

# On Average Throughput Benefits of Network Coding

Christina Fragouli\*

School of Comput. and Commun. Sciences

EPFL, Lausanne

christina.fragouli@epfl.ch

Emina Soljanin†

Bell Labs, Lucent

Murray Hill, NJ

emina@lucent.com

## Abstract

We calculate the average throughput benefit that network coding can offer as compared to routing for different classes of multicast configurations over directed graphs. The averaging is over the throughput each individual receiver experiences. For all the cases we examined, network coding at most doubles the average throughput.

## 1 Introduction

The min-cut, max-flow theorem states that a source node can send a commodity through a network to a sink node at the rate determined by the flow of the min-cut separating the source and the sink. By min-cut we refer to the minimum number of edges we need to remove to disconnect the source and the sink. Recently, Ahleswede et al. [1] have shown that if the nodes in the network can decode and re-encode incoming bits, the min-cut rate can be also achieved in multicasting to several sinks. Shortly afterwards Li et al. [2] showed that linear coding suffices to achieve the optimal rate.

A central question in this new area is what are the throughput benefits that network coding can offer as compared to routing. Routing refers to that each node in the network can only forward and not linearly re-encode incoming packets. In [3] it was shown that, for undirected graphs and allowing fractional routing, network coding can at most double the throughput each receiver experiences. Moreover, in [4] it was shown that for random undirected graphs the min-cut value concentrates around its average for every node in the graph. Thus, from Edmonds theorem [6], since the mincut to each node is approximately the same, there exist spanning trees that can route throughput comparable to the throughput that network coding achieves.

This result does not transfer for directed graphs: In fact, in [5] an example was provided which showed that, if we compare the minimum rate guaranteed to all receivers under routing, with the rate that network coding can offer, then network coding can offer benefits proportional to the number of sources  $h$ .

However, the minimum rate might be a pessimistic measure of performance, especially when the number of receivers is large, and the throughput they experience tends to

---

\*C. Fragouli was supported by Swiss National Foundation Grant No. 200021-103836/1.

†E. Soljanin was in part supported by the NSF grant No. CCR-0325673.

the average throughput achieved with routing, where the averaging is performed over the rate that each individual receiver experiences. Equivalently, we compare the sum rate of the receivers for network coding and routing. Using this performance measure, we will show that for different configurations, including the example in [5], network coding can offer only a constant factor benefit as compared to routing. Actually, we were not able to find an example where network coding more than doubles the average throughput.

Moreover, the example in [5] implicitly assumes a fixed over time routing scheme. We show that, for the same example, if we are interested in maximizing the long-term-average minimum throughput  $\lambda$ , there exists a routing scheme that guarantees to each receiver at least half the rate that is achieved with network coding. The quantity  $\lambda$  was used in [11] in the context of wireless ad hoc networks, to study the scaling of throughput with the number of nodes. It is also interesting that, in terms of throughput, network coding performed comparably to tree packing on the network graphs of six Internet service providers in [7].

In view of these results, we believe that it is still an open question whether network coding can offer significant throughput benefits as compared to routing, even for directed graphs.

The paper is organized as follows. Section 2 presents the problem formulation and briefly reviews previous results. Section 3 examines in detail a generalization of the example in [5]. Section 4 contains results for a number of different configurations and Section 5 concludes the papers.

## 2 Problem Formulation

We consider a communications network represented by a directed acyclic graph  $G = (V, E)$  with unit capacity edges. There are  $h$  unit rate information sources  $S_1, \dots, S_h$  and  $N$  receivers  $R_1, \dots, R_N$ . The number of edges of the min-cut between the source and each receiver is at least  $h$ . The  $h$  sources multicast information simultaneously to all  $N$  receivers at rate  $h$ .

We are interested in investigating the throughput benefits that network coding can offer as compared to routing (uncoded transmission). To measure throughput, we examine the rate sum of the receivers. More specifically, let  $T_{nc}^i$  denote the rate that receiver  $i$  experiences when network coding is used, and  $T_u^i$  the rate when only uncoded transmission is allowed. We are interested in comparing the total aggregate rate when network coding is used  $T_{nc} = \sum_{i=1}^N T_{nc}^i$  with the total aggregate rate when only uncoded transmission is allowed  $T_u = \sum_{i=1}^N T_u^i$ . Equivalently, we are interested in comparing the average throughput when network coding is used  $T_{nc}/N$  to the average throughput when only uncoded transmission is allowed  $T_u/N$ .

For a multicast configuration with  $h$  sources and  $N$  receivers, where the min-cut condition is satisfied for every receiver, it holds that

$$T_{nc} = Nh, \tag{1}$$

from the main theorem in network coding [1, 2]. Also, because the mincut to each node of the graph is greater than one, there exists a spanning tree [6], and thus the uncoded throughput is at least  $N$

$$N \leq T_u \leq hN. \tag{2}$$

Consider a unit-capacity directed graph, a node  $S$  that acts as a source (root), and  $N$  nodes that act as receivers. Given that the mincut between the source and each receiver is  $h$ , we are interested in calculating

$$\frac{T_u}{T_{nc}}. \quad (3)$$

In other words, we are interested in identifying trees, rooted at  $S$ , each tree spanning  $S$  and a subset of the  $N$  receivers, such that the total number of leaves (not counting the source) is maximized.

## Subtree Decomposition

For the development in this paper, we need to use terminology and results on the subtree decomposition that are available at [8, 9, 10]. Following we very briefly review the main points.

The subtree decomposition allows to group together different network configurations that are equivalent from a network coding point of view. The main idea is that we can “contract” the parts of the network through which the same information flows. To do that, we proceed as following. Starting from the graph that represents the physical network, we first take the associated line graph. For a given graph  $G = (V, E)$ , the associated line graph  $L(G)$  is the graph with vertex set  $E(G)$  in which two vertices are joined if and only if they are adjacent as edges in  $G$ .

Without loss of generality we may assume that the line graph contains a node corresponding to each of the  $h$  sources. We refer to these nodes as *source nodes*. Each node with a single input edge merely forwards its input symbol to its output edges. Each node with two or more input edges performs a coding operation (linear combination) on its input symbols, and forwards the result to all of its output edges. We refer to these last nodes as *coding points*. We also refer to the node corresponding the last edge of the path  $(S_i, R_j)$ , as the *receiver node* for receiver  $R_j$  and source  $S_i$ . For a configuration with  $h$  sources and  $N$  receivers there exist  $hN$  receiver nodes.

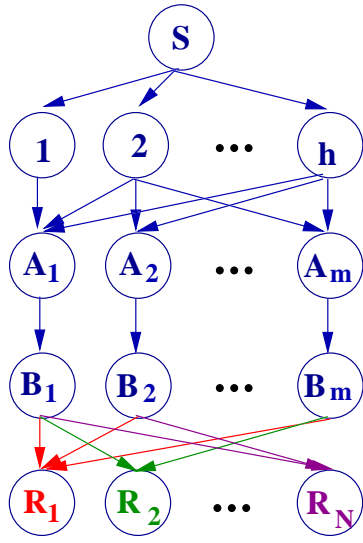
We partition the line graph into a disjoint union of subsets  $T_i$  so that the following properties hold:

1. each  $T_i$  contains exactly one source node or a coding point, and
2. every other node belongs to the  $T_i$  containing its first ancestral coding or source node.

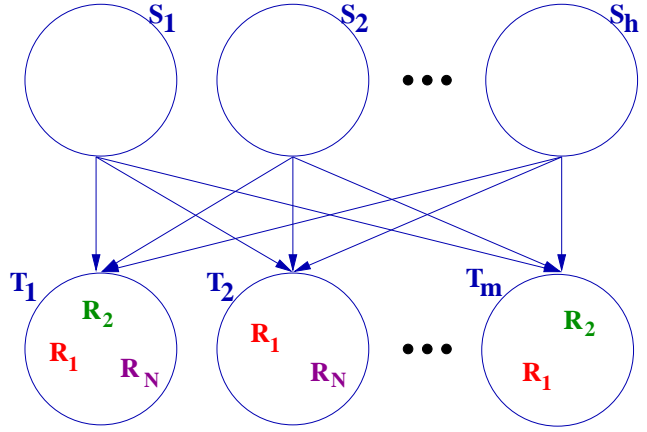
It is easy to see that the above conditions imply that each  $T_i$  is a tree because the only nodes with two or more input edges in the line graph are the coding points. We shall call the subset  $T_i$  a *source subtree* if it starts with a source node or a *coding subtree* if it starts with a coding point.

For the network code design problem, we only need to know how the subtrees are connected and which receiver nodes are in each  $T_i$ , whereas the structure of the network inside a subtree does not play any role. Thus we can contract each subtree to a node and retain only the edges that connect the subtrees, to get the *subtree graph*  $\Gamma$ .

Fig. 1 depicts a network configuration, and the corresponding subtree graph. For example, subtree  $T_1$  is the tree with edges  $\{(A_1, B_1), (B_1, R_1), \dots (B_1, R_N)\}$ .



(a) Multicast Configuration



(b) Associated Subtree Graph

Figure 1: A multicast configuration and the associated bipartite subtree graph, with  $h$  sources,  $N$  receivers and  $m$  coding subtrees. Each coding subtree has  $h$  parents.

### 3 Symmetric Distribution of Receivers

Consider a bipartite multicast configuration, with  $h$  sources and  $N$  receivers. Such a configuration is depicted in Fig. 1. Intuitively, one expects network coding to offer most throughput benefit when each subtree combines all  $h$  sources and the receiver nodes contained in each subtree are evenly distributed among the  $h$  sources. Indeed, this is the example [5]. The following proposition shows that this intuition may be inaccurate when we are interested in maximizing the average throughput.

**Proposition 1** *Consider a bipartite subtree configuration with  $h$  sources and  $N$  receivers. Assume that each coding subtree has  $h$  parents, and that every subset of  $h$  subtrees shares a receiver. Then*

$$\frac{T_u}{T_{nc}} \geq a, \text{ with } 0 < \frac{1}{2} < 1 - \frac{1}{e} \leq a. \quad (4)$$

**Proof**

Note that the min-cut condition is satisfied for every receiver, thus  $T_{nc} = Nh$ . Assume that the total number of subtrees is  $kh$ , that is, there exist  $h$  source subtrees and  $(k-1)h$  coding subtrees<sup>1</sup>. Since there exists one receiver that observes each  $h$  subtrees, the aggregate throughput when using network coding is equal to

$$T_{nc} = Nh = \binom{kh}{h} h. \quad (5)$$

We are going to calculate the throughput achieved when transmitting each of the  $h$  sources to exactly  $k$  subtrees, which will give us a lower bound on the uncoded throughput

<sup>1</sup>If the total number of subtrees, say  $M$ , is not a multiple of  $h$ , we can use as  $k = \lfloor \frac{M}{h} \rfloor$  which does not cause any significant difference in the result

receive source  $S_i$ .

The total loss of throughput, as compared to  $T_{nc}$ , will be equal to  $\sum_{i=1}^h M_i$ . Since source  $S_i$  is transmitted to  $k$  subtrees, there exist  $M_i = \binom{kh-k}{h}$  receivers that do not receive source  $S_i$ . Using symmetry, the total loss in throughput is  $h \binom{kh-k}{h}$  and

$$T_u = h \binom{kh}{h} - h \binom{kh-k}{h}. \quad (6)$$

The fraction of the throughput loss can be calculated as

$$a = \frac{T_u}{T_{nc}} = \frac{h \binom{kh}{h} - h \binom{kh-k}{h}}{h \binom{kh}{h}} = 1 - \frac{\binom{kh-k}{h}}{\binom{kh}{h}}. \quad (7)$$

So it is sufficient to show that the term  $\frac{\binom{kh-k}{h}}{\binom{kh}{h}}$  does not become zero. But its easy to see that

$$\begin{aligned} \frac{\binom{kh-k}{h}}{\binom{kh}{h}} &= \prod_{i=0}^{k-1} \left(1 - \frac{h}{(kh-i)}\right) \leq \left(1 - \frac{1}{k}\right)^k \leq \frac{1}{e} \\ \frac{\binom{kh-k}{h}}{\binom{kh}{h}} &= \prod_{i=0}^{h-1} \left(1 - \frac{k}{(kh-i)}\right) \leq \left(1 - \frac{1}{h}\right)^h \leq \frac{1}{e}, \end{aligned} \quad (8)$$

and thus

$$a = \frac{T_u}{T_{nc}} \geq 1 - \frac{1}{e} > \frac{1}{2} > 0. \quad (9)$$

□

## Discussion

Given that the example in Proposition 1 is at the foundation of the belief that network coding can offer significant throughput benefits for directed graphs, it is worth making the following points.

1. *Looking at the average throughput is a reasonable choice.* This is especially so when the number of receivers is large, and the throughput they experience tends to concentrate around a much larger value than the minimum. For example, Fig. 2 plots how the throughput is distributed among the receivers for two bipartite configurations with  $h$  sources and  $kh$  subtrees (special cases of Proposition 1). In both cases the fraction of receivers that observe throughput  $T_u^i = 1$  is very small as compared to the number of receivers that experience throughput  $T_u^i \geq h/2$ .
2. *We can achieve long-term-average throughput  $\lambda = \frac{h}{2}$  for each receiver.*

The example in [5] uses a fixed over time routing scheme. Assume that, for the same example, we are interested in maximizing the minimum average throughput  $\lambda = \min_i E(T_u^i)$  that every receiver is experiencing, where averaging is over time. The quantity  $\lambda = \min_i E(T_u^i)$  was used in [11] in the context of wireless ad hoc networks, to study the scaling of throughput with the number of nodes.

---

<sup>2</sup>It is easy to show that this choice maximizes  $T_u$ , and that the lower bound is infact tight.

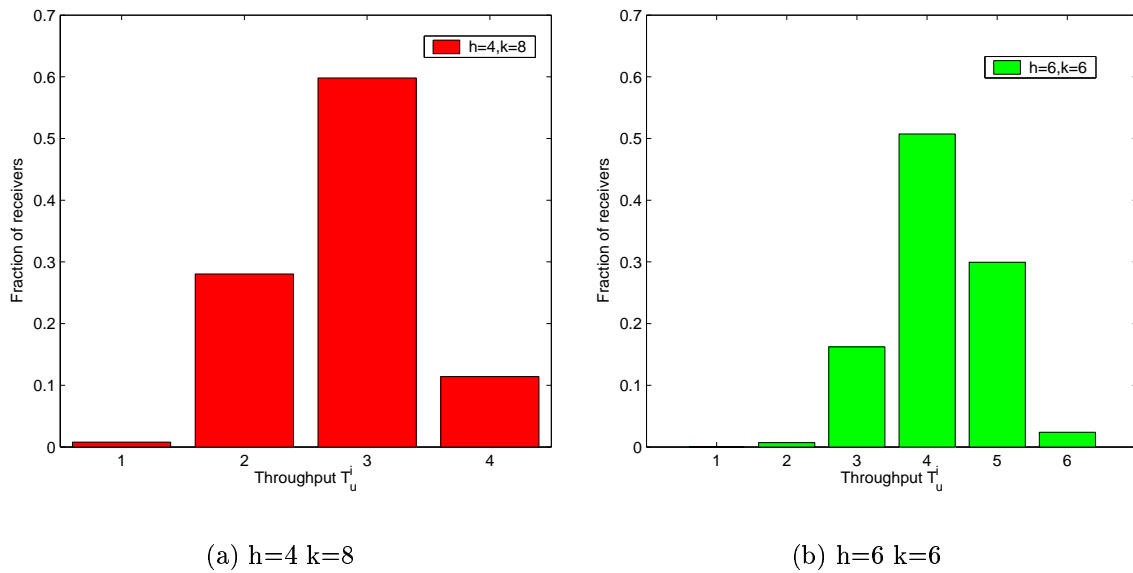


Figure 2: Histogram depicting on the y-axis the normalized number of receivers and on the x-axis the throughput  $T_u^i$  the receivers experience with routing, for two bipartite multicast configurations with  $h$  sources and  $kh$  subtrees.

As we saw in the proof of Proposition 1, in a configuration with  $kh$  subtrees, the throughput  $T_u$  does not depend either on the specific partition of the  $kh$  subtrees in  $h$  sets, or on the “1-1” mapping from the  $h$  sources to the  $h$  subsets of the partition. Averaging over all the possible partitions/source allocations - we denote this averaging by  $E_P(\cdot)$ - we get that

$$T_u = E_P(T_u) = \sum_{i=1}^N E_P(T_u^i) = N E_P(T_u^i) \implies E_P(T_u^i) = \frac{T_u}{N} \implies \frac{E_P(T_u^i)}{T_{nc}^i} \geq \frac{1}{2}.$$

This result hinges on the fact that the configuration is symmetric with respect to the receivers, and thus  $E_P(T_u^i) = E_P(T_u^j)$  for any two receivers  $i$  and  $j$ . Thus a routing scheme that cycles among the different partitions/source allocations, would guarantee to each receiver an average over time throughput that is at least half to that achieved with network coding.

3. *If each receiver  $R_i$  has mincut  $C_i \leq h$ , with routing we can always achieve at least  $T_u = \frac{1}{2} \sum_i C_i$  while with network coding it is no longer guaranteed that we can achieve  $T_{nc} = \sum_i C_i$ . Here I will add a sentence saying why this is so Thus, routing seems to be more flexible in terms of adapting to different receivers requirements. For example, if in the example in Fig. 1 we have some receivers that have mincut 2, while the remaining receivers have mincut  $h$ , to use linear network coding we can either design the network code so that all receivers receive throughput two, or have the receivers with mincut two receive zero throughput and the rest of the receivers throughput  $h$ .*
4. *For the configurations in Proposition 1 there exist polynomial-time algorithms for routing that achieve the prescribed throughput. This is obvious as it is sufficient to*

Polynomial time algorithms also exist in a more general setting, as discussed in Section 4.2. Thus for this particular example, network coding does not even offer reduced complexity benefits, as is the case in general networks [3].

## 4 Further Results

In this section we present results for a number of different classes of multicast configurations.

### 4.1 The case of two sources or two receivers

**Theorem 1** *For all networks with  $h = 2$  sources and  $N$  receivers, if the min-cut condition is satisfied for every receiver, it holds that*

$$\frac{T_u}{T_{nc}} \geq \frac{1}{2} + \frac{1}{2N}. \quad (10)$$

**Proof**

Since the mincut condition is satisfied for every receiver,  $T_{nc} = 2N$ . Moreover, the mincut to each node of the graph is at least one, thus there exists a tree that spans the source and the receivers [6]. Finally, since each source subtree contains at least one receiver node ([10] Theorem 3), at least one of the receivers will be able to receive both sources. Thus a lower bound on the achievable  $T_u$  throughput is  $N + 1$ .  $\square$ .

Note that for every  $N$ , there exist minimal configurations where without network coding we can not achieve throughput better than  $N + 1$ , *i.e.*, the bound is tight. Such configurations are the minimal subtree graphs with  $N - 1$  receivers and  $N - 1$  coding subtrees, which are described in ([10] Theorem 4). Indeed, for these configurations, each of the two source subtree contains one receiver node, thus we immediately start with rate  $T_u = 2$ . Moreover each of the  $N - 1$  coding subtrees contains exactly two receiver nodes. Using routing, one of the two receiver nodes in each subtree will collect incremental information. Thus  $T_u = 2 + N - 1 = N + 1$ . *I will try to make this argument more clear.*

Note that even if the number of receivers goes to infinity, the throughput ratio does not increase.

As a toy example, consider also the case where we have  $h \rightarrow \infty$  sources and  $N = 2$  receivers. Consider receiver  $R_1$ . Because the mincut to receiver  $R_1$  is  $h$ , there exist  $h$  edge-disjoint paths from the source to  $R_1$ , and thus, transmitting only to  $R_1$  allows to get throughput  $T_u = h$ .

### 4.2 Generalizations of proposition 1

In Proposition 1 we examined bipartite configurations, where each coding subtree has  $h$  parents, and where every subset of  $h$  subtrees shares a receiver. In this section we relax these conditions and examine more general configurations.

The following theorem relaxes the bipartite graph assumption.

**Theorem 2** *Consider a subtree configuration with  $h$  sources and  $N$  receivers. Assume that each coding subtree has  $h$  parents, and that each subset of  $h$  subtrees shares a receiver. Then*

$$\frac{T_u}{T_{nc}} \geq a, \text{ with } 0 < 1 - \frac{1}{e} \leq a. \quad (11)$$

*This proof need to be made clear*

Assume that the number of subtrees is  $kh$ . It is sufficient to show that we can assign each source to  $k$  coding subtrees (because of symmetry it doesn't matter which  $k$ ) and then apply the result in Proposition 1.

We partition in  $hk$  steps the  $hk$  subtrees into  $h$  sets (one for each source) of size  $k$ . At each step, we add a subtree  $T$  to a set  $E$  so that the mincut towards the rest of the subtrees is not affected. To do that, it is sufficient to add to  $E$  a subtree  $T$  that has a parent in the set and has no child in common with any other subtree in the set. Such a subtree  $T$  always exists; thus this procedure will work.  $\square$

Next we examine the case, where we still have a bipartite graph, and every subtree has  $h$  parents, but no constraint is placed on how the receivers are distributed. We show that the same result still holds.

**Theorem 3** *Consider a bipartite subtree configuration with  $h$  sources and  $N$  receivers. Assume that each coding subtree has  $h$  parents. Then*

$$\frac{T_u}{T_{nc}} \geq a, \text{ with } 0 < 1 - \frac{1}{e} \leq a. \quad (12)$$

**Proof**

To prove this result, we are going to calculate the average rate that receiver  $R_i$  is going to observe, if we uniformly randomly decide which source is going to each subtree. Since

$$T_u^i = \sum_{i=1}^h I(\text{Source } C_i \text{ assigned to } i \text{ subtree has not appeared before}), \quad (13)$$

where  $I()$  is the indicator function, we can calculate the average rate as

$$E(T_u^i) = \sum_{i=1}^h Pr(\text{Source } C_i \text{ assigned to } i \text{ subtree has not appeared before}). \quad (14)$$

On the other hand,

$$Pr(C_i \text{ not appeared before}) = \sum_{j=1}^h Pr(C_i \text{ not appeared before} | C_i = j) \frac{1}{h} = \left(\frac{h-1}{h}\right)^{i-1}, \quad (15)$$

thus

$$\sum_{i=1}^h Pr(C_i \text{ not appeared before}) = \sum_{i=1}^h \left(\frac{h-1}{h}\right)^{i-1} = \frac{1 - \left(\frac{h-1}{h}\right)^h}{1 - \frac{h-1}{h}} = h\left(1 - \left(1 - \frac{1}{h}\right)^h\right), \quad (16)$$

and

$$\frac{E(T_u^i)}{T_{nc}^i} \geq \left(1 - \left(1 - \frac{1}{h}\right)^h\right). \quad (17)$$

$\square$

There are three interesting points to be made regarding Theorem 3.



probability each receiver will experience a throughput close to the average. Thus, the average throughput gives a better indication of what is the throughput the different receivers observe than the minimum throughput.

### Concentration around the average in Theorem 3

*Need to write this proof clearly*

Consider a Doob-type martingale. Assume that we uniformly randomly choose which source is allocated to each subtree. Consider a receiver. Let the process  $t_i$  denote the throughput for the receiver under examination, if we know which sources were randomly allocated to the first  $i$  subtrees that the receiver observes. Then,  $t_h$  will denote the uncoded throughput the receiver will observe from the  $h$  subtrees. From Azuma's inequality, given that  $|t_{i+1} - t_i| \leq 1$ ,

$$\Pr(t_h < \frac{h}{2}) < e^{-\frac{h}{8}} \quad (18)$$

Thus, the probability that the receiver will observe throughput smaller than  $\frac{h}{2}$  reduces exponentially with  $h$ .  $\square$

- *To be written*
- Finally the third point is that, even if the number of parents is  $h - C$ , where  $C$  is a constant, for  $h \rightarrow \infty$  we can get as expected the same result as the following proposition shows.

**Proposition 2** *Consider a bipartite subtree configuration with  $h$  sources and  $N$  receivers. Assume that each coding subtree has  $h - C$  parents, where  $C$  is as constraint. Then as  $h \rightarrow \infty$*

$$\frac{T_u}{T_{nc}} \geq a, \text{ with } a \rightarrow 1 - \frac{1}{e}. \quad (19)$$

#### Scetch of Proof

Use same uniform random assignment as before, and the lower bound

$$\Pr(C_i \text{ not appeared before—} C_i = j \geq (\frac{h - C - 1}{h - C})^i. \quad (20)$$

$\square$

### 4.3 The case of two parents

Intuitively, the uniform random assignment in the proof of Theorem 3 works, because, if the number of parents of each subtree is large, there exist many alternative path choices for each receiver to receive the sources. This is no longer the case when the number of parents of each subtree is restricted. However, if each subtree has exactly two parents, we can still prove the following.

**Proposition 3** *Consider a bipartite subtree configuration with  $h$  sources and  $N$  receivers. Assume that each coding subtree has two parents. Then*

$$\frac{T_u}{T_{nc}} \geq \frac{1}{2}. \quad (21)$$

Since each coding subtree has two parents, connecting it to sources  $S_i$  and  $S_j$ , it contains  $N_1 \geq 1$  nodes that want to receive source  $S_i$  and  $N_2 \geq 1$  sources that want to receiver source  $S_j$ . If  $N_1 \geq N_2$  assign to the subtree source  $S_i$ , else source  $S_j$ . Since for each subtree, half the receivers nodes will contribute to the uncoded throughput, for the whole configuration as well, half the receiver nodes will contribute to the uncoded throughput.  $\square$

## 5 Conclusions

In this paper we examined the average throughput benefit that network coding can offer as compared to uncoded transmission for different classes of multicast configurations. For all the cases we examined, network coding could at most double the throughput.

## References

- [1] R. Ahlswede, N. Cai, S-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, Vol. 46, pp. 1204–1216, July 2000.
- [2] S-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inform. Theory*, Vol. 49, pp. 371–381, Feb. 2003.
- [3] Z. Li, B. Li, "Network coding in undirected networks", *Allerton 2003*.
- [4] S. Ramamoorthy, J. Shi, and R. Wesel, "Achievable capacity of network coding for random networks", *GlobeCom 2003*.
- [5] P. Sanders, S. Egner, and L. Tolhuizen, "Polynomial Time Algorithms For Network Information Flow," *Proc. 15th ACM Symposium on Parallel Algorithms and Architectures*, 2003.
- [6] J. Edmonds, "Edge-disjoint branchings" *Combinatorial Algorithms*, ed. R. Rustin, pp 91-96, Academic Press, NY 1973.
- [7] Y. Wu, P. A. Chou, and K. Jain, "A comparison of network coding and tree packing",
- [8] C. Fragouli, E Soljanin and A. Shokrollahi, "Network coding as a coloring problem", CISS 2004, Princeton University.
- [9] C. Fragouli, E Soljanin "A Connection between network coding and convolutional codes", ICC 2004, Paris.
- [10] C. Fragouli, E Soljanin "Information flow decomposition for network coding", submitted to *Transactions in Information theory*, available at: "<http://cm.bell-labs.com/cm/ms/who/emina>" and "<http://algo.epfl.ch>".
- [11] Gupta, P., Kumar, P. R. , "The Capacity of Wireless Networks " *IEEE Transactions on Information Theory*, Volume: 46 Issue: 2, Pages: 388 -404, Mar 2000.