Random Linear Intersession Network Coding With Selective Cancelling

Chih-Chun Wang
Center of Wireless Systems and Applications (CWSA)
School of ECE, Purdue University

Ness B. Shroff
Departments of ECE and CSE
The Ohio State University

***Abstract*—The network coding capacity of a single multicast traffic is characterized by the min-cut/max-flow (mcMF) theorem, which can be achieved by random linear network coding (RLNC). Nonetheless, the graph-theoretic characterization for multiple unicast/multicast traffic remains an open problem. This paper proposes and studies a new class of intersession-network-coding schemes: *RLNC with selective cancelling (SC)*, which inherits the complexity advantage of RLNC once the set of selective cancelling edges is decided. A graph-theoretic characterization is provided for the *achievable rates* of RLNC with SC for the *general multiple multicast setting*. The findings contain most existing achievability results as special cases, including the mcMF theorem of the single multicast traffic and the existing characterization of pairwise intersession network coding. One prominent feature of the proposed approach is its focus on the achievability analysis with arbitrary network topology, arbitrary inter-session packet-mixing capability, and arbitrary traffic demands, which distinguishes the results from the special case analysis, capacity outer bound constructions, and the pattern-based (butterfly-based) superposition arguments.**

I. Introduction

Network coding allows *packet mixing* at intermediate nodes and provides strict throughput improvements for modern communication networks. The capacity region of network coding is well understood when only a single multicast session exists in the network [1]. Network coding across multiple sessions, termed *intersession network coding*, also demonstrates significant throughput improvement in the *butterfly network* [2] and empirically enhances the network performance by 50–200% [3]. Nonetheless, the problem of a unifying characterization of intersession network coding remains largely open.

The lack of progress on intersession network coding is mainly attributed to its intrinsic hardness. For example, checking the existence of intersession network coding solutions is an NP-complete problem [4], its information-theoretic characterization is heavily intertwined with the notoriously elusive *fundamental regions of the entropy function* [5], and linear coding is insufficient to achieve the intersession coding capacity [6]. Even when restricting our focus to only linear codes on directed acyclic networks, it is shown [7], [8] that the feasibility of linear intersession network coding is alphabet $\mathsf{GF}(q)$ dependent, and the complexity of determining the feasibility of linear intersession network coding is similar to that of the long standing problems of finding solutions of multi-variate polynomials. It is worth mentioning that with a fixed number of N coexisting sessions and a fixed alphabet size $\mathsf{GF}(q)$, finding a network coding solution is a polynomial-time task with respect to the network size [9]. However, the complexity grows exponentially with respect to N and $b = \log_2(q)$, the number of bits representing each alphabet.

To better understand the capacity of intersession network coding, many ongoing works focus on more tractable outer/inner bounds analysis for networks of general topology. Capacity outer bounds are often devised based on *generalized cut conditions* (see [10] and the references therein).

For the achievable rate regions (capacity inner bounds), existing works can be categorized into three major approaches. Since the butterfly network is well-known to admit intersession coding benefits, many works focus on *decomposing* the original network into many butterfly substructures [11] and use linear programming for the corresponding network resource allocation [12]. The second type of approaches classifies the coded traffic by the *participating sessions.* The traffic flow that is a mixture of N' unicast sessions can then be treated as generated from a single multicast session with N' symbols, to which one can apply the min-cut/max-flow (mcMF) theorem [13], [14]. Recently, a new characterization has been discovered for *pairwise intersession network coding*, which mixes only two symbols of two coexisting unicast/multicast sessions [2]. When combined with the *superposition principle,* the pairwise coding results are used to derive new achievable rates for N coexisting sessions [15].

This paper proposes and studies a new class of intersession-network-coding schemes: *random linear network coding (RLNC) with selective cancelling (SC)* for networks of general topology. A graph-theoretic characterization is provided for the feasibility of RLNC w. SC in the setting of N multicast sessions. The results include as special cases all existing achievability approaches and characterize a strictly larger achievable rate region. Although focusing only on achievable rates, RLNC w. SC is capable of achieving the capacity in many instances including the setting of a single multicast session. This work can thus be viewed as a generalization of *the achievability part of the mcMF theorem* from the single multicast session to multiple multicasts. The random construction of our scheme inherits the complexity advantages of RLNC [16] once the set of selective cancelling edges is decided. With the complexity advantage and the larger achievable throughput, RLNC w. SC could have impact on designing high-performance intersession network coding protocols.

II. The Formulation

A. Graph-Theoretic Notation

We consider only directed acyclic graphs $G = (V, E)$. For any node $v \in V$, use $\text{In}(v) \subseteq E$ to denote the incoming edges $(u, v) \in E$. Similarly, $\text{Out}(v) \subseteq E$ contains all $(v, w) \in E$. For any edge $e = (u, v)$, the tail and head of e are denoted by $\mathsf{tail}(e) = u$ and $\mathsf{head}(e) = v$, respectively. A path is an ordered set of edges $P = e_1 e_2 \cdots e_n$ such that $\mathsf{head}(e_i) = \mathsf{tail}(e_{i+1})$ for $i \in [1, n-1]$. For any v used by P, we define vP as the path segment of P from v to $\mathsf{head}(e_n)$. $\mathsf{tail}(e_1)$ and $\mathsf{head}(e_n)$ are the *beginning node* and the *ending node* of P, respectively.

B. The Network Setting

The network is modelled as a directed acyclic graph G with each edge capable of carrying one symbol per time slot. High-rate links are modelled by parallel edges. There are N distinct *source edges* $S = \{s_1, \cdots, s_N\}$ and each source is generating symbols Y_1 to Y_N, respectively. There are N disjoint sets of *destination edges* $\mathbf{d}_1$ to $\mathbf{d}_N$. Any destination edge $d \in \mathbf{d}_i$ requests/desires the symbol Y_i. Namely, there are N coexisting multicast $(s_i, \mathbf{d}_i)$ sessions. Collectively, we term $\{(s_i, \mathbf{d}_i)\}$ as the *traffic demand* of the network.[1] This form of traffic demand is general and can describe any traffic pattern. For example, if a single source s_i would like to send a collection of symbols $\{Y_{i,j} : j\}$ to a destination d_i, we simply add multiple $s_{i,j}$ sources and multiple $d_{i,j}$ destinations such that $\forall j, \mathsf{head}(s_{i,j}) = \mathsf{head}(s_i)$ and $\forall j, \mathsf{tail}(d_{i,j}) = \mathsf{tail}(d_i)$.

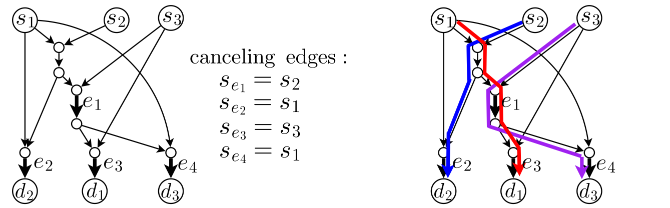
For any $e \in E$, M_e is the N-dimensional *global coding vector* transmitted over e, of which the coordinates correspond to the coefficients for symbols Y_1 to Y_N. For any edge $e = (u, v)$, one has the freedom to choose the *local kernel* (local transfer matrix) $x_{w,e}$ such that $M_e = \sum_{(w,u) \in \text{In}(u)} x_{w,e} M_{w,u}$. The global coding vectors of the source edges are simply the *elementary unit vectors*, e.g., $M_{s_1} = (1, 0, \cdots, 0)$. For an M_e coding vector, we use the "s_i-coordinate" to refer to the coordinate that corresponds to the i-th symbol Y_i.

III. The Scheme

We consider a simple class of intersession network coding schemes: Random Linear Network Coding (RLNC) with Selective Cancelling (SC). The scheme works as follows.

We have the freedom to choose an arbitrary set of K edges e_1 to e_K in G, termed the *cancelling edges*. Each cancelling edge e_k is associated with a source s_{e_k}. RLNC [16] is performed for all edges other than the e_k edges, i.e., the local kernels $x_{w,e}$ are chosen uniformly randomly from the finite field $\mathsf{GF}(q)$. *For the e_k edges, the local kernels x_{w,e_k} are chosen such that M_{e_k} has zero s_{e_k}-coordinate.* Namely, the s_{e_k}-coordinate of M_{e_k} is *cancelled*. Cancelling the s_{e_k}-coordinate is always feasible as we can simply let x_{w,e_k} be all zero, which results in $M_{e_k} = \mathbf{0}$. If more than one set of

[1] We use interchangeably the source/destination *edge* and source/destination *node* to refer to the edge/node that generates/requests Y_i. While the edge-based notation is more suitable for rigorous discussion, in many figures we still express a source/destination by a *node* for its intuitive representation.



(a) The cancelling edges

(b) Routing solutions for individual unicast sessions

Fig. 1. A 3-sources/3-destination network with cancelling edges.

x_{w,e_k} can cancel the s_{e_k}-coordinate, choose randomly[2] from all eligible x_{w,e_k}. Both the normal mixing and the cancelling operations depend only on the incoming coding vectors of a given node, which greatly reduces the control overhead as network-wide coordination is unnecessary (cf. RLNC in [16]).

Take Fig. 1(a) for example, which consists of three unicast sessions (s_1, d_1) to (s_3, d_3). If we perform RLNC on the entire network (without any cancelling edges), then the packets received by d_1, d_2, and d_3 will be contaminated by $\{Y_2, Y_3\}$, Y_1, and $\{Y_1, Y_2\}$, respectively. None of the d_i can recover its desired symbol Y_i. RLNC is hence not a solution for intersession network coding. We overcome this drawback of RLNC by introducing *cancelling edges*. Suppose we assign four cancelling edges e_1 to e_4 in Fig. 1(a), which cancel the s_2, s_1, s_3, and s_1 coordinates, respectively. It is easy to check that *for d_2, the interference Y_1 is removed by the cancelling edge e_2. Therefore, d_2 is free from any interference.* Similarly, for d_1 the interferences Y_2 and Y_3 are removed by e_1 and e_3, and for d_3, the interferences Y_1 and Y_2 are removed by e_1 and e_4. Therefore both d_1 and d_3 are free from any interference.

Nonetheless, a byproduct of removing interference by cancelling edges is that the desired information may also be removed, which means that the traffic demand cannot be met. *One thus has to consider the topological relationship among all cancelling edges to ensure that the desired information is not removed accidentally by the collective effect of all cancelling edges.* This turns out to be a non-trivial task, which is illustrated as follows. Continue our example in Fig. 1(a). If we *route* the packet along the (s_2, d_2) path in Fig. 1(b) (without any coding) while keeping the rest of the network silent, then we can indeed transmit Y_2 from s_2 to d_2. Moreover, since this pure routing solution uses only the (s_2, d_2) path, we have $M_{e_1} = \mathbf{0}$, $M_{e_2} = (0, 1, 0)$, $M_{e_3} = \mathbf{0}$, and $M_{e_4} = \mathbf{0}$. This pure routing solution *complies with* the cancelling edges e_1 to e_4 in the sense that the s_{e_k}-coordinate of the cancelling edge M_{e_k} is indeed zero for $k = 1$ to 4. Similarly, if we route the Y_1 (resp. Y_3) along the (s_1, d_1) path (resp. the (s_3, d_3) path) in Fig. 1(b), then d_1 (resp. d_3) receives Y_1 (resp. Y_3)

[2] Note that cancelling a coordinate is different from simply *blocking an input*. For example, if the two incoming vectors are $M_1 = (1, 2, 3)$ and $M_2 = (4, 4, 7)$, blocking any one of them will not *cancel* any coordinate. Suppose we would like to *cancel* the s_1-coordinate. The output needs to be $4M_1 - M_2$, which has to use both vectors. Given the input coding vectors $M_{w,\mathsf{tail}(e_k)}$, all x_{w,e_k} that *cancel* the s_{e_k}-coordinate form a linear space, from which we can choose x_{w,e_k} randomly.

successfully and the pure routing solution again complies with all cancelling edges. Nonetheless, it can be shown by [10] that there is no solution that satisfies the traffic demand for all three unicast sessions in this network. We thus have

> *Observation 1: Even if there exist N individual solutions for (s_1, d_1) to (s_N, d_N) (one for each session, respectively) that comply with the cancelling edges, it is not guaranteed that a single solution exists that meets the traffic demand for all sessions (s_1, d_1) to (s_N, d_N) while complying with the cancelling edges.*

This observation is in sharp contrast with the elegant RLNC theorem for a single multicast session [16], [1]:

> *Observation 2: Consider a single multicast session, if there exist N individual solutions for d_1 to d_N (one for each destination respectively), it is guaranteed that a single solution exists that meets the traffic demand for all destinations d_1 to d_N.*

We need new theorems to analyze RLNC w. SC.

Our goal is to identify a graph-theoretic characterization for the general multiple multicast problem such that *for all i, all $d \in \mathbf{d}_i$ are able to recover Y_i with close-to-one probability using the RLNC w. SC scheme when a sufficiently large $\mathsf{GF}(q)$ is used.* In this case, we say that RLNC w. SC is *feasible*.

IV. The Theorem

Theorem 1: RLNC w. SC is feasible for multiple $\{(s_i, \mathbf{d}_i)\}$ sessions if the following graph-theoretic conditions hold.

1) There exists a set of cancelling edges $E_K = \{e_k\}$ such that $\forall i, \mathbf{d}_i \subseteq E_K$. Each $e_k \in E_K$ is associated with a source s_{e_k}. If $e_k \in \mathbf{d}_i$, then we require $s_{e_k} = s_i$. This choice of E_K and $\{s_{e_k}\}$ is fixed for the remaining conditions.
2) For all i, all $d_{i,j} \in \mathbf{d}_i$, and any $s'_i \in S$ with $s'_i \neq s_i$, the edge set $\{e \in E_K : s_e = s'_i\}$ is an edge cut separating s'_i and $d_{i,j}$.
3) For each destination $d_{i,j}$ (we often drop the subscript and use d as shorthand), there exists an $E_K^{(d)} = \{e_k : k = 1, \cdots, K^{(d)}\} \subseteq E_K$ with $d \in E_K^{(d)}$, termed the *active cancelling edges* for destination d, and a set of $\{P_1, \cdots, P_{K^{(d)}}\}$ paths for each $e_k \in E_K^{(d)}$, such that jointly these paths satisfy the following conditions:
   a) Any path P_k must end at $\mathsf{tail}(e_k)$, must not use any $e_h \in E_K^{(d)}$ with $s_{e_h} = s_{e_k}$, and must not use any $e_h \in E_K \backslash E_K^{(d)}$.

   For any P_k and any node $v \in P_k$, define $S(P_k, v)$ as

$$S(P_k, v) \triangleq \bigcup_{e \in (vP_k) \cap E_K^{(d)}} \{s_e\} \cup \{s_{e_k}\}.$$

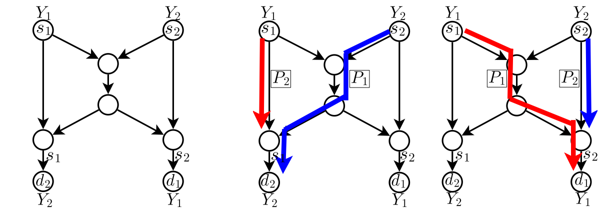
   The remaining conditions are based on $S(P_k, v)$.

   b) The beginning node v of P_k is either s_{e_k} (or equivalently $\mathsf{head}(s_{e_k})$); Or belonging to some P_h satisfying the following: $s_{e_h} = s_{e_k}$, $S(P_h, v) \subseteq S(P_k, v)$, and v is not a beginning node of P_h.
   c) Any two distinct paths P_k and P_h with $s_{e_k} = s_{e_h}$ do not share an edge.
   d) Consider any two distinct paths P_k and P_h with $s_{e_k} \neq s_{e_h}$. These two paths share an edge e only if $S(P_k, \mathsf{tail}(e)) = S(P_h, \mathsf{tail}(e))$.

The conditions of Theorem 1 are best interpreted as follows.

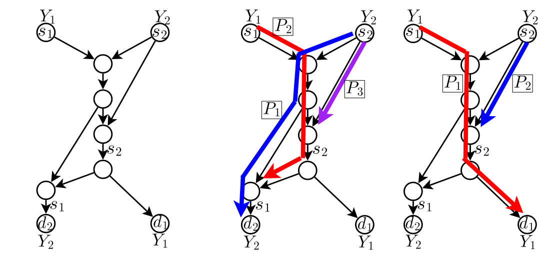
- Condition (1) expands the cancelling edge set E_K to include the destination edges $\mathbf{d}_i$. The reason behind this expansion will be clear in the end of our discussion.
- Condition (2): The purpose of selective cancelling is to remove the interference of $s'_i \neq s_i$ from the packet received by $d_{i,j}$. Therefore, we require the cancelling edges $\{e \in E_K : s_e = s'_i\}$ to form an edge cut that *shields* the interference of s'_i from $d_{i,j}$.
- Condition (3) has a flavor similar to Observation 2 in the previous section. Namely, in the broadest sense, Theorem 1 shows that the existence of $\sum_{i=1}^{N} |\mathbf{d}_i|$ solutions such that *each solution satisfies Conditions (3.a)–(3.d) for one destination d, respectively,* guarantees the existence of a single solution satisfying *all* destinations simultaneously (i.e., meeting the multiple multicast traffic demand). The Observation 2 of RLNC is thus generalized to the multiple multicast setting. The challenges described in Observation 1 are addressed by requiring each individual solution (for each destination d) to satisfy the stronger Conditions (3.a)–(3.d) instead of the weaker condition "complying with the cancelling edges."
- One necessary condition for $d_{i,j}$ to receive Y_i is the existence of a *principal information path* P from s_i to $d_{i,j}$ that uses no cancelling e_k with $s_{e_k} = s_i$ (cf. Condition (3.a)), otherwise the desired s_i-coordinate is canceled along the principal information path.
- Nonetheless, the principal information path P may use some cancelling edge e_k with $s_{e_k} \neq s_i$. Unlike using a normal edge e, using a cancelling edge e_k has its price. Specifically, whenever e_k is used, we need to create another path P_k such that jointly P_k and P satisfy some feasibility conditions (3.b)-(3.d). The more cancelling edges that are used, the more paths P_k need to be considered jointly to satisfy the feasibility conditions.
- We thus view $E_K^{(d)}$ as the active cancelling edges e_k that are used by at least one P_k. Having $d \in E_K^{(d)}$ ensures that one of the P_1 to $P_{K^{(d)}}$ is indeed the principal information P path connecting s_i and $d_{i,j}$, which in turn guarantees successful Y_i packet recovery at destination $d_{i,j}$.

Remark: Due to the intrinsic hardness of finding the capacity (see Section I), the stronger the capacity inner bound is, the less likely it will admit a simple form. As will be seen shortly after, Theorem 1 provides a strong achievable rate region for general networks that strictly supersedes many existing results, which thus results in the convoluted description of Conditions (3.a)–(3.d). A much simpler (and weaker) version of Theorem 1 can be obtained by replacing Conditions (3.b)–(3.d) by: (3.b') the beginning node of P_k must be s_{e_k}, (3.c') any P_k and P_h with $s_{e_k} = s_{e_h}$ do not share any edge except



(a) A butterfly network and the corresponding cancelling edges (b) The two P_k paths for destination d_2 (c) The two P_k paths for destination d_1

Fig. 2. The butterfly network is captured by Theorem 1.



(a) A grail network and the corresponding cancelling edges (b) The three P_k paths for destination d_2 (c) The two P_k paths for destination d_1

Fig. 3. The grail network is captured by Theorem 1.

for the common source edge $s_{e_k} = s_{e_k}$, and (3.d') any P_k and P_h with $s_{e_k} \neq s_{e_h}$ do not share any edge.

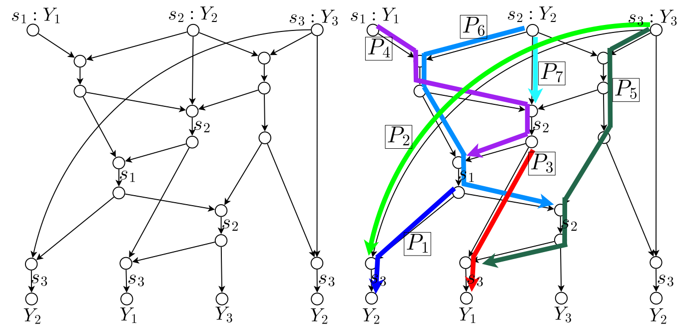
V. Examples

We provide the following examples to demonstrate the application of Theorem 1.

Example 1: The butterfly network. Consider the classic butterfly network in Fig. 2(a). If we choose two cancelling edges in the network (those edges in Fig. 2(a) with the to-be-cancelled coordinate s_{e_k} marked on the side), then we can choose $E_K^{(d)}$ and P_k for d_1 and d_2 respectively as described in Figs. 2(b) and 2(c). In Fig. 2(b), $E_K^{(d_2)}$ contains one active intermediate cancelling edge (the edge entering d_2) and one destination edge (node) d_2. The corresponding two paths in Fig. 2(b) satisfy all (3.a)–(3.d) conditions. Fig. 2(c) describes the $E_K^{(d_1)}$ and the two path P_k for d_1, which contains one active intermediate cancelling edges (the edge entering d_1) and one destination edge (node) d_1. By Theorem 1, the traffic demand in Fig. 2(a) is attainable by RLNC w. SC. Theorem 1 captures the butterfly example as a special instance.

Example 2: The grail network. In [17], it is shown that the feasibility of transmitting two symbols (one for each (s_i, d_i) session) is equivalent to the existence of either a butterfly (Fig. 2(a)) or a grail structure (Fig. 3(a)) in the network G. Namely, a butterfly and a grail are the only two structures that admit pairwise intersession network coding.

For the grail network, if we choose two cancelling edges in the network (those edges in Fig. 3(a) with the to-be-cancelled coordinate s_{e_k} marked on the side), then we can choose $E_K^{(d)}$ and P_k for d_1 and d_2 respectively as described in Figs. 3(b) and 3(c). In Fig. 3(b), $E_K^{(d_2)}$ contains two active intermediate cancelling edges and one destination edge (node) d_2. The corresponding three paths in Fig. 3(b) satisfy all (3.a)–(3.d) conditions. Fig. 3(c) describes the $E_K^{(d_1)}$ and the two path P_k for d_1, which contains one active intermediate cancelling edges and one destination edge (node) d_1. By Theorem 1, the traffic demand in Fig. 3(a) is attainable by RLNC w. SC. Theorem 1 captures the grail structure as well.



(a) A example network and the corresponding cancelling edges (b) The seven P_k paths for the left two destinations

Fig. 4. An example network demonstrating Theorem 1.

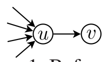
Example 3: A 3-sources/4-destinations example. Theorem 1 extends beyond pairwise coding. Consider a network in Fig. 4(a) that has three sources s_1 to s_3 that generate symbols Y_1 to Y_3 respectively. If we choose 6 cancelling edges within the network (those edges in Fig. 4(a) with the to-be-cancelled coordinate s_{e_k} marked on the side), then we can choose $E_K^{(d)}$ and P_k satisfying Conditions (3.a)–(3.d) for each destination respectively. Due to the limit of space, we illustrate only the choice of paths for the left two destinations. In Fig. 4(b), $E_K^{(d)}$ contains five active intermediate cancelling edges and two destination edges (nodes). If we choose the corresponding seven paths as in Fig. 4(b), then Conditions (3.a)–(3.d) are satisfied for any one of the left two destinations. For the right two destinations, we can also construct $E_K^{(d)}$ and P_k satisfying on Conditions (3.a)–(3.d). By Theorem 1, the traffic demand in Fig. 4(a) is attainable by RLNC w. SC. It can be shown that this example cannot be characterized by any of the existing works in [13], [12], [15], [11], [2], [14].

VI. Generality of The Main Results

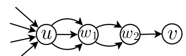
The following two techniques convert any network G to a *coding-equivalent* network G'. Namely, whether there exists a coding solution on G is equivalent to whether there exists a coding solution on G'. The proof of the coding-equivalence between G and G' is omitted due to space limitation.

Graph Conversion 1 (GC1): For any directed acyclic network $G = (V, E)$, let G' contain exactly $2|E|$ distinct nodes $u_e, v_e, \forall e \in E$. The edge set E' contains $\{(u_e, v_e) : \forall e \in E\}$ and $\{(v_e, u'_e) : \forall e, e' \in E, \mathsf{head}(e) = \mathsf{tail}(e')\}$.

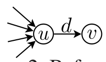
Graph Conversion 2 (GC2): Iteratively apply the following edge-based conversion for an arbitrary number of times. (See Fig. 5 for illustration.) Choose an arbitrary $e \in E$ and continue according to one of the following two cases. Case 1: $e = (u, v)$ is not a destination edge. If $|\mathrm{In}(u)| \leq 2$, make no changes. If



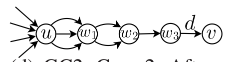
(a) GC2, Case 1: Before conversion.



(b) GC2, Case 1: After conversion.



(c) GC2, Case 2: Before conversion.



(d) GC2, Case 2: After conversion.

Fig. 5. Illustration of the cases of Graph Conversion 2 (GC2).

$|\text{In}(u)| \geq 3$, remove e from G. Add $|\text{In}(u)| - 2$ intermediate nodes w_1 to $w_{|\text{In}(u)|-2}$. Add $|\text{In}(u)| - 1$ parallel edges between u and w_1. For $i = 2$ to $|\text{In}(u)| - 2$, add $|\text{In}(u)| - i$ parallel edges between w_{i-1} and w_i. Add one edge between $w_{|\text{In}(u)|-2}$ and v. Case 2: $e = (u, v)$ is a destination edge $d \in \mathbf{d}_i$ for some i. If $|\text{In}(u)| = 1$, make no changes. If $|\text{In}(u)| \geq 2$, remove e from G. Add $|\text{In}(u)| - 1$ intermediate nodes w_1 to $w_{|\text{In}(u)|-1}$. Add $|\text{In}(u)| - 1$ parallel edges between u and w_1. For $i = 2$ to $|\text{In}(u)| - 1$, add $|\text{In}(u)| - i$ parallel edges between w_{i-1} and w_i. Add one edge between $w_{|\text{In}(u)|-1}$ and v. Reassign $(w_{|\text{In}(u)|-1}, v)$ as the new destination edge.

Proposition 1: Consider the *achievability part of the results* in the following works: the mcMF theorem for single multicast [16], [1], two single-symbol unicasts [2], [17], decoding-&-remixing schemes [13], [11], pollution-free treatment [14], and all their superposition-principle-based generalizations [13], [15], [11], [14]. For any G that is characterized by any of the above works as a network admitting intra- or inter-session coding solutions, one can construct the cancelling edges E_K, $E_K^{(d)}$, and P_k on a coding-equivalent network G' (following GC1 and GC2) such that Theorem 1 is satisfied.

The above proposition and Example 3 show that Theorem 1 describes a capacity inner bound that is strictly larger than those of the above works.

For the following, we provide some intuition behind Proposition 1. One important ingredient of the results in [13], [11], [14] is that one can *decode* some symbols in the intermediate nodes and then *remix* them with other network traffic. Since decoding Y_{i_0} and then sending it over an edge e is no different than cancelling all other $\{s_i : i \neq i_0\}$ coordinates on e, decoding-&-remixing is a special case of successively cancelling coordinates in $\{s_i : i \neq i_0\}$ when applying RLNC w. SC on a coding-equivalent G' constructed from GC2. Any decode-&-remix scheme is thus captured by Theorem 1. Note that cancelling is strictly more general than decoding. For example, if node u receives two 3-dimensional global coding vectors $M_1 = (1, 2, 3)$ and $M_2 = (4, 4, 7)$, then u cannot decode any of the three symbols Y_1 to Y_3. On the other hand, u can send out one of the three coding vectors $4M_1 - M_2$, $2M_1 - M_2$, and $7M_1 - 3M_2$, which cancel, respectively, the s_1, s_2, and s_3 coordinates of the outgoing coding vector.

VII. Conclusion & Future Directions

We have studied RLNC w. SC and provided new achievable rate characterization for the general multiple multicast setting, which is strictly more powerful than the existing results. The RLNC w. SC scheme possesses many practical advantages and could have new impact on designing intersession network coding protocols.

One critical question of applying Theorem 1 is how to identify efficiently the *optimal cancelling edge set* E_K, which is an open problem for our future research. On the other hand, as an achievability result, *any* good E_K will lead to a new achievable rate, and for practical applications one may not need to identify the *optimal* E_K. For more specialized settings such as pairwise intersession coding [17] or multi-resolution video multicast that have special structure on the traffic demand $\{(s_i, \mathbf{d}_i)\}$, Theorem 1 can be further simplified, which might enables new methods of searching for E_K. We are currently investigating how to effectively identify good/optimal E_K in a practical setting and the results will be used to develop new intersession network coding protocols with the help of distributed resource allocation algorithms.

References

[1] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 782–795, October 2003.

[2] C.-C. Wang and N. Shroff, "Beyond the butterfly — a graph-theoretic characterization of the feasibility of network coding with two simple unicast sessions," in *Proc. IEEE Int'l Symp. Inform. Theory*. Nice, France, June 2007.

[3] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: Practical wireless network," in *Proc. ACM Special Interest Group on Data Commun. (SIGCOMM)*, 2006.

[4] A. Lehman, "Network coding," Ph.D. dissertation, MIT, 2005.

[5] X. Yan, R. Yeung, and Z. Zhang, "The capacity region for multi-source multi-sink network coding," in *Proc. IEEE Int'l Symp. Inform. Theory*. Nice, France, 2007.

[6] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," *IEEE Trans. Inform. Theory*, vol. 51, no. 8, pp. 2745–2759, August 2005.

[7] ——, "Networks, matroids, and non-Shannon information inequality," *IEEE Trans. Inform. Theory*, vol. 53, no. 6, pp. 1949–1969, June 2007.

[8] ——, "Linear network codes and systems of polynomial equations," *IEEE Trans. Inform. Theory*, vol. 54, no. 5, pp. 2303–2316, May 2008.

[9] A. Subramanian and A. Thangaraj, "A simple algebraic formulation for the scalar linear network coding problem," in *Proc. 46th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, Illinois, USA, September 2008, pp. 177–184.

[10] X. Yan, J. Yang, and Z. Zhang, "An outer bound for multisource multisink network coding with minimum cost consideration," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2373–2385, June 2006.

[11] D. Traskov, N. Ratnakar, D. Lun, R. Koetter, and M. Médard, "Network coding for multiple unicasts: An approach based on linear optimization," in *Proc. IEEE Int'l Symp. Inform. Theory*. Seattle, USA, July 2006.

[12] A. Eryilmaz and D. Lun, "Control for inter-session network coding," in *Proc. 3rd Workshop on Network Coding, Theory, & Applications (NetCod)*, January 2007.

[13] T. Cui, L. Chen, and T. Ho, "Energy efficient opportunistic network coding for wireless networks," in *Proc. 27th IEEE Conference on Computer Communications (INFOCOM)*. Phoenix, USA, April 2008.

[14] Y. Wu, "On constructive multi-source network coding," in *Proc. IEEE Int'l Symp. Inform. Theory*. Seattle, USA, July 2006, pp. 1349–1353.

[15] A. Khreishah, C.-C. Wang, and N. Shroff, "An optimization-based rate control for communication networks with inter-session network coding," in *Proc. 27th IEEE Conference on Computer Communications (INFOCOM)*. Phoenix, USA, April 2008.

[16] T. Ho, M. Médard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inform. Theory*, vol. 52, no. 10, pp. 4413–4430, October 2006.

[17] C.-C. Wang and N. Shroff, "Intersession network coding for two simple multicast sessions," in *Proc. 45th Annual Allerton Conf. on Comm., Contr., and Computing*. Monticello, IL, September 2007.