On the Capacity of Multiple Unicast Sessions in Undirected Graphs

Kamal Jain *Vijay V. Vazirani* *Raymond Yeung* *Gideon Yuval*

***Abstract*— Li and Li conjectured that in an undirected network with multiple unicast sessions, network coding does not lead to any coding gain. Surprisingly enough, this conjecture could not so far be verified even for the simple network consisting of $K_{3,2}$ with four source-sink pairs. Using entropy calculus, we provide the first verification of the Li–Li conjecture for this network. We extend our bound to the case of an arbitrary directed bipartite network.**

I. Introduction

Network coding is a simple though powerful idea put forth recently by Ahlswede, Cai, Li and Yeung [ACLY00]. They showed how this idea can be used to improve the broadcast rate from a single sender s to several receivers, $t_1, \ldots, t_n$. The traditional bound on the broadcast rate is the maximum number of Steiner trees, rooted at s and containing all the receivers as required nodes, that can be packed in the network. [ACLY00] show the improved bound of the minimum cut separating s from a t_i. They gave instances of networks where the latter bound is strictly better than the former.

Koetter and Medard [KM] showed how network coding could be implemented with deterministic functions over a finite field. However, the field size they require is exponential in n and the capacity of the minimum cut. Jaggi et. al. [JSC$^+$] brought the field size down to linear in n.

The ratio of the information rate with and without network coding is called the *coding advantage* of the network. Charikar and Agarwal [CA04] showed that for the broadcast problem, the coding advantage is precisely equal to the integrality gap of the Steiner tree LP for the given network. The latter has been extensively studied within approximation algorithms (e.g., see [Vaz01]).

Although the coding advantage of the broadcast problem has been precisely understood, the same is not the case for other information flow problems. In this paper we consider one such problem: let $G = (V, E)$ be an undirected network with capacities on edges and with specified source-sink pairs $s_i, t_i, 1 \leq i \leq k$. We need to send information at the maximum possible rate r from each s_i to t_i. The problem is to determine the coding advantage of G for this task.

If network coding is not used then the maximum rate r is given by the throughput of the given network which is determined by viewing this as a multicommodity flow problem. An obvious upper bound on the rate if network coding is used is the sparsity of the sparsest cut in the network, where sparsity of a cut is defined to be the ratio of its capacity and the demand disconnected. The seminal result of Leighton and Rao [LR88] places a bound of $O(\log n)$ on the ratio of sparsest cut and throughput. This ratio is also the integrality gap of the sparsest cut LP.

Where in this range does the true answer lie? Li and Li [LL04] conjecture that for undirected networks, the coding advantage is always 1, i.e., network coding does not help. The smallest graph we know of for which the sparsest cut LP has an integrality gap is $K_{3,2}$ with all edges of unit capacity and with every pair of vertices on the same side as a source-sink pair (hence there are 4 source-sink pairs). For this graph, the sparsest cut has capacity 1 and the maximum throughput is $3/4$. Interestingly enough, upper bounding maximum information rate for this graph, assuming network coding, is in itself a non-trivial task.

In this paper, we show that the maximum information rate for this graph is $3/4$ hence confirming the conjecture of Li and Li for this graph. Our proof is purely information theoretic. We introduce two key inequalities, the input-output inequality and the crypto inequality, for deriving our proofs. These two inequalities have a particularly simple form. This was intentional – we chose not involve the notion of time in these inequalities and thereby arrive at simpler, though weaker, constraints that still yield the right bound for the $K_{3,2}$ network. It is also not that we ignored the notions of time and causality relationships, instead we used these notions in proving our simple inequalities.

We further use these inequalities to show the following general result: Let $G(U.V, E)$ be a bipartite graph with bipartition (U, V) in which k_1 (k_2) edges are are directed from U to V (from V to U). Furthermore, assume that there are n_1 (n_2) source sink pairs that have source in U (V) and sink in V (U), and n source-sink pairs that are one the same side of the bipartition (either both source-sink in U or both in V). Then, the maximum information rate in G is bounded by:

$$\min\left(\frac{k_1}{n+n_1}, \frac{k_2}{n+n_2}\right).$$

We believe that our proof technique should extend to the Li and Li conjecture; however, this still remains a challenging problem. Once this conjecture is resolved, there are more general scenario to be considered, e.g., broadcasts over multiple sessions where network coding can be used across sessions and network coding for correlated sources via Slepian-Wolf theorem [SW73].

II. The network $K_{3,2}$

We are given an undirected graph $G = (V, E)$, whose edges have unit capacity. We are also given a set of source-sink

pairs, $\{(s_1,t_1),(s_2,t_2),\cdots,(s_k,t_k)\}$. Associated with each pair we are given an independent random variable which is observed at the source and is required to be communicated to the sink. We are interested in knowing the maximum rate r at which all sources can simultaneously communicate their random variables to their corresponding sinks. We allow the use of Network Coding. From a mathematical point of view we would like to characterize the maximum rate, r. From a computational point of view, we would like to compute this maximum rate, preferably in polynomial time. It is not even known whether the computational question is decidable.

Consider this question when Network Coding is not permitted. In this case we can write a linear program to characterize the maximum rate. This linear program can then be solved in polynomial time. Li and Li [LL04] conjectured that the maximum rate with or without network coding is exactly the same. If this conjecture is true then it answers positively all of the questions stated above.

Since Network Coding only provides extra flexibility, clearly the rate can only be higher when Network Coding is permitted, hence this conjecture requires us to upper bound the rate when Network Coding is allowed. However, there is a lack of effective upperbounding techniques on the rate when Network Coding is allowed.

When Network Coding is not permitted a natural upperbounding quantity is the *Sparcity* of the graph. *Sparcity of a cut* is defined as the number of edges crossing the cut divided by the number of source-sink pairs separated by the cut (if the denominator is zero then we assume the value of the division to be infinity). *Sparcity of the graph* is the minimum sparcity over all the cuts. Clearly sparcity of the graph is an upperbound on the maximum rate but this is not tight even for the case when Network Coding is not allowed.

The simplest counterexample is $K_{3,2}$, the complete bipartite graph with three vertices on one side and two vertices on the other. Let $\{a,b,c\}$ denote the left hand side of the graph and $\{d,e\}$ denote the right hand side. Edges of the graphs are then $\{ad,ae,bd,be,cd,ce\}$. Four source-sink pairs are given; these are $\{(a,b),(b,c),(c,a),(d,e)\}$. We are associating random variables X_1, X_2, X_3 and X_4 with these source sink pairs, respectively.

One can manually check all the cuts to see that the sparcity of this graph is 1. On the other hand the best rate in the absence of Network Coding is only 3/4. One way to observe this is that the shortest path between any source to its sink is two. Hence, every unit of rate will consume eight edges, two per source-sink pair. But we are given only six edges hence the maximum rate could be only 6/8.

To prove the Li-Li conjecture, we will require some way of upperbounding the rate more effectively than sparcity. Note that in the presence of Network Coding, each source-sink pair can consume capacity on the edges on a non-exclusive basis. Hence the 3/4 argument given above does not work when Network Coding is allowed. This seems to be a major stumbling block. We are not aware of any upperbound other than sparcity (Harvey et. al. [HLK2005] introduce meagerness as an upperbound, but in the case of undirected graph, there is no difference between sparcity and meagerness).

On the graph $K_{3,2}$ we can define two qualitatively different networks by changing the role of the source and the sink in a source-sink pair. If the source-sink pairs on $\{a,b,c\}$ form a cycle as defined above, we will call this network $K_{3,2}$ *cyclic*. On the other hand, if we interchange the roles of the source and the sink in the third source-sink pair, i.e., we remove the source-sink pair (c,a) and add the source-sink pair (a,c) we will get a different network. Let us call this the $K_{3,2}$ *acyclic* network.

If network coding is not allowed, then each source-sink pair is utilitzing capacity on an exclusive basis. Hence, it is easy to see that the rate does not change when we change the role of the source and the sink in a source-sink pair (the two linear programs give the same objective function value). However, in the presence of network coding, it is not clear if the rate of the network remains unchanged under this operation. Of course, if the Li-Li conjecture is true then the rate does remain unchanged.

In this paper we develop an information theoretic technique for more effectively upperbounding the rate in the presence of Network Coding. For the network $K_{3,2}$, we show that our upperbound is tight. We first derive a 3/4 bound for the $K_{3,2}$ acyclic network and then for the $K_{3,2}$ cyclic network. We believe that our upperbound is tight even in the most general case.

III. Entropy Calculus

The technique behind our upperbounding comes from entropy calculus. Suppose $T=\{X_1,X_2,\cdots\}$ is a set of random variables. For any set $S\subseteq T$, let $H(S)$ be the joint entropy of the random variables in S. It is well known that H satisfies polymatroid axioms:

1) $H(\emptyset)=0$.
2) $\forall S_1\subseteq S_2,\ H(S_1)\leq H(S_2)$.
3) $\forall S1,S2,\ H(S_1)+H(S_2)\geq H(S_1\cap S_2)+H(S_1\cup S_2)$.

The second axiom is called *monotonicity* and the third is called *submodularity*. Further if the random variables in S_1 are independent of the random variables in S_2 then we also have $H(S_1,S_2)=H(S_1)+H(S_2)$. This axiom is called *independence*. A very useful notation in entropy is *conditional entropy*. The conditional entropy of S_1 subject to S_2 is denoted by $H(S_1/S_2)$ and is equal to $H(S_1,S_2)-H(S_2)$. A well known theorem in entropy calculus is the following: if the variables in S_1 are the functions of the variables in S_2 then the conditional entropy of S_1 subject to S_2 is zero i.e., $H(S_1,S_2)=H(S_2)$ (or simply, $H(S_1,S_2)\leq H(S_2)$, because equality follows from the monotonicity axiom.)

We take every edge of the graph and replace it by two directed edges, one in each direction. The two directed edges corresponding to each undirected edge of the original graph do not have separate capacities instead their total capacity is equal to the original capacity of the edge, which we have assumed w.l.o.g. to be one. We associate one random variable with each directed edge. We also associate one random variable with

each source-sink pair. This random variable can be written as an incoming edge into the source and an outgoing edge from the sink. Now we consider T be the set of all these random variables. We then extend the above entropy calculus over T. Besides these axioms, we develop two kinds of inequalities to extend these axioms. The first we are calling *Input-Output inequalities* and the second *Crypto inequalities*.

IV. The input-output inequality

Consider a cut $S \subseteq V$. Let $\delta_{in}(S)$ be the set of random variables on the edges incoming into S and $\delta_{out}(S)$ be the set of random variables on the edges outgoing from S. Clearly what goes out of S is a function of what S has received so far. One can define a notion of time and formally write a constraint corresponding to this statement. One of our contributions is to not involve time and thereby arrive at simpler, though weaker constraints that still yield the right bound for the $K_{3,2}$ network. We consider the following a weaker statement: what goes out of S is a function of what S has received and what S will be receiving, i.e., what goes out of S is a function of what comes into S. Hence we have the following inequality.

Input-Output inequality:

$\forall S \subseteq V: \ H(\delta_{in}(S), \delta_{out}(S)) \leq H(\delta_{in}(S))$

Remark: One can adapt the proof of this theorem when randomized network coding [J2004] is allowed. For the sake of space the proof is kept for a full version of the paper.

Theorem 1: For the $K_{3,2}$ acyclic network, the maximum rate possible with Network Coding is $3/4$.

Proof : We will associate random variables X_1, X_2, X_3 and X_4 with the source-sink pairs $\{(a,b),(b,c),(a,c),(d,e)\}$, respectively.

Let us write the input-output inequality for the cut $\{d\}$. We get $H(\delta_{in}(d), \delta_{out}(d)) \leq H(\delta_{in}(d))$, i.e., $H(X_4, X_{ad}, X_{bd}, X_{cd}, X_{da}, X_{db}, X_{dc}) \leq H(X_4, X_{ad}, X_{bd}, X_{cd})$.

Similarly we write the input-output inequality for the cut $\{e\}$ (observe that X_4 is an outgoing variable on e whereas it is an incoming variable on d). We get

$H(X_4, X_{ae}, X_{be}, X_{ce}, X_{ea}, X_{eb}, X_{ec}) \leq H(X_{ae}, X_{be}, X_{ce})$.

Adding these two inequalities and applying submodularity we get:

$H(X_4) + H(X_4, X_E) \leq H(X_4, X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce})$,

where X_E is the set of random variables associated with all edges of the network. Applying submodularity once more to cancel out X_4 from both the sides we get:

$H(X_4, X_E) \leq H(X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce})$.

Now applying the input-output inequality at node c to note that both X_2 and X_3 are the functions of X_{dc} and X_{ec}. Since both X_{dc} and X_{ec} are present in the left hand side of the above inequality, using submodularity we can derive,

$H(X_2, X_3, X_4, X_E) \leq H(X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce})$.

Now applying the input-output inequality at node b to note that X_1 is a function of X_2 and X_{db} and X_{eb}, all these three terms are present in the left hand side of the above inequality hence by submodularity we get (here the reason for considering the $K_{3,2}$ acyclic network is clearfied; in $K_{3,2}$ cyclic network, X_1 is a function of X_2 at node b, X_2 is a function of X_3 at node c and X_3 is a function of X_1 at node a):

$H(X_1, X_2, X_3, X_4, X_E) \leq H(X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce})$.

Using monotonicity and independence on the left hand side we get:

$H(X_1)+H(X_2)+H(X_3)+H(X_4) \leq H(X_{ad}, X_{bd}, X_{cd})+ H(X_{ae}, X_{be}, X_{ce})$.

Using submodularity on the right hand side we get:

$H(X_1)+H(X_2)+H(X_3)+H(X_4) \leq H(X_{ad})+H(X_{bd})+ H(X_{cd}) + H(X_{ae}) + H(X_{be}) + H(X_{ce})$.

Similarly by starting on the left partition of the graph we could get:

$H(X_1)+H(X_2)+H(X_3)+H(X_4) \leq H(X_{da})+H(X_{ea})+ H(X_{db}) + H(X_{eb}) + H(X_{dc}) + H(X_{ec})$.

Adding the last two inequalities we get

$2(H(X_1) + H(X_2) + H(X_3) + H(X_4)) \leq \sum_e$ is an edge $H(X_e) \leq 6$.

Hence, $H(X_i) \leq 3/4$. □

V. The crypto inequality

Now let us go back to our original network without changing the roles of any source-sink pair. The only place where the proof fails is when we try to apply the input-output inequality at node a, b or c and we can't effectively apply at any of these nodes first. To circumvent this situation we develop another inequality as follows.

Crypto inequality

$\forall S \subseteq V: \ H(CUT(S), DEM(S)) \leq H(CUT(S))$

where $CUT(S)$ is the set of random variables on edges coming into S and going out of S, excluding the random variables of all the source-sink pairs and $DEM(S)$ is the set of random variables of those source-sink pairs which are separated by S, that is those source-sink pair for which either the source belongs to S and the sink does not or the sink belongs to S and the source does not.

Let us first give an intuitive reason why this inequality must hold. Suppose Alice and Bob have two independent messages which they want to exchange. They follow a protocol and exchange several packets and are able to communicate their messages to each other. Eve is a passive listener. She sees all the packets exchanged between them. We are interested in information-theoretic bounds, so we assume that Eve has unlimited compting power.

The crypto inequality is saying that by looking at the packets exchanged by Alice and Bob, Eve can determine the messages which Alice and Bob exchanged. The reason is these packets must carry Bob's message to Alice and Alice's message to Bob. In the real world, these messages are hidden in computationally hard problems, such as factoring or discrete log. But since Eve lives in an information theory world, one

can't really hide these message behind the curtain of hard problems.

Theorem 2: The crypto inequality holds.

Proof : Let us first prove this inequality for a very simple network. Suppose the network consists of just two nodes, u and v, and one undirected edge (u, v). Further suppose that u wants to communicate a message M_u to v and v wants to communicate a message M_v to u. Assume that M_u and M_v are completely independent messages. Suppose u and v exchange some finite number of packets, say k. Let these be $p_1, p_2, \cdots, p_k$. After the exchange v gets to know M_u and u gets to know M_v. Let P_i denote the set of packets exchanged exchanged till packet i, i.e., $P_i = \{p_1, p_2, \cdots, p_i\}$ (assume $P_0 = \{\}$). We will prove the crypro inequality by induction.

Let us introduce a potential function, which we denote for the i-th step by $I(M_u, P_i; M_v, P_i/P_i) = H(M_u, P_i) + H(M_v, P_i) - H(M_u, M_v, P_i) - H(P_i)$. For our proof this syntatic definition is sufficient. For a greater understanding let us provide a semantics of this. $I(M_u, P_i; M_v, P_i/P_i)$ denotes the amount of information which both u and v have but P_i does not have. Intially at time $i = 0$, we have $I(M_u, P_0; M_v, P_0/P_0) = H(M_u) + H(M_v) - H(M_u, M_v)$. Since M_u and M_v are independent we have $H(M_u, M_v) = H(M_u) + H(M_v)$. Hence at $i = 0$ our potential function is 0.

Now let us prove by induction that it remains 0. Suppose consider time $i + 1$. A packet p_{i+1} is communicated. W.l.o.g. assume that this packet was sent from u to v. From our input-output inequality we have: $H(p_{i+1}, P_i, M_u) \leq H(P_i, M_u)$. By induction we also have $H(M_u, P_i) + H(M_v, P_i) \leq H(M_u, M_v, P_i) + H(P_i)$ (Note that because of submodularity this is the same as saying that $I(M_u, P_i; M_v, P_i/P_i) = 0$). Adding these two inequalities we get $H(p_{i+1}, P_i, M_u) + H(M_v, P_i) \leq H(M_u, M_v, P_i) + H(P_i)$. Note that we can add p_{i+1} in the first term on the right hand side too. Hence we get, $H(p_{i+1}, P_i, M_u) + H(M_v, P_i) \leq H(M_u, M_v, P_i, p_{i+1}) + H(P_i)$. Again using submodularity we can also add p_{i+1} in the remaining two terms of this inequality. Hence we get $H(M_u, P_{i+1}) + H(M_v, P_{i+1}) \leq H(M_u, M_v, P_{i+1}) + H(P_{i+1})$. This is the same as saying $I(M_u, P_{i+1}; M_v, P_{i+1}/P_{i+1}) = 0$.

By induction we then have, $I(M_u, P_k; M_v, P_k/P_k) = 0$ i.e., $H(M_u, P_k) + H(M_v, P_k) \leq H(M_u, M_v, P_k) + H(P_k)$. At the end of the communication of the k-th packet, u has M_v and v has M_u. This can be written as $H(M_u, P_k) = H(M_v, M_u, P_k) = H(M_v, P_k)$. This together with our potential inequality gives us: $H(M_u, M_v, P_k) \leq H(P_k)$. Hence the crypto inequality is proved for this simple network.

We could generalize this proof to arbitrary networks. For the sake of space we are keeping this generalization for a full version of the paper.

□

Remark: Crypto inequality holds even if we allow randomized protocols. For the sake of space the proof is kept for a full version of the paper.

We next use the crypto inequality to derive a tight bound for the $K_{3,2}$ cyclic network.

Theorem 3: For the $K_{3,2}$ cyclic network, the maximum rate possible with Network Coding is $3/4$.

Proof : We will associate random variables X_1, X_2, X_3 and X_4 with the source-sink pairs $\{(a, b), (b, c), (c, a), (d, e)\}$, respectively. As in the proof of Theorem 1 we apply the input-output inequality for cuts $\{d\}$ and $\{e\}$ first to obtain:

$$H(X_2, X_3, X_4, X_E) \leq H(X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce}).$$

Observe that whereas we cannot apply the input-output inequality at any of the nodes a, b, and c, we can apply the crypto inequality at any of these nodes. Let us apply it at node c:

$H(X_2, X_3, X_{dc}, X_{cd}, X_{ec}, X_{ce}) \leq H(X_{dc}, X_{cd}, X_{ec}, X_{ce})$.

Adding this to the inequality we have already derived and using submodularity we get:

$H(X_2, X_3, X_4, X_E) \leq H(X_{ad}, X_{bd}, X_{cd}) + H(X_{ae}, X_{be}, X_{ce})$, where X_E is the set of random variables associated with all edges of the graph. This is the next step of the proof of Theorem 1. At this point, we can continue with the rest of that proof. □

VI. A BOUND FOR DIRECTED BIPARTITE NETWORKS

We use the techniques developed above to prove the following theorem. Observe that Theorems 1 and 3 can also be seen as a corollaries of Theorem 4.

Theorem 4: Let $G(U.V, E)$ be a directed bipartite graph with bipartition (U, V) in which k_1 (k_2) edges are directed from U to V (from V to U). Furthermore, assume that there are n_1 (n_2) source sink pairs that have source in U (V) and sink in V (U), and n source-sink pairs that are one the same side of the bipartition (either both source-sink in U or both in V). Then, the maximum information rate in G is bounded by:

$$\min\left(\frac{k_1}{n + n_1}, \frac{k_2}{n + n_2}\right).$$

Proof : Let us apply the input-output inequality on each node v on the V side. We get:

$$\forall v \in V: \; H(source_v, sink_v, \delta_{in}(v), \delta_{out}(v)) \leq H(source_v, \delta_{in}(v)),$$

where by $source_v$ we mean the random processes of all source-sink pairs which have v as its source. Similarly define $sink_v$. $\delta_{in}(v)$ denotes the random variables on the incoming edges of the bipartite graph into v. Similarly define $\delta_{out}(v)$.

Now consider the vertices of V in any order. Start with the inequality of the first vertex. Add into it the inequality of the second vertex. Apply submodularity. This generates an intersection term and a union term. Now add into it the inequality of the third vertex. Call the left hand side of the third inequality the new term. Apply submodularity between the union term (from the previous application of submodularity)

and the new term (introduced by the inequality of the next vertex). This will again generate a union term. Continue in this fashion. We will eventually get:

$$\sum_{X_i \in Source_V \cap Sink_V} H(X_i) + H(Source_V, Sink_V, X_E) \leq \sum_{v \in V} H(source_v, \delta_{in}(v),$$

where $Source_V$ is the set of pairs which have source in V, $Sink_V$ is the set of pairs which have sink in V, and X_E is the set of random variables assiciated with all edges of G. We simplified all the intersection terms generated because the random variables for all the source-sink pairs are independent. Now we apply submodularity to simplify the right hand side summation too. We get:

$$\sum_{X_i \in Source_V \cap Sink_V} H(X_i) + H(Source_V, Sink_V, X_E) \leq \sum_{X_i \in Source_V} H(X_i) + \sum_{e \in U \times V} H(X_e).$$

This gives us:

$$H(Source_V, Sink_V, X_E) \leq \sum_{X_i \in Source_V - Sink_V} H(X_i) + \sum_{e \in U \times V} H(X_e).$$

Now look at the left hand side term. It has the random variables of all the edges. Consider every vertex, u, of U one by one. Apply the crypto inequality on the signleton cut $\{u\}$. Random variables on all the edges in this cut implies the random variables in $Source_u$, which u was trying to communicate with the rest of the vertices. Hence we can strengthen the above inequality as:

$$H(X_R, X_E) \leq \sum_{X_i \in Source_V - Sink_V} H(X_i) + \sum_{e \in U \times V} H(X_e),$$

where X_R is the set of random variables associated with all the sources in our network. Applying monotonocity to this inequality we get:

$$H(X_R) \leq \sum_{X_i \in Source_V - Sink_V} H(X_i) + \sum_{e \in U \times V} H(X_e).$$

Using the fact that all sources are independent we get:

$$\sum_{X_i \in X_R} H(X_i) \leq \sum_{X_i \in Source_V - Sink_V} H(X_i) + \sum_{e \in U \times V} H(X_e).$$

Cancelling the first set of terms of the right hand side gives us:

$$\sum_{X_i \in X_R - (Source_V - Sink_V)} H(X_i) \leq \sum_{e \in U \times V} H(X_e).$$

Note that the number of edges on the right hand side is k_1. Let us carefully count the number of terms in the first summation (note that $X_R - (Source_V - Sink_V) = X_R - Source_V + Sink_V$ does not make any sense). Note that since $(Source_V - Sink_V) \subseteq X_R$, we get $|X_R - (Source_V - Sink_V)| = |X_R| - |(Source_V - Sink_V)|$. We know that $|X_R| = n + n_1 + n_2$.

Now let us count the number of variables in $Source_V - Sink_V$. $Source_V - Sink_V = Source_V - (Source_V \cap Sink_V)$, i.e., those terms which have sources in V but sink in U. The number of such terms is n_2. Hence the number of terms on the left hand side of this inequality is $n + n_1$. This gives us that the information rate is bounded above by $k_1/(n + n_1)$. Interchaning the roles of U and V we get another upper bound $k_2/(n + n_2)$, hence proving the theorem.

□

Remark: In the recent Network Coding workshop at DIMACS we learned that Harvey et. al. [HLK2005] independently have similar results through a different kind of inequality, which they call down-stream inequality. Based on their talk at DIMACS, we would like to point out that down-stream inequality is implied by our input-output inequality and crypto inequality together. The converse is not yet clear. They also presented examples of graphs, on which they obtain the tight results. Those examples can also be solved by our directed bipartite graph theorem.

VII. Acknowledgment

We like to thank Christian Borgs, Jennifer Chayes, Uriel Feige, Andras Frank, Laszlo Lovasz, Nicole Immorlica, Mohammad Mahdian, Aranyak Mehta, Peter Montogomory, Amin Saberi and Kunal Talwar for valuable discussions. We specially thanks to Philip A. Chou for the guidance he provided.

References

[ACLY00] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Trans. on Information Theory*, 46:1204–1216, 2000.

[CA04] M. Charikar and A. Argawal. On the advantage of network coding for improving network throughput. In *Proceedings, 2004 IEEE Information Theory Workshop, San Antonio*, 2004.

[HLK2005] N. J. A. Harvey and R. D. Kleinberg and A. R. Lehman. On the Capacity of Information Networks. Presentation at DIMACS workshop on Network Coding. Jan 28, 2005.

[JSC+] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen. Polynomial time algorithms for multicast network code construction. *IEEE Transactions on Information Theory*.

[J2004] Kamal Jain. Security Based on Network Topology Against the Wiretapping Attack. IEEE Magazine, Special issue on Topics in Wireless Security, 2004.

[KM] R. Koetter and M. Medard. An algebraic approach to network coding. *Transactions on Networking*.

[LL04] Z. Li and B. Li. Network coding in undirected networks. In *Proceedings, CISS*, 2004.

[LR88] F.T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with application to approximation algorithms. In *Proceedings, 29th Annual IEEE Symposium on Foundations of Computer Science*, pages 422–431, 1988.

[SW73] D. Slepian and J. K. Wolf. Noiseless Coding of Correlated Information Sources. IEEE Trans. Inform. Theory. vol. IT-19, pp. 471–480, Mar. 1973.

[Vaz01] V. V. Vazirani. *Approximation Algorithms*. Springer-Verlag, 2001.