Simplified Composite Coding for Index Coding

Yucheng Liu, Parastoo Sadeghi
Research School of Engineering
Australian National University
{yucheng.liu, parastoo.sadeghi}@anu.edu.au

Fatemeh Arbabjolfaei, Young-Han Kim
Department of Electrical and Computer Engineering
University of California, San Diego
{farbabjo, yhk}@ucsd.edu

Abstract—Simplification methods are introduced for composite coding, which is an existing layered random coding technique for the index coding problem. As the problem size grows, the original number of composite indices grows exponentially and the number of possible decoding configurations (decoding sets) grows super exponentially, leading to considerably high computational complexity. The proposed simplifications address both issues and do not affect the performance (tightness) of the coding scheme. Removing composite indices is achieved by pairwise comparison of any two indices and removing one if its corresponding rate can be transferred without loss to the other in the expressions of the achievable rate region. Decoding configurations are reduced by establishing a baseline or natural decoding configuration, where no smaller decoding configuration can provide a strictly larger rate region. A heuristic method is also proposed for reducing the number of composite indices even further, but possibly with some performance loss. Numerical results demonstrate good performance with substantial reduction in complexity. To achieve the capacity region for all 9608 non-isomorphic index coding problems with $n = 5$, a single natural decoding configuration per problem and less than 3 out of $2^5 - 1 = 31$ composite indices are sufficient, on average. In only 31 problems, 7 to at most 10 composite indices are used.

I. Introduction

The index coding problem studies the optimal broadcast rate of n messages from a server to multiple receivers with side information. Since its introduction by Birk and Kol [1] in 1998, the index coding problem has been recognized as a canonical problem in network information theory and has drawn considerable attention from many research communities. Various index coding schemes, broadly categorized into linear or non-linear codes, have been proposed, which have in turn established upper bounds on the optimal broadcast rate (or inner bounds on the capacity region) of the index coding problem; see [2] and the references therein. However, none of the existing schemes is generally optimal. Particularly, it has been shown in [3], [4] that linear index codes can be in general outperformed by non-linear index codes.

One promising non-linear coding scheme is the *composite coding*, a two-layer random coding scheme first proposed in [5] and later enhanced in [6]. For obtaining an explicit rate region or a weighted sum rate for a given index coding problem, one has to solve a Fourier–Motzkin elimination (FME) problem [7, Appendix D] or a linear program (LP). As n increases, the computational complexity can become a practical issue for composite coding. First, the number of composite messages and their corresponding rates in the expressions of the achievable rate region grows exponentially with n. Second, the number of possibilities for the sets of messages that each receiver can decode, which we collectively

refer to as decoding configurations, is on average super exponential in the problem size. Therefore, it is imperative to determine whether some of the composite messages and their corresponding rates, and some of the decoding configurations can be removed from the coding scheme and the computation of the rate region achieved by it.

The main contribution of this paper is to introduce simplification techniques for composite coding that address the aforementioned issues. In Section IV-A, we propose a pairwise test between any two composite index rates, based on examining the decoding constraints of composite coding, to determine whether one of them can be removed without sacrificing achievable rate performance. Besides, we present a heuristic algorithm that can reduce the number of composite indices even further, albeit with possible performance loss. For a given problem, sufficiency of the heuristic algorithm can be checked by comparing the obtained achievable rates with necessary conditions on the rates, such as those given by the maximal acyclic induced subgraph (MAIS) bound [8] or a generally tighter polymatroidal bound [5]. In Section IV-B, we introduce a minimal *natural* decoding configuration and show that it suffices to consider only decoding configurations that are supersets of the natural decoding configuration. In Section V, we present numerical results for all proposed simplification methods, which demonstrate their efficacy.

II. System Model

Consider the index coding problem with n messages, $x_i \in \{0,1\}^{t_i}, i \in [n] \doteq \{1, 2, \ldots, n\}$. For brevity, when we say message i, we mean message x_i. Let X_i be the random variable corresponding to x_i. We assume that $X_1, \ldots, X_n$ are uniformly distributed and independent of each other. For any $K \subseteq [n]$, we use the shorthand notation $\mathbf{x}_K$ to denote the collection of messages whose index is in K. By convention $\mathbf{x}_\emptyset = \emptyset$. There is a single server that contains all messages $\mathbf{x}_{[n]}$ and is connected to all receivers via a noiseless broadcast link of normalized capacity $C = 1$. Let y be the output of the server, which is a function of $\mathbf{x}_{[n]}$. There are n receivers, where receiver $i \in [n]$ wishes to obtain x_i and knows $\mathbf{x}_{A_i}$ as side information for some $A_i \subseteq [n] \setminus \{i\}$.

We define a $(\mathbf{t}, r) = ((t_i, i \in [n]), r)$ *index code* by

- an encoder $\phi : \prod_{i \in [n]} \{0,1\}^{t_i} \to \{0,1\}^r$, which maps the messages $\mathbf{x}_{[n]}$ to an r-bit sequence y, and
- n decoders, one for each receiver $i \in [n]$, such that $\psi_i : \{0,1\}^r \times \prod_{k \in A_i} \{0,1\}^{t_k} \to \{0,1\}^{t_i}$ maps the received sequence y and the side information $\mathbf{x}_{A_i}$ to $\hat{x}_i$.

We say that a rate tuple $\mathbf{R} = (R_i, i \in [n])$ is achievable if for every $\epsilon > 0$, there exists a $(\mathbf{t}, r)$ index code such that

$$R_i \leq \frac{t_i}{r}, \quad i \in [n], \tag{1}$$

and the probability of error satisfies

$$\mathsf{P}\{(\hat{X}_1, \ldots, \hat{X}_n) \neq (X_1, \ldots, X_n)\} \leq \epsilon. \tag{2}$$

The capacity region $\mathscr{C}$ of this index coding problem is the closure of the set of all achievable rate tuples $\mathbf{R}$. The optimal broadcast rate β is the reciprocal of the symmetric capacity

$$\beta = \frac{1}{C_{\mathrm{sym}}} = \frac{1}{\max\{R : (R, \ldots, R) \in \mathscr{C}\}}. \tag{3}$$

We will compactly represent an index coding instance by a sequence $(i | j \in A_i), i \in [n]$. For example, for $A_1 = \emptyset$, $A_2 = \{3\}$, and $A_3 = \{2\}$, we write $(1|-), (2|3), (3|2)$.

III. A BRIEF REVIEW OF COMPOSITE CODING

We briefly review the composite coding scheme from [5], [6]. For clarity of exposition, we first consider the case in which each receiver performs the decoding operation using a fixed decoding configuration (fixed decoding message sets). Then in Subsection III-B, we discuss the case of a variable decoding configuration. The motivation for a simplified composite coding scheme is described in Section III-C.

A. Fixed Decoding Configuration

Let $r \in \mathbb{N}$, $t_i = \lceil rR_i \rceil$, $i \in [n]$, and $s_K = \lceil rS_K \rceil$, $K \subseteq [n]$. Here R_i is the rate of message i and S_K is the rate of composite index K (to be defined shortly). We set $S_\emptyset = 0$ by convention.

Codebook generation. *Step 1.* For each $K \subseteq [n]$, generate a composite index $w_K(\mathbf{x}_K)$ drawn uniformly at random from $[2^{s_K}]$. For brevity, when we say composite index K, we mean composite index $w_K(\mathbf{x}_K)$. *Step 2.* Generate the codeword $y(w_K, K \subseteq [n])$ drawn uniformly at random from $[2^r]$. The codebook $\{(w_K(\mathbf{x}_K), K \subseteq [n]), y(w_K, K \subseteq [n])\}$ is revealed to all parties.

Encoding. To communicate messages $\mathbf{x}_{[n]}$, the server transmits $y(w_K(\mathbf{x}_K), K \subseteq [n])$.

Receiver i decodes for a subset of messages indexed by $D_i \subseteq [n] \setminus A_i$, such that $i \in D_i$. The tuple of decoding message sets is denoted by $\mathbf{D} = (D_i, i \in [n])$ and referred to as the decoding configuration. In this subsection, we assume that the decoding configuration $\mathbf{D}$ is fixed.

Decoding. *Step 1.* Receiver i finds the unique composite index tuple $(\hat{w}_K, K \subseteq [n])$ such that $y = y(\hat{w}_K, K \subseteq [n])$. If there is more than one such tuple, it declares an error. *Step 2.* Assuming that $(\hat{w}_K, K \subseteq [n])$ is correct, receiver i finds the unique message tuple $\hat{\mathbf{x}}_{D_i}$ such that $\hat{w}_K = w_K(\hat{\mathbf{x}}_K)$ for every $K \subseteq D_i \cup A_i$. If there is more than one such tuple, it declares an error.

The achievable rate tuple of this coding scheme [5] is summarized as follows. The proof can be found in [2].

Proposition 1: A rate tuple $\mathbf{R}$ is achievable for the index coding problem $(i|A_i)$, $i \in [n]$, under a given decoding configuration $\mathbf{D}$ if

$$\sum_{j \in L} R_j < \sum_{\substack{K \subseteq D_i \cup A_i, \\ K \cap L \neq \emptyset}} S_K, \quad \forall L \subseteq D_i, i \in [n], \tag{4}$$

$$\sum_{K \subseteq [n], K \nsubseteq A_i} S_K < 1, \qquad \forall i \in [n], \tag{5}$$

for some $S_K \geq 0$, $K \subseteq [n]$.

Here the inequalities in (4) signify the second-step decoding constraints for the messages in D_i. The inequalities in (5) signify the first-step decoding constraints for the composite indices to be recovered from the server output, except those that can be generated from receiver side information.

B. Variable Decoding Configuration

In this part, we allow the decoding configuration to vary and compute the achievable rate region across all such configurations. Let $\mathcal{D}_i = \{D_i \subseteq [n] \setminus A_i : i \in D_i\}$ be the set of all possible decoding message sets at receiver i. Any decoding configuration $\mathbf{D} = (D_i, i \in [n]) \in \mathcal{D} = \prod_{i=1}^n \mathcal{D}_i$. For each decoding configuration $\mathbf{D}$, define message rates $R_i(\mathbf{D}), i \in [n]$, fractional server capacity $C(\mathbf{D}) \leq C = 1$, and composite index rates $S_K(\mathbf{D})$, $K \subseteq [n]$ as a function of $\mathbf{D}$. Using the composite coding scheme presented in Section III-A, and combining the corresponding rate tuples for each $\mathbf{D}$, the following was established in [6].

Proposition 2: A rate-capacity tuple $\mathbf{R}$ is achievable for the index coding problem $(i|A_i)$, $i \in [n]$, if

$$R_i = \sum_{\mathbf{D}} R_i(\mathbf{D}), \qquad \forall i \in [n], \tag{6}$$

$$1 = \sum_{\mathbf{D}} C(\mathbf{D}), \tag{7}$$

for some $R_i(\mathbf{D})$, $S_K(\mathbf{D})$, and $C(\mathbf{D})$ such that for every $\mathbf{D}$

$$\sum_{j \in L} R_j(\mathbf{D}) < \sum_{\substack{K \subseteq D_i \cup A_i, \\ K \cap L \neq \emptyset}} S_K(\mathbf{D}), \forall L \subseteq D_i, i \in [n], \tag{8}$$

$$\sum_{K \subseteq [n], K \nsubseteq A_i} S_K(\mathbf{D}) < C(\mathbf{D}), \qquad \forall i \in [n]. \tag{9}$$

For any $\mathbf{D}$ and corresponding $C(\mathbf{D}) \leq 1$, we use $\mathscr{R}(\mathbf{D})$ to denote the rate region achievable through (8) and (9). Note that if $C(\mathbf{D}) = 1$, $\mathscr{R}(\mathbf{D})$ simply denotes the rate region given by Proposition 1 (achievable through (4) and (5)).

C. Computation Complexity Issues

To motivate the problem, consider the achievable rate region for a given decoding configuration $\mathbf{D}$ in Proposition 1, which has to be computed by eliminating the intermediate variables S_K, $K \subseteq [n]$, from (4) and (5). However, the complexity of FME is between linear and doubly-exponential in the number of variables to eliminate [9]. Since the number of S_K variables is $2^n - 1$ and is already exponential in n, without any simplification, the FME computational complexity for problems with large n is prohibitive. If only the optimal

broadcast rate β or the sum capacity rather than the whole capacity region $\mathscr{C}$ is of interest, LP can be used, which typically has a much lower computational complexity than FME. However, it is still not clear how to choose the decoding configuration.

If a "lucky" $\mathbf{D}$ yields achievable rates that match necessary conditions given by the MAIS or polymatroidal bound, the capacity (region) is established. If there is a gap, a tighter result may be obtained by recomputing Proposition 1 through varying $\mathbf{D}$ or by computing Proposition 2 using a collection of $\mathbf{D}$.[1] For a given index coding problem, starting from $D_i = \{i\}$ and taking into account the side information A_i, the number of decoding configurations to consider grows super-exponentially in n as

$$|\mathcal{D}| = \left| \prod_{i=1}^{n} \mathcal{D}_i \right| = 2^{n^2 - \sum_{i=1}^{n} |A_i| - n}.$$

It is therefore imperative to reduce the number of S_K variables and the search range over $\mathbf{D}$ as much as possible.

IV. Simplified Composite Coding

Our first contribution is to show in Subsection IV-A how composite coding scheme can be simplified for a fixed decoding configuration. Our second contribution is to show in Subsection IV-B how one can ignore some decoding configurations without affecting the achievable rate region.

A. Reducing Composite Indices

In this subsection, our goal is to determine whether it is possible to remove S_K for some $K \subseteq [n]$ in Proposition 1 without affecting its performance. Our proposed simplification involves pairwise comparison of any two composite index rates, say S_K and $S_{K'}$, $K, K' \subseteq [n]$. Roughly speaking, if S_K appears in less second-step inequalities (4) and in more first-step inequalities (5) relative to $S_{K'}$, then it can be safely removed from computations.

More formally, fix an ordering for all inequalities identified in Proposition 1. Enumerate all inequalities in (4) by indices $\ell_2 \in [m_2]$ and all inequalities in (5) by indices $\ell_1 \in [m_1]$. Note that including possibly inactive inequalities, there is one first-step inequality and $2^{|D_i|} - 1$ second-step inequalities due to each receiver. Now, assume S_K respectively appears in first-step and second-step inequalities that are identified by indices $Q_1(K) \subseteq [m_1]$ and $Q_2(K) \subseteq [m_2]$ and $S_{K'}$ respectively appears in first-step and second-step inequalities identified by $Q_1(K') \subseteq [m_1]$ and $Q_2(K') \subseteq [m_2]$.

Theorem 1: If $Q_2(K) \subseteq Q_2(K')$ and $Q_1(K') \subseteq Q_1(K)$, then S_K can be removed from the inequalities identified by Proposition 1 without affecting the resulting rate region.

Proof: Assume that $S_K = a$ and $S_{K'} = b$ in the full set of expressions in Proposition 1. Since $Q_2(K) \subseteq Q_2(K')$, whenever S_K appears in any second-step inequality, so does $S_{K'}$. Therefore, transferring the rate of S_K to $S_{K'}$, that is setting $S_K = 0$ and $S_{K'} = a + b$, cannot decrease message

rates. Since $Q_1(K') \subseteq Q_1(K)$, whenever $S_{K'}$ appears in any first-step inequality, so does S_K. Hence, transferring the rate of S_K to $S_{K'}$ cannot result in an invalid composite index rate assignment in (5) and one can remove S_K from expressions in Proposition 1 without affecting performance. ∎

Note that for any given problem, simplification via Theorem 1 requires no more than $(2^n - 1)^2$ pairwise comparisons.

Example 1: Consider the index coding problem $(1|4)$, $(2|3,4)$, $(3|1,2)$, and $(4|2,3)$. Set $\mathbf{D}$ as $D_j = [n] \setminus A_j$, $j = 2, 3, 4$, and $D_1 = \{1\}$. We compare the relative presence of $S_{\{1,3\}}$ and $S_{\{1,2,3\}}$ in the decoding inequalities. Since for any $i \in [n]$, $\{1,3\} \not\subseteq A_i$ and $\{1,2,3\} \not\subseteq A_i$, $S_{\{1,3\}}$ and $S_{\{1,2,3\}}$ are present in all first-step inequalities. Writing second-step decoding inequalities of Proposition 1 yields

$$R_1 < S_{\{1\}} + S_{\{1,4\}},$$
$$R_1 < S_{\{1\}} + S_{\{1,2\}} + S_{\{1,3\}} + S_{\{1,2,3\}} + \cdots + S_{[n]},$$
$$R_2 \overset{(a)}{<} S_{\{2\}} + S_{\{1,2\}} + S_{\{2,3\}} + S_{\{1,2,3\}} + \cdots + S_{[n]},$$
$$\cdots\cdots$$
$$R_1 + R_4 < S_{\{1\}} + S_{\{1,2\}} + \cdots + S_{\{4\}} + \cdots + S_{[n]}.$$

Now we observe that $S_{\{1,2,3\}}$ is present in one more second-step decoding inequality compared with $S_{\{1,3\}}$ (in the inequality (a) above, $S_{\{1,2,3\}}$ is present, but $S_{\{1,3\}}$ is not). Hence, $S_{\{1,3\}}$ can be removed without affecting the achievable rate performance. Continuing this procedure for all distinct $K, K' \subseteq [n]$, the only remaining composite index rates are S_K, $K \in \{\{1\}, \{2\}, \{1,2\}, \{3\}, \{2,3\}, \{4\}, \{1,4\}, \{3,4\}, \{1,2,3,4\}\}$, reducing the number of rate variables from 15 to 9.[2]

1) Heuristic Reduction: We present a heuristic algorithm that can result in further reductions in the number of composite index rates, albeit with possible performance loss. It is based upon Theorem 1 and allows reduction even when some of the conditions are violated in a controlled manner.

Specifically, we still require that $Q_2(K) \subseteq Q_2(K')$, which will ensure that the corresponding composite index $w_{K'}$ is at least as useful as w_K in the second-step decoding for all receivers. Now, note that $Q_1(K') \subseteq Q_1(K)$ in Theorem 1 implies that there does not exist any receiver for which $S_{K'}$ appears in their first-step decoding inequalities while S_K does not. We define the subset of receivers who know all messages in $\mathbf{x}_K$, but not all messages in $\mathbf{x}_{K'}$ as follows,

$$M(K, K') = \{j \in [n] : K \subseteq A_j, K' \not\subseteq A_j\}. \tag{10}$$

The condition $Q_1(K') \subseteq Q_1(K)$ of Theorem 1 implies $M(K, K')$ is empty. In the heuristic, we allow S_K to be removed in comparison to $S_{K'}$ even when $M(K, K') \neq \emptyset$, provided that $w_{K'}$ is useful in the second-step decoding for all receivers in $M(K, K')$. Intuitively, although $w_{K'}$ has to be decoded in the first-step decoding at those receivers, it is not an interference and is useful in their second-step decoding.

Simplification via Algorithm 1 can be implemented in such a way that no more than $(2^n - 1)^2$ pairwise comparisons (same

---

[1] If a discrepancy exists after exhausting composite coding, one should examine other coding schemes or look for tighter necessary conditions (e.g., non-Shannon-type inequalities [10]), which is beyond the scope of our paper.

[2] Further reductions may be possible if we first remove inactive inequalities and then apply Theorem 1, but we will not pursue this further in this paper.

as Theorem 1) are required for any n-message index coding problem. The algorithm is shown below.

Algorithm 1: Heuristic composite index rate reduction

Input : Two composite coding rates S_K and $S_{K'}$ that appear respectively in (5) and (4) indexed by $Q_1(K), Q_2(K)$ and by $Q_1(K'), Q_2(K')$. Receiver subset $M(K, K')$ in (10).

Output: A flag indicating whether S_K can be removed from the expressions of Proposition 1.

1 **If** Theorem 1 holds (i.e., $Q_2(K) \subseteq Q_2(K')$ and $M(K, K') = \emptyset$), then flag = TRUE.

2 **Else if** $Q_2(K) \subseteq Q_2(K')$, $M(K, K') \neq \emptyset$, and $K' \subseteq D_j \cup A_j, \forall j \in M(K, K')$, then flag = TRUE.

3 **Else** flag = FALSE.

Applying Algorithm 1 to the problem in Example 1 with the same **D**, all composite index rates, but $S_{\{1,4\}}$ and $S_{\{1,2,3,4\}}$ can be eliminated. The achievable rate region of Proposition 1 using only $S_{\{1,4\}}$ and $S_{\{1,2,3,4\}}$ coincides with the MAIS bound, which establishes the capacity region and confirms sufficiency of Algorithm 1 for this problem.

Note that Theorem 1 and Algorithm 1 can be used to remove composite index rates $S_K(\mathbf{D})$ for any **D** from the expressions of Proposition 2 as well.

B. Reducing Decoding Configurations

The main idea behind removing some decoding configurations is as follows. Given an index coding problem $(i|A_i)$, $i \in [n]$, if $A_i \subseteq A_j \cup D_j$ for some $j \neq i$, then receiver j already knows or will know more than receiver i does. Therefore, one can update $D_j \leftarrow D_j \cup \{i\}$ at no cost to the achievable rate region. This is the basis for iteratively building a decoding configuration in Algorithm 2, which we will refer to as the *natural decoding configuration* and denote it by $\underline{\mathbf{D}}$. Note that one can equivalently update $\underline{D}_j$ in Step 2 of the algorithm as follows $\underline{D}_j \leftarrow \underline{D}_j \cup \underline{D}_i$.

Algorithm 2: Natural decoding configuration

Input : Index coding problem $(i|A_i)$, $i \in [n]$.

Output: Natural decoding configuration $\underline{\mathbf{D}} = (\underline{D}_i, i \in [n])$.

1 Initialize $\underline{D}_i = \{i\}$, $i \in [n]$.

2 For as long as there exists $i, j \in [n]$ such that $A_i \subseteq A_j \cup \underline{D}_j$, update $\underline{D}_j \leftarrow \underline{D}_j \cup \{i\}$. If no such i, j exist, terminate the algorithm.

Theorem 2: Let $\mathbf{D} = (D_i, i \in [n])$ be a decoding configuration such that $\underline{D}_k \setminus D_k \neq \emptyset$ for some $k \in [n]$. Then there exists another decoding configuration $\mathbf{D}' = (D'_i, i \in [n])$, for which $\underline{D}_i \subseteq D'_i$ for all $i \in [n]$, such that for any $C(\mathbf{D}) = C(\mathbf{D}') \leq 1$, $\mathscr{R}(\mathbf{D}) \subseteq \mathscr{R}(\mathbf{D}')$.

Theorem 2 can be proved by applying the following lemma at most n times.

Lemma 1: Let $\mathbf{D} = (D_i, i \in [n])$ be a decoding configuration such that $\underline{D}_k \setminus D_k \neq \emptyset$ for some $k \in [n]$. Then there

exists some $\mathbf{D}' = (D'_i, i \in [n])$ satisfying

$$\underline{D}_i \subseteq D'_i, \quad i = k, \tag{11}$$

$$D'_i = D_i, \quad i \neq k, \tag{12}$$

such that for any $C(\mathbf{D}) = C(\mathbf{D}') \leq 1$, $\mathscr{R}(\mathbf{D}) \subseteq \mathscr{R}(\mathbf{D}')$.

Proof: Let $\underline{D}_k \setminus D_k = \{k_1, \cdots, k_m\}$. Assume that $k_1, \cdots, k_m$ are added to $\underline{D}_k$ using Algorithm 2 in order. Therefore, for $\ell \in [m]$ we have

$$A_{k_\ell} \subseteq A_k \cup D_k \cup \{k_1, \cdots, k_{\ell-1}\} \tag{13}$$

$$\subseteq A_k \cup D_k \cup D_{k_1} \cup \cdots \cup D_{k_{\ell-1}}. \tag{14}$$

Define $\mathbf{D}' = (D'_i, i \in [n])$ by

$$D'_i = \begin{cases} D_i, & i \neq k, \\ D_k \cup D_{k_1} \cup \cdots \cup D_{k_m}, & i = k. \end{cases} \tag{15}$$

Since $k_\ell \in D_{k_\ell}$, $\ell \in [m]$, we have $\underline{D}_k \subseteq D'_k = D_k \cup D_{k_1} \cup \cdots \cup D_{k_m}$. So it remains to prove that $\mathscr{R}(\mathbf{D}) \subseteq \mathscr{R}(\mathbf{D}')$.

For any achievable rate tuple $\mathbf{R}(\mathbf{D}) \in \mathscr{R}(\mathbf{D})$, there exists some $(S_K(\mathbf{D}), K \subseteq [n])$ such that they satisfy (8) and (9) with **D** and $C(\mathbf{D})$. Now we are going to show that $\mathbf{R}(\mathbf{D}) \in \mathscr{R}(\mathbf{D}')$ by showing that $\mathbf{R}(\mathbf{D})$ and $(S_K(\mathbf{D}), K \subseteq [n])$ also satisfy (8) and (9) with $\mathbf{D}'$ and $C(\mathbf{D}')$.

As $C(\mathbf{D}) = C(\mathbf{D}')$, the inequalities in (9) do not depend on whether **D** or $\mathbf{D}'$ is used. Also, for $i \neq k$, $D'_i = D_i$ and the corresponding inequalities in (8) are the same. Therefore, it suffices to show that $\mathbf{R}(\mathbf{D})$ and $(S_K(\mathbf{D}), K \subseteq [n])$ satisfy

$$\sum_{j \in L'} R_j(\mathbf{D}) < \sum_{\substack{K \subseteq D'_k \cup A_k, \\ K \cap L' \neq \emptyset}} S_K(\mathbf{D}), \quad \forall L' \subseteq D'_k. \tag{16}$$

Consider partitioning L' as $L' = L \cup L_1 \cup \cdots \cup L_m$, where

$$L \subseteq D_k,$$

$$L_\ell \subseteq D_{k_\ell} \setminus ((\bigcup_{1 \leq j \leq \ell-1} D_{k_j}) \cup D_k), \quad \forall \ell \in [m].$$

Sets $L, L_1, \cdots, L_m$ are disjoint and the LHS of (16) is

$$\sum_{j \in L'} R_j(\mathbf{D}) = \sum_{j \in L} R_j(\mathbf{D}) + \sum_{\ell \in [m]} \sum_{j \in L_\ell} R_j(\mathbf{D}). \tag{17}$$

For any $\ell \in [m]$, as $L_\ell \subseteq D_{k_\ell}$, according to (8), we can write

$$\sum_{j \in L_\ell} R_j(\mathbf{D}) < \sum_{\substack{K \subseteq A_{k_\ell} \cup D_{k_\ell}, \\ K \cap L_\ell \neq \emptyset}} S_K(\mathbf{D})$$

$$< \sum_{\substack{K \subseteq A_k \cup D_k \cup D_{k_1} \cup \cdots \cup D_{k_{\ell-1}} \cup D_{k_\ell}, \\ K \cap L_\ell \neq \emptyset}} S_K(\mathbf{D})$$

$$= \sum_{\substack{K \subseteq A_k \cup D_k \cup D_{k_1} \cup \cdots \cup D_{k_{\ell-1}} \cup D_{k_\ell}, \\ K \not\subseteq A_k \cup D_k \cup D_{k_1} \cup \cdots \cup D_{k_{\ell-1}}, \\ K \cap L_\ell \neq \emptyset}} S_K(\mathbf{D}), \tag{18}$$

where the second inequality follows from (13). Again, according to (8), we have

$$\sum_{j \in L} R_j(\mathbf{D}) < \sum_{\substack{K \subseteq D_k \cup A_k, \\ K \cap L \neq \emptyset}} S_K(\mathbf{D}). \tag{19}$$

Summing up (19) and (18) for $\ell \in [m]$ yields the desired result in (16). ■

Example 2: Consider the same problem as in Example 1. We start with $\underline{D}_i = \{i\}$, $i \in [n]$. Since $A_1 \subset A_2$, receiver 2 can also decode message 1 at no cost. Hence, $\underline{D}_2 = \{1, 2\}$. Similarly, since $A_4 \subset A_3 \cup \underline{D}_3$, $\underline{D}_3 = \{3, 4\}$, and since $A_1 \subset A_4 \cup \underline{D}_4$, $\underline{D}_4 = \{1, 4\}$. Algorithm 2 stops here. This decoding configuration coincides with the one in Example 1 and is sufficient to achieve the capacity region for this problem.

V. NUMERICAL RESULTS

To show the efficacy of our proposed simplifications, we first consider all 218 and 9,608 non-isomorphic problems with $n = 4$ and 5, respectively. For each problem, we only use the natural decoding configuration $\underline{\mathbf{D}}$ obtained through Algorithm 2. Hence for each problem, the reduction in the number of decoding configurations is $(100 - \frac{100}{|\mathcal{D}|})\%$, where $|\mathcal{D}|$ was given in Section III-C. We then apply Algorithm 1 to all possible rate pairs S_K and $S_{K'}$ in Proposition 1. If for a given problem, Algorithm 1 only retains m out of $2^n - 1$ such rates, the reduction rate for that problem is $(100 - \frac{100m}{2^n - 1})\%$. Finally, we use FME [11] to eliminate the only m remaining $S_{K'}$ variables and compute the rate region of Proposition 1. All tests in this section were run on an Apple iMac 4GHz Intel Core i7 with 16 GB memory and using Matlab® R2017b.

According to the test results, the natural decoding configuration $\underline{\mathbf{D}}$ together with Algorithm 1 are sufficient to establish the capacity region for all problems with $n = 4$ and 5. Table I indicates the average computational savings. Although not strictly needed, the average reduction using the more conservative Theorem 1 is shown for comparison. As shown in the last column, it takes on average 5.6 and 11.63 times longer to eliminate all $(2^n - 1)$ S_K variables in the non-reduced problems with $n = 4$ and 5, respectively, as compared to eliminating only the few rates retained by Algorithm 1. Finally, the distribution of composite rate reduction among problems is quite good. For $n = 4$, only in 7 problems a maximum of $m = 3$ rates were retained, where for all other problems, $m \leq 2$. For $n = 5$, in only 31 of 9,608 problems, $m = 7$ to at most $m = 10$ composite rates were used.

In Table II, we show results for some problems with $n = 6$. We have chosen the set of 10,634 potential canonical problems as described in [2]. We only compute the achievable symmetric rate of the composite coding. For a vast majority of problems, $\underline{\mathbf{D}}$ together with Algorithm 1 were sufficient to obtain β. In 114 problems, Theorem 1 was required. In 460 problems, the natural decoding configuration was not sufficient to achieve β. Instead, we grouped all supersets of $\underline{\mathbf{D}}$ into batches of $\mathbf{D}$ of size 256, labelled as B_1 to B_e and were able to achieve β using Algorithm 1 and Proposition 2 using a certain batch B_b. If b batches are tested for a problem until finding β, the decoding configuration reduction is $(100 - \frac{100 \times 256b}{|\mathcal{D}|})\%$. Finally, for 110 problems we were not able to achieve β using composite coding with any of the methods described above.

The efficacy of the reduction techniques is not limited to $n \leq 6$ or small number of decoding configura-

tions. As an illustration, for the problem $(1|2, 3, 4, 5, 7)$, $(2|1, 3, 5, 6, 7)$, $(3|1, 2, 4, 6, 7)$, $(4|1, 2, 5, 7)$, $(5|2, 3, 6, 7)$, $(6|1, 3, 4, 7)$, $(7|-)$, applying Algorithm 1 to Proposition 2 to compute the achievable symmetric rate using all $4,096$ supersets of $\underline{\mathbf{D}}$ takes 128 seconds to obtain $R_{\mathrm{Sym}} = 1/3.25$. The total number of variables in the LP (including all $R_i(\mathbf{D})$ variables) is around $58,000$. In comparison, if Proposition 2 without any composite rate reduction is used across all $4,096$ decoding configurations, the program takes about 31 minutes (15 times slower) to compute the same symmetric rate. The total number of variables in the LP is around $553,000$. Both methods used sparse matrix representation in Matlab®.

Table I
RESULTS FOR ALL PROBLEMS WITH $n = 4$ AND 5.

Problem Size	Ave. dec. config. reduction	Ave. S_K reduction by Alg. 1	Ave. S_K reduction by Thm. 1	Not-reduced time / Alg. 1 time
$n = 4$	96.13%	89.6%	63.06%	560%
$n = 5$	99.64%	92.36%	71.42%	1163%

Table II
RESULTS FOR 10,634 CANONICAL PROBLEMS WITH $n = 6$.

Number of Problems	Ave. dec. config. reduction	Ave. S_K reduction by Alg. 1	Ave. S_K reduction by Thm. 1	How β was obtained
9,950	99.97%	90.90%	71.93%	$\mathbf{D}$ & Alg.1
114	99.99%	-	73.88%	$\mathbf{D}$ & Thm. 1
460	96.70%	84.99%	68.21%	Batches of $\mathbf{D}$ & Alg.1
110	-	-	-	Unable to achieve β

REFERENCES

[1] Y. Birk and T. Kol, "Informed-source coding-on-demand (ISCOD) over broadcast channels," in *IEEE INFOCOM*, Mar. 1998, pp. 1257–1264.
[2] F. Arbabjolfaei and Y.-H. Kim, *Fundamentals of Index Coding*. To appear in Foundations and Trends in Communications and Information Theory, 2018.
[3] A. Blasiak, R. Kleinberg, and E. Lubetzky, "Lexicographic products and the power of non-linear network coding," in *Proc. Symp. on Foundations of Computer Science (FOCS)*, 2011, pp. 609–618.
[4] H. Maleki, V. R. Cadambe, and S. A. Jafar, "Index coding: An interference alignment perspective," *IEEE Trans. Inf. Theory*, vol. 60, no. 9, pp. 5402–5432, 2014.
[5] F. Arbabjolfaei, B. Bandemer, Y.-H. Kim, E. Sasoglu, and L. Wang, "On the capacity region for index coding," in *Proc. IEEE Int. Symp. on Information Theory (ISIT)*, 2013, pp. 962–966.
[6] Y. Liu, P. Sadeghi, F. Arbabjolfaei, and Y.-H. Kim, "On the capacity for distributed index coding," in *Proc. IEEE Int. Symp. on Information Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 3055–3059.
[7] A. El Gamal and Y.-H. Kim, *Network Information Theory*. Cambridge: Cambridge University Press, 2011.
[8] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, "Index coding with side information," *IEEE Trans. Inf. Theory*, vol. 57, pp. 1479–1494, 2011.
[9] D. Monniaux, "Quantifier elimination by lazy model enumeration," in *International Conference on Computer Aided Verification*. Springer, 2010, pp. 585–599.
[10] H. Sun and S. A. Jafar, "Index coding capacity: How far can one go with only shannon inequalities?" *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3041–3055, 2015.
[11] I. B. Gattegno. FME-IT package for MATLAB. [Online]. Available: http://www.ee.bgu.ac.il/~fmeit/download.html