On the Complementary Index Coding Problem

M. A. R. Chaudhry¹, Z. Asad¹, A. Sprintson¹, and M. Langberg²

¹Department of Electrical & Computer Engineering, Texas A & M University.
²Computer Science Division, Open University of Israel.
email: {masadch,zasad,spalex}@tamu.edu, mikel@openu.ac.il.

Abstract—The Index Coding problem is one of the basic problems in wireless network coding. In this problem, a server needs to deliver a set \( P \) of packets to several clients through a noiseless broadcast channel. Each client needs to obtain a certain subset of \( P \) and has prior side information about a different subset of \( P \). The objective is to satisfy the requirements of all clients with the minimum number of transmissions. Recently, it was shown that the Index Coding problem is NP-hard. Furthermore, this problem was shown to be hard to approximate under a widely accepted complexity assumption.

In this paper, we consider a complementary problem whose goal is to maximize the number of saved transmissions, i.e., the number of transmissions that are saved by combining packets compared to the solution that does not involve coding. We refer to this problem as the Complementary Index Coding problem. It turns out that the complementary problem can be approximated in certain cases of practical importance.

We consider the multiple unicast and multiple multicast scenarios. In the multiple unicast scenario, each packet is requested by a single client; while in the multiple multicast scenario, each packet can be requested by several clients. For the multiple unicast scenario, we present approximation algorithms for finding scalar and vector linear solutions. For the multiple multicast scenario, we show that finding an approximation solution is NP-hard.

I. INTRODUCTION

The Index Coding problem [1], [2] is one of the basic problems in the wireless network coding. Recently, it has attracted a significant attention from the research community (see e.g., [3]–[6] and references therein).

An instance of the Index Coding problem comprises of a server, a set \( X = \{c_1, \ldots, c_m\} \) of \( m \) wireless clients, and a set \( P = \{p_1, \ldots, p_n\} \) of \( n \) packets that need to be delivered to the clients. Each client is interested in a certain subset of packets available at the server, and has a (different) subset of packets as side information. The server can transmit the packets to clients via a noiseless wireless channel. The goal is to find a transmission scheme that requires the minimum number \( \mu \) of transmissions to satisfy the requests of all clients.

Figure 1 depicts a simple instance of the Index Coding problem with a server that needs to deliver five packets \( P = \{p_1, \ldots, p_5\} \) to five clients. Each client requires a unique packet in \( P \) and has access to a subset of \( P \). It can be verified that the demands of all clients can be satisfied by broadcasting three packets: \( p_1 + p_2, p_5 + p_4, \) and \( p_5 \) (all additions are over \( GF(2) \)). Note that with the traditional approach (without coding) all five packets \( p_1, \ldots, p_5 \) need to be transmitted.

The research on the Index Coding problem can be classified into two main directions. The first direction focuses on achievable rate bounds, as well as on the connections between the Index Coding problem and the Network Coding problem [7]–[9]. The second direction focuses on analyzing the computational complexity of the Index Coding problem as well as developing heuristic approaches to this problem [3]–[6], [10]. In particular, the Index Coding problem has been shown to be NP-hard (in the linear setting). Moreover, finding an approximation solution for this problem is hard under a certain complexity assumption [10].

In this paper we focus on the Complementary Index Coding (CIC) problem. In this problem, instead of minimizing the number of transmissions \( \mu \), our goal is to maximize the number of “saved” transmissions, i.e., \( n - \mu \), where \( n \) is the number of packets that need to be delivered to the clients. Thus, the CIC problem seeks to maximize the benefit obtained by employing the coding technique, e.g., for the problem instance shown in Figure 1, two transmissions can be saved by using coding. Note that, if \( OPT_{IC} \) is the optimum of the Index Coding problem and \( OPT_{CIC} \) is the optimum of the Complementary Index Coding problem, then it holds that \( |OPT_{CIC}| = n - |OPT_{IC}| \). This implies that the CIC problem is also NP-hard. However, as we show in this paper, the CIC problem can be successfully approximated in many cases of practical importance.

There are several well-known optimization problems which are complementary to one another. Examples include the Vertex Cover and Independent Set problems [11] and the problems of coloring (finding the minimum chromatic number of a graph) and that of color saving (see e.g., [12]). In the context of approximation, complementary problems may have...
a significantly different behavior as a good approximation algorithm to one of the problems does not imply one for its companion (e.g. [13]).

A. Contributions

In this paper we study scalar and vector linear solutions for the Complementary Index Coding problem. In a vector-linear solution, each packet can be split into smaller sub-packets, such that sub-packets originated from different packets can be combined together. In the scalar-linear solution, the packets cannot be split. We consider two scenarios for Complementary Index Coding, i.e., the Multiple Unicast scenario and the Multiple Multicast scenario. In the multiple unicast scenario, each packet is requested by a single client; in the multiple multicast scenario, each packet can be requested by more than one client. For the multiple unicast scenario, we present approximation algorithms for finding scalar and vector linear solutions that achieve approximation ratios of \( \Omega(\sqrt{m} \cdot \log n \cdot \log \log n) \) and \( \Omega(\log n \cdot \log \log n) \), respectively. For the multiple multicast scenario, we show that finding an approximation solution is NP-hard.

Our algorithms for multiple unicast strongly build on the notion of a dependency graph \( G(V,E) \) that captures the combinatorial properties of the IC and CIC problems. Loosely speaking, we show that any cycle in this graph \( G \) enables to save a single transmission. Thus, finding many such disjoint cycles, via the algorithms presented in [14] and [15], one may save several transmission and obtain an approximation to Problem CIC. To quantify the amount of transmissions saved in this manner we tie the value of the optimal coding scheme to the feedback vertex set of \( G(V,E) \). For multiple multicast, we show a reduction from the CIC problem to the Independent Set problem, implying hardness of approximation in this case.

The rest of this paper is organized as follows. In Section II we present the network model. Section III introduces the notion of the dependency graph and related problems. Sections IV and V focus on multiple unicast and multiple multicast cases, respectively. Conclusions are presented in Section VI.

II. Model

An instance of the Index Coding (IC) problem includes a server \( s \), a set of \( m \) wireless clients \( X = \{c_1, \ldots, c_m\} \), and a noiseless broadcast channel. The server holds a set of \( n \) packets, \( P = \{p_1, \ldots, p_n\} \), that need to be transmitted to the clients. A client \( c_i \in X \) is represented by a pair \((w_i, H(c_i))\), where \( w_i \in P \) is the packet required by client \( c_i \) and \( H(c_i) \subseteq P \) is the side information set available to client \( c_i \).

Note that our assumption that each client requires a single packet does not limit the generality of the problem. Indeed, a client that requires more than one packet can be represented by multiple clients that share the same side information set, but require different packets.

In a scalar-linear solution, each packet is considered to be an element of the Galois field of order \( q \), i.e., \( p_i \in GF(q) \). A scalar linear solution includes \( \mu \) transmissions such that the packet \( p^i = \sum_{j=1}^{n} g^j \cdot p_j \) transmitted at iteration \( i \), \( 1 \leq i \leq \mu \) is a linear combination of packets in \( P \), where \( g^i = \{g^j_i\} \in GF(q)^n \) is the encoding vector for iteration \( i \). To decode packet \( w_i \), client \( c_i \) uses a linear decoding function \( r_i \), such that \( w_i = r_i(p^1, \ldots, p^n, H(c_i)) \). The goal of the Index Coding problem is to find the minimum value of \( \mu \) such that there exists a set of \( \mu \) encoding vectors \( g_1, g_2, \ldots, g_\mu \) and a set of \( m \) decoding functions \( r_1, \ldots, r_m \) that allow each client to decode the required packet. The goal of the CIC problem is to maximize the value of \( n - \mu \), i.e., the number of transmissions that were “saved” by using the encoding scheme. We denote by \( OPT_{IC} \) and \( OPT_{CIC} \) the optima of the Index Coding and Complementary Index Coding problems, respectively.

In a vector-linear solution, each packet \( p_i \) is subdivided into \( k \) smaller size subpackets \( p^1_i, \ldots, p^k_i \). Then, each each transmitted packet is a linear combination of the subpackets \( \{p^j_i\} \leq n, 1 \leq j \leq k \), rather than the original packets. With vector-linear network coding our goal is to find encoding and decoding schemes that minimize the ratio of \( \frac{\mu}{\tau} \), where \( \mu \) is the number of times a combination of subpackets is transmitted. For the CIC problem, the goal is to maximize the value of \( n - \frac{\mu}{\tau} \).

For example, consider the setting depicted in Figure 1. As mentioned in the Introduction, the scalar-linear solution requires three transmissions (i.e., saves 2 transmissions). However, there exists a vector-linear solution that achieves \( \frac{\mu}{\tau} = 2.5 \) by dividing each packet into two parts (i.e., \( k = 2 \)) and sending five combinations of subpackets, each subpacket is half the size of the original packet. Specifically, each packet \( p_i \) is divided into two sub-packets \( p^1_i \) and \( p^2_i \) and then the following five linear combinations of the resulting packets are transmitted: \( p^1_1 + p^2_2, p^2_1 + p^1_3, p^3_2 + p^4_1, p^4_2 + p^3_1, \) and \( p^5_2 + p^5_3 \).

III. Preliminaries

We start by introducing a notion of the Dependency Graph. The dependency graph is defined for the multiple unicast case, in which each packet is requested by a single client.
Definition 3.1 (Dependency Graph): Given a multi-
unicast instance of the CIC problem we define a graph $G(V, E)$ as follows:
- For each client $c_i \in X$ there is a corresponding vertex $v_{c_i}$ in $V$.
- There is a directed edge from $v_{c_i}$ to $v_{c_j}$ if and only if it holds that $w_i \in H(c_j)$.

Figure 2(a) depicts the dependency graph that corresponds to the instance of the CIC problem depicted in Figure 1.

We proceed to define the notion of the Maximum Induced Acyclic Subgraph (MAIS) [2].

Definition 3.2 (Maximum Induced Acyclic Subgraph (MAIS)): For a given graph $G(V, E)$, we define Maximum Induced Acyclic Subgraph (MAIS(G)) as the maximum acyclic induced subgraph of $G$. We denote by $|\text{MAIS}(G)|$ the number of vertices in MAIS(G).

It was shown in [2] that in the multiple unicast case the optimal solution to the Index Coding problem is larger or equal to MAIS(G).

Other related problems are the minimum Feedback Vertex Set (FVS) and the minimum Feedback Edge Set (FES) of a graph.

Definition 3.3 (Problems FVS and FES): For a given graph $G$, find the least number of vertices (edges) whose removal makes the graph acyclic. We denote the optimal integral solution to the FVS (FES) problem for a graph $G(V, E)$ by $|FVS(G)|$ ($|FES(G)|$).

Note that, for a given graph $G$, by definition of the Feedback Vertex Set, it holds that $|\text{MAIS}(G)| = n - |FVS(G)|$.

Both the FVS and FES problems can be formulated as integer programs. By relaxing the integrality constraints on the decision variables, we can get fractional version of these problems. The fractional feedback vertex set ($FVS^f(G)$) for the graph $G(V, E)$ is a function $t : V \rightarrow [0, 1]$ such that every cycle, $c$, is covered by $t$, i.e., $\sum_{e \in c} t(v) \geq 1$. The fractional feedback edge set ($FES^f(G)$) for the graph $G(V, E)$ is a function $t : E \rightarrow [0, 1]$ such that every cycle, $c$, is covered by $t$, i.e., $\sum_{e \in c} t(e) \geq 1$. An optimum feedback vertex (edge) set has minimizes $\sum_{v \in V} t(v)$. The values of optimal fractional solutions to these problems are denoted by $|FVS^f(G)|$ and $|FES^f(G)|$, respectively.

Proposition 3.4: $|OPT_{CIC}| \leq |FVS(G)|$

Proof: In [2] it was shown that $|OPT_{CIC}| \geq |\text{MAIS}(G)|$. Since $|\text{MAIS}(G)| = n - |FVS(G)|$, we get $n - |OPT_{CIC}| \leq |FVS(G)|$. Furthermore, by the definition of the CIC problem $|OPT_{CIC}| = n - |OPT_{CIC}|$. Hence, $|OPT_{CIC}| \leq |FVS(G)|$.

In our work we also use a notion of the Vertex Split Graph, defined as follows:

Definition 3.5 (Vertex Split Graph): Given a graph $G(V, E)$ we construct a corresponding Vertex Split Graph $G'(V', E')$ as follows: (1) For each node $v \in V$: (a) create two nodes $v_{in}, v_{out}$ in $V'$ (b) create an edge $(v_{in}, v_{out})$ in $E'$; (2) For each edge $(u, v) \in E$ create an edge $(v_{out}, v_{in})$ in $E'$.

Note that the size of the Feedback Vertex Set of $G$ is equal to the size of Feedback Edge Set of its vertex split graph $G'$, i.e., $|FVS(G)| = |FES(G')|$. We proceed to describe Edge-disjoint Cycle Packing (ECP) and Vertex-Disjoint Cycle Packing (VCP) problems. Problems ECP and VCP ask for the largest set of directed edge (node) disjoint cycles in a given graph $G(V, E)$. We denote the maximum number of edge (vertex) disjoint cycles in a graph $G$ by $|ECP(G)|$ ($|VCP(G)|$). We can also define fractional versions of these problems as follows. Let $C$ be a set that includes all cycles in the graph and let $\psi, C \rightarrow R$ be a function that maps each cycle $c \in C$ to a real number. Our goal is to find a function $\psi$ that maximizes $\sum_{c \in C} \psi(c)$ subject to the following constraints:

- For each $v \in V$, it holds that $\sum_{e \in c e \in C} \psi(c) \leq 1$ (for node-disjoint case);
- For each $e \in E$, it holds that $\sum_{e \in c e \in C} \psi(e) \leq 1$ (for edge-disjoint case).

We denote by $|ECP^f(G)|$ ($|VCP^f(G)|$) the optimal value of the fractional edge-disjoint (node-disjoint) cycle packing problem.

IV. MULTIPLE UNICAST CASE

In this section we first present an approximation algorithm for the scalar version of the Complementary Index Coding (CIC) problem. Then, we present an algorithm for the vector version of this problem.

A. An Approximation Algorithm for Finding Scalar-Linear Solution

The main idea of our algorithm is to find a vertex disjoint cycle packing in the dependency graph. Note that for each vertex-disjoint cycle in the dependency graph we can save at least one transmission. To see this, consider the example depicted in Figure 2(b). In this example, we have a cycle that involves five clients, such that client $c_i$ requires packet $p_i$. For $i = 2, \ldots, 5$ it holds that the client $c_i$ has the packet required by client $c_{i-1}$. It is easy to verify that all clients can be satisfied by four transmissions: $p_1 + p_2, p_2 + p_3, p_3 + p_4,$ and $p_4 + p_5$ (all operations are over GF(2)). Indeed, the client $c_5$ will be satisfied by the transmission $p_1 + p_2$, the client $c_3$ will be satisfied by transmission $p_2 + p_3$, and so on. The client $c_1$ will add all the transmissions to obtain $p_1 + p_5$, which will allow it to decode packet $p_2$.

Thus, a vertex-disjoint cycle packing of size $k$ will allow to find a solution that saves $k$ transmissions. We summarize our result by the following lemma.

Lemma 4.1: Let $\alpha$ be a set of vertex disjoint cycles in the dependency graph $G(V, E)$. Then, it is possible to construct a feasible solution to the CIC problem that saves at least $|\alpha|$ transmissions, where $|\alpha|$ is the size of $\alpha$.

For convenience, we convert the problem of finding vertex-disjoint cycle packing in the dependency graph $G(V, E)$ to the problem of finding edge-disjoint cycle packing in the vertex split graph $G'(V', E')$ of $G(V, E)$. As mentioned above, the
size of the edge-disjoint cycle packing in graph $G'(V', E')$ is equal to the size of vertex-disjoint cycle packing in $G(V, E)$.

Our algorithm, referred to as Algorithm $sCIC$, performs the following steps. First, the algorithm constructs the dependency graph $G(V, E)$ for the problem at hand. Next, the vertex split graph $G'(V', E')$ of $G(V, E)$ is constructed. Finally, we apply the approximation algorithm due to Krivelevich et al. [15] to find an approximate cycle packing in $G'(V', E')$. Next, we identify the set of vertex-disjoint cycles in $G(V, E)$ that correspond to edge-disjoint cycles in $G(V, E)$. Finally, for each cycle in the dependency graph we identify the set of encoding vectors such that one transmission is saved per cycle.

We proceed to analyze the correctness of Algorithm $sCIC$.

**Lemma 4.2:** Algorithm $sCIC$ finds a scalar-linear solution to the Complementary Index Coding problem with approximation ratio of $\Omega(\sqrt{n} \cdot \log n \cdot \log \log n)$.

**Proof:** By Proposition 3.4, $\text{OPT}_{CIC} \leq |\text{FVS}(G)|$. As discussed above, $|\text{FVS}(G)| = |\text{FES}(G')|$, where $G'$ is the vertex split graph of $G$, hence $\text{OPT}_{CIC} \leq |\text{FES}(G')|$. By [16], the integrality gap of the Feedback Edge Set problem is bounded by $\log n \cdot \log \log n$, hence $|\text{FES}(G')| \leq |\text{FES}(G')| \cdot \log n \cdot \log \log n$. Hence, $\text{OPT}_{CIC} \leq |\text{FES}(G')| \cdot \log n \cdot \log \log n$. The algorithm due to [15] yields an edge disjoint cycle cover $G'$, whose size is at least $|\text{FES}(G')| / \sqrt{n}$. Since for each cycle we save a transmission, the total number of saved transmissions is at least

$$\frac{|\text{FES}(G')|}{\sqrt{n}} \geq \frac{|\text{FES}(G')|}{\sqrt{n} \cdot \log n \cdot \log \log n} \geq \frac{\text{OPT}_{CIC}}{\sqrt{n} \cdot \log n \cdot \log \log n}.$$

\[\Box\]

**B. An Approximation Algorithm for Finding Vector-Linear Solution**

In this section we present an algorithm, referred to as Algorithm $vCIC$, for finding vector-linear solutions for the $CIC$ problem. The algorithm achieves the approximation ratio of $\Omega(\log n \cdot \log \log n)$. As mentioned in the model, with a vector-linear solution, each packet can be divided into several smaller-size subpackets and the server can transmit linear combinations of these subpackets.

The algorithm includes the following steps. Given an instance of the $CIC$ problem, we first construct the dependency graph $G(V, E)$, and the corresponding Vertex Split Graph $G'(V', E')$. Then, we apply the algorithm due to Yuster and Nutov [14] to find the fractional cycle packing in $G'(V', E')$, which, in turn, yields a fractional vertex-disjoint cycle packing $\psi : C \rightarrow R$ in $G(V, E)$. Then, we find the minimum integer number $k$ that satisfies that $k\psi(c)$ is an integer for any $c \in C$. Such number exists because for each $c \in C$, $\psi(c)$ is a rational number. Next, we divide each packet into $k$ smaller subpackets $p_1^1, \ldots, p_k^k$.

Next, we create an auxiliary dependency graph $\hat{G}(\hat{V}, \hat{E})$. This graph is constructed similarly to the regular dependency graph, but for subpackets, instead of the original packets. Specifically, graph $G'(\hat{V}, \hat{E})$ is defined as follows:

- For each subpacket $p_i^k$ of a packet $p_i \in P$ there is a corresponding vertex $v_i^k$ in $V$.
- There is a directed edge from $v_i^k$ and $v_j^k$ if and only if it holds that $p_i \in H(e)$, $e$ is a client requesting packet $p_i$.

Graph $\hat{G}(\hat{V}, \hat{E})$ has the following property. For each fractional cycle packing $\psi$ of graph $G(V, E)$ of size $\alpha$, there exists a set of vertex-disjoint cycles $\hat{C}$ in $\hat{G}(\hat{V}, \hat{E})$ of size $\alpha k$. The integer cycle packing $\hat{C}$ in $G(V, E)$ can be identified through the following procedure. For each cycle $c \in C$ for which $\psi(c) > 0$ we can identify $k \cdot \psi(c)$ vertex-disjoint cycles in $G$ such that for node $v_i \in C$, each of the corresponding cycles use one of the nodes in $\{v_i^1, \ldots, v_i^k\}$. We then remove $k \cdot \psi(c)$ vertex-disjoint cycles from $G$ and repeat the procedure for the next cycle in $C$.

Now, for each cycle $\hat{c} \in \hat{C}$ we generate $|\hat{c}| - 1$ linear combinations that satisfy the demands for packets that correspond to vertices in $\hat{c}$. Each such cycle will save one subpacket, so in total $\alpha k$ subpackets will be saved. This corresponds to saving $\alpha$ original packets.

We proceed to analyze the correctness of the Algorithm $vCIC$.

**Lemma 4.3:** Algorithm $vCIC$ finds a vector-linear solution to the Complementary Index Coding problem with approximation ratio of $\Omega(\log n \cdot \log \log n)$.

**Proof:** Let $\text{OPT}_{vCIC}^f$ be the value of the optimal fractional solution to the $CIC$ problem. It is easy to verify that, similar to the integer case, $\text{OPT}_{vCIC}^f \leq |\text{FVS}(G)|$. This follow from the fact that the optimal fractional solution to IC is greater or equal than the size of $\text{MAIS}(G)$.

Since $|\text{FVS}(G)| = |\text{FES}(G')|$, it holds that $\text{OPT}_{vCIC}^f \leq |\text{FES}(G')|$. By [16], the integrality gap of the Feedback Edge Set problem is bounded by $\log n \cdot \log \log n$, hence $|\text{FES}(G')| \leq |\text{FES}(G')| \cdot \log n \cdot \log \log n$. Hence, $\text{OPT}_{vCIC}^f \leq |\text{FES}(G')| \cdot \log n \cdot \log \log n$. As discussed above, our solution saves $|\text{FES}(G')| k$ subpackets, which is equivalent to saving $|\text{FES}(G')| k$ packets for the CIC problem. Thus, the total number of saved transmissions is at least $\frac{\text{OPT}_{vCIC}^f}{\log n \cdot \log \log n}$ and the lemma follows.

\[\Box\]

V. MULTIPLE MULTICAST CASE

In this section we allow multiple clients to require the same packet i.e., $m \geq n$, and show that in this case Complementary Index Coding (CIS) problem is not only NP-hard but is also hard to approximate.

We prove the results by reducing the Independent Set (IS) problem (where the objective is to find the Independent Set

\[1\] This can be done assuming that each packet is sufficiently large and the size of each packet in $P$ is a multiple of $k$. Thus, in practice, a fractional solution can be suitable for large packets. This assumption is standard for vector-linear network coding schemes.
of maximum cardinality) into Problem CIC. We denote the value of the optimal solution of the IS problem by $OPT_{IS}$. The instance of the IS problem is given by a graph $G(V, E)$.

Given an instance of the IS problem we define an instance of the CIC problem as following:

- For each vertex $v \in V$, we define a packet $p_v$, and for each edge $e = (u, v) \in E$, we define packet $p_e$, i.e., we define a total of $|V| + |E|$ packets.
- For each edge $e = (u, v) \in E$ we define the following three clients $c_{e1}, \ldots, c_{e3}$ such that:
  1. Client $c_{e1}$ requires packet $p_u$ and has packet $p_v$;
  2. Client $c_{e2}$ requires packet $p_v$ and has packet $p_e$;
  3. Client $c_{e3}$ requires packet $p_e$ and has packets $p_u$ and $p_v$.

i.e., we define a total of $3 \cdot E$ clients.

This construction is similar to that used in our previous work [17].

**Lemma 5.1:** $OPT_{IS} = OPT_{CIC}$.

**Proof:** We proceed to prove that $OPT_{IS} = OPT_{CIC}$. Let $OPT_{VC}$ be value of the minimum Vertex Cover in $G(V, E)$. In [17] it was shown that

$$OPT_{VC} + |E| = OPT_{CIC}$$

(1)

Since a set of vertices in $G$ that do not belong to a vertex cover constitute an independent set, it holds that

$$OPT_{VC} + OPT_{IS} = |V|$$

(2)

Also, since the total number of packet in our instance of Problem CIC is equal to $|V| + |E|$, it holds that

$$OPT_{CIC} = |V| + |E| - OPT_{IC}$$

(3)

By combining Equations 1-3 we get $OPT_{IS} = OPT_{CIC}$.

By Lemma 5.1, the optimal value of Problem CIC is equal to the maximum size of the independent set in $G(V, E)$, Thus, our reduction implies that the Complementary Index Coding is NP-hard. Furthermore, since it is NP-hard to approximate the independent set problem within a ratio of $n^{1-\epsilon}$ [18] the same holds for Problem CIC.

We conclude our discussion by the following theorem.

**Theorem 5.2:** The Complementary Index Coding is NP-hard, and it is NP-hard to approximate within a ratio of $n^{1-\epsilon}$ for any constant $\epsilon > 0$.

VI. CONCLUSION

In this paper, we focused on the complementary problem of the Index Coding problem (referred to as the Complementary Index Coding (CIC) problem). The goal of the CIC problem is to maximize the number of saved transmissions, i.e., the number of transmissions that are saved by using the Index Coding technique compared to the traditional solution that does not involve coding.

We considered two scenarios for the CIC problem, i.e., the multiple unicast scenario and the multiple multicast scenario. In the multiple unicast scenario, each packet is requested by a single client; whereas in the multiple multicast scenario, each packet can be requested by more than one client. We also considered scalar and vector linear solutions for the problem at hand. In the scalar linear solution, the packets cannot be split. In a vector linear solution, each packet can be split into a smaller sub-packets, such that sub-packet originated from different packets can be combined together. For the multiple unicast scenario, we presented approximation algorithms for finding scalar and vector linear solutions. The approximation ratios of the scalar and vector linear solutions are $\Omega(\sqrt{n} \cdot \log n \cdot \log \log n)$ and $\Omega(\log n \cdot \log \log n)$, respectively. For the multiple multicast scenario, we showed that finding an approximate solution within ratio $n^{1-\epsilon}$ is NP-hard.

**REFERENCES**


