evidenced, without IR-MS, the distribution of the min-cut is wider, but with IR-MS, the distribution is closer to M (with a peak for $M = 20$, the targeted min-cut). Still, there is some room for improvement, because a few nodes have a min-cut around 15.

The Fig. 4.2 provides additional information about the value of the min-cut depending on the position of the nodes, for IR-MS and IRON respectively. The distance of each nodes to the border of the network is computed, and statistics are made for the min-cut of nodes with same distance to the border. Fig. 4.2 gives the average min-cut and the minimum min-cut for nodes at a given distance from the border.

It evidences that the border effect is key, with a lower min-cut when the node is near the border: again, we see that IR-MS improves the minimum min-cut but does not always achieve the target min-cut= M .

4.3 Random Unit Disk Graphs N, M

The Fig. 4.3 and Fig. 4.4 show different perspectives on the performance. The performance measured on these graphs, is E_{cost} , the number of retransmissions per broadcast packet.

First, different algorithms are compared:

- Network coding with IRON.
- Network coding with IR-MS.
- MPR-based dominating sets from [5] (with performance close to MultiPoint-Relays (MPR)-based flooding of [3]). It is representative of the performance of algorithms, with only local topology information, without network coding.
- Connected dominating set from [4] (efficient variant of a greedy algorithm). It is representative of the performance of centralized connected dominating sets algorithms (and to some extent, representative of what could be achieved without network coding).

• for Fig. 4.3, the optimal solution with network coding.

Fig. 4.3 displays the performance of each algorithm when the density of the network increases.

One result is that the performance of the optimal network coding is about $\frac{1}{3}$ better than the performance of the connected dominating set from [4], without network coding.

Then it appears that IR-MS is also close to this efficient dominating set, which is already an interesting result. The gap between IR-MS and the optimal with network coding however indicates that some improvements are possible.

Last, MPR-based dominating sets, which use only local topology information are have the lowest performance: this indicates that network coding, with IR-MS, would be useful in practice, if a fully distributed solution could be designed (with distributed min-cut computation [18]).

The Fig. 4.4 shows the results for increasing density ; they are essentially similar.

4.4 Difficulties for Distributed Rate Selection

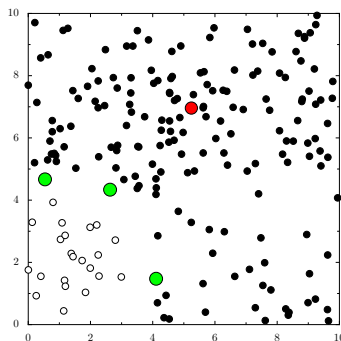


Figure 6: Example of cut

Finally, the Fig. 6 gives an example of an instance of a random unit disk graph, where the performance of IR-MS was found to be low. The cut (S/T) corresponding to the min-cut for a node at the bottom-left corner is represented: the green dots are the only nodes which are connecting the part in the corner to the rest of the network. With IR-MS, the green nodes do not have extremely starving nodes as neighbors, and hence increase their rate by little. But data from the source is only transmitted through these green nodes to the sets of white dots, hence for this reason their rate should be greater. This example perfectly illustrates the difficulties found in sparse networks: notice how coordination between the green nodes would require multi-hop communication to detect the issue.

5. CONCLUSION

We proposed and experimentally studied heuristics for efficient broadcasting with network coding only using static local information: one hop or two hop neighbors. We showed excellent performance of this rate selection, and detailed reasons for variations of performance. Future work includes the use of dynamic information for the heuristics in complement of the static local topology information.

6. REFERENCES

- [1] R. Ahlswede, N. Cai, S.-Y. R. Li and R. W. Yeung, "Network Information Flow", IEEE Trans. Inf. Theory, vol. 46, no.4, pp. 1204-1216, Jul. 2000
- [2] M. Cagalij, J.-P. Hubaux, C. Enz, "Minimum-energy broadcast in all wireless networks: NP-completeness and distribution issues", ACM/IEEE Mobicom 2002
- [3] T. Clausen and P. Jacquet (eds.), C. Adjih, A. Laouiti, P. Minet, P. Mühlethaler, A. Qayyum, L. Viennot "Optimized Link State Routing Protocol (OLSR)", IETF RFC 3626, October 2003.
- [4] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets", European Symposium on Algorithms, 1996.
- [5] C. Adjih, P. Jacquet, and L. Viennot, "Computing Connected Dominated Sets with Multipoint Relays", Ad Hoc & Sensor Wireless Networks, Jan. 2005
- [6] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "Further Results on Coding for Reliable Communication over Packet Networks" International Symposium on Information Theory (ISIT 2005), Sept. 2005
- [7] T. Ho, R. Koetter, M. Médard, D. Karger and M. Effros, "The Benefits of Coding over Routing in a Randomized Setting", International Symposium on Information Theory (ISIT 2003), Jun. 2003
- [8] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks", Tech. Report #2741, MIT LIDS, Jan. 2007
- [9] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-Cost Multicast over Coded Packet Networks", IEEE/ACM Trans. Netw., Jun. 2006
- [10] Y. Wu, P. A. Chou, and S.-Y. Kung, "Minimum-energy multicast in mobile ad hoc networks using network coding", IEEE Trans. Commun., vol. 53, no. 11, pp. 1906-1918, Nov. 2005
- [11] C. Adjih, S. Y. Cho and P. Jacquet, "Near Optimal Broadcast with Network Coding in Large Sensor Networks", First Workshop on Information Theory for Sensor Networks, Sante Fe, Jun. 2007
- [12] C. Adjih, S. Y. Cho and P. Jacquet, "Near Optimal Broadcast with Network Coding in Large Homogeneous Networks", INRIA Research Report RR-6188, May 2007
- [13] C. Fragouli, J. Widmer, and J.-Y. L. Boudec, "A Network Coding Approach to Energy Efficient Broadcasting", INFOCOM 2006, Apr. 2006
- [14] S. Deb, M. Effros, T. Ho, D. Karger, R. Koetter, D. Lun, M. Médard, N. Ratnakar, "Network Coding for Wireless Applications: A Brief Tutorial", IWWAN, May 2005.
- [15] A. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of Wireless Erasure Networks", IEEE Trans. Inf. Theory, vol. 52, no.3, pp. 789-804, Mar. 2006
- [16] Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision", PAMI, Sept. 2004
- [17] Pushmeet Kohli and Philip H.S. Torr, "Efficiently Solving Dynamic Markov Random Fields Using Graph Cuts.", ICCV 2005, Oct. 2005
- [18] Z. Li and B. Li, "Efficient and Distributed Computation of Maximum Multicast Rates", INFOCOM 2005, March. 2005