A Generalisation of Interlinked Cycle Structures and Their Index Coding Capacity

Mahesh Babu Vaddi and B. Sundar Rajan
Department of Electrical Communication Engineering, Indian Institute of Science, Bengaluru 560012, KA, India
E-mail: {vaddi, bsrajan}@iisc.ac.in

***Abstract*—Cycles and Cliques in a side-information graph reduce the number of transmissions required in an index coding problem. Thapa, Ong and Johnson defined a more general form of overlapping cycles, called the interlinked-cycle (IC) structure, that generalizes cycles and cliques. They proposed a scheme, that leverages IC structures in digraphs to construct scalar linear index codes. In this paper, we extend the notion of interlinked cycle structure to define more generalised graph structures called overlapping interlinked cycle (OIC) structures. We prove the index coding capacity of OIC structures by giving an index code with length equal to the order of maximum acyclic induced subgraph (MAIS) of OIC structures.**

I. Introduction and Background

A single unicast index coding problem, comprises a transmitter that has a set of K messages, $X = \{x_1, x_2, \ldots, x_K\}$, and a set of K receivers, $R = \{R_1, R_2, \ldots, R_K\}$. Each receiver, $R_k = (\mathcal{W}_k, \mathcal{K}_k)$, knows a subset of messages, $\mathcal{K}_k \subseteq X$, called its side-information, and wants to know one message, $W_k = \{x_k\}$, called its *Want-set*. The transmitter can take cognizance of the side-information of the receivers and broadcast coded messages, called the index code. The objective is to minimize the number of coded transmissions, called the length of the index code, such that each receiver can decode its demanded message using its side-information and the coded messages.

The index coding with side-information was introduced by Birk and Kol in [1]. Single unicast index coding problems were studied in [2]. A single unicast index coding problem (SUICP) can be represented by using a graph G with K vertices $\{x_1, x_2, \ldots, x_K\}$. In G, there exists an edge from x_i to x_j if the receiver wanting x_i knows x_j. This graph is called the side-information graph of SUICP.

The broadcast rate [6] of an index coding problem is the minimum (minimization over all mapping including nonlinear and all dimensions) number of index code symbols required to transmit such that every receiver can decode its wanted message by using the broadcasted index code symbols and its side-information. The capacity of an index coding problem is the reciprocal of the broadcast rate.

In this paper, we refer the capacity of a single unicast index coding problem with side-information graph G as the index coding capacity of side-information graph G. In [5], Maleki *et.al.* found the index coding capacity of some side-information graphs which have a circular symmetry by using interference alignment technique. However, in general, finding the index coding capacity is a complicated problem because one need to consider all possible linear and non linear mappings and dimensions to evaluate capacity.

For a graph G, the order of an induced acyclic sub-graph formed by removing the minimum number of vertices in G, is called Maximum Acyclic Induced Subgraph ($MAIS(G)$). In [2], it was shown that $MAIS(G)$ lower bounds the broadcast rate of the index coding problem described by G. That is,

$$\beta(G) \geq MAIS(G). \tag{1}$$

A. Interlinked Cycles and Optimal Index Codes

In [3], Thapa, Ong and Johnson defined a special graph structures called interlinked cycle (IC) structure. The interlinked cycle structures generalises the notion of cycles and cliques. Consider a graph G with K vertices $\{x_1, x_2, \ldots, x_K\}$ having the following property: G has a vertex set V_I such that for any ordered pair ($x_i \in V_I, x_j \in V_I$) and $x_i \neq x_j$, there is a path from x_i to x_j, and the path does not include any other vertex in V_I except x_i and x_j. The set V_I is called inner vertex set and the vertices in V_I are called inner vertices. A path in which only the first and the last vertices are from V_I, and they are distinct, is called an I-path. If the first and last vertices are the same, then it is called an I-cycle. If the directed graph G satisfies the four conditions given below, it is called an interlinked-cycle structure.

- There is no I-cycle in G.
- Every non-inner vertex must be present in at least one I-path.
- For all ordered pairs of inner vertices (x_i, x_j), $x_i \neq x_j$, there is only one I-path from x_i to x_j in G.
- There exist no cycles among non-inner vertices

Let G be the IC structure with K vertices $\{x_1, x_2, \ldots, x_K\}$ and N inner vertices $V_I = \{x_1, x_2, \ldots, x_N\}$. Let the $K - N$ non-inner vertices be $V_{NI} = \{x_{N+1}, x_{N+2}, \ldots, x_K\}$. The following coded symbols for an IC structure G with $|V(G)| = K$ was proposed in [3].

- A code symbol is obtained by the bitwise XOR (denoted by $\oplus$) of messages present in the inner vertex set V_I, i.e.,

$$y_I = \bigoplus_{i=1}^{N} x_i. \tag{2}$$

- For each $x_j \in V_{NI}$, for $j \in [N+1 : K]$, a code symbol

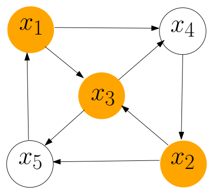

Fig. 1: Interlinked cycle structure with $V_I = \{x_1, x_2, x_3\}$.

is obtained as given below.

$$y_j = x_j \bigoplus_{x_q \in N_G^+(x_j)} x_q. \tag{3}$$

where $N_G^+(x_j)$ is the out-neighborhood of x_j in the IC structure G.

The length of index code constructed above is $K - N + 1$. Thapa, Ong and Johnson proved that the constructed codes in (2) and (3) are of optimal length.

The following decoding procedure is given in [3] to decode the index codes given by ICC scheme.

- For $j \in [N+1 : K]$, the message x_j (x_j corresponding to a non-inner vertex) can be decoded from y_j given in (3).
- For each $x_k \in V_I$, a directed rooted tree (denoted by T_k) in G can be found with x_k as the root vertex and all other inner vertices $V_I \setminus \{x_k\}$ are the leaves. The inner vertex $x_k \in V_I$ is decoded by computing the XOR of all index code symbols corresponding to the non-leaf vertices at depth greater than zero in T_k and y_I, where T_k is the rooted tree with x_k as the root node and all other inner vertices as the leaves.

Example 1. Consider an SUICP with side-information graph G given in Fig. 1. G is an IC structure with $K = 5, N = 3$ and inner vertices $V_I = \{x_1, x_2, x_3\}$. Hence, for this side-information graph, we have $K - N + 1 = 3$. An optimal length index code for this ICP obtained from (2) and (3) is

$$\mathfrak{C} = \{\underbrace{\underbrace{x_1 + x_2 + x_3}_{\in V_I}}_{y_I}, \underbrace{\underbrace{x_4}_{\in V_{NI}} + \underbrace{x_2}_{N_G^+(x_4)}}_{y_4}, \underbrace{\underbrace{x_5}_{\in V_{NI}} + \underbrace{x_1}_{N_G^+(x_5)}}_{y_5}\}.$$

In [8], we provided an addition to interlinked cycle structure class by providing optimal length index codes for IC structures with one cycle among non-inner vertex set. We gave a modified code construction and modified decoding method for the IC structure with one cycle among non-inner vertex set. In [9], we disproved the two conjectures given in [3] regarding the optimality of IC structures.

The below mentioned two examples are useful to understand the motivation behind studying overlapping interlinked cycle structures.

Motivating Example I

Consider the side-information graph G given in Figure 2. The broadcast rate of the index coding problem described by this side-information graph is three. To get an index code of length three, the side-information graph must have an inner vertex set with four inner vertices in it. But, no subset of size four of $\{x_1, x_2, \ldots, x_6\}$ satisfies the necessary conditions required for V_I. Hence, the side-information graph G is not an interlinked cycle structure.

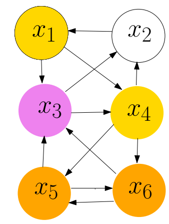

Fig. 2: Overlapping IC structure with capacity $\frac{1}{3}$.

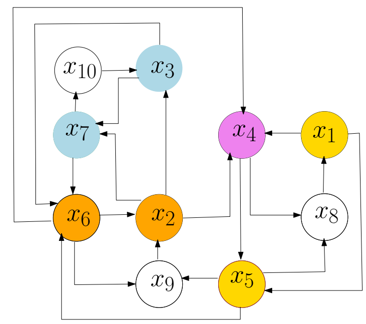

Fig. 3: Overlapping IC structure with capacity $\frac{1}{6}$.

In this paper we show that G given in Figure 2 is an overlapping interlinked cycle structure and give an index code with length MAIS(G).

Motivating Example II

Consider the side-information graph G given in Figure 3. The broadcast rate of the index coding problem described by this side-information graph is six. To get an index code of length six, the side-information graph must have an inner vertex set with five inner vertices in it. But, no subset of size five of $\{x_1, x_2, \ldots, x_{10}\}$ satisfies the necessary conditions required for V_I. Hence, the side-information graph G is not an interlinked cycle structure.

In this paper we show that G given in Figure 3 is an overlapping interlinked cycle structure give an index code with length MAIS(G).

B. Contributions

The contributions of this paper are summarized as below:

- We extend the notion of interlinked cycle structure to define more generalised graph structures called overlapping interlinked cycle (OIC) structures. We give an index code for OIC structures whose length is equal to MAIS of OIC structure.

The proofs of all lemmas in this paper have been omitted due to space constraints. All the proofs and more examples can be found in [10].

II. Overlapping Cycle Structures

In this section, we generalise the notion of interlinked cycle structure to define overlapping interlinked cycle structure.

A tree is an undirected graph in which any two vertices are connected by exactly one path. That is, a tree is an acyclic connected graph. A polytree [7] is a directed acyclic graph whose underlying undirected graph is a tree. In a polytree, there exits only one directed path from any vertex to any other vertex.

A graph G is called overlapping interlinked cycle structure if there exists a collection of subsets of vertices of $V(G)$ double indexed as $V_I^{(i,j)}$ for $i \in [0:d]$ and $j \in [1:w_i]$ such that the following conditions are satisfied.

Condition 1. The vertex subsets $V_I^{(i,j)}$ for $i \in [0:d]$ and $j \in [1:w_i]$ form a polytree with edges existing only between a parent and child when there is a single common vertex between the parent and child. Also, the number of vertices in $V_I^{(i,j)}$ must be greater than the sum of number of parents and children of $V_I^{(i,j)}$ in the polytree. The polytree is shown in Figure 4.

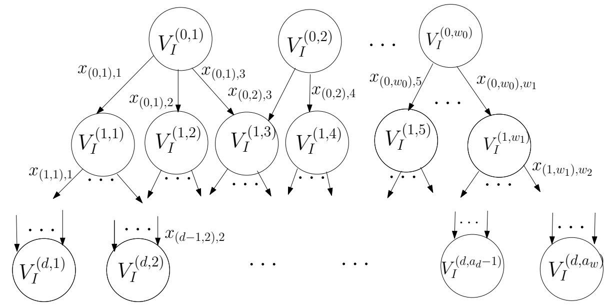

Fig. 4: Polytree structure of vertex subsets.

To define the second condition, we need to define some sets related to the vertex sets present in polytree. Let $V_I^{\text{Total}} = \cup_{i=0}^{d} \cup_{j=1}^{w_i} V_I^{(i,j)}$ and $V_{NI} = V(G) \setminus V_I^{\text{Total}}$. We refer vertices in V_I^{Total} as inner vertices and the vertices in V_{NI} as non-inner vertices. If the vertex set $V_I^{(i,j)}$ has p number of parents, then, $V_I^{(i,j)}$ is having one common vertex with each of these p parents. Let the set $\tilde{V}_I^{(i,j)}$ be the set after removing all the p vertices from $V_I^{(i,j)}$ which are common with their p parents. We have

$$V_I^{\text{Total}} = \cup_{i=0}^{d} \cup_{j=1}^{w_i} V_I^{(i,j)} = \cup_{i=0}^{d} \cup_{j=1}^{w_i} \tilde{V}_I^{(i,j)}.$$

If the vertex set $V_I^{(i,j)}$ has c number of children, then, $V_I^{(i,j)}$ is having one common vertex with each of these c children and $\tilde{V}_I^{(i,j)}$ consists of c common vertices with the c children of $V_I^{(i,j)}$ and $|\tilde{V}_I^{(i,j)}| - c$ number of vertices which are not common to any other vertex set in the polytree. Let the c children of $V_I^{(i,j)}$ in polytree be $V_I^{(i+1,j_1)}, V_I^{(i+1,j_2)}, \ldots, V_I^{(i+1,j_c)}$ and the corresponding common vertices be $x_{(i,j),j_1}, x_{(i,j),j_2}, \ldots, x_{(i,j),j_c}$ respectively.

Let $S^{(i,j),j_k}$ for $k \in [1:c]$ be the collection of all nodes $V_I^{(i',j')}$ in the polytree to which there exists a path from $V_I^{(i,j)}$ to $V_I^{(i',j')}$ through $V_I^{(i+1,j_k)}$. For every $V_I^{(i',j')} \in S^{(i,j),j_k}$, $i' \in [i+1:d]$, there exists $i'-i+1$ nodes of polytree present in this path including the first node $V_I^{(i,j)}$ and the last node $V_I^{(i',j')}$ and these $i'-i+1$ nodes are connected by $i'-i$ edges. Note that any two nodes in the polytree which are connected by an edge have a common vertex. Let this path be as given below.

$$V_I^{(i,j)} \xrightarrow[x_{(i,j),j_k}]{} V_I^{(i+1,j_k)} \xrightarrow[x_{(i+1,j_k),k_2}]{} V_I^{(i+2,k_2)} \xrightarrow[x_{(i+2,k_2),k_3}]{} \ldots \xrightarrow[x_{(i'-2,k_{i'-i-2}),k_{i'-i-1}}]{} V_I^{(i'-1,k_{i'-i-1})} \xrightarrow[x_{(i'-1,k_{i'-i-1}),k'}]{k'=j_{i'-i}} V_I^{(i',j')}.$$

Let

$$V_P^{(i,j),(i',j')} = V_I^{(i+1,j_k)} \bigcup \left(\bigcup_{s=2}^{i'-i} V_I^{(i+s,k_s)} \right) \setminus \{x_{(i,j),j_k}, x_{(i+1,j_k),k_2}, x_{(i+2,k_2),k_3}, \ldots, x_{(i'-1,j_{i'-i-1}),j'}\}.$$

That is, $V_P^{(i,j),(i',j')}$ is the union of the $i'-i$ vertex sets present in the path from $V_I^{(i,j)}$ to $V_I^{(i',j')}$ excluding the vertex set $V_I^{(i,j)}$ and after removing $i'-i$ common vertices.

Condition 2. In G, for every $\tilde{V}_I^{(i,j)}$, from every vertex in $\tilde{V}_I^{(i,j)}$, there should be only one path to every non-common vertex in that $\tilde{V}_I^{(i,j)}$ such that the path does not include any other inner vertices other than the first and last vertex. From every vertex in $\tilde{V}_I^{(i,j)}$, either there can be only one path to the common vertex $x_{(i,j),j_k}$ for $k \in [1:c]$ or there can be only one path to every vertex in $V_P^{(i,j),(i',j')}$ for any $V_I^{i',j'}$ present in $S^{(i,j),j_k}$ such that the path does not include any other inner vertices other than the first and last vertex. All the paths mentioned in this condition are referred as I-paths in this paper.

Condition 3. For every inner vertex $x_{(i,j),k_1} \in V_I^{(i,j)}$, there exists no cycle in G that includes $x_{(i,j),k_1}$ and vertices only from the set $V(G) \setminus V_I^{(i,j)}$. The graph G should not have any cycle with only non-inner vertices in it.

Condition 4. Every non-inner vertex must be present in at least one I-path. All the outgoing paths from a non-inner vertex terminate at the vertices of only one vertex set $V_I^{(i,j)}$ for $i \in [0:d]$ and $j \in [1:w_i]$.

In this paper, we refer the vertex subsets $V_I^{(i,j)}$ for $i \in [0:d]$ and $j \in [1:w_i]$ as semi-inner vertex sets. The following two examples illustrate Condition 2.

Example 2. Consider the side-information graph given in Figure 3. In the graph G, we have $V_I^{(0,1)} = \{x_1, x_4, x_5\}, V_I^{(1,1)} = \{x_2, x_4, x_6\}$ and $V_I^{(2,1)} = \{x_3, x_6, x_7\}$. The polytree structure of the three semi-inner vertex sets are shown on Figure 5. In G, the vertex x_5, instead of having an I-path to x_4, it has I-paths to $\{x_2, x_4, x_6\} \setminus \{x_4\} = \{x_2, x_6\}$. Similarly, the vertex x_2, instead of having an I-path to x_6, it has I-paths to $\{x_3, x_6, x_7\} \setminus \{x_6\} = \{x_3, x_7\}$. The details of I-paths in G are summarised in Table I.

x_k	I-path	Depth at which I-path originates	Depth at which I-path terminates
x_1	x_4, x_5	0	0
x_4	x_1, x_5	0	0
x_5	$x_1, \{x_2, x_6\}$	0	1
x_2	x_4, x_6	1	1
x_6	$x_2, \{x_3, x_7\}$	1	2
x_7	x_4, x_3	2	2
x_3	x_4, x_7	2	2

TABLE I: I-paths present in Figure 3

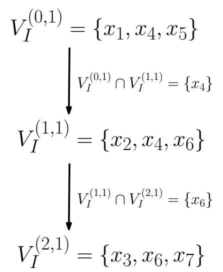

Fig. 5: Polytree of semi-inner vertex sets of OIC structure given in Figure 3 and Figure 6.

Example 3. Consider the side-information given in Figure 6. In the graph G, we have $V_I^{(0,1)} = \{x_1, x_4, x_5\}, V_I^{(1,1)} = \{x_2, x_4, x_6\}$ and $V_I^{(2,1)} = \{x_3, x_6, x_7\}$. The polytree structure of these side-information graph is shown in Figure 5. Note that the only difference between Figure 3 and Figure 6 is that the vertex x_5 have I-paths to $\{x_3, x_7\}$ instead of having an I-path to x_6. Hence, the I-path of x_5 is terminated at depth two instead of depth one. The details of I-paths are mentioned in Table I.

x_k	I-path	Depth at which I-path originates	Depth at which I-path terminates
x_1	x_4, x_5	0	0
x_4	x_1, x_5	0	0
x_5	$x_1, x_2, \{x_3, x_7\}$	0	2
x_6	x_2, x_4	1	1
x_2	$x_6, \{x_3, x_7\}$	1	2
x_7	x_6, x_3	2	2
x_3	x_6, x_7	2	2

TABLE II: I-paths present in Figure 6

The graphs given in Figure 2, Figure 3 and Figure 6 satisfy all the four conditions given above. Hence, they are

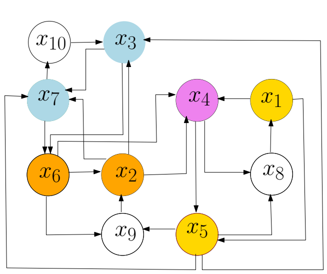

Fig. 6: Overlapping IC structure with capacity $\frac{1}{5}$.

overlapping interlinked cycle structures.

A. Index code construction for Overlapping Interlinked Cycle Structure

Consider an index coding problem whose side-information graph is a Overlapping Interlinked Cycle (OIC) structure. Let $s = \sum_{i=0}^{d} w_i$. That is, s is the number of nodes in polytree (note that each node in polytree represents a semi-inner vertex set). In the following two steps, we give an index code of length $|V_{NI}| + s$. The index code comprises one code symbol for every semi-inner vertex set (total s index code symbols for s semi-inner vertex sets) and one index code symbol for every non-inner vertex ($|V_{NI}|$ index code symbols for $|V_{NI}|$ non-inner vertices).

- A code symbol is obtained by the bitwise XOR of messages present in the semi-inner vertex set $V_I^{(i,j)}$ for every $i \in [0:d]$ and every $j \in [1:w_i]$, i.e.,
$$y_I^{(i,j)} = \bigoplus_{k=1}^{|V_I^{(i,j)}|} x_{(i,j),k}. \quad (4)$$
- For each $x_k \in V_{NI}$, a code symbol is obtained as given below.
$$y_k = x_k \bigoplus_{x_q \in N_G^+(x_k)} x_q. \quad (5)$$
where $N_G^+(x_k)$ is the out-neighborhood of x_k in the IC structure G.

B. Decoding procedure

In this subsection, we give the decoding procedure for the index code constructed from (4) and (5) for OIC structures.

To establish decoding procedure for OIC structures, we define the tree $T_k^{(i,j)}$ for every vertex $x_{(i,j),k} \in \tilde{V}_I^{(i,j)}$ for every $i \in [0:d]$ and $j \in [1:w_i]$. Let the I-paths originating from $x_{(i,j),k}$ pass through b_k semi-inner vertex sets at depth $i+k$ for $k \in [0:t-i]$. Note that $b_0 = 1$ follows from the fact that I-paths originating from $x_{(i,j),k}$ pass through only one semi-inner vertex set $V_I^{(i,j)}$ at depth i. Let $t = \sum_{k=0}^{t-i} b_k$. From the definition of polytree, these t semi-inner vertex sets are represented by t nodes and these t nodes are connected by $t-1$ directed edges in the polytree. Note that every edge in the semi-inner vertex set polytree represents one vertex in G which is common to both the parent and child connected by this edge. Let $V_k^{(i,j)}$ be the union of t semi-inner vertex sets after deleting the $t-1$ common vertices belonging to $t-1$ edges connecting the t semi-inner vertex sets. That is, the cardinality of $V_k^{(i,j)}$ is t less than that of the cardinality of union of t semi-inner vertex sets. For $x_{(i,j),k} \in V_I^{(i,j)}$, because of the presence of I-paths from $x_{(i,j),k}$ to all other vertices in $V_k^{(i,j)}$, a directed rooted tree (denoted by $T_k^{(i,j)}$) in G can be found with $x_{(i,j),k)}$ as the root vertex and all other inner vertices $V_k^{(i,j)} \setminus \{x_{(i,j),k}\}$ as the leaves.

From (4), all the message symbols in a semi-inner vertex set $V_I^{(i,j)}$ are encoded into one index code symbol $y_I^{(i,j)}$. Let

$w_k^{(i,j)}$ be the XOR of t index code symbols corresponding to the t semi-inner vertex sets through which the I-paths originating from $x_{(i,j),k}$ are pass through. That is, $w_k^{(i,j)}$ is the XOR of the message symbols present in $V_k^{(i,j)}$.

Theorem 1 given below gives the decoding procedure for the index code constructed for OIC structures. The decoding procedure is same as that of the decoding procedure given by Thapa, Ong and Johnson in [3] for IC structures except that the tree T_k needs to be replaced by tree $T_k^{(i,j)}$ and y_k needs to be replaced with $w_k^{(i,j)}$.

Theorem 1. For any OIC structure, the index code constructed from (4) and (5) can be decoded by using the given below method.

- For $x_k \in V_{NI}$, the message x_k can be decoded from y_k given in (5).
- The inner vertex $x_{(i,j),k} \in \tilde{V}_I^{(i,j)}$ is decoded by computing the XOR of all index code symbols corresponding to the non-leaf vertices at depth greater than zero in $T_k^{(i,j)}$ and $w_k^{(i,j)}$, where $T_k^{(i,j)}$ is the rooted tree with $x_{(i,j),k}$ as the root node and all other inner vertices in $V_k^{(i,j)}$ as the leaves.

The following theorem establish the index coding capacity and broadcast rate of OIC structures.

Theorem 2. The index coding capacity $C(G)$ of an OIC structure G with s semi-inner vertex set ($s = \sum_{i=0}^{d} w_i$ = number of nodes in polytree) is given by $C(G) = \frac{1}{|V_{NI}|+s}$.

In the OIC structures, if the number of semi-inner vertex sets is one ($s = 1$), then the OIC structure becomes an IC structure. Hence, when $s = 1$, the encoding, decoding and optimality results in this paper exactly match the results given by Thapa, Ong and Johnson in [3]. The following example illustrates the encoding and decoding of OIC structures.

Example 4. Consider the index coding problem with side-information graph given in Figure 2. In G, we have $V_I^{(0,1)} = \{x_1, x_3, x_4\}$ and $V_I^{(1,1)} = \{x_3, x_5, x_6\}$. The polytree structure of two inner vertex sets are shown on Figure 7. For G, the index code obtained from (4) and (5) is given below.

$$\mathfrak{C} = \{\underbrace{\underbrace{x_1+x_3+x_4}_{\in V_I^{(0,1)}}}_{y_I^{(0,1)}},\ \underbrace{\underbrace{x_3+x_5+x_6}_{\in V_I^{(1,1)}}}_{y_I^{(1,1)}},\ \underbrace{\underbrace{x_2}_{\in V_{NI}} + \underbrace{x_1}_{N_G^{+}(x_2)}}_{y_2}\}.$$

Trees $T_1^{(0,1)}, T_3^{(0,1)}, T_4^{(0,1)}, T_5^{(1,1)}$ and $T_6^{(1,1)}$ corresponding to the inner vertices x_1, x_3, x_4, x_5 and x_6 are given in Figure 8. The decoding of each message symbol from $\mathfrak{C}$ is summarised in Table III. In Table III, we use γ_k to denote the index code symbols used by R_k to decode x_k and τ_k to denote the sum of index code symbols present in γ_k.

Acknowledgement

This work was supported partly by the Science and Engineering Research Board (SERB) of Department of Science and Technology (DST), Government of India, through J.C. Bose National Fellowship to B. Sundar Rajan.

x_k	Tree	γ_k	τ_k
x_1	T_1	$y_I^{(0,1)}$	$x_1 + \underbrace{x_3+x_4}_{\text{side-information}}$
x_2	$\in V_{NI}$	y_2	$x_2 + \underbrace{x_1}_{\text{side-information}}$
x_3	T_3	$y_I^{(0,1)}, y_2$	$x_3 + \underbrace{x_2+x_4}_{\text{side-information}}$
x_4	T_4	$y_I^{(0,1)}, y_I^{(1,1)}, y_2$	$x_4 + \underbrace{x_2+x_5+x_6}_{\text{side-information}}$
x_5	T_5	$y_I^{(1,1)}$	$x_5 + \underbrace{x_3+x_6}_{\text{side-information}}$
x_6	T_6	$y_I^{(1,1)}$	$x_6 + \underbrace{x_3+x_5}_{\text{side-information}}$

TABLE III: Decoding of ICP described by Figure 2

$$\begin{array}{c} V_I^{(0,1)} = \{x_1, x_3, x_4\} \\ \Big\downarrow {\scriptstyle V_I^{(0,1)} \cap V_I^{(1,1)} = \{x_3\}} \\ V_I^{(1,1)} = \{x_3, x_5, x_6\} \end{array}$$

Fig. 7: Polytree of semi-inner vertex sets of Figure 2.

References

[1] Y. Birk and T. Kol, "Informed-source coding-on-demand (ISCOD) over broadcast channels", in Proc. IEEE Conf. Comput. Commun., San Francisco, CA, 1998, pp. 1257-1264.

[2] Z. Bar-Yossef, Z. Birk, T. S. Jayram and T. Kol, "Index coding with side-information", in Proc. 47th Annu. IEEE Symp. Found. Comput. Sci., Oct. 2006, pp. 197-206.

[3] C. Thapa, L. Ong and J. Johnson, "Interlinked cycles for index coding: Generalizing cycles and cliques", *in IEEE Trans. Inf. Theory,*, vol. 63, no.9, pp.3692-3711, Jun. 2017.

[4] C. Thapa, L. Ong and J. Johnson, "Corrections to "Interlinked Cycles for Index Coding: Generalizing Cycles and Cliques" [Jun 17]", *in IEEE Trans. Inf. Theory,*, vol. 64, no.9, pp.6460-6460, Sept. 2018.

[5] H. Maleki, V. Cadambe, and S. Jafar, "Index coding-an interference alignment perspective", in IEEE Trans. Inf. Theory, vol. 60, no.9, pp.5402-5432, Sep. 2014.

[6] A. Blasiak, R. Kleinberg and E. Lubetzky, "Broadcasting with side-information: Bounding and approximating the broadcast rate" in IEEE Trans. Inf. Theory, vol. 59, no.9, pp.5811-5823, Sep. 2013.

[7] R. Dechter, "Reasoning with probabilistic and deterministic graphical models: Exact Algorithms," Morgan and Claypool Publishers, 2013.

[8] M. B. Vaddi and B. S. Rajan, "Optimal index codes for a new class of interlinked cycle structure," in IEEE Communication Letters, volume 22, issue 4, pp. 684-687, April 2018.

[9] M. B. Vaddi and B. S. Rajan, "On a Conjecture on Optimality of Index Codes from Interlinked Cycle Cover Scheme," in Proc. IEEE ISIT 2018, Vail, Colorado, USA, pp.606-610.

[10] M. B. Vaddi and B. S. Rajan, "A Generalisation of Interlinked Cycle Structures and Their Index Coding Capacity," in arXiv: 1901.06804v1 [cs.IT] 21 Jan 2019.

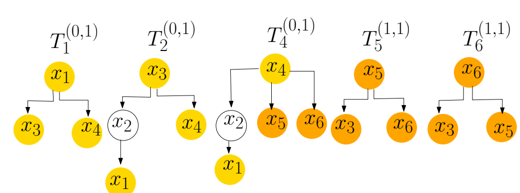

Fig. 8: Trees of inner vertices in Figure 2.