Interactive Function Computation

Changho Suh
KAIST, Daejeon, South Korea
Email: chsuh@kaist.ac.kr

Michael Gastpar
EPFL, Lausanne, Switzerland
Email: michael.gastpar@epfl.ch

***Abstract*—We investigate the role of *interaction* for computation problem settings where nodes intend to compute *functions* of the raw messages generated at other nodes. In this work, we make some progress on a more elementary research component: *feedback*. Specifically we characterize the feedback computing capacity of a two-transmitter two-receiver linear deterministic network in which both receivers wish to decode a linear function (modulo-2 sum) of Bernoulli sources generated at the transmitters. Inspired by the concept of interference alignment and compute-and-forward, we develop a new achievable scheme called *interactive function alignment*. A new converse theorem is established that is tighter than cut-set based and genie-aided bounds. As a consequence of this result, we show that interaction can provide an arbitrarily large gain for computation, as in classical communication settings.**

I. Introduction

The inherent two-way nature of communication links provides an opportunity to enable *interaction* between nodes. It allows the nodes to adapt their transmitted signals to the past received signals that can be fed back through backward communication links. This opportunity occurs more frequently in large and dense networks, and the growth of networks in size and density has been driving the need to characterize the fundamental limits of the gains reaped by interaction.

In this paper, interaction is modeled via the classical information-theoretic notion of noiseless output feedback. For a memoryless point-to-point channel, Shannon elegantly proved that feedback has no bearing on capacity [1], but subsequent work showed that the situation is different for point-to-point channels with memory as well as for many multi-user channels. For many scenarios, capacity improvements due to feedback are rather modest. One notable exception is the result in [2], which shows significant capacity gains for the Gaussian interference channel. Finally, we point out that while feedback is known to enlarge the capacity region, in most cases, the precise feedback capacity region remains unknown.

Our interest is to examine the benefit of interaction for more general scenarios in which nodes now intend to compute *functions* of the raw messages rather than the messages themselves. The general settings include many of the realistic scenarios such as sensor networks [3] and cloud computing scenarios [4], [5]. In a sensor network, a fusion node may be interested in computing a relevant function of the measurements from various data nodes. In a cloud computing scenario, a client may download a function or part of the original source information that is distributed (e.g., using a maximum distance separable code) across multiple data nodes. In the setting of computation, we seek to understand the role of feedback as an initial step. Specifically we are interested in addressing the following question: Can feedback provide a significant gain for computation, as in the setting of classical communications [2]?

For classical communications, it was shown in [2], [6] that the significant feedback gain comes from the many-to-many channel structure. This leads us to consider a *two*-transmitter *two*-receiver scenario for computation as a natural model, in which two receivers wish to compute a function of the sources generated at the transmitters. For simplicity, we assume two independent Bernoulli sources and a linear function (modulo-2 sum). We also assume that each receiver feeds back its received signal to both transmitters. Specifically we consider the Avestimehr-Diggavi-Tse (ADT) deterministic single-hop network model [7] which captures superposition and broadcast properties of wireless Gaussian networks and is a generalization of networks of orthogonal links. For this model, we develop a new achievable scheme termed *interactive function alignment*, inspired by the concept of interference alignment [8], [9] and compute-and-forward [10], [11]. We also derive a matching upper bound that is tighter than cut-set based and genie-aided bounds. As a consequence of this result, we show that interaction provides significant gain for computation – qualitatively similar to that in [2].

Related Work: Orlitsky and Roche [12] explored the benefit of interaction for a point-to-point source coding problem setting in which two nodes wish to compute functions of $f_1(S_1, S_2)$ and $f_2(S_1, S_2)$ respectively, where S_ℓ denotes node ℓ's source. While interaction was shown to reduce the transmission rate for general target functions, it has been shown that feedback does not help when the desired functions are modulo-2 sums of the Bernoulli sources. On the other hand, we consider a *many-to-many* source-channel coding problem, and find that interaction can provide a capacity increase even for the modulo-2-sum function of the Bernoulli sources.

II. Model

We consider a two-transmitter two-receiver ADT deterministic network as depicted in Fig. 1. This network is described by four integer parameters n_{ij} which indicates the number of signal bit levels from transmitter i $(i = 1, 2)$ to receiver j $(j = 1, 2)$. Let $X_\ell \in \mathbb{F}_2^q$ be transmitter ℓ's encoded signal

This research was funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2013.

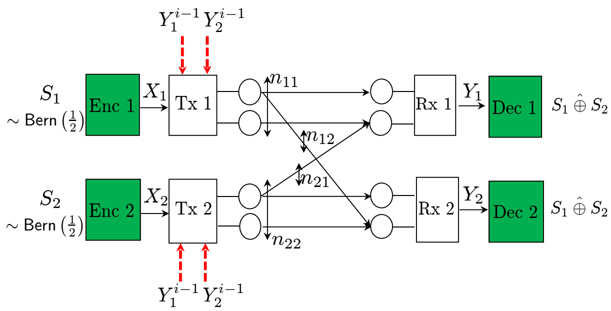

Fig. 1. Two-transmitter two-receiver Avestimehr-Diggavi-Tse (ADT) deterministic network with feedback.

where $q = \max_{ij} n_{ij}$. The received signals are then given by

$$\begin{aligned} Y_1 &= \mathbf{G}^{q-n_{11}} X_1 \oplus \mathbf{G}^{q-n_{21}} X_2, \\ Y_2 &= \mathbf{G}^{q-n_{12}} X_1 \oplus \mathbf{G}^{q-n_{22}} X_2, \end{aligned} \tag{1}$$

where $\mathbf{G}$ is the q-by-q shift matrix, i.e., $[\mathbf{G}]_{ij} = \mathbf{1}\{i = j+1\}$ $(1 \le i \le q; 1 \le j \le q)$, and operations are performed in $\mathbb{F}_2$. We focus on a simple setting where $n := n_{11} = n_{22}$ and $m := n_{12} = n_{21}$. Each receiver wishes to compute modulo-2 sums of the two Bernoulli sources S_1^K and S_2^K, generated at the two transmitters, with N uses of the network. Here we use shorthand notation to indicate the sequence up to K, e.g., $S_1^K := (S_{11}, \cdots, S_{1K})$. We assume that S_1^K and S_2^K are independent and identically distributed with Bern$(\frac{1}{2})$. The encoded signal $X_{\ell i}$ of transmitter ℓ at time i is a function of its own source S_ℓ^K and past feedback signals (Y_1^{i-1}, Y_2^{i-1}). Receiver ℓ uses a decoding function d_ℓ to estimate $S_1^K \oplus S_2^K$ from its received signal Y_ℓ^N. An error occurs whenever $d_\ell \neq S_1^K \oplus S_2^K$. The average probabilities of error are given by $\lambda_\ell = \mathbb{E}\left[P(d_\ell \neq S_1^K \oplus S_2^K)\right]$, $\ell = 1, 2$.

We say that the computing rate $R_{\text{comp}} = \frac{K}{N}$ is achievable if there exists a family of codebooks and encoder/decoder functions such that the average decoding error probabilities of λ_1 and λ_2 go to zero as code length N tends to infinity. The computing capacity C_{comp} is the supremum of the achievable rates.

III. Main Result

Theorem 1: The feedback computing capacity is

$$C_{\text{comp}} = \begin{cases} \frac{2}{3}\max(m,n), & m \neq n; \\ n, & m = n. \end{cases}$$

Proof: Section IV establishes the achievablity proof. The converse is proved in Section V. ∎

Feedback Gain: From the above theorem, it can be readily seen that feedback provides a significant gain. To see this clearly, let us plot the normalized computing capacity C_{comp}/q as a function of $\alpha := \min(m,n)/q$. See Fig. 2. Here the parameter α indicates a signal strength difference between direct and cross channel links. We compare this to the nonfeedback computing capacity characterized in [13] as: $C^{\text{no}}_{\text{comp}}/q = \min\{\alpha, \frac{2}{3}\}$ for $\alpha \neq 1$; $C^{\text{no}}_{\text{comp}}/q = 1$ for $\alpha = 1$. Notice a feedback gain for the regime of $0 \le \alpha < \frac{2}{3}$. This

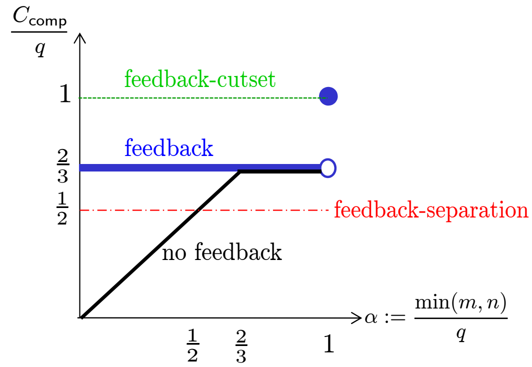

Fig. 2. Feedback computing capacity. Here $n := n_{11} = n_{22}$, $m := n_{12} = n_{21}$ and $q = \max(m,n)$. The parameter α captures a signal similarity between m and n.

gain is multiplicative – qualitatively similar to that of the Gaussian interference channel [2]. The gap between the feedback and nonfeedback capacities can be arbitrarily large with an increase in q. Moreover, the factor gain can be unbounded. Note that for $\alpha = 0$, the normalized feedback capacity is $\frac{2}{3}$, while the nonfeedback capacity is 0. In Section IV-B, we will provide an intuition behind this gain while describing an achievable scheme.

Remark 1 (Comparison to Separation Scheme): Compare to the separation scheme where both receivers decode all of the sources and then compute the desired function. Note that the computing rate of the separation scheme can be readily derived from the feedback capacity region of the ADT traditional multicast network: $\mathcal{C}_{\text{mult}} = \{(R_1, R_2) : R_1 + R_2 \le \max(m,n)\}$ [6]. Specifically it is derived as $R_{\text{sep}}/q = C_{\text{sym}}/q = 1/2$, where $C_{\text{sym}} := \sup\{R : (R,R) \in \mathcal{C}_{\text{mult}}\}$. Notice that our optimal scheme outperforms the separation approach for an entire range of α. It turns out that feedback enables in-network computation, thus achieving the desired function more efficiently. Details will be explained in Section IV-B. □

Remark 2 (Comparison to Cut-set Bound): For the whole range of α (except for $\alpha \neq 1$), the feedback computing capacity is strictly less than the cut-set bound of $\max(m,n)$. In Section V, we will provide a formal proof of the converse bound. □

IV. Proof of Achievability

By symmetry, focus on the case of $m \le n$. The other case of $m \ge n$ is a mirror image in which receivers 1 and 2 are swapped. Since in our problem both receivers have the same demand, it suffices to consider only one case. For the case of $\alpha = 1$, simple uncoded transmission can yield $R_{\text{comp}} = n$. For $\frac{2}{3} \le \alpha \le 1$, we can achieve $R_{\text{comp}} = \frac{2}{3}n$ with the nonfeedback scheme in [13]. Hence, our focus is the case of $0 \le \alpha \le \frac{2}{3}$.

Our achievability consists of two parts. First we employ *network decomposition* (developed in the nonfeedback case [13]) to identify elementary subnetworks that can constitute an original network without loss of optimality. Next we develop an achievable scheme for the identified elementary subnetworks. From these two parts, we can then derive an achievable rate of an original (m, n) network.

A. Achievability via Network Decomposition

We state the network decomposition established for the nonfeedback case [13]. We focus on the regime of $0 \le \alpha \le \frac{2}{3}$.

Lemma 1 (Network Decomposition): The following network decompositions hold:

(1) For $0 \le \alpha \le \frac{1}{2}$:

$$(m, n) = (1, 2)^m \times (0, 1)^{n-2m}.$$

(2) For $\frac{1}{2} \le \alpha \le \frac{2}{3}$:

$$(m, n) = (2, 3)^{2m-n} \times (1, 2)^{2n-3m}.$$

Here we use the symbol $\times$ for the concatenation of orthogonal models, just like in $\mathbb{R}^2 = \mathbb{R} \times \mathbb{R}$.

Proof: Fig. 3(a) shows how the (m, n) model can be decomposed into $(1, 2)^m \times (0, 1)^{n-2m}$ for the regime of $0 \le \alpha \le \frac{1}{2}$. The idea of the decomposition is to use graph coloring. Start with assigning a color (say, blue) to the first level at transmitter 1. We then assign the same blue color to all the levels that are connected with the first level at transmitter 1. These are the first level at receiver 1 and the second bottom level at receiver 2. Now do the same procedure starting from transmitter 2. Specifically, assign the blue color to the first level at transmitter 2, then assign the same color to all of the connected levels: the first level at receiver 2 and the second bottom level at receiver 1. We then obtain an independent graph of model $(1, 2)$ and are left with model $(m-1, n-2)$. For the remaining graph of model $(m-1, n-2)$, repeat the above procedure. We then obtain $(1, 2)^2$ and are left with model $(m-2, n-2)$. Here we used the same blue color, as the additionally obtained graph is of the same model $(1, 2)$. Repeating this procedure m times, we finally obtain $(1, 2)^m \times (0, 1)^{n-m}$. Using the same graph coloring idea, we can also prove the second decomposition. See Fig. 3(b) for details. ■

Lemma 1 implies that fundamental building blocks constitute only three models: $(0, 1)$, $(1, 2)$ and $(2, 3)$. Hence, we focus on the computing rates of these models.

Lemma 2: The following rates are achievable:

(1) For the model $(0, 1)$: $R_{\text{comp}} = \frac{2}{3}$.
(2) For the model $(1, 2)$: $R_{\text{comp}} = \frac{4}{3}$.
(3) For the model $(2, 3)$: $R_{\text{comp}} = 2$.

Proof: See Section IV-B. ■

By Lemmas 1 and 2, we can now readily prove the achievability of Theorem 1. For $0 \le \alpha \le \frac{1}{2}$,

$$R_{\text{comp}} = \frac{4}{3} \cdot m + \frac{2}{3} \cdot (n - 2m) = \frac{2}{3} n.$$

For $\frac{1}{2} \le \alpha \le \frac{2}{3}$,

$$R_{\text{comp}} = 2 \cdot (2m - n) + \frac{4}{3} \cdot (2n - 3m) = \frac{2}{3} n.$$

Remark 3 (Decomposition is optimal): In [13], we showed that for our model (but without feedback), it is without loss of

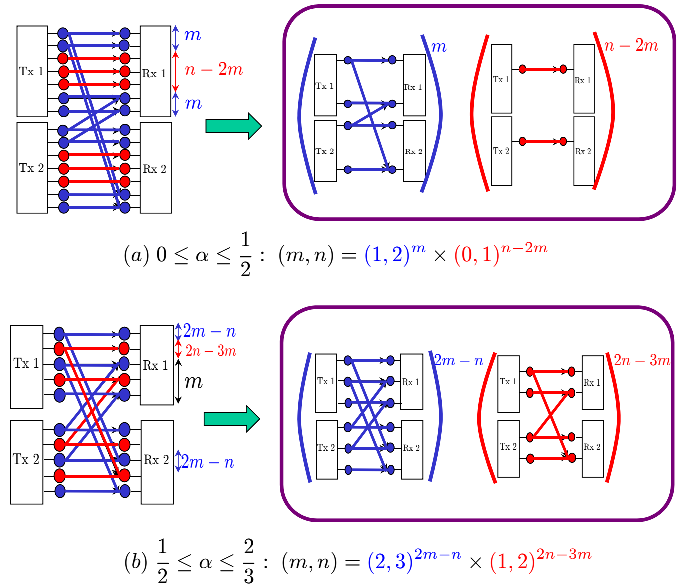

Fig. 3. Network Decomposition.

optimality to first decompose the network into components as in Lemma 1 and then to *separately* code over each component. This is non-trivial since for general parallel interference channels, optimum performance requires *joint* coding across the components. Lemma 2 (together with the converse bound that completes Theorem 1) somewhat surprisingly establishes that for our model, even if there is feedback, coding separately over each of the component channels is sufficient to attain optimum performance. □

B. Proof of Lemma 2

(0,1) model: In the nonfeedback case, each transmitter cannot deliver its source to a cross-linked receiver. Hence, $C_{\text{comp}}^{\text{no}} = 0$. With feedback, however, we can create a new signal path which helps linking the crossed transmitter-receiver pair. Using this created path due to feedback, we can obtain a positive computing rate.

A naive way to achieve a positive rate is to employ the separation scheme. In time 1, transmitters 1 and 2 send a_1 and b_1 respectively. With feedback, transmitter 1 can then get the other transmitter's symbol b_1. Similarly transmitter 2 can get a_1. Forwarding these fed-back symbols at time 2, each receiver can get a_1 and b_1, from which $a_1 \oplus b_1$ can be computed. This gives a positive rate of $\frac{1}{2}$.

A smarter approach enables us to achieve a higher rate. The idea is to employ *compute-and-forward* at transmitters with the help of feedback. The achievable scheme is illustrated in Fig. 4. It consists of three time slots. In time 1, transmitter 1 sends a_1 while transmitter 2 sends b_2. Note that b_2 is transmitted instead of b_1. This is one of the key parts in the scheme. With feedback, transmitter 1 can now decode b_2. The key observation here is that this b_2 enables transmitter 1 to compute the desired function of $a_2 \oplus b_2$. Transmitter 1 forwarding the $a_2 \oplus b_2$ in time 2, receiver 1 can get the $a_2 \oplus b_2$.

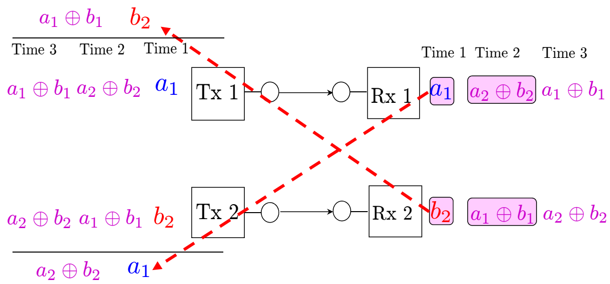

Fig. 4. An achievable scheme for the $(m, n) = (0, 1)$ model.

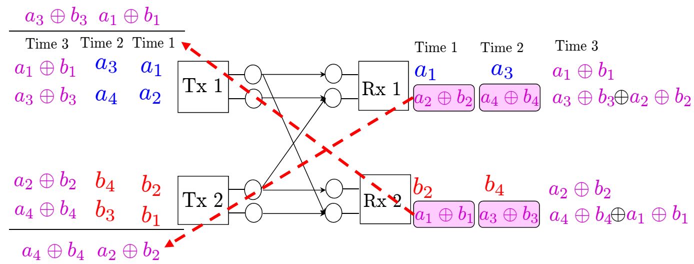

Fig. 5. An achievable scheme for the $(m, n) = (1, 2)$ model.

Similarly receiver 2 can get $a_1 \oplus b_1$. Until the end of time 2, $a_1 \oplus b_1$ is not delivered yet at receiver 1; similarly $a_2 \oplus b_2$ is missing at receiver 2. Using one more time slot, we can accomplish the transmission of these signals. With feedback, transmitter 1 can get $a_1 \oplus b_1$. Transmitter 1 forwarding this in time 3, receiver 1 can obtain the $a_1 \oplus b_1$. Similarly receiver 2 can get $a_2 \oplus b_2$. In summary, both receivers can decode $(a_1 \oplus b_1, a_2 \oplus b_2)$ during three time slots, thus achieving $R_{\text{comp}} = \frac{2}{3}$.

Remark 4 (Interactive function alignment): Note that receiver 1's signal $a_2 \oplus b_2$ at time 2 occupies only one-dimensional linear subspace. Two different symbols a_2 and b_2 are *aligned* onto a single-dimensional linear subspace. This coincides with the concept of *interference alignment*. In the setting of computation, this alignment phenomenon was named as *function alignment* [13], as the alignment is w.r.t a function. Also the function alignment here is enabled through the use of feedback (interaction). Hence, we call this scheme *interactive function alignment*. □

(1,2) model: See Fig. 5. Similar to the previous scheme, our scheme consists of three time slots. In this model, observe that the bottom level at each receiver naturally forms the mod-2 sum function of interest. This enables us to achieve function alignment. In time 1, we achieve the function alignment on the bottom level: achieving $a_2 \oplus b_2$ at receiver 1; achieving $a_1 \oplus b_1$ at receiver 2. In time 2, we repeat this w.r.t new symbols: achieving $a_4 \oplus a_4$ and $a_3 \oplus b_3$ at receivers 1 and 2 respectively. Until the end of time 2, $(a_1 \oplus b_1, a_3 \oplus b_3)$ are not delivered yet to receiver 1. Similarly $(a_2 \oplus b_2, a_4 \oplus b_4)$ are missing in receiver 2. With feedback, however, we can accomplish the transmission of these signals very efficiently.

With feedback, each transmitter can now obtain the desired functions which were received at only one receiver. Transmitter 1 can get $(a_1 \oplus b_1, a_3 \oplus b_3)$ which were received only at receiver 2. Similarly transmitter 2 can get $(a_2 \oplus b_2, a_4 \oplus b_4)$. Now the idea is to forward all of these functions in time 3. Receiver 1 then gets $a_1 \oplus b_1$ cleanly on the top level. But it gets a mixture of the two desired functions on the bottom level: $(a_3 \oplus b_3) \oplus (a_2 \oplus b_2)$. Here the key observation is that the $a_2 \oplus b_2$ in the mixture was already received in time 1. So using $a_2 \oplus b_2$, receiver 1 can clean up $a_3 \oplus b_3$. Similarly receiver 2 can decode $(a_2 \oplus b_2, a_4 \oplus b_4)$. In summary, both receivers can compute four bits during three time slots, thus achieving $R_{\text{comp}} = \frac{4}{3}$.

Remark 5 (Exploiting side information): Note that receiver 1's signal $a_2 \oplus b_2$ at time 1 is used to clean up $a_3 \oplus b_3$ from $(a_3 \oplus b_3) \oplus (a_2 \oplus b_2)$ received at time 3. One can see that feedback enables each receiver to exploit their previously received signals as *side information*, thus increasing the nonfeedback capacity. This coincides with the role of feedback found in interference channels [2], two-way relay channels [14] and many other examples. □

(2,3) model: We use the nonfeedback scheme in [13]. Transmitter 1 sending (a_2, a_1, a_2) on the three levels in order from the top, transmitter 2 sending (b_1, b_2) on the first and second levels, both receivers can decode $(a_1 \oplus b_1, a_2 \oplus b_2)$, thus yielding $R_{\text{comp}} = 2$.

V. Proof of Converse

The proof for the case of $m = n$ is straightforward. Using the standard cut-set argument, we get: $N(R_{\text{comp}} - \epsilon_N) \le I(S_1^K \oplus S_2^K; Y_1^N) \le \sum H(Y_{1i}) \le N \max(m, n)$. If R_{comp} is achievable, then $\epsilon_N \to 0$ as N tends to infinity, and hence $R_{\text{comp}} \le \max(m, n) = n$.

The main focus is to prove the bound for $m \neq n$. Starting with Fano's inequality, we get

$$
\begin{aligned}
&N(3R_{\text{comp}} - \epsilon_N) \\
&\le I(S_1^K \oplus S_2^K; Y_1^N) + I(S_1^K \oplus S_2^K; Y_2^N) + I(S_1^K \oplus S_2^K; Y_1^N) \\
&\overset{(a)}{\le} [H(Y_1^N) - H(Y_1^N | S_1^K \oplus S_2^K)] \\
&\quad + [H(Y_2^N) - H(Y_2^N | S_1^K \oplus S_2^K, Y_1^N)] + I(S_1^K \oplus S_2^K; Y_1^N) \\
&\overset{(b)}{\le} H(Y_1^N) + H(Y_2^N) \\
&\quad - H(Y_1^N, Y_2^N | S_1^K \oplus S_2^K) + I(S_1^K; Y_1^N, Y_2^N) \\
&\overset{(c)}{\le} H(Y_1^N) + H(Y_2^N) \\
&\quad - H(Y_1^N, Y_2^N | S_1^K \oplus S_2^K) + I(S_1^K; Y_1^N, Y_2^N | S_1^K \oplus S_2^K) \\
&\le \sum [H(Y_{1i}) + H(Y_{2i})] \le 2N \max(m, n)
\end{aligned}
$$

where (a) follows from the fact that conditioning reduces entropy; (b) follows from $I(S_1^K \oplus S_2^K; Y_1^N) \le I(S_1^K; Y_1^N, Y_2^N)$ (See Claim 1 below); (c) follows from the fact that S_1^K is independent of $S_1^K \oplus S_2^K$. Therefore, we get the desired bound.

Claim 1: $I(S_1^K \oplus S_2^K; Y_1^N) \le I(S_1^K; Y_1^N, Y_2^N)$.

Proof:

$$
\begin{aligned}
&I(S_1^K \oplus S_2^K; Y_1^N) \overset{(a)}{=} H(S_1^K) - H(S_1^K \oplus S_2^K | Y_1^N)\\
&\overset{(b)}{\leq} H(S_1^K) - H(S_1^K \oplus S_2^K | Y_1^N, Y_2^N, S_2^K)\\
&= H(S_1^K) - H(S_1^K | Y_1^N, Y_2^N, S_2^K)\\
&\overset{(c)}{=} H(S_1^K) - H(S_1^K | Y_1^N, Y_2^N)
\end{aligned}
$$

where (a) follows from the fact that S_1^K and S_2^K are i.i.d. Bernoulli sources; (b) follows from the fact that conditioning reduces entropy; and (c) follows from $S_1^K - (Y_1^N, Y_2^N) - S_2^K$ (See Claim 2 below). ■

Claim 2: $S_1^K - (Y_1^N, Y_2^N) - S_2^K$

Proof: It suffices to prove that $I(S_1^K; S_2^K | Y_1^N, Y_2^N) \leq 0$ as below, due to the non-negativity of mutual information.

$$
\begin{aligned}
&I(S_1^K; S_2^K | Y_1^N, Y_2^N)\\
&= I(S_1^K; S_2^K, Y_1^N, Y_2^N) - I(S_1^K; Y_1^N, Y_2^N)\\
&= I(S_1^K; Y_1^N, Y_2^N | S_2^K) - I(S_1^K; Y_1^N, Y_2^N)\\
&= -H(Y_1^N, Y_2^N) + H(Y_1^N, Y_2^N | S_1^K) + H(Y_1^N, Y_2^N | S_2^K)\\
&\overset{(a)}{=} -H(X_1^N, X_2^N) + H(X_1^N, X_2^N | S_1^K) + H(X_1^N, X_2^N | S_2^K)\\
&= -\sum H(X_{1i}, X_{2i} | X_1^{i-1}, X_2^{i-1})\\
&\quad + \sum H(X_{1i}, X_{2i} | S_1^K, X_1^{i-1}, X_2^{i-1})\\
&\quad + \sum H(X_{1i}, X_{2i} | S_2^K, X_1^{i-1}, X_2^{i-1})\\
&\overset{(b)}{=} -\sum H(X_{1i}, X_{2i} | X_1^{i-1}, X_2^{i-1})\\
&\quad + \sum H(X_{2i} | S_1^K, X_1^{i-1}, X_2^{i-1})\\
&\quad + \sum H(X_{1i} | S_2^K, X_1^{i-1}, X_2^{i-1})\\
&= -\sum [H(X_{1i} | X_1^{i-1}, X_2^{i-1}) - H(X_{1i} | S_2^K, X_1^{i-1}, X_2^{i-1})]\\
&\quad + \sum [H(X_{2i} | X_1^i, X_2^{i-1}) - H(X_{2i} | S_1^K, X_1^{i-1}, X_2^{i-1})]\\
&\overset{(c)}{\leq} 0
\end{aligned}
$$

where (a) follows from the fact that (X_1, X_2) is a function of (Y_1, Y_2) (See Claim 3 below); (b) follows from the fact that $X_{\ell i}$ is a function of $(S_\ell^K, Y_1^{i-1}, Y_2^{i-1})$; and (c) follows from the fact that conditioning reduces entropy. ■

Claim 3: For $m \neq n$, (X_1, X_2) is a function of (Y_1, Y_2).

Proof: Consider the case of $m < n$. From (1), we get

$$
Y_1 \oplus (\mathbf{G}^{n-m} Y_2) = (\mathbf{I}_n \oplus \mathbf{G}^{2(n-m)}) X_1.
$$

Note that $\mathbf{I}_n \oplus \mathbf{G}^{2(n-m)}$ is invertible when $m \neq n$. Hence, X_1 is a function of (Y_1, Y_2). By symmetry, X_2 is a function of (Y_1, Y_2). Similarly we can show this for the case of $m > n$. ■

VI. Conclusion

We developed a new achievable scheme and derived a new outer bound, thereby establishing the computing capacity of the two-transmitter two-receiver ADT network with feedback. Our future work is along several new directions: (1) extending to more realistic scenarios where feedback is offered through rate-limited bit-piped links [15] or a corresponding backward channel [16]; (2) generalizing to four-source scenarios in which transmitters also want to compute a function of the sources generated at receivers; (3) extending to arbitrary multi-hop networks [17], [18].

References

[1] C. E. Shannon, "The zero error capacity of a noisy channel," *IRE Transactions on Information Theory*, vol. 2, pp. 8–19, Sept. 1956.

[2] C. Suh and D. Tse, "Feedback capacity of the Gaussian interference channel to within 2 bits," *IEEE Transactions on Information Theory*, vol. 57, pp. 2667–2685, May 2011.

[3] A. Giridhar and P. R. Kumar, "Computing and communicating functions over sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 755–764, Apr. 2005.

[4] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 56, pp. 4539–4551, Sept. 2010.

[5] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proceedings of the IEEE*, vol. 99, pp. 476–489, Mar. 2011.

[6] C. Suh, N. Goela, and M. Gastpar, "Approximate feedback capacity of the Gaussian multicast channel," *Proceedings of the IEEE International Symposium on Information Theory*, July 2012.

[7] S. Avestimehr, S. Diggavi, and D. Tse, "Wireless network information flow: A deterministic approach," *IEEE Transactions on Information Theory*, vol. 57, pp. 1872–1905, Apr. 2011.

[8] M. A. Maddah-Ali, S. A. Motahari, and A. K. Khandani, "Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis," *IEEE Transactions on Information Theory*, vol. 54, pp. 3457–3470, Aug. 2008.

[9] V. R. Cadambe and S. A. Jafar, "Interference alignment and the degree of freedom for the K user interference channel," *IEEE Transactions on Information Theory*, vol. 54, pp. 3425–3441, Aug. 2008.

[10] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Transactions on Information Theory*, vol. 53, pp. 3498–3516, Oct. 2007.

[11] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Transactions on Information Theory*, vol. 57, pp. 6463–6486, Oct. 2011.

[12] A. Orlitsky and J. R. Roche, "Coding for computing," *IEEE Transactions on Information Theory*, vol. 47, pp. 903–917, Mar. 2001.

[13] C. Suh, N. Goela, and M. Gastpar, "Computation in multicast networks: Function alignment and converse theorems," *submitted to the IEEE Transactions on Information Theory*, Oct. 2012.

[14] Y. Wu, P. A. Chou, and S. Y. Kung, "Information exchange in wireless networks with network coding and physical-layer broadcast," *CISS 39th Annual Conference*, Mar. 2005.

[15] A. Vahid, C. Suh, and A. S. Avestimehr, "Interference channels with rate-limited feedback," *IEEE Transactions on Information Theory*, vol. 58, pp. 2788–2812, May 2012.

[16] C. Suh, I.-H. Wang, and D. Tse, "Two-way interference channels," *arXiv:1202.5014*, Feb. 2012.

[17] A. Ramamoorthy and M. Langberg, "Communicating the sum of sources over a network," *arXiv:1001.5319*, Jan. 2010.

[18] B. Rai and B. Dey, "On network coding for sum-networks," *IEEE Transactions on Information Theory*, vol. 58, pp. 50–63, Jan. 2012.