A “Chicken & Egg” Network Coding Problem

Nicholas J.A. Harvey
MIT

Robert Kleinberg
Cornell

Chandra Nair
Microsoft

Yunnan Wu
Microsoft

***Abstract*— We consider the multi-source network coding problem in cyclic networks. This problem involves several difficulties not found in acyclic networks, due to additional causality requirements. This paper highlights the difficulty of these causality conditions by analyzing two example cyclic networks. The networks appear quite similar at first glance, and indeed both have invalid rate-1 network codes that violate causality. However, the two networks are actually quite different: one also has a valid rate-1 network code obeying causality, whereas the other does not. This unachievability result is proven by a new information inequality for causal coding schemes in a simple cyclic network.**

1. Introduction

The multi-source network coding problem, where multiple communicating sessions share a network of lossless links with rate constraints, has proven to be a challenging problem involving many subtle and counterintuitive phenomena. These difficulties are present in both acyclic and cyclic networks, although cyclic networks are generally harder, due to the additional causality issues. This paper studies two example cyclic networks that illustrate an interesting “chicken and egg” phenomenon.

To explain our work, we begin with some informal definitions; more formal definitions are given in Section 1-A. A natural way to specify a coding function in a network is to associate a single symbol to be transmitted with each edge of the network; these symbols are simply functions of the sources’ data. We say that such a code meets the *single-letter criterion for validity* if the information on edges leaving a vertex can be computed from the information on edges entering a vertex. For acyclic networks, a code satisfying the single-letter criterion is indeed a valid solution. However, the problem is more complicated when there are cycles. For a cyclic network, a code is called *valid* only if there is a sequence of messages corresponding to the associated symbols, such that each message leaving a vertex can computed from *previous* messages entering the vertex. Generally, the single-letter criterion does not imply validity because there may be circular dependencies among edge symbols which can not be resolved into a valid sequence of transmissions; we describe such invalid coding solutions as “*chicken and egg*” solutions. The problem of determining when a single-letter-valid code is valid can be quite subtle; this note presents an example illustrating the subtlety.

Our example comprises the networks $\mathcal{G}_1$ and $\mathcal{G}_2$, shown in Figure 1 and Figure 2. Figure 3 shows network codes for these networks meeting the single-letter criterion for validity. For $\mathcal{G}_2$, a valid rate-1 coding solution is presented in Section 3. For $\mathcal{G}_1$, it is shown via information theoretic inequalities that any valid coding solution has rate at most $4/5$. For the sake of comparison,

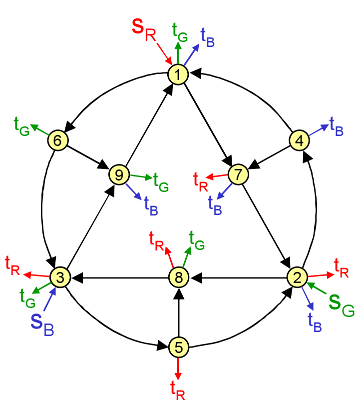


Fig. 1. $\mathcal{G}_1$, the first communication problem studied in this paper.

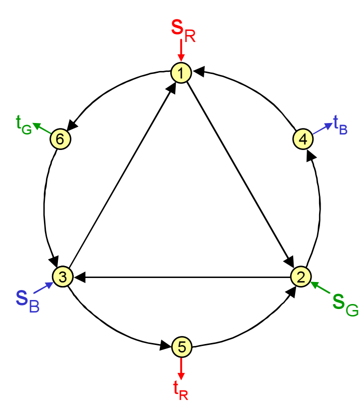


Fig. 2. $\mathcal{G}_2$, the second communication problem studied in this paper.

a standard volume argument shows that the maximum routing rate in $\mathcal{G}_1$ is at most $2/3$, whereas for $\mathcal{G}_2$ it is at most $3/4$.

In proving our unachievability result for $\mathcal{G}_1$, a new single-letter information inequality is proven for the simplest cyclic network structure — two edges with opposite directions. Similar to $\mathcal{G}_1$, this inequality appears to have a counterexample, whose flaw is revealed only by expanding the network as transmissions over a sequence of time steps. The moral of this analysis is that analyzing networks in time-dimension is crucial for establishing tight converse theorems for multi-source network coding.

There is much prior work considering multi-source network coding in cyclic networks, e.g., [7], [1], [4], [5], [6], [3]. The treatment of causality in these prior works focused on single-letter information inequalities expressing the fact that sources can be completely recovered from the information contained in an edge set when that edge set separates the source from the sink in a suitably strong sense. One key innovation in the present work is a new information inequality which can be used to preclude chicken-and-egg solutions and yet is derived by using a technique different from the prior art.

A. *Definitions and Notation*

Networks $\mathcal{G}_1$ and $\mathcal{G}_2$ have three information streams (or commodities), which we denote by R, G and B. Each information stream, say R, is produced at its source node, which is denoted s_R. A source can be viewed as a process which produces a random variable at each time step. The objective is to transmit the information from the sources to their respective sinks, which are denoted t_R, etc.

An edge in the network is denoted e_{ij}, where i is the tail and j is the head of the edge. Suppose that the information is transmitted through the network for n consecutive time steps. Let $e_{ij}^{(k)}$ denote the random variable corresponding to the symbol transmitted on edge e_{ij} at time step k. Let $e_{ij}^{(1..k)} = \{e_{ij}^{(1)}, \ldots, e_{ij}^{(k)}\}$. For convenience, we use $\mathbf{e}_{ij}$ as shorthand for $e_{ij}^{(1..n)}$. We assume that each edge has unit capacity, meaning that $H(e_{ij}^{(k)}) \leq 1$.

We restrict our attention to deterministic coding. That is, for an arbitrary edge e whose tail has sources $S_1, \ldots, S_n$ and has incoming edges $e_1, \ldots, e_m$, we require the output at time k on edge e to be a function of those sources and the past symbols on the incoming edges. As an equation, we have

$$e^{(k)} = g\left(S_1, \ldots, S_n, e_1^{(1..k-1)}, \ldots, e_m^{(1..k-1)}\right). \quad (1.1)$$

This requirement also applies when e is entering a sink.

It can be argued that a randomized coding scheme, where $e^{(k)}$ in Eq. (1.1) can be a function of other source of randomness, does not enlarge the rate regions of the problem. An example argument can be found in the Appendix of Ahlswede et al. [2]. The probability of error of a randomized scheme is the average probability of error over the difference realizations of the additional independent sources of randomness. As a result, there must exist a deterministic scheme that drives the error probability to zero. Thus we will henceforth assume that the only sources of randomness are R, G, and B.

That being said, for simplicity, this paper considers only zero-error solutions, i.e., the sinks must exactly recover the sources. * A coding solution achieves rate r if it transmits the sources to the sinks with zero error for n time steps and $H(s_R) = H(s_G) = H(s_B) = rn$.

2. Analysis of $\mathcal{G}_1$

In this section, we will show that any valid rate-r network code in $\mathcal{G}_1$ has $r \leq 4/5$. To this end, suppose we have such a network code with $r \geq 1 - \lambda$. We now derive two inequalities on the information transmitted through the network.

Lemma 2.1: The following inequalities hold:

$$I(\mathbf{e}_{35}; GB|R) \leq \lambda n \qquad I(\mathbf{e}_{58}\mathbf{e}_{28}; B|RG) \leq 2\lambda n$$

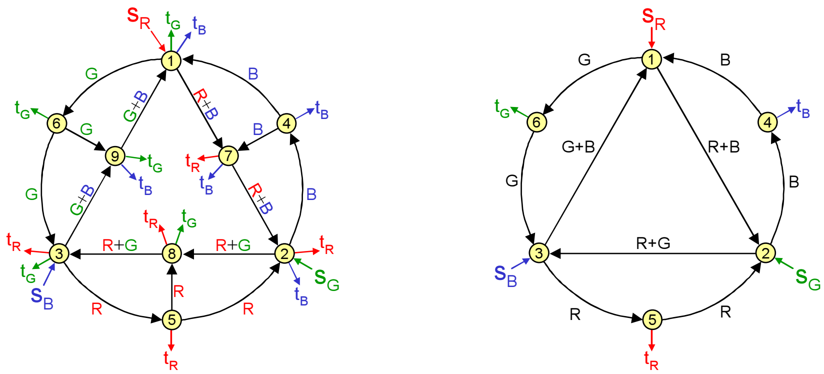
Proof. Fano's inequality (along with zero error criterion) implies that $I(R; \mathbf{e}_{35}) = n(1-\lambda)$ and $I(RG; \mathbf{e}_{58}\mathbf{e}_{28}) = 2n(1-\lambda)$. Now

$$\begin{aligned} \underbrace{I(RGB; \mathbf{e}_{35})}_{\leq H(\mathbf{e}_{35}) = n} &= \underbrace{I(R; \mathbf{e}_{35})}_{=n(1-\lambda)} + I(GB; \mathbf{e}_{35}|R) \\ \implies I(GB; \mathbf{e}_{35}|R) &\leq \lambda n. \end{aligned}$$

$$\begin{aligned} &\underbrace{I(RGB; \mathbf{e}_{58}\mathbf{e}_{28})}_{\leq H(\mathbf{e}_{58}\mathbf{e}_{28}) = 2n} \\ &\quad = \underbrace{I(RG; \mathbf{e}_{58}\mathbf{e}_{28})}_{=2n(1-\lambda)} + I(B; \mathbf{e}_{58}\mathbf{e}_{28}|RG) \\ &\implies I(B; \mathbf{e}_{58}\mathbf{e}_{28}|RG) \leq 2\lambda n. \qquad \blacksquare \end{aligned}$$

Next, from the graph structure, without loss of generality, we can assume that $\mathbf{e}_{58}$ and $\mathbf{e}_{52}$ are delayed versions of $\mathbf{e}_{35}$, i.e., $e_{58}^{(k)} = e_{52}^{(k)} = e_{35}^{(k-1)}$. This is because $\mathbf{e}_{35}$ is the only input for $\mathbf{e}_{58}$ and $\mathbf{e}_{52}$ and their capacity are the same. Any operations at $\mathbf{e}_{58}$ can be deferred to a later point. Hence any solution can be transformed into a solution where $e_{58}^{(k)} = e_{52}^{(k)} = e_{35}^{(k-1)}$. This observation and Lemma 2.1 together imply that $I(\mathbf{e}_{52}; GB|R) \leq \lambda n$ and $I(\mathbf{e}_{52}\mathbf{e}_{28}; B|RG) \leq 2\lambda n$.

*Using Fano's inequality it is quite straightforward to convert these results to the more conventional asymptotically zero error coding schemes.



(a) $\mathcal{G}_1$ (b) $\mathcal{G}_2$

Fig. 3. An invalid "coding solution" for the two instances $\mathcal{G}_1$ and $\mathcal{G}_2$. The solution appears to have rate 1 but, due to causality issues, it cannot be implemented over a sequence of time steps.

A. Reduction to a Smaller Instance

In this section we modify the graph $\mathcal{G}_1$ by simplifying it to a smaller instance $\mathcal{G}_1^*$. (See Figure 4.) We identify the nodes in each of the following sets: $\{1,6,9\}$, $\{2,4,7\}$, $\{3,5,8\}$. (Effectively, the edges induced by those vertex sets have been given infinite capacity and zero delay.) The resulting "supernodes" will henceforth be denoted 1, 2, and 3.

Any network coding solution of rate r in $\mathcal{G}_1$ can obviously be viewed as a solution of rate r in $\mathcal{G}_1^*$, since the connectivity has improved whereas the communication requirements have not changed. Furthermore, when we map a solution from $\mathcal{G}_1$ to $\mathcal{G}_1^*$, Lemma 2.1 implies (via symmetry) that the following inequalities hold:

$$\begin{array}{ll} I(\mathbf{e}_{32};GB|R) \le \lambda n & I(\mathbf{e}_{23}\mathbf{e}_{32};B|RG) \le 2\lambda n \\ I(\mathbf{e}_{13};RB|G) \le \lambda n & I(\mathbf{e}_{31}\mathbf{e}_{13};R|BG) \le 2\lambda n \\ I(\mathbf{e}_{21};RG|B) \le \lambda n & I(\mathbf{e}_{12}\mathbf{e}_{21};G|RB) \le 2\lambda n \end{array} \tag{2.1}$$

B. Analysis

The argument proceeds as follows. Lemma 2.6 is an important result whose statement and proof is deferred to the end of the analysis to retain the flow. In Lemma 2.2, we apply this general inequality to the particular example. In the subsequent lemmas, we obtain lower bounds and upper bounds of relevant terms as a function of λ. This culminates in a lower bound on λ in Lemma 2.5.

Lemma 2.2:

$$I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|G) \;\ge\; I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|G\mathbf{e}_{12}).$$

Proof. Apply Lemma 2.6 with $a = \mathbf{e}_{23}$, $b = \mathbf{e}_{32}$, $x = \mathbf{e}_{12}$, $x' = G$, $y = B\mathbf{e}_{13}$, $y' = 0$. ∎

Lemma 2.3: $I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|G) \le 3\lambda n$.

Proof. Using the inequality $I(X;Y) \le I(Y;Z) + I(X;Y|Z)$, we have

$$\begin{aligned} &I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|G) \\ &\quad\le\; I(R;B\mathbf{e}_{13}|G) \;+\; I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|RG). \end{aligned} \tag{2.2}$$

The first term in Eq. (2.2) is upper bounded as follows.

$$\begin{aligned} I(R;B\mathbf{e}_{13}|G) \;&=\; \underbrace{I(R;B)}_{=0} + I(R;\mathbf{e}_{13}|GB) \\ &\le\; I(RB;\mathbf{e}_{13}|G) \;\le\; \lambda n, \end{aligned}$$

where the first inequality follows from the inequality $I(W;XY) \ge I(W;X|Y)$ and the second is due to 2.1.

To upper bound the second term in Eq. (2.2), we write it as

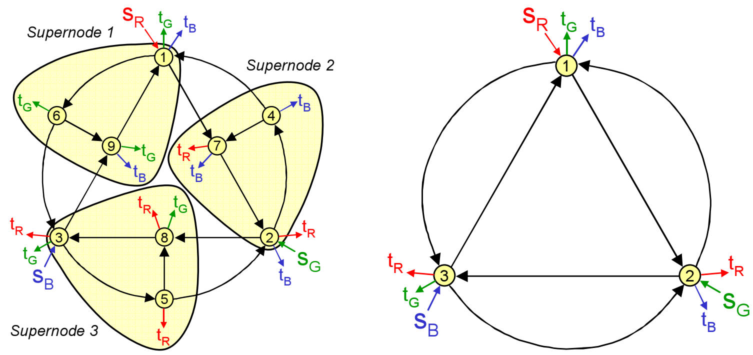
$$\begin{aligned} &I(\mathbf{e}_{23}\mathbf{e}_{32};B\mathbf{e}_{13}|RG) \\ &\quad=\; I(\mathbf{e}_{23}\mathbf{e}_{32};B|RG) \;+\; I(\mathbf{e}_{23}\mathbf{e}_{32};\mathbf{e}_{13}|RGB). \end{aligned}$$

Clearly $I(\mathbf{e}_{23}\mathbf{e}_{32};\mathbf{e}_{13}|RGB) = 0$, since no entropy remains after conditioning on all sources. Now by 2.1, $I(\mathbf{e}_{23}\mathbf{e}_{32};B|RG) \le 2\lambda n$. Thus the second term in Eq. (2.2) is at most $3\lambda n$. ∎

Lemma 2.4:

$$I(\mathbf{e}_{23},\mathbf{e}_{32};\mathbf{e}_{13}B|\mathbf{e}_{12}G) \ge n - 2\lambda n.$$

Proof. Since decoding error is zero, we have $H(RB|\mathbf{e}_{12},\mathbf{e}_{32},G) = 0$; and we know that $H(\mathbf{e}_{12},\mathbf{e}_{32}|G) \ge 2n(1-\lambda)$. Now



(a) $\mathcal{G}_1$ (b) $\mathcal{G}_1^*$

Fig. 4. (a) Identifying vertices in $\mathcal{G}_1$. (b) The resulting instance $\mathcal{G}_1^*$.

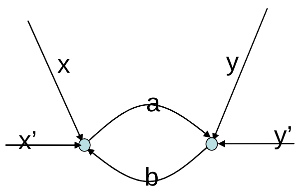


Fig. 5. The graph structure for Lemma 2.6.

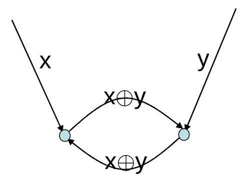


Fig. 6. An invalid counter example to Lemma 2.6.

$$
\begin{aligned}
&2n(1-\lambda)\\
&\quad \leq H(\mathbf{e}_{12}|G) + H(\mathbf{e}_{32}|\mathbf{e}_{12}, G)\\
&\quad \leq n + H(\mathbf{e}_{32}, \mathbf{e}_{23}|\mathbf{e}_{12}, G)\\
&\quad = n + I(\mathbf{e}_{23}, \mathbf{e}_{32}; \mathbf{e}_{13}, B|\mathbf{e}_{12}, G)\\
&\qquad + H(\mathbf{e}_{23}, \mathbf{e}_{32}|\mathbf{e}_{13}, B, \mathbf{e}_{12}, G)\\
&\quad = n + I(\mathbf{e}_{23}, \mathbf{e}_{32}; \mathbf{e}_{13}, B|\mathbf{e}_{12}, G) + 0. \qquad \blacksquare
\end{aligned}
$$

Lemma 2.5: $\lambda \geq 1/5$.

Proof. From Lemma 2.2, Lemma 2.3, and Lemma 2.4, we have $3\lambda n \geq n - 2\lambda n$. Thus $\lambda \geq 1/5$. ■

C. An Useful Lemma

Lemma 2.6: Suppose that a, b, x, y, x', y' are random variables corresponding to the graph structure in Figure 5. Then for any deterministic coding scheme, we have

$$I(a, b; x, y|x', y') \leq I(a, b; x|x', y') + I(a, b; y|x', y').$$

Note that equivalent forms of this inequality include:

$$
\begin{aligned}
I(a, b; x|x', y') &\geq I(a, b; x|x', y', y),\\
I(a, b; y|x', y') &\geq I(a, b; y|x', y', x).
\end{aligned}
$$

Discussion: Before proving Lemma 2.6, let us first examine a well-known example, shown in Figure 6, illustrating that the lemma is not trivial. Here suppose x and y are independent binary strings, and $x' = 0$, $y' = 0$. This network code appears to be locally consistent if we neglect the causality requirement: each node appears to be able to generate $x \oplus y$ from its inputs. The proof of Lemma 2.6 resolves the causality problem of this example by explicitly considering the time dimension.

Proof (of Lemma 2.6). Expand the random variables in $I(xy; ab|x'y')$ into their components at each time step.

$$I(xy; ab|x'y') = \sum_{k=1}^{n} I(xy; a^{(k)}b^{(k)}|a^{(1..k-1)}b^{(1..k-1)}x'y').$$

This can be divided into the following two sums:

$$
\begin{aligned}
&\sum_{k=1}^{n} I(xy; a^{(k)}|a^{(1..k-1)}b^{(1..k-1)}x'y')\\
&+ \sum_{k=1}^{n} I(xy; b^{(k)}|a^{(1..k)}b^{(1..k-1)}x'y').
\end{aligned} \tag{2.3}
$$

We now analyze the first sum.

$$\sum_{k=1}^{n} I(xy; a^{(k)}|a^{(1..k-1)}b^{(1..k-1)}x'y') \tag{2.4}$$

$$\geq \sum_{k=1}^{n} I(x; a^{(k)}|a^{(1..k-1)}b^{(1..k-1)}yx'y') \tag{2.5}$$

$$\geq \sum_{k=1}^{n} I(x; a^{(k)} | a^{(1..k-1)} y x' y') \quad (2.6)$$

$$= I(x; a | y x' y') \quad (2.7)$$

$$= I(x; ab | y x' y') \quad (2.8)$$

where Eq. (2.5)≥Eq. (2.6) holds because $b^{(1..k-1)}$ is a function of $a^{(1..k-1)}$, y and y', and Eq. (2.7)=Eq. (2.8) holds because b is a function of a, y and y'. This gives a lower bound on the first sum in Eq. (2.3). A similar argument applies to the second sum (note that $a^{(1..k)}$ is a function of $b^{(1..k-1)}$, x and x') , so we obtain

$$I(xy; ab|x'y') \geq I(x; ab|yx'y') + I(y; ab|xx'y').$$

Since $I(xy; ab|x'y') = I(x; ab|x'y') + I(y; ab|xx'y')$, we have shown that $I(x; ab|x'y') \geq I(x; ab|yx'y')$, as required. ■

3. Analysis of $\mathcal{G}_2$

To illustrate the subtle issues at play in the "chicken and egg" phenomenon, we now demonstrate that rate 1 is asymptotically achievable in the communication problem $\mathcal{G}_2$, despite its superficial similarity with $\mathcal{G}_1$ and the fact that the invalid "coding solutions" for both problems in Figure 3 are virtually identical. Our solution which asymptotically achieves rate 1 in $\mathcal{G}_2$ reveals a key difference between the two communication problems. There is a *second* "chicken-and-egg" solution in $\mathcal{G}_2$, in which each of the edges $\mathbf{e}_{12}, \mathbf{e}_{23}, \mathbf{e}_{31}$ transmits the message $R \oplus G \oplus B$. Unlike the invalid coding solution presented in Figure 3(b), this invalid solution can be "unraveled" into a valid coding solution. (Note that the comparable modification of the coding solution in Figure 3(a) — i.e., transmitting $R \oplus G \oplus B$ on each of the edges $\mathbf{e}_{17}, \mathbf{e}_{72}, \mathbf{e}_{28}, \mathbf{e}_{83}, \mathbf{e}_{39}, \mathbf{e}_{91}$ — produces a solution which does not even meet the single-letter criterion for validity, e.g. node 8 does not receive sufficient information to output message G to sink t_G.)

Let n be any positive integer. Suppose that sources R, G, B generate independent uniformly-distributed n-bit messages $r^{(1..n)}, g^{(1..n)}, b^{(1..n)}$. We first describe a solution which transmits R, G, B over a series of $9n+2$ rounds; in each round exactly one of the edges of the network transmits a single bit and the other edges are idle. Let us adopt the convention that $r^{(0)} = b^{(0)} = g^{(0)} = 0$.

Rounds $1, \ldots, 3n$: For $0 \leq i < n$,

$$\begin{aligned} e_{12}^{(3i+1)} &= r^{(i+1)} \oplus g^{(i)} \oplus b^{(i)} \\ e_{23}^{(3i+2)} &= r^{(i+1)} \oplus g^{(i+1)} \oplus b^{(i)} \\ e_{31}^{(3i+3)} &= r^{(i+1)} \oplus g^{(i+1)} \oplus b^{(i+1)} \end{aligned}$$

Rounds $3n+1, 3n+2$:

$$e_{35}^{(3n+1)} = e_{52}^{(3n+2)} = b^{(n)}$$

Rounds $3n+3, \ldots, 9n+2$: For $0 \leq i < n$, let $j = 6i + 3n + 3$.

$$\begin{aligned} e_{24}^{(j)} &= e_{41}^{(j+1)} = b^{(n-i)} \\ e_{16}^{(j+2)} &= e_{63}^{(j+3)} = g^{(n-i)} \\ e_{35}^{(j+4)} &= e_{52}^{(j+5)} = r^{(n-i)} \end{aligned}$$

It is easy to check that this satisfies the definition of a coding solution, i.e. every message transmitted along an edge $e = (u, v)$ is a function of the messages received earlier at u together with the sources originating at u. Over the course of $9n + 2$ rounds, edges $\mathbf{e}_{35}, \mathbf{e}_{52}$ each transmit $n + 1$ bits and the remaining edges each transmit n bits. Using a standard interleaving trick, we can eliminate the idle periods on edges and achieve a rate approaching 1 as n tends to infinity. More precisely, for any positive integer m we can construct a network code in which each source generates $(9n + 2)nm$ bits and these are transmitted to the sinks over the course of $(9n + 2)(n + 1)(m + 1)$ time steps, with each edge sending at most one bit per time step.† Thus there is a coding solution achieving rate $\left(\frac{n}{n+1}\right)\left(\frac{m}{m+1}\right)$, which approaches 1 as n, m simultaneously tend to infinity.

References

[1] M. Adler, N. J. A. Harvey, K. Jain, R. Kleinberg, and A. R. Lehman. On the capacity of information networks. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, Jan. 2006.

[2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, 2000.

[3] N. J. A. Harvey, K. Jain, L. C. Lau, C. Nair, and Y. Wu. Conservative network coding. In *44th Annual Allerton Conference on Communication, Control, and Computing*, Sept. 2006.

[4] N. J. A. Harvey, R. Kleinberg, and A. R. Lehman. On the capacity of information networks. *IEEE Transactions on Information Theory*, 52(6):2345–2364, June 2006.

[5] K. Jain, V. Vazirani, and G. Yuval. On the capacity of multiple unicast sessions in undirected graphs. *IEEE Transactions on Information Theory*, 52(6):2805–2809, June 2006.

[6] G. Kramer and S. Savari. Edge-cut bounds on network coding rates. *Journal of Network and Systems Management*, 14(1), 2006.

[7] R. Yeung. *A First Course in Information Theory*. Springer, 2006.

†Treat the ordered triple of $(9n + 2)nm$-bit source messages as $9n+2$ separate triples of nm-bit messages, numbered $1, 2, \ldots, 9n+2$. Each of these triples of nm-bit messages can be transmitted over a sequence of $(9n + 2)m$ rounds, using the coding solution specified above repeated m times sequentially. We divide time into phases numbered $1, 2, \ldots, (9n + 2)(m + 1)$. In phase p, for each $s = 1, 2, \ldots, 9n+2$, each edge participates in round $p - s$ of the protocol for sending the s^{th} triple of messages; this does not require transmitting any bits if $p - s \leq 0$ or if the edge is idle in round $p - s$ of the protocol for sending the s^{th} triple of messages. In a given phase, each edge transmits at most $n+1$ bits and these bits depend only on information which is received in prior phases or which originates at the tail of the edge. Thus all $(9n+2)(m+1)$ phases can be scheduled in a sequence of $(9n+2)(n+1)(m+1)$ time steps, without violating any causality or edge capacity constraints.