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 which are structurally similar. Both networks have an essentially identical network code which appears to transmit all information from the sources to the sinks; however, this network code is invalid since it violates causality. We show that, in one of the networks, the invalid code can be modified to obey causality, whereas in the other network this is impossible. 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, is 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 essentially identical 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, a standard volume argument shows that the maximum routing rate (i.e., without coding) 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

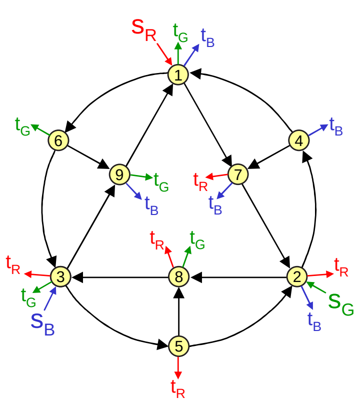


Fig. 1. $\mathcal{G}_1$, the first communication problem studied in this paper.

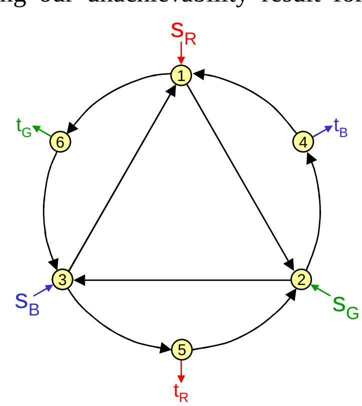
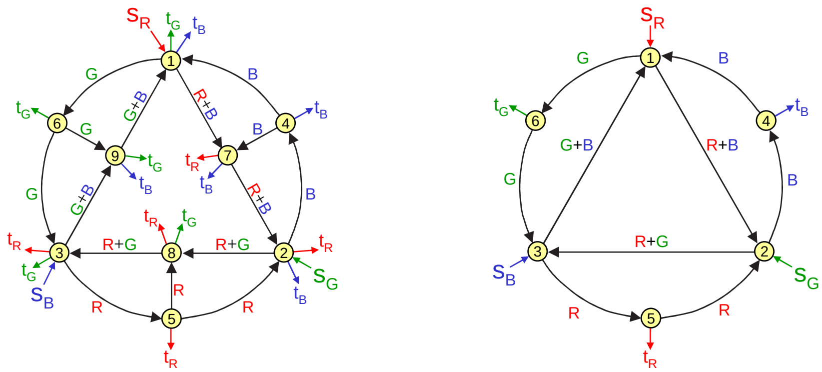


Fig. 2. $\mathcal{G}_2$, the second communication problem studied in this paper.



(a) $\mathcal{G}_1$ (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.

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 communication problems in cyclic networks, e.g., [7], [9], [1], [4], [5], [6], [3]. The treatment of causality in these prior works focused on single-letter information inequalities capturing the following fact: sources can be decoded from the information transmitted on a set of edges when those edges separate the source from the sink in a suitably strong sense. One key innovation in our work is a new information inequality which can be used to preclude chicken-and-egg solutions and 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 $e_{ij}^{(1..k)} = \{e_{ij}^{(1)}, \dots, e_{ij}^{(k)}\}$. 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 $S_1, \dots, S_a$ and has incoming edges $e_1, \dots, e_b$, 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, for some function g,

$$e^{(k)} = g\left(S_1, \dots, S_a, e_1^{(1..k-1)}, \dots, e_b^{(1..k-1)}\right). \quad (1.1)$$

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 [2] or [8]. 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 show that any valid rate-r network code in $\mathcal{G}_1$ has $r \leq 4/5$. Before doing so, we present the information inequality at the crux of our proof.

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

A. An Information Inequality

Lemma 2.1: Suppose that a, b, x, y are random variables corresponding to the network structure in Figure 4. (The variables x and y may be arbitrarily correlated.) Then for any deterministic coding scheme, we have

$$I(ab; xy) \leq I(ab; x) + I(ab; y).$$

Equivalent forms of this inequality include:

$$I(ab; x) \geq I(ab; x|y) \quad \text{and} \quad I(ab; y) \geq I(ab; y|x).$$

Discussion: Lemma 2.1 has the following informal interpretation. Suppose an individual wishes to learn x by eavesdropping on edges a and b. The lemma states that knowing y does not help the eavesdropper. This fact is rather subtle, as the following well-known example shows. Let x and y be independent binary strings, and let a and b both transmit the symbol $x \oplus y$. This network code meets the single-letter criterion for validity; furthermore, knowing y in this case *does* help the eavesdropper. However, this network code is not valid so it does not give a counterexample to Lemma 2.1.

Proof (of Lemma 2.1). Expand the random variables in $I(xy; ab)$ into their components at each time step.

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

This can be divided into the following two sums:

$$\begin{aligned} &\sum_{k=1}^{n} I(xy; a^{(k)} | a^{(1..k-1)} b^{(1..k-1)}) \\ +\; &\sum_{k=1}^{n} I(xy; b^{(k)} | a^{(1..k)} b^{(1..k-1)}). \end{aligned} \tag{2.1}$$

We now analyze the first sum.

$$\begin{aligned} &\sum_{k=1}^{n} I(xy; a^{(k)} | a^{(1..k-1)} b^{(1..k-1)}) \\ &\quad \geq \sum_{k=1}^{n} I(x; a^{(k)} | a^{(1..k-1)} b^{(1..k-1)} y) && (2.2) \\ &\quad = \sum_{k=1}^{n} I(x; a^{(k)} | a^{(1..k-1)} y) && (2.3) \\ &\quad = I(x; a|y) && (2.4) \\ &\quad = I(x; ab|y) && (2.5) \end{aligned}$$

where Eq. (2.2) = Eq. (2.3) holds because $b^{(1..k-1)}$ is a function of $a^{(1..k-1)}$ and y, and Eq. (2.4)=Eq. (2.5) holds because b is a function of a and y. This gives a lower bound on the first sum in Eq. (2.1). A similar argument applies to the second sum (note that $a^{(1..k)}$ is a function of $b^{(1..k-1)}$ and x) , so we obtain

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

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

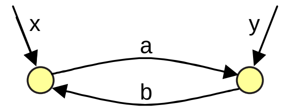


Fig. 4. The network structure analyzed in Lemma 2.1.

B. Preliminary Analysis of $\mathcal{G}_1$

Suppose we have a valid network code in $\mathcal{G}_1$ with $r \geq 1-\lambda$. We now derive two inequalities on the information transmitted through the network.

Lemma 2.2: 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

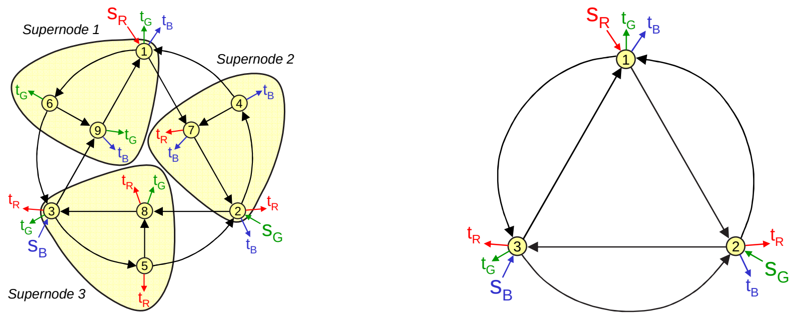
$$\begin{aligned} &\underbrace{I(RGB; \mathbf{e}_{35})}_{\leq H(\mathbf{e}_{35}) \leq n} \;=\; \underbrace{I(R; \mathbf{e}_{35})}_{=n(1-\lambda)} + I(GB; \mathbf{e}_{35}|R) \\ \implies &I(GB; \mathbf{e}_{35}|R) \;\leq\; \lambda n. \end{aligned}$$

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

Without loss of generality, we may use the network structure to 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 capacities 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 2.2 together imply that $I(\mathbf{e}_{52}; GB|R) \leq \lambda n$ and $I(\mathbf{e}_{52}\mathbf{e}_{28}; B|RG) \leq 2\lambda n$.

C. 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 5.) This is done to eliminate portions which need no further analysis and to facilitate derivation of further information inequalities. $\mathcal{G}_1^*$ is constructed by identifying 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



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

Fig. 5. (a) Identifying vertices in $\mathcal{G}_1$. (b) The resulting instance $\mathcal{G}_1^*$.

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 2.2 implies (via symmetry) that the following inequalities hold:

$$\begin{array}{ll} I(\mathbf{e}_{32}; GB|R) \leq \lambda n & I(\mathbf{e}_{23}\mathbf{e}_{32}; B|RG) \leq 2\lambda n \\ I(\mathbf{e}_{13}; RB|G) \leq \lambda n & I(\mathbf{e}_{31}\mathbf{e}_{13}; R|BG) \leq 2\lambda n \\ I(\mathbf{e}_{21}; RG|B) \leq \lambda n & I(\mathbf{e}_{12}\mathbf{e}_{21}; G|RB) \leq 2\lambda n \end{array} \tag{2.6}$$

D. Analysis

The argument proceeds as follows. In Lemma 2.3, we apply the inequality of Lemma 2.1 to our particular network. 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 2.6.

Lemma 2.3:

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

Proof. Conditioning on each realization of G, apply Lemma 2.1 with $a = \mathbf{e}_{23}$, $b = \mathbf{e}_{32}$, $x = \mathbf{e}_{12}$ and $y = B\mathbf{e}_{13}$. ∎

Lemma 2.4: $I(\mathbf{e}_{23}\mathbf{e}_{32}; B\mathbf{e}_{13}|G) \leq 3\lambda n$.

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

$$\begin{aligned} &I(\mathbf{e}_{23}\mathbf{e}_{32}; B\mathbf{e}_{13}|G) \\ &\quad\leq I(R; B\mathbf{e}_{13}|G) + I(\mathbf{e}_{23}\mathbf{e}_{32}; B\mathbf{e}_{13}|RG). \end{aligned} \tag{2.7}$$

The first term in Eq. (2.7) is upper bounded as follows.

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

where the first inequality follows from the inequality $I(W;XY) \geq I(W;X|Y)$ and the second is in Eq. (2.6).

To upper bound the second term in Eq. (2.7), we write

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

Clearly $I(\mathbf{e}_{23}\mathbf{e}_{32}; \mathbf{e}_{13}|RGB) = 0$, since no entropy remains after conditioning on all sources. Now by Eq. (2.6), $I(\mathbf{e}_{23}\mathbf{e}_{32}; B|RG) \leq 2\lambda n$. ∎

Lemma 2.5:

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

Proof. Considering vertex 2, any network code must have $H(RB|\mathbf{e}_{12}\mathbf{e}_{32}G) = 0$. Thus $H(\mathbf{e}_{12}\mathbf{e}_{32}|G) = H(\mathbf{e}_{12}\mathbf{e}_{32}RB|G) \geq 2n(1-\lambda)$. Thus

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

But $H(\mathbf{e}_{23}\mathbf{e}_{32}|\mathbf{e}_{13}B\mathbf{e}_{12}G) = 0$, by the network structure, so we conclude that

$$2n(1-\lambda) \leq n + I(\mathbf{e}_{23}\mathbf{e}_{32}; \mathbf{e}_{13}B|\mathbf{e}_{12}G).$$ ∎

Lemma 2.6: Any valid network code in $\mathcal{G}_1$ has rate at most $4/5$.

Proof. The preceding analysis assumes existence of a network code with rate $1-\lambda$. Lemma 2.3, Lemma 2.4, and Lemma 2.5 together show that $3\lambda n \geq n - 2\lambda n$. Thus $\lambda \geq 1/5$. ∎

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 $0 \le i < n$,

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

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

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

Rounds $3n+3, \ldots, 9n+2$: For $0 \le i < n$, 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 to transform this solution into a convolutional code as in [2], 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. (The construction uses standard techniques, but we describe it in concrete terms in the following paragraph for the purpose of making our exposition self-contained.) 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.

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 $p - s \le 0$ 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.

There is a deceptive similarity between $\mathcal{G}_2$ and the example analyzed in Section VI of [2]; the main difference between our example and theirs is that their network admits a solution which exactly achieves the maximum possible rate, and nodes only need to remember a bounded number of received symbols in order to perform the required encoding and decoding operations. In contrast, our solution achieves rate 1 only as a supremum (as n tends to infinity) and nodes must store $O(n)$ symbols in memory in order to perform the encoding and decoding operations required in rounds $3n+3$ to $9n+2$ of our protocols. It is doubtful that there exists a solution for $\mathcal{G}_2$ without these two features.

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] E. C. van der Meulen. *Transmission of information in a T-terminal discrete memoryless channel*. PhD thesis, UC Berkeley, 1968.

[8] F. M. J. Willems and E. C. van der Meulen. The discrete memoryless multiple-access channel with cribbing encoders. *IEEE Transactions on Information Theory*, 31(3):313–327, 1985.

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