2004 International Workshop on Wireless Ad-Hoc Networks

On the utility of network coding in dynamic environments

Tracey Ho†, Ben Leong†, Muriel Médard†, Ralf Koetter‡, Yu-Han Chang†, and Michelle Effros*

†*Massachusetts Institute of Technology*
‡*University of Illinois, Urbana-Champaign*
**California Institute of Technology*

***Abstract*— Many wireless applications, such as ad-hoc networks and sensor networks, require decentralized operation in dynamically varying environments. We consider a distributed randomized network coding approach that enables efficient decentralized operation of multi-source multicast networks. We show that this approach provides substantial benefits over traditional routing methods in dynamically varying environments.**

We present a set of empirical trials measuring the performance of network coding versus an approximate online Steiner tree routing approach when connections vary dynamically. The results show that network coding achieves superior performance in a significant fraction of our randomly generated network examples. Such dynamic settings represent a substantially broader class of networking problems than previously recognized for which network coding shows promise of significant practical benefits compared to routing.

***Index Terms*— Multicast, ad-hoc networks, network coding, Steiner tree**

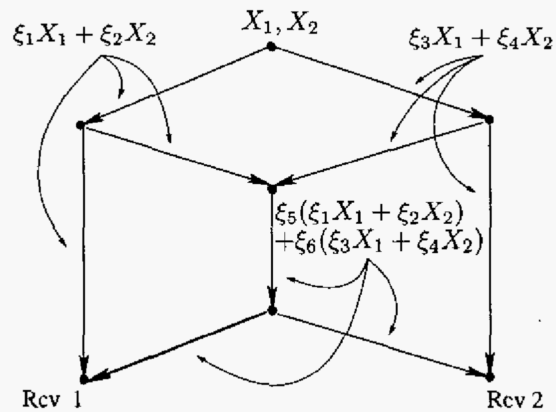

Fig. 1. An example of distributed randomized network coding. X_1 and X_2 are the source processes being multicast to the receivers, and the coefficients ξ_i are randomly chosen elements of a finite field. The label on each link represents the signal being carried on the link.

I. INTRODUCTION

In this paper, we consider the utility of network coding compared to routing for multi-input multicast in distributed, dynamically changing environments. This set-up encompasses a rich family of problems, such as the delivery of multicast content and the reachback problem for sensor networks, in which several sources transmit to a single receiver.

Network coding, as a superset of routing, has been shown to offer significant capacity gains for specially constructed networks [1], [15]. Apart from such examples, however, the benefits of centralized network coding over centralized optimal routing have not been as clear.

On the other hand, distributed or dynamic settings, such as in mobile ad-hoc or sensor networks, make optimal centralized control more costly or inconvenient. Such environments pose more challenges for routing-only approaches. For instance, in networks with large numbers of nodes and/or changing topologies, it may be expensive or infeasible to reliably maintain routing state at network nodes. For networks with dynamically varying multicast connections, it may be desirable to avoid recomputing distribution trees for existing connections to accommodate new connections.

Reference [6] proposed a distributed randomized network coding approach, and showed analytically that, for a completely decentralized single transmitter multicast system over a regular grid, distributed randomized coding outperformed distributed randomized flooding without coding. This paper significantly widens the scope of scenarios in which network coding presents benefits by extending consideration to randomly generated geometric graphs, to multiple transmitters and, most importantly, by comparing to quasi-optimal online routing, where, for each transmitter sequentially, a multicast tree is selected in a centralized fashion.

In this randomized network coding approach, all nodes other than the receiver nodes independently choose random linear mappings from inputs onto outputs over some field. An illustration is given in Figure I. Note that such an approach is intrinsically very different from traditional routing approaches. Data originating at different sources can be mixed through linear algebraic operations. Moreover, no coordination among nodes in their selection of input to output mappings is required. The receivers need only know the overall linear combination of source processes in each of their incoming signals. This information can be sent with each signal or packet as a vector of coefficients corresponding to each of the source processes, and updated at each coding node by applying the same linear mappings to the coefficient vectors as to the information signals. The required overhead of transmitting these coefficients decreases with increasing length of blocks over which the codes and network state are expected to remain constant.

This distributed coding approach achieves optimal network capacity asymptotically in the length of the code [6]. The dis-

[1]This research is supported in part by NSF Grants CCR-0325324, CCR-0220039 and CCR-0325496, University of Illinois subaward #03-25673, Hewlett-Packard 008542-008, and Caltech's Lee Center for Advanced Networking.

tributed nature of this approach also ties in well with considerations of robustness to changing network conditions. Moreover, issues of stability, such as those arising from propagation of routing information, are obviated by the fact that each node selects its code independently from the others.

While theoretical performance bounds have been derived for this randomized coding approach [6], [7], exact theoretical analysis of optimal online multicast routing is difficult. Multicast routing is closely related to the NP-complete Steiner-tree problem, for which various heuristic and approximate algorithms have been considered. We compare, with simulations on randomly generated graphs, the relative performance of distributed randomized coding and a Steiner heuristic algorithm presented by Kodialam [10], in which, for each transmitter, a tree is selected in a centralized fashion. Kodialam states that this heuristic's performance is comparable to or better than many alternative algorithms for centralized tree selection. These trees represent multicast routes in our setting. The networks we consider in this paper are random geometric graphs with degree constraints. We seek to model the kinds of topologies encountered in wireless ad-hoc networks with a limited number of channels, and nodes which may turn on and off intermittently. We do not assume omnidirectional transmissions in this paper, though the randomized coding approach could be adapted for this scenario.

This paper does not consider aspects such as resource and energy allocation, but focuses on optimally exploiting a given set of resources. There are also many issues surrounding the adaptation of protocols, which generally assume routing, to random coding approaches. We do not address these here, but rather seek to establish that the potential benefits of randomized network coding justify future consideration of protocol compatibility with or adaptation to network codes.

II. Background and Related Work

Network coding was introduced by Ahlswede et al. [1], who showed that as the network coding symbol size approaches infinity, a source can multicast information at a rate approaching the smallest minimum cut between the source and any receiver, which is not always possible with routing alone. Li et al. [12] showed that linear coding with finite symbol size is sufficient for multicast. Koetter and Médard [11] presented an algebraic framework for linear network coding that extended previous results to arbitrary networks and robust networking, and proved the achievability with time-invariant solutions of the min-cut max-flow bound for networks with delay and cycles. Using this algebraic framework, Ho et al. presented and analyzed distributed randomized network coding in [6], and gave further theoretical analysis in [7]. Concurrent independent work by Sanders et al. [15] and Jaggi et al. [8] considered single-source multicast on acyclic delay-free graphs, giving centralized deterministic and randomized polynomial-time algorithms for finding network coding solutions over a subgraph consisting of flow solutions to each receiver. Various protocols for and experimental demonstrations of randomized network coding [5] and non-randomized network coding [17], [13] have also been presented. Reference [5] considers single source multicast on Internet service provider network topologies.

Input: A directed graph $G = (N, E)$ with edge costs, a source node s and a set R of receiver ndoes.
Output: A low cost directed Steiner tree rooted at s and spanning all the nodes in R.
Method:

```
1 X ← R ;
2 while X ≠ Φ do
3    Run Djikstra's shortest path algorithm
     with source s until a node r ∈ X is reached ;
4    Add the path from s to r to the Steiner tree
     built so far ;
5    Set the costs of all the edges along this path to zero ;
6    X ← X − r ;
7 endwhile
```

Fig. 2. Pseudo-code for the Nearest Node First (NNF) Algorithm.

Approximation algorithms for undirected Steiner tree problem are given in [3], [9]. Waxman [16] considers undirected Steiner tree heuristics in the context of multicast routing. The Steiner tree problem for directed graphs is considered in [4], [14]. The online case is further discussed by Awerbuch et al. [2] for undirected graphs, and by Kodialam et al. [10] for directed graphs.

III. Problem Statement and Algorithms

We consider an online multi-source multicast problem in which source turn on and off dynamically. Thus, multicast connection requests are presented and accommodated sequentially. Existing connections are not disrupted or rerouted in trying to accommodate new requests. The algorithms are evaluated on the basis of the number of connections that are rejected or blocked owing to capacity limitations, and the multicast throughput supported.

For simplicity, we run our trials on directed acyclic networks, assuming that there exist mechanisms, e.g. based on geographical position or connection information, to avoid transmitting information in cycles. We also assume integer edge capacities and integer source entropy rates.

The online routing algorithm we consider finds a multicast tree for each new source using the Nearest Node First (NNF) heuristic for Steiner tree computation from [10], which uses Dijkstra's shortest path algorithm to reach receiver nodes in order of increasing distance from the source. Dijkstra's shortest path algorithm is run until a receiver node is reached. The corresponding source-receiver path is added to the Steiner tree and the costs of all the edges along this path are set to zero. The algorithm is then applied recursively on the remaining receiver nodes. This algorithm is described formally in Figure 2.

The coding algorithm we use is from [6]; we give a brief description here. The algorithm assumes that information is transmitted as vectors of bits. Linear coding[1] is carried out on vectors of length u in the finite field $\mathbb{F}_{2^u}$. The signal $Y(j)$ on a link j is a linear combination of processes X_i generated at node $v = \text{tail}(j)$ and signals $Y(l)$ on incident incoming links l. This is represented by the equation

$$Y(j) = \sum_{\{i \,:\, X_i \text{ generated at } v\}} a_{i,j} X_i + \sum_{\{l \,:\, \text{head}(l) = v\}} f_{l,j} Y(l)$$

[1] which is sufficient for multicast [12]

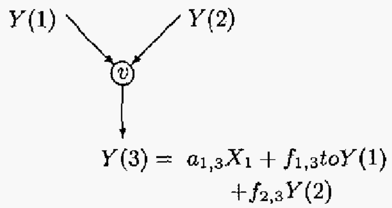

Fig. 3. Illustration of linear coding at a node.

An output process $Z(\beta, i)$ at receiver node β is a linear combination of signals on its terminal links, represented as

$$Z(\beta, i) = \sum_{\{l\,:\,\text{head}(l)=\beta\}} b_{\beta_{i,l}} Y(l)$$

An illustration of linear coding at a network node is given in Figure 3.

A randomly chosen network code is successful if each receiver obtains as many linearly independent combinations as the number of source processes. This enables it to decode each source process.

In order for random network coding to be attractive, the particular size of the field (code length) we use is important. Ho et al. [6] provides a lower bound for the success probability of randomized coding, $\left(1 - \frac{d}{q}\right)^{\nu}$, where q is the finite field size, d is the number of receivers, and ν is the number of links involved in the randomized coding. While this bound provides a worst-case guarantee over all possible network topologies with particular values of d, q, and ν, they may be pessimistic for most topologies. Thus, we wish to investigate what code lengths are necessary in practice to match or surpass the performance of traditional routing approaches.

The basic randomized network coding approach requires no coordination among nodes in the selection of code coefficients. If we allow for retrials to find successful codes, we in effect trade code length for some rudimentary coordination. Implementations for various applications may not be completely protocol-free, but the roles and requirements for protocols may be substantially redefined in this new environment.

IV. Experimental Setup

We run our trials on randomly generated geometric graphs, which model wireless ad-hoc network topologies. Test networks are generated with the following parameters: number of nodes n, number of sources r, number of receivers d, transmission range ρ, maximum in-degree and out-degree i. The parameter values for the tests are chosen such that the resulting random graphs would in general be connected and able to support some of the desired connections, while being small enough for the simulations to run efficiently. For each trial, n nodes are scattered uniformly over a unit square. To create an acyclic graph we order the nodes by their x-coordinate and choose the direction of each link to be from the lower numbered to the higher numbered node. Any pair of nodes within a distance ρ of each other is connected by a unit capacity link, and any pair within distance $\rho/\sqrt{2}$ of each other is connected by a link of capacity 2, provided this does not violate the degree constraints. The receiver nodes are chosen to be the d highest numbered nodes, and r source nodes are chosen randomly (with replacement) from among the lower-numbered half of the nodes. An example topology is given in Figure 4.

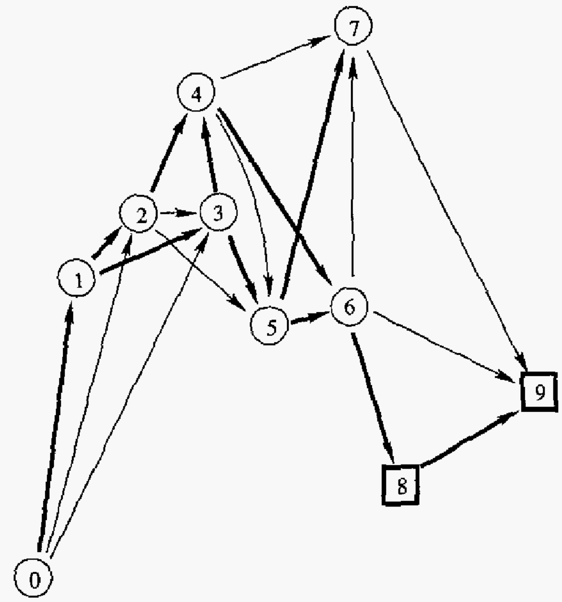

Fig. 4. An example of a randomly generated network used in our trials. This network was generated with parameters $n = 10$, $s = 6$, $r = 2$, $i = 4$, $\rho = 0.6$. Nodes are labeled as circles, and the receivers are squares; thick lines denote links with capacity two, and thin lines denote links with capacity one.

Each trial consists of a number of periods during which each source is either on (i.e. is actively transmitting) or off (i.e. not transmitting). During each period, any currently-on source turns off with probability p_o, and any currently-off source turns on with probability p_o if it is able to reach all the receivers. A source that is unable to reach all the receivers is blocked from turning on.

Initially all sources are off. For routing, in order for a source to turn on, it would need to find a tree connecting it to all the receivers using spare network capacity unreserved by other sources, and would then reserve capacity corresponding to the tree. A source that turns off frees up its reserved links for new connections. For coding, each network node that tries to turn on initiates up to three random choices of code coefficients within the network. If the receivers are able to decode the new source in addition to all the sources that are already on, the new source is allowed to turn on. A source that is not allowed to turn on is considered a blocked request.

The frequency of blocked requests and the average throughput are calculated for windows of 250 periods until these measurements reach steady-state, i.e. measurements in three consecutive periods being within a factor of 0.1 from each other. This avoids transient initial startup behavior.

V. Results and Discussion

We ran simulations on 242 networks generated randomly using 45 different parameter combinations. In 44 of these networks, coding outperformed routing in both blocking rate and throughput, doing better by more than 10% in at least one of

TABLE I

A SAMPLE OF RESULTS ON GRAPHS GENERATED WITH THE FOLLOWING PARAMETERS: NUMBER OF NODES n, NUMBER OF SOURCES r, NUMBER OF RECEIVERS d, TRANSMISSION RANGE ρ, MAXIMUM IN-DEGREE AND OUT-DEGREE i. b_r AND b_c ARE THE RATE OF BLOCKED CONNECTIONS FOR ROUTING AND CODING, RESPECTIVELY, AND t_r AND t_c ARE THE CORRESPONDING THROUGHPUTS.

Parameters						Results				
nodes n	srcs s	rcvrs d	deg i	range ρ	prob p_0	Network	b_r	t_r	b_c	t_c
8	6	1	4	0.5	0.6	1	1.54	1.46	1.55	1.46
						2	0.72	2.27	0.74	2.31
						3	0.26	2.78	0.23	2.74
9	6	2	3	0.5	0.7	1	2.14	0.84	2.17	0.83
						2	0.70	2.31	0.68	2.28
						3	0.90	2.05	0.71	2.26
10	4	2	4	0.5	0.6	1	0.61	1.43	0.50	1.45
						2	1.62	0.53	1.52	0.54
						3	0.14	1.96	0.00	2.05
10	6	2	4	0.5	0.5	1	1.31	1.63	0.71	2.28
						2	0.74	2.17	0.64	2.42
						3	1.51	1.54	1.49	1.61
10	9	3	3	0.5	0.7	1	1.05	2.37	1.14	2.42
						2	1.36	2.22	1.06	2.39
						3	2.67	0.87	2.56	0.89
12	6	2	4	0.5	0.6	1	1.44	1.67	0.71	2.31
						2	0.28	2.72	0.29	2.75
						3	0.75	2.28	0.73	2.31
12	8	2	3	0.5	0.7	1	2.39	1.73	2.34	1.74
						2	2.29	1.73	2.23	1.74
						3	1.57	2.48	1.52	2.51

these parameters. In 15 of these, coding outperformed routing in both parameters by more than 10%. In the rest, routing and coding showed comparable performance. Some results for various randomly generated networks are given in table I.

These simulations do not attempt to quantify precisely the differences in performance and overhead of randomized coding and online routing. However, they serve as useful illustrations in two ways.

Firstly, they show that the performance of the Steiner tree heuristic is exceeded by randomized coding over a non-negligible proportion of our randomly constructed graphs, indicating that when connections vary dynamically, coding offers advantages that are not circumscribed to carefully constructed examples. This is in contrast to static settings with optimal centralized control.

Secondly, the simulations illustrate the kinds of field sizes needed in practice for networks with a moderate number of nodes. Field size is important, since it affects memory and complexity requirements. To this end, the simulations use a small field size that still allows randomized coding to generally match the performance of the Steiner heuristic, and to surpass it in networks whose topology makes coding desirable over trees. The theoretical bounds of [6], [7] guarantee the optimality of randomized coding for large enough field sizes, but they are tight for worst-case network scenarios. In our trials, a field size of 17 with up to three retrials proved sufficient to achieve equal or better performance compared to the Steiner heuristic. The simulations show the applicability of short network code lengths for moderately-sized networks.

VI. Conclusions and Further Work

We have compared a distributed randomized network coding approach to an approximate online Steiner routing algorithm on multi-source multicast networks with dynamically varying connections. Our results show that for a significant proportion of randomly generated networks, the coding approach achieves superior performance over the routing-based approach. Such dynamic settings represent a substantially wider class of networking problems than previously recognized for which network coding shows promise of substantial benefits compared to routing. Our results suggest that the decentralized nature and robustness of randomized network coding can offer significant advantages in settings that hinder optimal centralized network control.

Further work includes investigation of other dynamically varying network scenarios, and extensions to non-uniform code distributions, possibly chosen adaptively or with some rudimentary coordination, to optimize different performance goals. Another question concerns selective placement of randomized coding nodes. The randomized and distributed nature of the approach also leads us naturally to consider applications in network security. It would also be interesting to consider protocol issues for different communication scenarios, and to compare specific coding and routing protocols over a range of performance metrics.

References

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

[2] B. Awerbuch, Y. Azar, and Y. Bartal. On-line generalized steiner problem. In *Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1996.

[3] P. Berman and V. Ramaiyer. Improved approximation algorithms for the steiner tree problem. *Journal of Algorithms*, 17:381–408, 1994.

[4] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li. Approximation algorithms for directed steiner problems. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

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

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

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

[8] S. Jaggi, P. Chou, and K. Jain. Low complexity algebraic network codes. In *Proceedings of the IEEE International Symposium on Information Theory*, 2003.

[9] M. Karpinsky and A. Zelikovsky. New approximation algorithms for the steiner tree problem. In *Technical Report, Electronic Colloquium on Computational Complexity (ECCC) TR95-030*, 1995.

[10] M. S. Kodialam, T. V. Lakshman, and S. Sengupta. Online multicast routing with bandwidth guarantees: a new approach using multicast network flow. In *Measurement and Modeling of Computer Systems*, pages 296–306, 2000.

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

[12] S.-Y. R. Li, R. W. Yeung, and N. Cai. Linear network coding. *IEEE Transactions on Information Theory*, 49:371–381, 2003.

[13] T. Noguchi, T. Matsuda, and M. Yamamoto. Performance evaluation of new multicast architecture with network coding. *IEICE Transactions on Communication*, E86-B, No.6, June 2003.

[14] S. Ramanathan. Multicast tree generation in networks with asymmetric links. *IEEE Transactions on Networking*, 4, August 1996.

[15] P. Sanders, S. Egner, and L. Tolhuizen. Polynomial time algorithms for network information flow. In *15th ACM Symposium on Parallel Algorithms and Architectures*, pages 286–294, 2003.

[16] B. M. Waxman. Performance evaluation of multipoint routing algorithms. In *Proceedings of IEEE INFOCOM*, 1993.

[17] Y. Zhu, B. Li, and J. Guo. Multicast with network coding in application-layer overlay networks. *IEEE Journal on Selected Areas in Communications*, 22(1), 2004.