Separating Distributed Source Coding from Network Coding

Aditya Ramamoorthy
Dept. of Electrical Engineering
University of California
Los Angeles, CA 90095
adityar@ee.ucla.edu

Kamal Jain
Microsoft Research
Redmond, WA 98052
kamalj@microsoft.com

Philip A. Chou
Microsoft Research
Redmond, WA 98052
pachou@microsoft.com

Michelle Effros
Dept. of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125
effros@caltech.edu

Abstract

This work considers the problem of distributed source coding of multiple sources over a network with multiple receivers. Work by Ho et. al [1] demonstrates that random network coding can solve this problem at the high cost of jointly decoding the source and the network code. Motivated by complexity considerations we consider the problem of separating the source coding from the delivery of an appropriate number of coded bits to each receiver. A multiplicative factor called the "price of separation" is defined that measures the gap to separability for a particular network and source distribution. Both networks with capacities and networks with costs on links are studied. We show that the problem with 2 sources and 2 receivers is always separable, and present counter-examples for other cases. Bounds are presented on the "price of separation". While examples can be constructed where separation does not hold, our experimental results show that in most cases separation holds implying that existing solutions to the classical Slepian-Wolf problem can be effectively used over a network as well.

1 Introduction

The Slepian-Wolf theorem [2] states that the lossless compression of two correlated sources that do not communicate with each other can be as efficient as the compression of the two sources when they do communicate with each other. Csiszár showed in [3] that linear codes were sufficient to achieve the Slepian-Wolf bounds and computed error-exponents for various decoders. In that paper, he also showed the existence of a universal decoder that is insensitive to the actual correlation structure of the sources. In recent years there has been a flurry of activity (see [4] [5] and their references) on code design for this distributed compression problem (hereby referred to as the S-W problem), spurred mainly by applications in sensor networks and video coding problems.

In a somewhat different line of work that can be broadly described as "Network Coding" a number of researchers have been investigating different network flow problems when intermediate nodes in the network have the ability to forward functions of received packets rather than simply adopting a replicate and forward strategy. The seminal work of Ahlswede et. al [6] showed that network coding achieves the capacity of single-source, multiple-terminal multicast. Subsequent work [7][8] showed that random linear network coding was an efficient distributed strategy to achieve this capacity. Variants of this problem involving multiple sources and multiple receivers are significantly harder and not much is known about them.

It is important to note that the classical S-W problem does not consider the sources to be communicating over a network i.e. there is a direct link from each source to the receiver. In addition the links do not have capacities on them. The S-W problem over a network has been considered by Razvan et. al [9] in the context of one receiver but they impose costs on links rather than considering capacities. In practical applications such as sensor networks, however one would expect that the sources communicate over a network with capacities on the edges to multiple receivers. This makes the problem of deciding the feasibility of a given distributed source coding problem with multiple sources and multiple receivers an interesting and important one. This problem was considered by Ho et. al [1]. They showed by using the approach pioneered by Csiszár that as long as the minimum cuts between all non-empty subsets of sources and a particular receiver were larger than the corresponding conditional entropies (more details follow), random linear network coding followed by appropriate decoding at the receivers would achieve the S-W bounds.

From a practical perspective one would like to leverage existing solutions to the classical S-W problem and thus separate the problem of sending the appropriate number of coded bits over a network from the source coding part. The solution proposed by Ho et. al comes at the cost of high complexity as the decoder at the receiver jointly decodes the code defined by the source code and the random network code. In general, the randomness in the network code destroys the structure in the source coder that allows tractable decoding. This paper formally defines the problem of separation between distributed source coding and network coding and investigates the conditions under which separation holds. We also define a parameter that quantifies the "price of separation" in terms of a multiplicative factor and study the range of values this parameter can take under different scenarios.

Section 2 explains the formulation of the problem. The notion of separation and the "price of separation" are formally defined in Section 2.1. Sections 3 and 4 present results on separation for networks with capacities and networks with costs on links respectively. Section 5 outlines the conclusions and suggests directions for future work. Due to lack of space we omit most of the proofs and refer to the reader to [10].

2 Problem Formulation

In this section we define an instance of the distributed source coding problem over a network. We are given

a) N_S discrete memoryless sources denoted by $X_i, i = 1, ..., N_S$ whose output values are drawn i.i.d. from a joint distribution $p(X_1, ..., X_{N_S})$. Each source alphabet is without loss of generality assumed to be a Galois field of a power of 2. All joint and conditional entropies are assumed to be rational, to keep the arguments simple. However this assumption is not essential and our arguments can be reformulated more generally.

b) A graph $G = (V, E, C)$, where V is the set of nodes, E is the set of edges and C is a function that gives the capacity of each edge, a set of source nodes $S \subset V, |S| = N_S$, a set of receiver nodes $T \subset V, |T| = N_R$. We assume that all the capacities are rational-valued (the comments in (a) apply).

With the above information as input, we can define the following items that help us in setting up the problem,

a) The S-W region of the sources, is denoted

$$\mathcal{R_{SW}} = \{[R_1 R_2 ... R_{N_S}] : \forall B \subseteq \{1, 2, ..., N_S\}, \sum_{i \in B} R_i > H(X_B / X_{B^c})\}$$

where X_B represents the vector of random variables $(X_{i_1}, X_{i_2}, ..., X_{i_{|B|}})$, for $i_k \in B, k = 1, ..., |B|$.

b) For a given $T_i \in T$ we can define a capacity region with respect to S. This is the region that defines the maximum flow from each subset of S to the receiver T_i. Formally,

$$C_{T_i} = \{[R_1 R_2 ... R_{N_S}] : \forall B \subseteq S, \sum_{i \in B} R_i \leq \text{min-cut}(B, T_i)\}.$$

An instance of a distributed source coding problem over a network is defined by

$$P =< \mathcal{R_{SW}}, G, S, T >$$

c) The network coding model used here is explained in detail in [1]. We communicate n symbols in a block. This means that each source X_i is encoded into some nR_i bits by its source encoder. This also means that links of capacity C bits/symbol can communicate $\lfloor nC \rfloor$ bits per block. Linear network coding is done over vectors of bits in the binary field. Conceptually each link can be thought of as multiple unit capacity links and each such new link corresponds to one bit in the code vectors.

d) We introduce N_S virtual nodes denoted $S'_1, ..., S'_{N_S}$, that can be thought of as the source encoders. According to Csiszár [3] it is sufficient for S'_i to perform linear encoding defined by a function $f_i^n : [X_{i,1}, X_{i,2}, ..., X_{i,n}] \rightarrow [U_{i,1} U_{i,2} ... U_{i,nR_i}]$. The vector $[U_{i,1} U_{i,2} ... U_{i,nR_i}]$ is denoted by U_i^n. We define an augmented graph denoted by $G' = (V \bigcup S', E \bigcup E', C')$. Here, E' represents the edges from S' to G and C' is a function that returns the capacities on the edges in G'.

e) We denote the graph G when considered over n time-steps by $G^n = (V, E_n, C_n)$. As explained above, when considered over n time-steps we can consider a unit-capacity link between 2 nodes in G to now consist of n unit-capacity links. E_n denotes the edge set at block length n and C_n denotes the function that returns the capacity of each edge in E_n. G'^n is similarly defined.

f) A solution to the problem $P =< \mathcal{R_{SW}}, G, S, T >$ is defined by the set of local encoding vectors on each link in G'^n. If g_e represents the local encoding vector on link e belonging to G'^n, then the solution to P at block length n, denoted by P_{sol}^n is given by $P_{sol}^n = \{g_1, g_2, ..., g_{|\{E \cup E'\}_n|}\}$.

Definition 1 *Feasibility Condition :- Consider an instance of a distributed source coding problem over a network defined by $P =< \mathcal{R_{SW}}, G, S, T >$. Let C_{T_i} be the capacity region of each receiver $T_i \in T$ with respect to S.If,*

$$C_{T_i} \bigcap \mathcal{R_{SW}} \neq \phi, \forall i = 1, ..., N_R \tag{1}$$

then the feasibility condition is said to be satisfied and P is said to be feasible.

Theorem 1 *[1] Consider an instance of a distributed source coding problem over a network defined by* $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$. *If the feasibility condition (Definition 1) is satisfied, then randomized network coding over* G'^n *followed by minimum-entropy [3] or maximum-likelihood decoding at each receiver causes the probability of decoding error to go to 0 as* $n \to \infty$.

This theorem follows from Theorem 1 in [1].

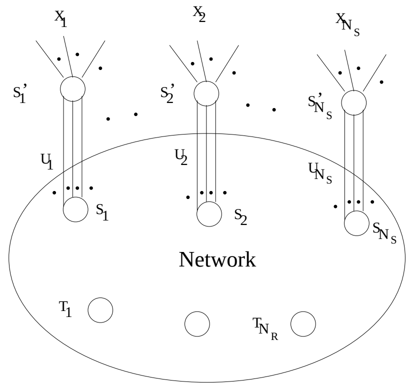


Figure 1: The figure shows a network with N_S sources (X_i's), source encoders (S_i''s) and source nodes (S_i's). The source coded bits are represented by the U_i's. There are N_R receivers (T_i's)

2.1 Notion of Separation of Distributed Source Coding and Network Coding

In the sequel we shall always work in the problem formulation presented in Section 2. As mentioned before the result of Theorem 1 assumes the existence of a minimum-entropy/maximum-likelihood decoder that can be arbitrarily complex when random network codes are used. In this paper we study the feasibility of performing these operations independent of each other. For this we need a formal definition of separation between distributed source coding and network coding that is presented below.

Definition 2 *Separable Problem:- Consider a distributed source coding problem over a network,* $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$. P *is said to be separable if for all* n *sufficiently large there exists a solution* P^n_{sol} *so that* $\forall T_i \in T$, *there exists a rate vector* $[R^{T_i}_{S_1}, ..., R^{T_i}_{S_{N_S}}] \in \mathcal{R}_{\mathcal{SW}}$ *such that for each* $S_j \in S$, *there exists* $B^{T_i}_{S_j} \subseteq U^n_j, |B^{T_i}_{S_j}| \geq nR^{T_i}_{S_j}$ *and the transfer function induced by* P^n_{sol} *from* $B^{T_i}_{S_1} \times B^{T_i}_{S_2} \times ... \times B^{T_i}_{S_{N_S}}$ *to* T_i *is one-to-one.*

For a given P it follows from [3] that if f^n_i is chosen to be a random linear block code and a solution P^n_{sol} is separable, reconstruction of the sources at each receiver is possible with probability of error going to 0 as $n \to \infty$. Thus a separable solution allows us to leverage existing solutions ([4][5]) for the classical S-W problem. One need not worry about the complexity of jointly decoding the source and the network code.

2.2 Price of Separation

It should be clear that the set of solutions that joint decoding can achieve is larger than the set of solutions that can be achieved by separability. By increasing the capacity of the network sufficiently it is always possible to achieve a separable solution e.g. if we increase the capacity of all links to the joint entropy of the sources then, surely separability will hold. With this in mind a multiplicative factor η_{cap} is defined which we call the "Price of Separation".

Definition 3 *Price of Separation - Consider a distributed source coding problem over a network,* $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$*. Let* $\alpha \geq 1$ *be a multiplicative factor by which the capacities of all the links in the network are increased so that a separable solution exists. We define* G_α *to be the graph* $G_\alpha = (V, E, \alpha C)$*, i.e. the graph* G *with capacities multiplied by* α*. The price of separation is defined to be,*

$$\eta_{cap} = \min_{<\mathcal{R}_{\mathcal{SW}}, G_\alpha, S, T> \text{ is separable}} \alpha \tag{2}$$

The factor η_{cap} characterizes the gap to separability as a single parameter. In the sequel we present worst case bounds on this parameter and also compute it for some typical instances.

3 Results for Networks with Capacities

In this section we present various results on separability and the "Price of Separation". We note that when we have $N_R = 1$, i.e. there is only one receiver in the system, then

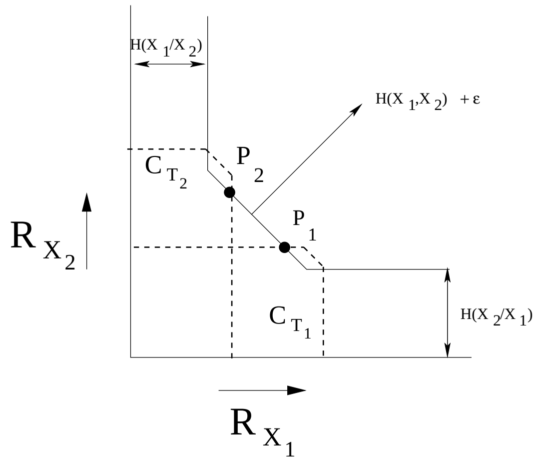


Figure 2: The two dotted regions are the capacity regions of T_1 and T_2 respectively. P_2 and P_1 are the closest operating points for each terminal on the S-W boundary.

separation trivially holds as the minimum cut conditions guarantee that there exist a sufficient number of edge-disjoint paths from each source node to the receiver so that routing itself would suffice to ensure the delivery of a terminal rate vector that lies in the S-W region of the sources. Of course the case corresponding to $N_S = 1$ is not a distributed source coding problem.

3.1 The 2-Sources, 2-Receivers Case

Theorem 2 *Consider a problem* $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$*, with* $|S| = 2, |T| = 2$ *. If* P *is feasible, then* P *is separable.*

Proof :- Since the connection is feasible the capacity regions of T_1 and T_2 intersect $\mathcal{R}_{\mathcal{SW}}$ as shown in Fig. 2. This further means that there exists a rational $\epsilon > 0$ such that the line $R_{X_1} + R_{X_2} = H(X_1, X_2) + \epsilon$ has a non-empty intersection with C_{T_1} and C_{T_2}. We force the terminals T_1 and T_2 to operate on the points marked P_1 and P_2 respectively on Fig. 2. Then, the following properties hold true,

1. $R_{S_1}^{T_1} \geq R_{S_1}^{T_2}, R_{S_2}^{T_2} \geq R_{S_2}^{T_1}$ and $R_{S_1}^{T_i} + R_{S_2}^{T_i} = H(X_1, X_2) + \epsilon$, for $i = 1, 2$

2. For $P_1' \in C_{T_1} \cap \mathcal{R}_{\mathcal{SW}} \cap \{(x_1, x_2) : x_1 + x_2 = H(X_1, X_2) + \epsilon\}$ and $P_2' \in C_{T_2} \cap \mathcal{R}_{\mathcal{SW}} \cap \{(x_1, x_2) : x_1 + x_2 = H(X_1, X_2) + \epsilon\}$,

$$dist(P_1, P_2) \leq dist(P_1', P_2') \quad \text{where } dist \text{ represents the distance function} \tag{3}$$

if $dist(P_1, P_2) = 0$, then a separable solution exists by the multicast result of Ahlswede et. al [6], so we focus on the case when $dist(P_1, P_2) > 0$.

We assume that we operate over n large enough so that $n(H(X_1, X_2)+\epsilon), nH(X_1)$ and $nH(X_2)$ and the capacities of all links in G'^n are integral. The proof below is inspired by the technique used in [11]. For now consider only $R_{S_1}^{T_1}, R_{S_1}^{T_2}$ and $R_{S_2}^{T_2}$. For ease of explanation we put $g = nR_{S_1}^{T_1}, r_1 = nR_{S_1}^{T_2}$ and $r_2 = nR_{S_2}^{T_2}$. Menger's theorem guarantees the existence of edge-disjoint paths in G'^n corresponding to these numbers. We denote the sets of edge-disjoint paths by $\mathbb{G}, \mathbb{R}_1$ and $\mathbb{R}_2$, so that $|\mathbb{G}| = g, |\mathbb{R}_1| = r_1, |\mathbb{R}_2| = r_2$. Note that paths in $\mathbb{R}_1 \cup \mathbb{R}_2$ are disjoint.

Each edge e in each path belonging to $\mathbb{G} \cup \mathbb{R}_1 \cup \mathbb{R}_2$ will be labelled (as explained below) with a pair of colors (c_1^e, c_2^e). Some edges may be labelled with just one color.

We label all edges in paths belonging to $\mathbb{G}$, "green" and edges belonging to paths in $\mathbb{R}_1$ and $\mathbb{R}_2$, "red". At the end of this procedure, some edges will be labelled by two colors whereas others would have just one.

We claim that we can always find $(g - r_1)$ exclusively green paths from S_1 to T_1. To prove this, we define an algorithm A that takes as input path $P_1 \in \mathbb{G}$.

Algorithm $A(P_1)$:-

1. Traverse P_1 starting at node S_1 and find the first edge e_1 that has color ("green","red").

2. If no such e_1 is found then **STOP**.

3. **ELSE** There are two possibilities,

   a) e_1 belongs to a path in $\mathbb{R}_2$.
   We claim that this is impossible. To see this, suppose that e_1 belonged to a path $P' \in \mathbb{R}_2$ such that $P' = P_1' - e_1 - P_2'$, where P_1' represents the portion of P' from S_2 to e_1 and P_2' represents the portion of P' from e_1 to T_2.

   We can color all edges on P_1 from S_1 to e_1, "red" (in addition to their existing color), and remove "red" from the color of edges in P_1'. This effectively means that we can increase a bit from S_1 to T_2 and reduce a bit from S_2 to T_2. But, doing so implies that P_2 and P_1 can be brought closer, which is a contradiction.

   b) e_1 belongs to a path in $\mathbb{R}_1$.
   If e_1 is the first edge of P_1, then **STOP**
   ELSE Again suppose that e_1 belonged to a path $P' \in \mathbb{R}_1$, such that $P' = P_1' - e_1 - P_2'$, where P_1' represents the portion of P' from S_1 to e_1 and P_2' represents the portion of the P' from e_1 to T_2. Color all edges on P_1 from S_1 to e_1, "red" (in addition to their existing color), and remove "red" from the color of the edges in P_1'.

Now we define a condition that each path $P_1 \in \mathbb{G}$ has to satisfy.

$$Cond(P_1) = \{\text{All edges in } P_1 \text{ are "green"}\} \\ \text{or } \{\text{the first edge of } P_1 \text{ is ("green", "red")}\} \tag{4}$$

We continue applying A to each path of $\mathbb{G}$ until all paths in $\mathbb{G}$ satisfy *Cond*. It is easy to see that A will always halt.

At the end of this process, we claim that there exist $(g - r_1)$ paths belonging to $\mathbb{G}$ that are colored exclusively with "green". This can easily be seen to be true, because if Algorithm A above changes a path $P' \in \mathbb{R}_1$, then it removes the color "red" from one outgoing edge of S_1 and places it on another outgoing edge. Thus, the number of outgoing edges that have the color "red" remains constant $= r_1$. Therefore, there have to be $(g - r_1)$ outgoing edges that are purely "green", which in turn means that there exist $(g - r_1)$ paths from S_1 to T_1 that are exclusively "green". To summarize, the above

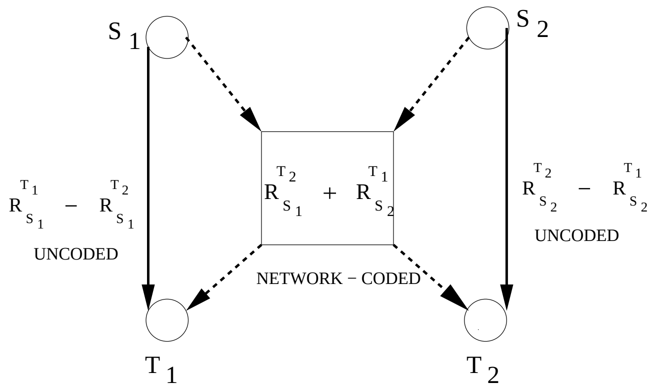


Figure 3: The figure shows that every 2-source 2-terminal distributed source coding problem can be decomposed into two uncoded flows and one coded flow.

argument shows that by choosing n sufficiently larger and carefully choosing paths, we can

a) Route $(R_{S_1}^{T_1} - R_{S_1}^{T_2})$ bits from S_1 to T_1.

b) A similar argument shows that we can route $(R_{S_2}^{T_2} - R_{S_2}^{T_1})$ bits from S_2 to T_2.

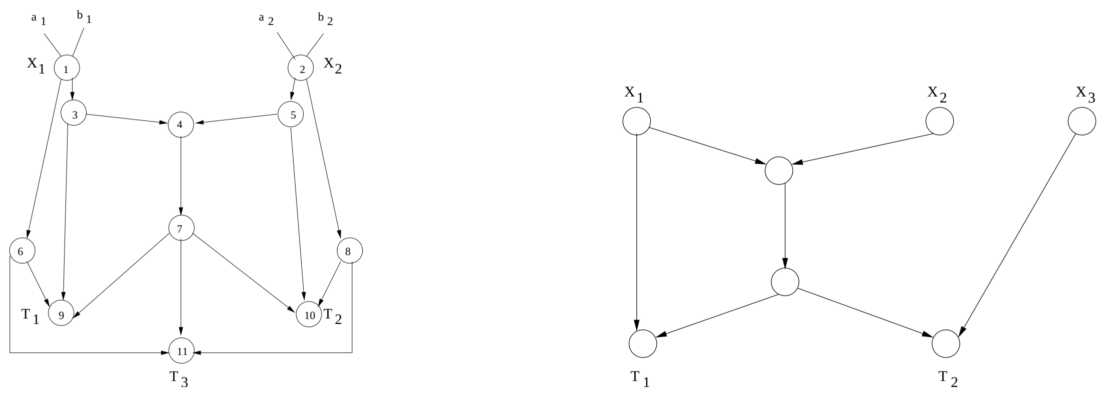
c) Each terminal needs exactly $(R_{S_1}^{T_2} + R_{S_2}^{T_1})$ bits more to satisfy it's requirement. But, we can send this via network coding, by invoking the multicast result of Ahlswede et. al [6].

Thus, the 2-sources, 2-receivers problem can always be decomposed as depicted in Fig. 3 which in turn implies separability. ■

3.2 Cases with Higher Number of Sources and Terminals

This above proof shows the surprising result that in the case of 2-sources and 2-receivers separability always holds. In fact for the case of 2-sources, 3-receivers (Fig. 4(a)) and 3-sources, 2-receivers (Fig. 4(b)) we have explicit counter-examples where even though the connection is feasible, separation does not hold. In Fig. 4(a), no coding strategy can result in separability at T_3 and in Fig. 4(b), T_2 cannot separate out the bits from X_1 and X_3.

Counter-examples for higher number of sources and receivers can be constructed by simply choosing the counter-examples above as appropriate subgraphs in the network. Upper bounds on η_{cap} based on the number of terminals in the system can also be found.



(a) (b)

Figure 4: (a)Counter example to separability for the case of 2 sources and 3 receivers. $H(X_1) = H(X_2) = 2$. $H(X_1, X_2) = 3$. (b)Counter example to separability for the case of 3 sources and 2 receivers. $H(X_1) = H(X_2) = H(X_3) = 1$. Correlation model - X_1 independent of X_2 and $X_2 = X_3$. In both (a) and (b) the capacity of all the links = 1.

Lemma 1 *Bound on the "Price of Separation" :- For any* $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$, *we have* $\eta_{cap} \leq |T|$.

Proof :- The proof follows from a simple time-sharing strategy and is omitted.

3.3 Results on Typical Instances

The previous results demonstrate that there exist networks and source distributions where separation does not hold in general. To test whether separation holds on typical instances of the problem we generated a large number of graphs and corresponding S-W regions. Our simulation methodology is explained below,

a) A total of M nodes were scattered randomly on the unit disk. To ensure acyclicity[1], an order was enforced whereby connections could only go from left to right, e.g. nodes v_1 and v_2 would be connected only if their distance was less than a parameter d and v_1 was to the left of v_2.

b) The first N_S nodes were declared to be the source nodes and the last N_R nodes were declared receiver nodes. In the simulation we were able to handle only small values ($N_S \leq 3, N_R \leq 3$).

c) Minimum cuts were computed between all subsets of the sources and each of the terminals. Based on these values a S-W region was generated, such that none of the constraints was trivial, and the problem was feasible (Definition 1).

d) To enforce separability, a linear program was developed that took the the graph and S-W region as input. The total flow from the sources to the terminals was broken up into $(2^{N_S} - 1)(2^{N_R} - 1)$ flows. The capacity on each edge was split into a portion for each flow, and within each flow, network coding was allowed. In addition the S-W constraints were enforced by summing the values of appropriate flows. The objective function to be minimized was the price of separation, η_{cap}.

[1]This constraint was enforced since network coding has been empirically found to be more effective for acyclic networks.

Since the number of flows is approximately exponential in $N_S + N_R$, it is hard to solve the LP for large values of this sum. A feasible solution for the LP implies the existence of a separable solution for the problem. This is because, each flow can transport its value (i.e. rate) to it's respective terminals using network coding. Since the capacity of each edge is split across the flows, we can assume that each flow is operating independent of others over the network. The notion of separability under which the LP operates is however slightly weaker than the definition in Section 2.1. Here, a receiver T_i is allowed to recover $R_{S_j}^{T_i}$ linear combination of the bits from U_j^n as long as the linear transformation specifying the combination is full-rank. We suspect that these results hold for the stronger definition as well.

In all the trials we ran (over 200 in number) we did not find a single instance where $\eta_{cap} > 1$. Thus, separation does seem to hold in most typical instances of the problem. It is important to point out that "if" the LP has a solution "then" we are guaranteed the existence of a separable solution, however the existence of a separable solution may not always imply the existence of feasible solution to the LP.

4 Results for Networks with Cost Constraints

The minimum cost version of the problem where each link in the network has a constant cost per bit of usage but no capacity constraint (as in [9]) was also investigated. Here the input graph is $G = (V, E, cost)$ where *cost* is a function that returns the cost on each link per bit. A problem instance is defined as before, $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$.

Definition 4 *Cost of a solution :- Suppose in a given solution P_{sol}^n, each link e has r_e bits flowing over it. The total cost of the solution is given by,*

$$\kappa(P_{sol}^n) = \frac{1}{n} \sum_{e \in G^n} r_e \times cost(e) \tag{5}$$

Definition 5 *Consider a distributed source coding problem over a network, $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$. Let P_{sol}^n be a solution to P. Let, η_{cost}^n be defined as,*

$$\eta_{cost}^n = \frac{\min_{\{P_{sol}^n \text{ is separable}\}} \kappa(P_{sol}^n)}{\min \kappa(P_{sol}^n)} \tag{6}$$

Separability is said to hold if for large enough n, η_{cost}^n is arbitrarily close to 1.

As before we can show that separability still holds in the case of 2 sources and 2 receivers and does not hold in other cases.

Theorem 3 *Consider a distributed source coding problem over a network, $P =< \mathcal{R}_{\mathcal{SW}}, G, S, T >$ with costs but no capacity constraints and $|S| = 2$ and $|T| = 2$. The costs on each link are assumed to be ≥ 0. Then P is separable.*

Proof :- It follows from the proof of Theorem 2 with minor modifications.

Counter-examples similar to ones in Section 3.2 can be found for the cost version as well.

5 Conclusion

The problem of distributed source coding of multiple sources over a network with multiple receivers was considered. In particular we focused on investigating whether the source coding part could be separated from the problem of transmitting an appropriate number of coded bits to each receiver. Both networks with capacities and networks with costs on links were considered. While in general the answer is negative, we showed that in the specific case of 2 sources and 2 receivers, a separable solution always exists. Our experiments on randomly generated networks show that in fact separation almost always holds[2]. In the full version of the paper we also consider separability issues when the network employs random network coding and show that separation depends upon the input bit rate in this situation and the "price of separation" may be much higher as compared to the case when we perform careful network coding. In the future we propose to look at selective network coding so that the complexity of joint source and network code decoding is tractable.

References

[1] T. Ho, M. Medard, M. Effros, and R. Koetter, "Network Coding for Correlated Sources," in *CISS*, 2004.

[2] T. Cover and J. Thomas, *Elements of Information Theory.* Wiley Series, 1991.

[3] I. Csiszar, "Linear Codes for Sources and Source Networks: Error Exponents, Universal Coding," *IEEE Trans. on Info. Th.*, vol. 28, no. 4, pp. 585–592, 1982.

[4] S. S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense sensor network," in *IEEE Signal Processing Magazine*, March 2003.

[5] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed Source Coding for Sensor Networks," in *IEEE Signal Processing Magazine*, Sept. 2004.

[6] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network Information Flow," *IEEE Trans. on Info. Th.*, vol. 46, no. 4, pp. 1204–1216, 2000.

[7] T. H. et. al, "Randomized Network Coding," in *41st Allerton Conference on Communication, Control, and Computing*, 2003.

[8] P. A. Chou, Y. Wu, and K. Jain, "Practical Network Coding," in *41st Allerton Conference on Communication, Control, and Computing*, 2003.

[9] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On Network Correlated Data Gathering," in *IEEE Infocom*, 2004.

[10] A. Ramamoorthy, K. Jain, P. A. Chou, and M. Effros, "Separating Distributed Source Coding from Network Coding," *MSR Tech. Rep., under preparation.*

[11] K. Jain, M. Mahdian, and M. R. Salavatipour, "Packing Steiner Trees," in *SODA*, 2004.

---
[2] We do not rule out a different model of generation where this might not be true