MULTI-SESSION NETWORK CODING CHARACTERIZATION USING NEW LINEAR CODING FRAMEWORKS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Jaemin Han

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2016

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

My deepest gratitude goes to my advisor, Professor Chih-Chun Wang. You guided me through this rough 6-year journey and inspired me to become a better personality as a researcher, as a husband, and as a father. Your intelligence, endless passion, and integrity have lead our works into academic completions. Meeting you as an advisor in the doctoral program has been one of the greatest things that happened in my life.

I thank my dissertation committee members, Prof. Ness Shroff, Prof. David Love, and Prof. Xiaojun Lin for their services and advices in my candidacy. I am especially grateful to Prof. Ness Shroff for your guidance and insightful discussions on my research. My first work would be stranded without your warm support and intelligence. My sincere gratitude also goes to Mrs. Wang for your heartfelt support and hospitality. I am also grateful to Wei-Cheng Kuo, Imad Ahmad, Po-Kai Huang, Xiaohang Li, Shizen Zhao, Jia Zhang, and Chih-Hua Chang for helpful discussions and hilarious chats in the lab. Special thanks to Wei-Cheng Kuo and Imad Ahmad for your support and friendship that helped me through ups and downs in different places. I have been very lucky to have Song Noh, Ilyong Chun, and Junil Choi around. You all helped me realize that the academia could be full of fun and wonder when I have good friends by my side. Thanks all members of the Wallyholics club for sharing weekly excitements and beers with me in Purdue! I especially thank Prof. Daesik Hong, my M.S. advisor in Yonsei University, for your support and encouragement to study abroad. You had taught me how to become a dedicated and thoughtful scholar.

Lastly, and most of all, I am profoundly grateful to my family. Eric, Jooyoung, and my parents, I am dedicating this dissertation to you. My achievements would not exist without your love.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Han, Jaemin Ph.D., Purdue University, May 2016. Multi-session Network Coding Characterization using New Linear Coding Frameworks. Major Professor: Chih-Chun Wang.

Recently, Network Coding (NC) has emerged as a promising technique in modern communication networks and has shown extensive potentials in practical implementations and theoretical developments. Nevertheless, the NC problem itself remains largely open especially where the multiple flows (sessions) exist. Unlike single-session where all receivers want the same information, they demand different set of information in multi-session and thus NC strategy should be carefully designed to avoid interferences. However, characterizing an optimal strategy (even a simple solution) has known to be of prohibitive complexity even we restrict to the linear network coding (LNC) problem.

This thesis provides a fundamental approach to overcome this multi-session complexity. We first consider the Directed Acyclic Integer-Capacity network model that characterizes the real-life instantaneous *Wireline Networks*. In this model, people recently applied the results of wireless interference channels to evade the multi-session difficulties. However, our NC understanding is still nascent due to different wireline channel characteristics to that of wireless. Therefore, motivated by the graph-theoretic characterizations of classic linear NC results, we first propose a new Precoding-based Framework and its fundamental properties that can bridge between the point-to-point network channel and the underlying graph structures. Such relationships turn out to be critical when characterizing graph-theoretically the feasibility of the Precoding-based solutions. One application of our results is to answer the

conjecture of the 3-unicast interference alignment technique and the corresponding graph-theoretic characterization conditions.

For *Wireless Networks*, we use the packet erasure network model that characterizes the real-life harsh wireless environment by the probabilistic arguments. In this model, we consider the multi-session capacity characterization problem. Due to the signal fading and the wireless broadcasting nature, the linear NC designer needs to optimize the following three considerations all together: LNC encoding operations; scheduling between nodes; and the feedback and packet reception probabilities. As a result, the problem itself is more convoluted than that of wireline networks where we only need to focus on how to mix packets, i.e., coding choices, and thus our understandings have been limited on characterizing optimal/near-optimal LNC strategies of simple network settings. To circumvent the intrinsic hardness, we have developed a framework, termed Space-based Framework, that exploits the inherent linear structure of the LNC problem and that can directly compute the LP(Linear Programming)-based LNC capacity outer bound. Motivated by this framework, this thesis fully characterizes more complex/larger network settings: The Shannon capacity region of the 3-node network with arbitrary traffic patterns; and The LNC capacity region of the 2-flow smart repeater network.

1. INTRODUCTION

In the communication network where multiple nodes are intertwined with each other, it was commonly believed that an information packet should be unchanged during delivery. As a result, the routing (store-and-forward) was an dominant form of distributing packets and thus network solution was approached as to optimize multi-commodities (flow demands) between nodes.

This long-lasting routing paradigm has been enlightened by the seminal work from Ahlswede *et al.*, the concept of *Network Coding* (NC) in 2000 [1]. The new concept that information packets can be mixed to be beneficial, not only achieved the single multicast capacity, but also broadened our understandings of the notoriously challenging network information problem. Network Coding has been further concreted by the follow-up works from theory to practice. Li *et al.* showed that linear network coding (LNC) suffices to achieve a *single-session* (also known as intra-session) capacity [2], which followed by the well-formulated framework for general *multi-session* (also known as inter-session) settings [3]. This classic framework bridged a straight connection between a given network information flow problem and a finite field algebraic variety (the set of solutions of a system of polynomial equations), providing a critical step in shifting Network Coding from knowledge to application. Network Coding became further implementation-friendly by the packet-header padding of mixing coefficients [4] along with the success of a polynomial-time algorithm [5] and the distributed random linear network coding [6], all in single-session scenario.

Thanks to these fundamental efforts, NC became an promising technique in modern communication systems. The numerous applications such as P2P file systems and recent wireless testbeds [7, 8] have also demonstrated that LNC can provide substantial throughput gains over the traditional 802.11 protocols in a practical environment. Several literatures also showed some potential extensions to the reliable communications from a security perspective [9, 10]; over network errors and erasures [11–14]; to the broadcasting systems for the multi-resolution support [15, 16]; to the resilient storage recovery [17, 18]; and even to the index coding problem [19, 20]

1.1 Limited Understandings in Multi-session Network Coding

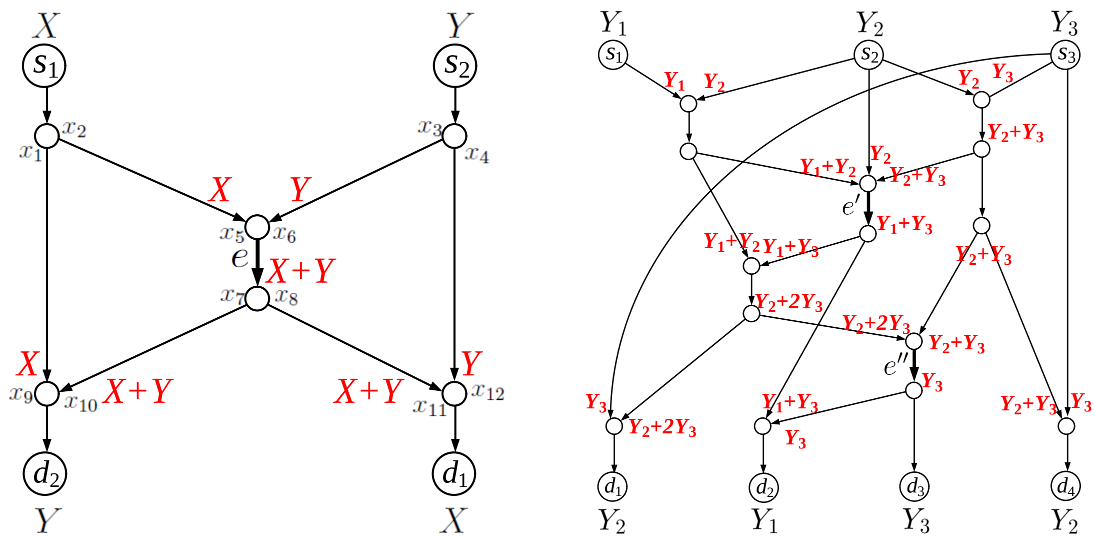
Despite its great potentials, the NC problem itself is largely open in general, especially where multiple flows (sessions) exist. Unlike single-session where all receivers want the same set of information, in multi-session scenario, receivers require different set of information from sources. Therefore, "how to mix information" should be carefully designed over the entire network, otherwise an inevitable interference from undesired senders may occur. Since the design needs to avoid interferences while satisfying the given traffic demands, our multi-session understandings in optimal/near-optimal NC strategies have been limited: over some special network topologies [21–23]; under restrictive rate constraints [24–26]; and by inner and outer bounding approaches [22,27,28]. Even we restrict our focus on the linear NC problem, the simplest scenarios of 2-unicast/multicast with single rates are only people have solved completely [24, 29, 30]. There are some achievability results for larger than single rates [25, 26] but still the LNC capacity for arbitrary 2-unicast/multicast has not been resolved up to date. This is also one reason why the simple form of 2-unicast instances, i.e., the famous Butterfly structure, has been exploited mostly in practical implementations and theoretical developments [7,31–35]. Therefore in this thesis, we propose two new frameworks that help us to characterize the multi-session NC problem. Both frameworks are built upon the linear structure of the packet-mixing nature, and are designed to provide an tractable analysis of the notorious multi-session Network Coding problems. Although it is known that there are some cases that the linear network coding (LNC) is not sufficient to achieve the multi-session capacity [36], the problem characterization based upon the linear structure will be invaluable in broadening our currently-limited understandings and in practical viewpoints as well. From the following section, we will introduce these two linear frameworks and develop our motivations in more depth.

1.2 Wireline Networks - Directed Acyclic Integer-Capacity Network

The NC problem in Wireline Networks has been considered in the Directed Acyclic Integer-Capacity network model [3]. Unlike the error-prone wireless environment, a packet transmission over a wired link (or edge) can be easily made error-free by forward error correcting codes. We can thus exclusively focus on the information delivery without worrying too much about erroneous receptions. There might be some topological changes in the network (such as a temporal link failure), but we focus on fixed topologies to understand the problem more clearly. We further assume that the network is directed acyclic (there are no cycles) and follow the widely-used instantaneous transmission model for the directed acyclic integer-capacity network [3].

1.2.1 Linear Network Coding : The Classic Algebraic Framework

Consider the following scenarios as shown in Fig. 1.1. The directed acyclic integer-capacity network model and the corresponding algebraic framework [3] for the LNC problem can be understood by looking into these examples. Fig. 1.1(a) illustrates the famous Butterfly topology where d_1 and d_2 wants to receive packets from s_1 and s_2, respectively. At each node, a packet transmitted through an outgoing link is a linear combination of the packets from all incoming links. For example, a packet transmitted through an link e is a linear combination of the packets from two incoming links, whose coefficients are x_5 and x_6, respectively. At each node, we have such coefficients for all incoming to outgoing relationships, and the collection of such coefficients in the network is called *local encoding kernels* (or *network variables*). For example, the network variables in Fig. 1.1(a) are $\{x_1, ..., x_{12}\}$. Then once transmitted, by the instantaneous transmission model, each destination will see the following linear combination of the packets X and Y whose coefficients are high-order polynomials



(a) (b)

Fig. 1.1. (a) The Butterfly structure (2-unicast) with the corresponding network variables $\{x_1, ..., x_{12}\}$ and the resulting LNC transmission that satisfies the traffic demand of $(R_{s_1\to d_1}, R_{s_2\to d_2}) = (1, 1)$; and (b) The 2-unicast and 1-multicast combination scenario and the resulting LNC transmission that satisfies the traffic demand $(R_{s_1\to d_2}, R_{s_2\to\{d_1,d_4\}}, R_{s_3\to d_3}) = (1, \{1, 1\}, 1)$

with respect to the network variables: $(x_1x_9 + x_2x_5x_7x_{10}) \cdot X + (x_3x_6x_7x_{10}) \cdot Y$ at d_2; and $(x_2x_5x_8x_{11}) \cdot X + (x_3x_6x_8x_{11} + x_4x_{12}) \cdot Y$ at d_1, respectively. The objective of the LNC problem is to find a specific assignment of the network variables that can satisfy the given traffic demand while being interference-free, i.e., solving the following feasibility equations:

$$d_1: \quad x_3x_6x_8x_{11} + x_4x_{12} = 0, \qquad x_2x_5x_8x_{11} \neq 0,$$
$$d_2: \quad x_1x_9 + x_2x_5x_7x_{10} = 0, \qquad x_3x_6x_7x_{10} \neq 0.$$

Note that the first column of equations are to be interference-free from the undesired packets (removing interferences) while the second column of equations are to receive the desired packets (satisfy the traffic demand).

In this example, we can easily find a solution that satisfies $(R_{s_1\to d_1}, R_{s_2\to d_2}) = (1, 1)$: set -1 to both x_9 and x_{12}, and set 1 to all the other variables. The resulting packet transmissions are shown by a red color in Fig. 1.1(a). Notice that without packet-mixing, an link e would be a bottleneck for each unicast. As a result, any routing solutions cannot simultaneously meet the rate $(1, 1)$ for this 2-unicast.

How about the scenario in Fig. 1.1(b)? For this 2-unicast and 1-multicast combination scenario, [37] has shown that $(R_{s_1\to d_2}, R_{s_2\to\{d_1,d_4\}}, R_{s_3\to d_3}) = (1, \{1, 1\}, 1)$ can be LNC-achievable. The feasibility equations and the corresponding LNC solutions (assignment of network variables) are left to the reader but one solution is shown by a red color. Notice that both d_2 and d_3 do not want Y_2 from s_2, and we are canceling Y_2 at two edges e' and e'' to be interference-free, while satisfying the multicast traffic from s_2 to $\{d_1, d_4\}$.

As you can see from these examples, finding a solution (or algebraic variety) that satisfies the feasibility equations directly tells us how to design an linear network code. This classic algebraic framework [3] thus bridges a straight connection between a given network information flow problem and an algebraic solution. Notice that it is easy to check whether the given solution is feasible but not easy to come up with a solution from the beginning. This is mainly due to the interference-free requirements of the multi-session problem that must be zero in the feasibility equations, unlike single-session where we only need to satisfy the non-zero-equations (satisfying the traffic demand), which can be done with high probability by choosing the values of local encoding kernels independently and randomly. It turned out that the complexity of finding a algebraic solution in multi-session scenarios becomes NP-hard for arbitrary communication demands [3, 38].

1.2.2 New Precoding-based Framework

To circumvent this NP-hard complexity, people recently focused on the analogy between the Directed Acyclic Wireline Network and the Wireless Interference Channel that the instantaneous transmission is assumed in the directed acyclic model as in wireless. Therefore, applying the techniques developed in Wireless Interference

Channels was a natural sequence. Such applications are the linear deterministic interference cancellation technique of 2-user Interference Channel [25, 26] and the interference alignment technique [39] to 3-unicast, called 3-unicast Asymptotic Network Alignment (ANA) scheme [40, 41].

This brings a new perspective on the multi-session LNC problem. As there is no control on wireless channels between two end points, the network designer can focus on designing the *precoding* and *decoding mappings* at the source and destination nodes while allowing randomly generated local encoding kernels [6] within the network. Compared to the classic algebraic framework that fully controls the local encoding kernels [3], this precoding-based approach trades off the ultimate achievable throughput with a distributed, implementation-friendly structure that exploits an *algebraic network channel* by a pure random linear NC in the interior of the network. These initial studies show that, under certain network topology and traffic demand, the precoding-based NC can perform as good as a few widely-used LNC solutions. Such results demonstrates a new balance between practicality and throughput enhancement.

However, due to different wireline channel characteristics to that of wireless, our NC understanding is still nascent, especially in a graph-theoretic sense. Notice that many known NC scenarios were characterized graph-theoretically. For example, if there exists only a single session $(s, \{d_i\})$ in the network, the existence of a NC solution is equivalent to that the rate being no larger than the minimum of min-cuts from a source s to each destination d_i. Another example is the 2-unicast with single rates. The existence of an LNC solution is equivalent to the conditions that the some cuts or paths are properly placed in certain ways [24, 29, 30]. Moreover, such graph-theoretic characterizations can be easily checked in polynomial time, which is not the case and intractable for the algebraic conditions as discussed above. Therefore, bridging a straight connection between an algebraic network channel and a graph-theoretic structure will be an influential direction in enlarging our understandings.

We believe that our work of establishing such connection will be a precursor along this leap.

1.3 Wireless Networks - Broadcast Packet Erasure Channel

In Wireless Networks, a packet transmission over a link suffers from a severe channel fading and thus a packet erasure is sometimes inevitable during delivery. Unlike Wireline Networks where an edge can be easily made error-free, an erasure-control mechanism such as Automatic Repeat-reQuest (ARQ) feedback is a common practice in Wireless Networks. We thus assume the casual network-wide channel state information feedback between nodes in the network. This can be accomplished by each node broadcasting its packet reception status (ACK/NACK) over the network via a very low-rate control channel or via piggybacking the forward traffic [42].

What makes the wireless multi-session LNC problem more intriguing is that, in addition to the feedback, we need to jointly consider the transmission orders between nodes as well. Unlike Wireline Networks where the packet transmissions are directive along the deployed links, in Wireless Networks, the transmission signals are dispersed/broadcasted around. Moreover, unlike Wireline Networks where simultaneous reception from different incoming edges can be processed separately, in Wireless Networks, simultaneous receptions are additive and thus may create severe interference from undesired senders such as the Hidden Node problem. As a result, the interference avoidance is a common baseline for most wireless advancements and thus *scheduling between nodes* needs to be jointly considered. Moreover, if there are multiple co-existing flows in a multi-hop network that go in different directions, then each node sometimes has to assume different roles (say, being a sender and/or being a relay) simultaneously. An optimal solution thus needs to balance the roles of each node either through scheduling [35, 43] or through ingenious ways of coding and co-operation [44, 45]. Also see the discussion in [46] for the very detailed case studies for a 3-node network. As a result, the linear NC designer needs to jointly optimize not only "how to mix the available packets for delivery" but also "how to schedule transmissions between nodes", both of which depend on the feedback and the packet erasure events of the wireless channel. Therefore, it becomes even harder to characterize and to design the optimal/near-optimal LNC strategy. Due to the wireless broadcasting nature, such erasure behaviors can be modeled by some probabilistic arguments, termed Broadcast Packet Erasure Channel (PEC). For the following subsections, we will look into some PEC example networks and develop these discussions more deeply.

1.3.1 Linear Network Coding : Illustrative Examples

Fig. 1.2(a) illustrates the 2-user Broadcast PEC scenario where a common node s would like to send different information to d_1 and d_2. If we let n be the total time budget and would like to achieve a specific rate tuple (R_1, R_2), then there are $n(R_1 + R_2)$ packets that need to be delivered over the course of n time slots. For the LNC design of "how to mix the available information", such coding choices can be as many as $q^{n(R_1+R_2)}$ if we use a packet size to be a finite field $\mathbb{F}_q$. Moreover, sending a specific coding choice out of $q^{n(R_1+R_2)}$ is coupled with the feedback and the reception probabilities. Thus, one can see that the characterization problem in Wireless Packet Erasure Channels are more convoluted that that of Wireline Networks. Recently, the LNC capacity region of the 2-user Broadcast PEC was fully characterized and proven that it is indeed the information-theoretic capacity [47]. Moreover, the LNC capacity region for arbitrary K-receiver extension of Fig. 1.2(a) was also fully characterized by the intelligent packet-evolution scheme [48].

Fig. 1.2(b) illustrates the 2-flow 1-hop relay scenario where two sources s_1 and s_2 would like to deliver packets to d_1 and d_2, respectively, via a relay node r. Unlike the previous literature where there is no scheduling consideration between nodes (the single source s is the only transmitting node), here we need to consider transmission orders between s_1, s_2, and r. Namely, the scheduling design is coupled with the

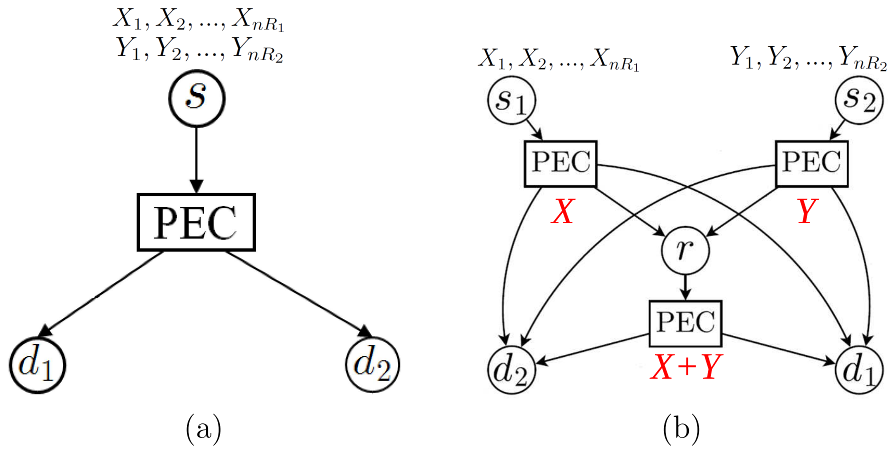


Fig. 1.2. The illustration of the wireless Packet Erasure Channel (PEC) scenarios: (a) 2-user broadcast channel; and (b) 2-flow 1-hop relay channel (Wireless Butterfly).

coding choices of "how to mix the available information", and also with the feedback and the reception probabilities. It is not hard to see the immediate throughput advantage because without Network Coding the relay r would require more time-slots to transmit X and Y to each receiver. However, creating NC opportunities and the use of smart coding choices is correlated to the scheduling decisions, the feedback, and the reception probabilities as explained above. Recently, the LNC capacity region and the achieving scheme of the 2-flow 1-hop relay network was fully characterized even with the direct overhearing between each source-receiver pair [35, 49]. Due to the inherent hardness of the problem, the network capacity understandings are limited to some simpler scenarios, most of which involve only 1-hop transmissions, say broadcast channels or multiple access channels, and/or with all co-existing flows in parallel directions (i.e., flows not forming cycles).

1.3.2 New Space-based Framework

One critical reason for the successful characterization of the simple PEC scenarios is that the network itself admits a strikingly simple solution that achieves the capacity. For example, the capacity-achieving scheme of Fig. 1.2(a) is that the source s

first transmits X and Y packets uncodedly, and then later perform the classic XOR operation of sending a packet mixture $[X + Y]$, for those which X is overheard by d_2 and Y is overheard by d_1. For the case of Fig. 1.2(b), the capacity approaching solution is to similarly take advantage of the classic butterfly-styles operations as much as possible at the relay r. Notice that there is a clear separation between the roles of source, relay, and receiver in these examples. However in real scenarios, there is no such distinct roles and nodes may communicate with each other in an arbitrary way. For such complex network with arbitrary multi-hop traffics in-between, one can imagine that an intelligent but rather simple solution would be extremely hard to find as more coding choices, scheduling decisions, feedback, and reception probabilities are convoluted with each other.

To circumvent this intrinsic hardness, we proposed a novel LNC framework, termed the Space-based Framework [50]. This framework incorporates the joint design of choosing the coding choices and the scheduling decisions into an easily-solvable linear programming (LP) problem. Specifically, the framework enables us to divide the entire set of the LNC choices into some necessary subspaces and formulate the evolution of the rank of each subspace to the scheduling decisions over the course of total time budget n. Once we carefully design the coding spaces to cover the entire LNC operations in a lossless way, then the LP solver directly finds the LNC capacity outer bound. This framework is innovative in a sense that not only it can be applied to arbitrary PEC network, but also the LNC capacity outer bound can be found without the need of finding any cut-condition.[1] This exhaustive search-based approach was previously not possible since there are already too many LNC design choices even in the simpler examples as in Fig. 1.2. Moreover, each variable in the LP formulation is associated to a subset of the entire linear space, i.e., an LNC operation that a sender can perform. Therefore, a careful analysis of the LP structure can lead us to design a simpler but intelligent LNC achievability strategy. Thanks to this framework, the LNC capacity (and even information-theoretic capacity) of many scenarios

[1]The cut-condition is usually for the traditional information-theoretic approach where we first finds a cut and an achievability scheme and later proves that both meet.

and the corresponding achieving schemes have been found [42, 43, 50]. Motivated by this Space-based Framework, this thesis characterizes the capacity and the simple achievability scheme of the larger/complex PEC networks: the 3-node multi-session PEC network with arbitrary traffic directions and the 2-flow smart repeater network.

1.4 Our Contributions

Our contributions consists of three parts. In the first part, this thesis, motivated by the proposed Space-based Framework, characterizes the full Shannon capacity of the 3-node multi-session PEC network with the most general traffic demands, i.e., when three nodes $\{1, 2, 3\}$ are communicating with each other and each node is a source, a relay, and a receiver simultaneously. Namely, there are six private-information flows with rates $(R_{1\to 2}, R_{1\to 3}, R_{2\to 1}, R_{2\to 3}, R_{3\to 1},\ R_{3\to 2})$, respectively, and three common-information flows with rates $(R_{1\to 23}, R_{2\to 31}, R_{3\to 12})$, respectively. We characterize the 9-dimensional Shannon capacity region within a gap that is inversely proportional to the packet size (bits). The gap can be attributed to exchanging reception status (ACK/NACK) and can be further reduced to zero if we allow such feedbacks to be transmitted via a separate control channel. For normal-sized packets, say 12000 bits, our results effectively characterize the capacity region for many important scenarios, e.g., wireless access-point networks with client-to-client cooperative communications, and wireless 2-way relay networks with packet-level coding and processing. Notice that most existing works on packet erasure networks have studied either ≤ 2 co-existing flows [7, 8, 35, 42, 43, 47] or all flows originating from the same node [43, 48, 50–54]. By characterizing the most general 9-dimensional Shannon capacity region with arbitrary flow directions, this work significantly improves our understanding for communications over the 3-node network. Technical contributions of this work also includes a new converse for many-to-many network communications and a new capacity-approaching scheme based on simple LNC operations.

In the second part of contributions, this thesis, motivated by the proposed Space-based Framework, characterizes the LNC capacity region of the 2-flow smart repeater PEC network. Namely, we consider a 4-node 2-hop relay network with one source s, two destinations $\{d_1, d_2\}$, and a common relay r inter-connected by two broadcast PECs. The smart repeater PEC network is a new topology by combining two sources s_1 and s_2 in the 2-flow wireless butterfly PEC network of Fig. 1.2(b). Unlike Fig. 1.2(b) where two separate sources s_1 and s_2 are not coordinating with each other and thus the LNC encoding operation of each source is limited to mixing its own packets at most, our single source s has no limitation for any LNC operation, thereby mixing packets of different sessions freely. As a result, our smart repeater problem is a strict generalization of the 2-flow wireless butterfly problem. In such a setting, we effectively characterize the LNC capacity with a new capacity-approaching scheme that utilizes the newly-identified LNC operations other than the previously known classic butterfly-style operations. Technical contributions of this work also includes a queue-based analysis of our capacity-approaching LNC scheme and the new correctness proof based on the properties of the queue invariance.

In the third part of contributions, this thesis, motivated by its practical advantages over the classic linear NC framework, focuses exclusively on the Precoding-based Framework and characterize its corresponding properties. To that end, we first formulate the Precoding-based Framework that embraces the results of Wireless Interference Channels, and compare it to the classic algebraic framework [3]. We then identify several fundamental properties which allow us to bridge the gap between the network channel gains and the underlying network topology. We then use the newly developed results to analyze the 3-unicast ANA scheme proposed in [40, 41]. Specifically, the existing results [40,41] show that the 3-unicast ANA scheme achieves asymptotically half of the interference-free throughput for each transmission pair when a set of algebraic conditions on the *channel gains* of the networks are satisfied. Note that for the case of Wireless Interference Channels, these algebraic feasibility conditions can be satisfied with close-to-one probability provided the channel gains are continuously distributed random variables [39]. For comparison, the "network channel gains" are usually highly correlated[2] discrete random variables and thus the algebraic channel conditions do not always hold with close-to-one probability. Moreover, except for some very simple networks, checking whether the algebraic channel conditions hold turns out to be computationally prohibitive. As a result, we need new and efficient ways to decide whether the network of interest admits a 3-unicast ANA scheme that achieves half of the interference-free throughput. Motivated by the graph-theoretic characterizations of classic linear NC results, this thesis answers this question by developing new graph-theoretic conditions that characterize the feasibility of the 3-unicast ANA scheme. The proposed graph-theoretic conditions can be easily computed and checked within polynomial time.

1.5 Thesis Outline

In the next chapter, we formulate the wireless multi-session PEC problems of the 3-node network and the smart repeater network, which incorporates the broadcast packet erasure channels with feedback and scheduling decisions all together. In Chapter 3, we describe the 9-dimensional Shannon capacity of the 3-node packet erasure network with a simple capacity-approaching LNC scheme. In Chapter 4, we propose the LNC capacity outer bound of the smart repeater problem based on the Space-based Framework, and provide a close-to-optimal LNC inner bound. In Chapter 5, we formulate the Precoding-based Framework with some necessary graph-theoretic and algebraic definitions. The comparison to the classic algebraic framework [3], and some applications and fundamental properties of the Precoding-based Framework are also discussed. In Chapter 6, we characterize the graph-theoretic feasibility conditions of one application of the Pecoding-based Framework, the 3-unicast Asymptotic Network Alignment (ANA) scheme. In Chapter 7, we conclude this thesis and discuss the possible extensions and future works.

[2]The correlation depends heavily on the underlying network topology.

2. MODEL FORMULATION FOR WIRELESS PACKET ERASURE NETWORKS

In this chapter, we will first mathematically formulate the 1-to-K broadcast packet erasure channel (PEC). Based on the PEC definition, we formulate the problems of the 3-node wireless packet erasure network and the wireless smart repeater packet erasure network, which incorporates the broadcast packet erasure channels with the network-wide feedback, encoding/decoding descriptions, and the scheduling decisions all together. We also define some useful channel probability notations.

2.1 The Broadcast Packet Erasure Channels

For any positive integer K, an 1-to-K broadcast packet erasure channel (PEC) is defined as to take an input X from a finite field $\mathbb{F}_q$ with size $q > 0$ and output a K-dimensional vector $\mathbf{Y} = (Y_1, Y_2, \cdots, Y_K)$. We assume that the input is either received perfectly or completely erased, i.e., each output Y_k must be either the input X or an erasure symbol ε, where $Y_k = \varepsilon$ means that the k-th receiver does not correctly receive the input X. As a result, the reception status can be described by a K-dimensional binary vector $\mathbf{Z} = (Z_1, Z_2, \cdots, Z_K)$ where $Z_k = 1$ and ε represents whether the k-th receiver successfully received the input X or not, respectively. Any given PEC can then be described by its distribution of the binary reception status $\mathbf{Z}$.

2.2 The 3-node Packet Erasure Network

Consider a network of three nearby nodes labeled as $\{1, 2, 3\}$, see Fig. 2.1(a). For the ease of exposition, we will use (i, j, k) to represent one of three cyclically shifted tuples of node indices $\{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$. The 3-node Packet Erasure

(a) 3 nearby nodes (b) The 3-node Packet Erasure Network

Fig. 2.1. Illustrations of the wireless 3-node Packet Erasure Network (PEN). There are nine co-existing flows possible in general.

Network (PEN) is then defined as the collection of three separate 1-to-2 broadcast PECs, each from node i to the other two nodes j and k for all $i \in \{1, 2, 3\}$, see Fig. 2.1(b).

The channel behaviors of the 3-node PEN can be described by the following definitions. For any time slot t, we use a 6-dimensional *channel reception status vector* $\mathbf{Z}(t)$ to represent the reception status of the entire network:

$$\mathbf{Z}(t) = (Z_{1\to2}(t), Z_{1\to3}(t), Z_{2\to1}(t), Z_{2\to3}(t), Z_{3\to1}(t), Z_{3\to2}(t)) \in \{1, \varepsilon\}^6,$$

where $Z_{i\to h}(t) = 1$ and ε represents whether node h can receive the transmission from node i or not, respectively. We assume that the 3-node PEN is memoryless and stationary,[1] i.e., we allow arbitrary joint distribution for the 6 coordinates of $\mathbf{Z}(t)$ but assume that $\mathbf{Z}(t_1)$ and $\mathbf{Z}(t_2)$ are independently and identically distributed for any $t_1 \neq t_2$. We use $p_{i\to jk} \triangleq \mathsf{Prob}(Z_{i\to j}(t) = 1, Z_{i\to k}(t) = 1)$ to denote the probability that the packet transmitted from node i is successfully received by both nodes j and k; and use $p_{i\to j\overline{k}}$ to denote the probability $\mathsf{Prob}(Z_{i\to j}(t) = 1, Z_{i\to k}(t) = \varepsilon)$ that node-i packet is successfully received by node j but not by node k. Probability $p_{i\to\overline{j}k}$ is defined symmetrically. Define $p_{i\to j\vee k} \triangleq p_{i\to\overline{j}k} + p_{i\to jk} + p_{i\to j\overline{k}}$ as the probability that at least one of nodes j and k receives the packet, and define $p_{i\to j} \triangleq p_{i\to jk} + p_{i\to j\overline{k}}$ (resp. $p_{i\to k} \triangleq p_{i\to jk} + p_{i\to\overline{j}k}$) as the marginal reception probability from node i to node j

[1]The 3-node PEN is a special case of the discrete memoryless network channel [44].

(resp. node k). We also assume that the random process $\{\mathbf{Z}(t) : \forall t\}$ is independent of any information messages.

Assume synchronized time-slotted transmissions. To model interference, we assume that only one node can successfully transmit at each time slot $t \in \{1, \cdots, n\}$. If ≥ 2 nodes transmit, then the transmissions of both nodes fail. More specifically, we define the following *scheduling decision* binary variable $\sigma_i(t)$ for any node $i \in \{1, 2, 3\}$. Namely, $\sigma_i(t) = 1$ represents that node i decides to transmit at time t and $\sigma_i(t) = 0$ represents not transmitting. Any transmission is completely destroyed if there are ≥ 2 nodes transmitting simultaneously. For example, suppose node i decides to transmit a packet $X_i(t) \in \mathbb{F}_q$ in time t (thus $\sigma_i(t) = 1$). Then, only when $\sigma_j(t) = \sigma_k(t) = 0$ can node i transmit without any interference. Moreover, only when $Z_{i\to h}(t) = 1$ will node $h \neq i$ receive $Y_{i\to h}(t) = X_i(t)$. In all other cases, node h receives an erasure $Y_{i\to h}(t) = \varepsilon$. To highlight this interference and erasure model, we sometimes write

$$Y_{i\to h}(t) = X_i(t) \circ Z_{i\to h}(t) \circ 1_{\{\sigma_i(t)=1, \sigma_j(t)=\sigma_k(t)=0\}}. \tag{2.1}$$

Over the 3-node PEN described above, we consider the following 9-dimensional traffic flows: 6 private-information flows with rates $(R_{1\to 2}, R_{1\to 3}, R_{2\to 1}, R_{2\to 3}, R_{3\to 1}, R_{3\to 2})$, respectively; and 3 common-information flows with rates $(R_{1\to 23}, R_{2\to 31}, R_{3\to 12})$, respectively. Namely, $R_{1\to 23}$ represents the rate of the common-information message from node 1 to both nodes 2 and 3. We use $\vec{R}_{i*} \triangleq (R_{i\to j},\ R_{i\to k}, R_{i\to jk})$ to denote the rates of all three 3 flows originated from node i, for all $i \in \{1, 2, 3\}$. We use a 9-dimensional rate vector $\vec{R} \triangleq (\vec{R}_{1*}, \vec{R}_{2*}, \vec{R}_{3*})$ to denote the rates of all possible flow directions.

Within a total budget of n time slots, node i would like to send $nR_{i\to h}$ packets (private-information messages), denoted by a row vector $\mathbf{W}_{i\to h}$, to node $h \neq i$, and would like to send $nR_{i\to jk}$ packets (common-information messages), denoted by a row vector $\mathbf{W}_{i\to jk}$, to the other two nodes simultaneously. Each uncoded packet is chosen independently and uniformly randomly from a finite field $\mathbb{F}_q$ with size $q > 0$.

For the ease of exposition, we define $\mathbf{W}_{i*} \triangleq \mathbf{W}_{i\to j} \cup \mathbf{W}_{i\to k} \cup \mathbf{W}_{i\to jk}$ as the collection of all messages originated from node i. Similarly, we define $\mathbf{W}_{*i} \triangleq \mathbf{W}_{j\to i} \cup \mathbf{W}_{j\to ki} \cup \mathbf{W}_{k\to i} \cup \mathbf{W}_{k\to ij}$ as the collection of all messages destined to node i. Sometimes we slightly abuse the above notation and define $\mathbf{W}_{\{i,j\}*} \triangleq \mathbf{W}_{i*} \cup \mathbf{W}_{j*}$ as the collection of messages originated from either node i or node j. Similar "collection-based" notation can also be applied to the received symbols and we can thus define $\mathbf{Y}_{*i}(t) \triangleq \{Y_{j\to i}(t), Y_{k\to i}(t)\}$ and $\mathbf{Y}_{i*}(t) \triangleq \{Y_{i\to j}(t), Y_{i\to k}(t)\}$ as the collection of all symbols received and transmitted by node i during time t, respectively. For simplicity, we also use brackets $[\cdot]_1^t$ to denote the collection from time 1 to t. For example, $[\mathbf{Y}_{*i}, \mathbf{Z}]_1^{t-1}$ is shorthand for the collection $\{Y_{j\to i}(\tau), Y_{k\to i}(\tau), \mathbf{Z}(\tau) : \forall \tau \in \{1, \cdots, t-1\}\}$.

To better understand the problem, we consider one of the following two scenarios.

Scenario 1: Motivated by the throughput benefit of the causal packet ACKnowledgment feedback for erasure networks [20,35,42,43,47–50,53–57], in this scenario we assume that the reception status is casually available to the entire network after each packet transmission through a separate control channel for free. Such assumption can be justified by the fact that the length of ACK/NACK is 1 bit, much smaller than the size of a regular packet.

Scenario 2: In this scenario we assume that there is no inherent feedback mechanism. Any ACK/NACK signal, if there is any, has to be sent through the regular forward channels along with information messages. As a result, any achievability scheme needs to balance the amount of information and control messages. For example, suppose a particular coding scheme chooses to divide the transmitted packet X into the header and the payload. Then it needs to carefully decide what the content of the control information would be and how many bits the header should have to accommodate the control information. The timeliness of delivering the control messages is also critical since the control information, sent through the forward erasure channel, may get lost as well. Therefore, the necessary control information may not arrive in time. Such a setting in Scenario 2 is much closer to practice as it considers the complexity/delay overhead of the coding solution. In Scenario 2, we also assume that the 3-node PEN is *fully-connected*, i.e, node i can always reach node j, possibly with the help of the third node k, for any $i \neq j$ pairs. The formal definition of fully-connectedness is provided in Definition 3.2.1. Note that *the fully-connectedness is assumed only in Scenario 2*. When the casual reception status is available for free (Scenario 1), our results do not need the fully-connectedness assumption.

In sum, the causal ACK/NACK feedback can be transmitted for free in Scenario 1 but has to go through the forward channel when in Scenario 2. For the following, we first focus on the detailed formulation under Scenario 2.

Given the rate vector $\vec{R}$, a joint scheduling and network coding scheme is described by $3n$ binary scheduling functions: $\forall\, t \in \{1, \cdots, n\}$ and $\forall\, i \in \{1, 2, 3\}$,

$$\sigma_i(t) = f^{(t)}_{\mathsf{SCH},\,i}([\mathbf{Y}_{*i}]_1^{t-1}) \tag{2.2}$$

plus $3n$ encoding functions: $\forall\, t \in \{1, \cdots, n\}$ and $\forall\, i \in \{1, 2, 3\}$,

$$X_i(t) = f^{(t)}_i(\mathbf{W}_{i*}, [\mathbf{Y}_{*i}]_1^{t-1}), \tag{2.3}$$

plus 3 decoding functions: $\forall\, i \in \{1, 2, 3\}$,

$$\hat{\mathbf{W}}_{*i} = g_i(\mathbf{W}_{i*}, [\mathbf{Y}_{*i}]_1^{n}). \tag{2.4}$$

To refrain from using the timing-channel[2] techniques [58], we also require the following equality

$$I([\sigma_1, \sigma_2, \sigma_3]_1^n \,;\, \mathbf{W}_{\{1,2,3\}*}) = 0, \tag{2.5}$$

where $I(\cdot\,;\,\cdot)$ is the mutual information and $\mathbf{W}_{\{1,2,3\}*} \triangleq \mathbf{W}_{1*} \cup \mathbf{W}_{2*} \cup \mathbf{W}_{3*}$ is all the 9-flow information messages as defined earlier.

Intuitively, at every time t, each node decides whether to transmit or not based on what it has received in the past, see (2.2). Note that the received symbols $[\mathbf{Y}_{*i}]_1^{t-1}$

[2]We believe that the use of timing channel techniques will not alter the capacity region much when the packet size is large. One justification is that the rate of the timing channel is at most 3 bits per slot, which is negligible compared to a normal packet size of 12000 bits.

may contain both the message information and the control information. (2.5) ensures that the "timing" of the transmission $\sigma_i(t)$ cannot be used to carry[3] the message information. Once each node decides whether to transmit or not,[4] it encodes $X_i(t)$ based on its information messages and what it has received from other nodes in the past, see (2.3). In the end of time n, each node decodes its desired packets based on its information messages and what it has received, see (2.4).

We can now define the capacity region.

Definition 2.2.1. *Fix the distribution of* $\mathbf{Z}(t)$ *and finite field* $\mathbb{F}_q$. *A 9-dimensional rate vector* $\vec{R}$ *is achievable if for any* $\epsilon > 0$ *there exists a joint scheduling and network code scheme with sufficiently large* n *such that* $\mathsf{Prob}(\hat{\mathbf{W}}_{*i} \neq \mathbf{W}_{*i}) < \epsilon$ *for all* $i \in \{1, 2, 3\}$. *The capacity region is the closure of all achievable* $\vec{R}$.

2.2.1 Comparison between Scenarios 1 and 2

The previous formulation focuses on Scenario 2. The difference between Scenarios 1 and 2 is that the former allows the use of causal ACK/NACK feedbacks for free. As a result, for Scenario 1, we simply need to insert the *causal* network-wide channel status information $[\mathbf{Z}]_1^{t-1}$ in the input arguments of (2.2) and (2.3), respectively; and insert the *overall* network-wide channels status information $[\mathbf{Z}]_1^n$ in the input argument of (2.4). The formulation of Scenario 1 thus becomes as follows: $\forall\, t \in \{1, \cdots, n\}$ and $\forall\, i \in \{1, 2, 3\}$,

$$\sigma_i(t) = \overline{f}^{(t)}_{\mathsf{SCH},\,i}([\mathbf{Y}_{*i}, \mathbf{Z}]_1^{t-1}), \tag{2.6}$$

$$X_i(t) = \overline{f}^{(t)}_i(\mathbf{W}_{i*}, [\mathbf{Y}_{*i}, \mathbf{Z}]_1^{t-1}), \tag{2.7}$$

$$\hat{\mathbf{W}}_{*i} = \overline{g}_i(\mathbf{W}_{i*}, [\mathbf{Y}_{*i}, \mathbf{Z}]_1^{n}), \tag{2.8}$$

[3]For example, one (not necessarily optimal) way to encode is to divide a packet $X_i(t)$ into the header and the payload. The messages $\mathbf{W}_{i*}$ will be embedded in the payload while the header contains control information such as ACK. If this is indeed the way we encode, then (2.5) requires that transmit decision depend only on the control information in the header, not the messages in the payload.

[4]If two nodes i and j decide to transmit simultaneously, then our channel model (2.1) automatically leads to full collision and erases both transmissions.

while we still impose no-timing channel information (2.5). Obviously, with more information to use, the capacity region under Scenario 1 is a superset of that of Scenario 2, which is why we use overlines in the above function descriptions. Following this observation, we will outer bound the (larger) capacity of Scenario 1 and inner bound the (smaller) capacity of Scenario 2 in the subsequent sections.

Without loss of generality, we can further replace the distributed scheduling computation in (2.6) (each node i computes its own scheduling) by the following centralized scheduling function

$$\sigma(t) = \overline{f}^{(t)}_{\mathsf{SCH}}([\mathbf{Z}]_1^{t-1}) \in \{1, 2, 3\}, \tag{2.9}$$

that takes the values in the set of three nodes $\{1, 2, 3\}$. That is, $\sigma(t) = i$ implies that only node i is scheduled to transmit in time t.

To prove why we can replace (2.6) by (2.9) without loss of generality, we first introduce the following lemma.

Lemma 2.2.1. *Without loss of generality, we can replace (2.6) by the following form:*

$$\sigma_i(t) = \overline{f}^{(t)}_{\mathsf{SCH},\,i}([\mathbf{Z}]_1^{t-1}), \tag{2.10}$$

which is still a binary scheduling function but the input argument $[\mathbf{Y}_{*i}]_1^{t-1}$ *in (2.6) is removed.*

The proof of Lemma 2.2.1 is relegated to Appendix F. The intuition behind the proof is to show that since the information equality (2.5) must hold, knowing the past reception status $[\mathbf{Z}]_1^{t-1}$ is sufficient for the scheduling purpose.

Lemma 2.2.1 ensures that we can replace the scheduling decision (2.6) of each individual node i by (2.10). We then observe that every node i makes its scheduling decision based on the same input argument $[\mathbf{Z}]_1^{t-1}$, which, in Scenario 1, is available to all three nodes for free via a separate control channel. Therefore, it is as if there is a centralized scheduler in Scenario 1 and the centralized scheduler will never induce

(a) 4-node 2-hop relay network (b) The BPEC network model

Fig. 2.2. The 2-flow wireless Smart Repeater network

any scheduling conflict. As a result, we can further replace the individual scheduler (2.10) by a centralized global scheduling function (2.9) where $\sigma(t) = i$ implies that node i is the only scheduled node in time t.

In sum, under Scenario 1, the joint network coding and scheduling solution is described by (2.7), (2.8), and (2.9). Here we do not impose (2.5) anymore since the centralized scheduler (2.9) satisfies (2.5) naturally.

2.3 The Smart Repeater Packet Erasure Network

The 2-flow wireless smart repeater network with broadcast PECs, see Fig. 2.2(b), can be modeled as follows. Consider two traffic rates (R_1, R_2) and assume slotted transmissions. Within a total budget of n time slots, source s would like to send nR_k packets, denoted by a row vector $\mathbf{W}_k$, to destination d_k for all $k \in \{1, 2\}$ with the help of relay r. Each packet is chosen uniformly randomly from a finite field $\mathbb{F}_q$ with size $q > 0$. To that end, we denote $\mathbf{W} \triangleq (\mathbf{W}_1, \mathbf{W}_2)$ as an nR_Σ-dimensional row vector of all the packets, and define the linear space $\Omega \triangleq (\mathbb{F}_q)^{nR_\Sigma}$ as the *overall message/coding space.*

To represent the reception status, for any time slot $t \in \{1, \cdots, n\}$, we define two *channel reception status vectors*:

$$\mathbf{Z}_s(t) = (Z_{s\to d_1}(t), Z_{s\to d_2}(t), Z_{s\to r}(t)) \in \{1, *\}^3,$$

$$\mathbf{Z}_r(t) = (Z_{r\to d_1}(t), Z_{r\to d_2}(t)) \in \{1, *\}^2,$$

where "1" and "$*$" represent successful reception and erasure, respectively. For example, $Z_{s\to d_1}(t) = 1$ and $*$ represents whether d_1 can receive the transmission from source s or not at time slot t. We then use $\mathbf{Z}(t) \triangleq (\mathbf{Z}_s(t), \mathbf{Z}_r(t))$ to describe the 5-dimensional channel reception status vector of the entire network. We also assume that $\mathbf{Z}(t)$ is memoryless and stationary, i.e., $\mathbf{Z}(t)$ is independently and identically distributed over the time axis t.

We assume that either source s or relay r can transmit at each time slot, and express the *scheduling decision* by $\sigma(t) \in \{s, r\}$. For example, if $\sigma(t) = s$, then source s transmits a packet $X_s(t) \in \mathbb{F}_q$; and only when $Z_{s\to h}(t) = 1$, node h (one of $\{d_1, d_2, r\}$) will receive $Y_{s\to h}(t) = X_s(t)$. In all other cases, node h receives an erasure $Y_{s\to h}(t) = *$. The reception $Y_{r\to h}(t)$ of relay r's transmission is defined similarly.

Assuming that the 5-bit $\mathbf{Z}(t)$ vector is broadcast to both s and r after each packet transmission through a separate control channel, a *linear network code* contains n scheduling functions

$$\forall\, t \in \{1, \cdots, n\}, \quad \sigma(t) = f_{\sigma,t}([\mathbf{Z}]_1^{t-1}), \tag{2.11}$$

where we use brackets $[\,\cdot\,]_1^\tau$ to denote the collection from time 1 to τ. Namely, at every time t, scheduling is decided based on the network-wide channel state information (CSI) up to time $(t-1)$. If source s is scheduled, then it can send a linear combination of any packets. That is,

$$\text{If } \sigma(t) = s, \text{ then } X_s(t) = \mathbf{c}_t \mathbf{W}^\top \text{ for some } \mathbf{c}_t \in \Omega, \tag{2.12}$$

where $\mathbf{c}_t$ is a row coding vector in Ω. The choice of $\mathbf{c}_t$ depends on the past CSI vectors $[\mathbf{Z}]_1^{t-1}$, and we assume that $\mathbf{c}_t$ is known causally to the entire network.[5] Therefore, decoding can be performed by simple Gaussian elimination.

[5]Coding vector $\mathbf{c}_t$ can either be appended in the header or be computed by the network-wide causal CSI feedback $[\mathbf{Z}]_1^{t-1}$.

We now define two important linear space concepts: The *individual message subspace* and the *knowledge subspace.* To that end, we first define $\mathbf{e}_l$ as an nR_Σ-dimensional elementary row vector with its l-th coordinate being one and all the other coordinates being zero. Recall that the nR_Σ coordinates of a vector in Ω can be divided into 2 consecutive "intervals", each of them corresponds to the information packets $\mathbf{W}_k$ for each flow from source to destination d_k. We then define the *individual message subspace* Ω_k:

$$\Omega_k \triangleq \mathsf{span}\{\mathbf{e}_l : l \in \text{"interval" associated to } \mathbf{W}_k\}, \tag{2.13}$$

That is, Ω_k is a linear subspace corresponding to any linear combination of $\mathbf{W}_k$ packets. By (2.13), each Ω_k is a linear subspace of the overall message space Ω and $\mathsf{rank}(\Omega_k) = nR_k$.

We now define the knowledge space for $\{d_1, d_2, r\}$. To that end, we first define the *reception subspace* in the end of time t by

$$\begin{aligned} RS_h(t) \triangleq \mathsf{span}\{\mathbf{c}_\tau : \forall \tau \leq t \text{ such that node } h \text{ receives the linear} \\ \text{combination } (\mathbf{c}_\tau \cdot \mathbf{W}^\top) \text{ successfully in time } \tau\} \end{aligned} \tag{2.14}$$

where $h \in \{d_1, d_2, r\}$. For example, $RS_r(t)$ is the linear space spanned by the packets successfully delivered from source to relay up to time t. $RS_{d_1}(t)$ is the linear space spanned by the packets received at destination d_1 up to time t, either transmitted by source or by relay. The *knowledge space*[6] $S_h(t)$ for $h \in \{d_1, d_2, r\}$ can be simply defined as

$$S_h(t) \triangleq RS_h(t). \tag{2.15}$$

For shorthand, we use $S_1(t)$ and $S_2(t)$ instead of $S_{d_1}(t)$ and $S_{d_2}(t)$, respectively. Then, by the above definitions, we quickly have that destination d_k can decode the desired

[6]The knowledge space $S_h(t)$ is a superordinate concept that contains not only the reception subspace $RS_h(t)$ but also the messages originated from node h, if any. In our problem of interest, the messages are originated only from source and thus its meaning is identical to the reception subspace as (2.15).

packets $\mathbf{W}_k$ as long as $S_k(n) \supseteq \Omega_k$. That is, when the knowledge space in the end of time n contains the desired message space.

With the above linear space concepts, we now can describe the packet transmission from relay. Recall that, unlike the source where the packets are originated, relay can only send a linear mixture of *the packets that it has known.* Therefore, the encoder description from relay can be expressed by

$$\text{If } \sigma(t) = r, \text{ then } X_r(t) = \mathbf{c}_t \mathbf{W}^\top \text{ for some } \mathbf{c}_t \in S_r(t-1). \tag{2.16}$$

For comparison, in (2.12), the source s chooses $\mathbf{c}_t$ from Ω. We can now define the LNC capacity region.

Definition 2.3.1. *Fix the distribution of $\mathbf{Z}(t)$ and finite field $\mathbb{F}_q$. A rate vector (R_1, R_2) is achievable by LNC if for any $\epsilon > 0$ there exists a joint scheduling and LNC scheme with sufficiently large n such that $\mathsf{Prob}(S_k(n) \supseteq \Omega_k) > 1 - \epsilon$ for all $k \in \{1, 2\}$. The LNC capacity region is the closure of all LNC-achievable (R_1, R_2).*

2.3.1 A Useful Notation

In the smart repeater network model, there are two broadcast PECs associated with s and r, respectively. For shorthand, we call those PECs the s-PEC and the r-PEC, respectively.

The distribution of the network-wide channel status vector $\mathbf{Z}(t) = (\mathbf{Z}_s(t), \mathbf{Z}_r(t))$ can be described by the probabilities $p_{s \to T\overline{\{d_1,d_2,r\}\backslash T}}$ for all $T \subseteq \{d_1, d_2, r\}$, and $p_{r \to U\overline{\{d_1,d_2\}\backslash U}}$ for all $U \subseteq \{d_1, d_2\}$. In total, there are $8 + 4 = 12$ channel parameters.[7]

For notational simplicity, we also define the following two probability functions $p_s(\cdot)$ and $p_r(\cdot)$, one for each PEC. The input argument of p_s is a collection of the

[7] By allowing some of the coordinates of $\mathbf{Z}(t)$ to be correlated (i.e., spatially correlated as the correlation is between coordinates, not over the time axis), our setting can also model the scenario in which destinations d_1 and d_2 are situated in the same physical node and thus have perfectly correlated channel success events.

elements in $\{d_1, d_2, r, \overline{d_1}, \overline{d_2}, \overline{r}\}$. The function $p_s(\cdot)$ outputs the probability that the reception event is compatible to the specified collection of $\{d_1, d_2, r, \overline{d_1}, \overline{d_2}, \overline{r}\}$. For example,

$$p_s(d_2\overline{r}) = p_{s\to\overline{d_1}d_2\overline{r}} + p_{s\to d_1d_2\overline{r}} \tag{2.17}$$

is the probability that the input of the source-PEC is successfully received by d_2 but not by r. Herein, d_1 is a dont-care receiver and $p_s(d_2\overline{r})$ thus sums two joint probabilities together (d_1 receives it or not) as described in (2.17). Another example is $p_r(d_2) = p_{r\to d_1d_2} + p_{r\to\overline{d_1}d_2}$, which is the probability that a packet sent by r is heard by d_2. To slightly abuse the notation, we further allow $p_s(\cdot)$ to take multiple input arguments separated by commas. With this new notation, $p_s(\cdot)$ then represents the probability that the reception event is compatible to at least one of the input arguments. For example,

$$\begin{aligned} p_s(d_1\overline{d_2}, r) = {} & p_{s\to d_1\overline{d_2r}} + p_{s\to d_1\overline{d_2}r} + p_{s\to d_1d_2r} \\ & + p_{s\to\overline{d_1}d_2r} + p_{s\to\overline{d_1d_2}r} \end{aligned}$$

That is, $p_s(d_1\overline{d_2}, r)$ represents the probability that $(Z_{s\to d_1}, Z_{s\to d_2}, Z_{s\to r})$ equals one of the following 5 vectors $(1, *, *)$, $(1, *, 1)$, $(1, 1, 1)$, $(*, 1, 1)$, and $(*, *, 1)$. Note that these 5 vectors are compatible to either $d_1\overline{d_2}$ or r or both. Another example of this $p_s(\cdot)$ notation is $p_s(d_1, d_2, r)$, which represents the probability that a packet sent by s is received by at least one of the three nodes d_1, d_2, and r.

2.4 Chapter Summary

In this chapter, we formulate the model of the 1-to-K broadcast packet erasure channel in Section 2.1. In Section 2.2, we construct a wireless 3-node network model including the encoding/decoding and scheduling descriptions, and the broadcast packet erasure channels with feedback. The corresponding Shannon capacity region is also defined in Section 2.2. Based on the feedback mechanism, two scenarios are considered, and their comparison is described in Section 2.2.1. In Section 2.3, we also construct a wireless 2-flow smart repeater network model including the LNC encoding/decoding and scheduling descriptions, and the broadcast packet erasure channels with feedback. The corresponding LNC capacity region is also defined in Section 2.3. A useful probability notations for the broadcast packet erasure channels are defined in Section 2.3.1.

3. ACHIEVING THE SHANNON CAPACITY OF THE 3-NODE PACKET ERASURE NETWORK

In Section 2.2, we formulated the problem of the wireless 3-node packet erasure network (PEN) with feedback, encoding/decoding descriptions, and scheduling decisions between the three nodes $\{1,2,3\}$. In this chapter, we propose the corresponding outer and inner bound. To that end, we will first provide the information-theoretic capacity outer bound of the 3-node PEN based upon Scenario 1. We then propose the capacity-achieving LNC scheme in Scenario 1 and the similar capacity-approaching inner bound in Scenario 2. In Scenario 1, both outer and inner bound will be further proven to be matched. Since both bounds are sufficient to describe the capacity, the LNC outer bound description based on the Space-based Framework will be relegated to Appendix A. The full details and arguments of the Space-based Framework can be found in [59]. Finally, we will discuss some related works as special examples and demonstrate the numerical results including the capacity region comparison.

3.1 The Shannon Capacity Outer Bound

Proposition 3.1.1. *For any fixed* $\mathbb{F}_q$*, a* 9*-dimensional* $\vec{R}$ *is achievable under*[1] *Scenario 1 only if there exist* 3 *non-negative variables* $s^{(i)}$ *for all* $i \in \{1,2,3\}$ *such that jointly they satisfy the following three groups of linear conditions:*

- Group 1, termed the *time-sharing condition*, has 1 inequality:

$$\sum_{\forall\, i\in\{1,2,3\}} s^{(i)} \leq 1. \tag{3.1}$$

[1]Proposition 3.1.1 is naturally an outer bound for Scenario 2, see Section 2.2.1.

• Group 2, termed the *broadcast cut-set condition*, has 3 inequalities: For all $i \in \{1,2,3\}$,

$$R_{i\to j} + R_{i\to k} + R_{i\to jk} \leq s^{(i)} \cdot p_{i\to j\vee k}. \tag{3.2}$$

• Group 3, termed the 3-*way multiple-access cut-set condition*, has 3 inequalities: For all $i \in \{1,2,3\}$,

$$\begin{aligned} R_{j\to i} + R_{j\to ki} + R_{k\to i} + R_{k\to ij} \leq s^{(j)} \cdot p_{j\to i} + s^{(k)} \cdot p_{k\to i} \\ - \left(\frac{p_{j\to i}}{p_{j\to k\vee i}} R_{j\to k} + \frac{p_{k\to i}}{p_{k\to i\vee j}} R_{k\to j} \right). \end{aligned}$$

Proposition 3.1.1 considers arbitrary, possibly non-linear ways of designing the encoding/decoding and scheduling functions in (2.7), (2.8), and (2.9), and is derived by entropy-based analysis. Proposition 3.1.1 can also be viewed as strict generalization of the results of the simpler settings [48, 53].

The brief intuitions behind (3.1) to (3.3) are as follows. Each variable $s^{(i)}$ counts the expected frequency (normalized over the time budget n) that node i is scheduled for successful transmissions. As a result, (3.1) holds naturally. (3.2) is a simple cut-set condition for broadcasting from node i. One main contribution of this work is the derivation of the new 3-way multiple-access outer bound in (3.3). The LHS of (3.3) contains all the information destined for node i. The term $s^{(j)}p_{j\to i} + s^{(k)}p_{k\to i}$ on the RHS of (3.3) is the amount of time slots that either node j or node k can communicate with node i. As a result, it resembles a multiple-access cut condition of a typical cut-set argument [60, Section 15.10]. What is special in our setting is that, since node j may have some private-information for node k and vice versa, sending those private-information has a penalty on the multiple access channel from nodes $\{j,k\}$ to node i. The last term on the RHS of (3.3) quantifies such penalty that is inevitable regardless of what kind of coding schemes being used. The proof of Proposition 3.1.1 and the detailed discussions are relegated to Section 3.4.

Remark: In addition to having a new penalty term on the RHS of (3.3), the 3-way multiple-access cut-set condition (3.3) is surprising, not because that it upper bounds the *combined information-flow rate* from nodes $\{j, k\}$ entering node i but because that, unlike the traditional multiple-access upper bounds, we do not need to upper bound the individual rate from node j (resp. k) to node i.

More specifically, a traditional multi-access channel capacity result will also upper bound the rate $R_{j\to i} + R_{j\to ki}$ by considering the cut from node j to node i (ignoring node k completely). If we follow the above logic and write down naively the "cut condition" from node j to i, then we will have

$$R_{j\to i} + R_{j\to ki} \leq s^{(j)} \cdot p_{j\to i} - \frac{p_{j\to i}}{p_{j\to k\vee i}} R_{j\to k}. \tag{3.3}$$

where $R_{j\to i} + R_{j\to ki}$ is the rate from nodes j to i, $s^{(j)} \cdot p_{j\to i}$ is the successful time slots, and $\frac{p_{j\to i}}{p_{j\to k\vee i}} R_{j\to k}$ is the penalty term. One might expect that (3.3) is also a legitimate outer bound if the naive cut condition arguments hold. It turns out that (3.3) is not an outer bound and one can find some LNC solution that contradicts (3.3).

The reason why (3.3) is false is as follows. The $\mathbf{W}_{j\to i}$ packets may not necessarily go directly from node j to node i and it is possible that node k can also help relay those packets. As a result, how frequently node k is scheduled can also affect the number of $\mathbf{W}_{j\to i}$ packets that one can hope to deliver from node j to node i. Since (3.3) does not involve $s^{(k)}$, it does not consider the possibility of node k relaying the packets for node j. In contrast, our outer bound (3.3) indeed captures such a subtle but critical phenomenon by grouping all $R_{j\to i}$, $R_{k\to i}$, $R_{j\to ki}$, $R_{k\to ij}$, $R_{j\to k}$, and $R_{k\to j}$ as a whole and upper bounds it with the (weighted) sum of scheduling frequencies of nodes j and k.

3.2 A LNC Capacity Achieving Scheme

Scenario 2 requires the network to be fully-connected, which is defined as follows.

Definition 3.2.1. *In Scenario 2, we assume the* 3*-node PEN is* fully-connected *in the sense that the given channel reception probabilities satisfy either* $p_{i_1 \to i_2} > 0$ *or* $\min(\, p_{i_1 \to i_3}, p_{i_3 \to i_2}) > 0$ *for all distinct* $i_1, i_2, i_3 \in \{1, 2, 3\}$.

Namely, node i_1 must be able to communicate with node i_2 either through the direct communication (i.e., $p_{i_1 \to i_2} > 0$) or through relaying (i.e., $\min(p_{i_1 \to i_3}, p_{i_3 \to i_2}) > 0$). Note that in Scenario 2, the control messages has to be sent through the regular forward channel as well. The fully-connectedness assumption guarantees that feedback/control information can be sent successfully from one node to any other node, either directly or through the help of another node.

We also need the following new math operator.

Definition 3.2.2. *For any* 2 *non-negative values* a *and* b*, the operator* $\mathsf{nzmin}\{a, b\}$*, standing for non-zero minimum, is defined as:*

$$\mathsf{nzmin}\{a, b\} = \begin{cases} \max(a, b) & \text{if } \min(a, b) = 0, \\ \min(a, b) & \text{if } \min(a, b) \neq 0. \end{cases}$$

Intuitively, $\mathsf{nzmin}\{a, b\}$ *is the minimum of the strictly positive entries.*

Proposition 3.2.1. *For any fixed* $\mathbb{F}_q$*, a* 9*-dimensional* $\vec{R}$ *is LNC-achievable in Scenario 2 if there exist* 15 *non-negative variables* $t^{(i)}_{[u]}$ *and* $\{t^{(i)}_{[c,\, l]}\}_{l=1}^{4}$ *for all* $i \in \{1, 2, 3\}$ *such that jointly they satisfy the following three groups of linear conditions:*

- Group 1, termed the *time-sharing condition*, has 1 inequality:

$$\sum_{\forall\, i \in \{1,2,3\}} t^{(i)}_{[\mathrm{u}]} + t^{(i)}_{[\mathrm{c},\,1]} + t^{(i)}_{[\mathrm{c},\,2]} + t^{(i)}_{[\mathrm{c},\,3]} + t^{(i)}_{[\mathrm{c},\,4]} \;\leq\; 1 - t_{\mathsf{FB}}, \tag{3.4}$$

where t_{FB} is a constant defined as

$$t_{\mathsf{FB}} \triangleq \sum_{\forall\, i \in \{1,2,3\}} \frac{3}{\log_2(q) \cdot \mathsf{nzmin}\{p_{i \to j}, p_{i \to k}\}}. \tag{3.5}$$

- Group 2 has 3 inequalities: For all $(i, j, k) \in \{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$,

$$R_{i\to j} + R_{i\to k} + R_{i\to jk} < t^{(i)}_{[\mathrm{u}]} \cdot p_{i\to j\vee k}. \tag{3.6}$$

- Group 3 has 6 inequalities: For all $(i, j, k) \in \{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$,

$$\Big(R_{i\to j} + R_{i\to jk}\Big)\frac{p_{i\to \overline{j}k}}{p_{i\to j\vee k}} < \Big(t^{(i)}_{[\mathrm{c},1]} + t^{(i)}_{[\mathrm{c},3]}\Big)\cdot p_{i\to j} + \Big(t^{(k)}_{[\mathrm{c},2]} + t^{(k)}_{[\mathrm{c},3]}\Big)\cdot p_{k\to j}, \tag{3.7}$$

$$\Big(R_{i\to k} + R_{i\to jk}\Big)\frac{p_{i\to j\overline{k}}}{p_{i\to j\vee k}} < \Big(t^{(i)}_{[\mathrm{c},1]} + t^{(i)}_{[\mathrm{c},4]}\Big)\cdot p_{i\to k} + \Big(t^{(j)}_{[\mathrm{c},2]} + t^{(j)}_{[\mathrm{c},4]}\Big)\cdot p_{j\to k}. \tag{3.8}$$

Proposition 3.2.2. *Continue from Proposition 3.2.1, if we focus on Scenario 1 instead, then the rate vector $\vec{R}$ is LNC-achievable if there exist 15 non-negative variables $t^{(i)}_{[u]}$ and $\{t^{(i)}_{[c,l]}\}^{4}_{l=1}$ for all $i \in \{1, 2, 3\}$ such that (3.4), (3.6) to (3.8) hold while we set $t_{\mathsf{FB}} = 0$ in (3.5).*

In short, the constant term t_{FB} in (3.5) quantifies the overhead of sending the ACK/NACK feedbacks through the forward erasure channel in Scenario 2 and can be set to 0 in Scenario 1.

The sketch of the proof for Proposition 3.2.2 (Scenario 1) is provided in Section 3.5 while the detailed construction for Proposition 3.2.1 (Scenario 2) is relegated to Appendix B.

Since both the outer bound and the achievable regions can be computed by an LP solver, one can numerically verify that for all possible channel parameters, the rate regions of Propositions 3.1.1 and 3.2.2 of Scenario 1 always match. We can actually prove this observation by analyzing the underlying linear algebraic structures of the two LP problems.

Proposition 3.2.3. *The outer bound in Proposition 3.1.1 and the closure of the achievable region in Proposition 3.2.2 match for all possible channel parameters $\{p_{i\to jk}, p_{i\to \overline{j}k}, p_{i\to j\overline{k}} : \forall (i, j, k)\}$. They thus describe the corresponding 9-dimensional Shannon capacity region under Scenario 1.*

The proof of Proposition 3.2.3 is relegated to Appendix C.

From the above discussions, one can see that even for the more practical Scenario 2, in which there is no dedicated feedback control channels, Proposition 3.2.1 is indeed capacity-approaching when the 3-node PEN is fully-connected. The gap to the outer bound is inversely proportional to $\log_2(q)$ and diminishes to zero if the packet size $\log_2(q)$ (bits) is large enough. In real life, the actual payload of each packet is roughly 10^4 bits and the gap is thus negligible unless the reception probabilities $p_{i\to j}$ or $p_{i\to k}$ is extremely small.

3.3 Comments On The Fully-Connected Assumption

We first consider Scenario 1, which does not require the fully-connected assumption. It is possible that in Scenario 1, we have $p_{i\to j\vee k} = 0$ for some (i, j, k), which implies that (3.7) and (3.8) being undefined. However, when $p_{i\to j\vee k} = 0$, it is simply impossible to send any messages out of node i. As a result, we can replace the (undefined) (3.7) and (3.8) by a hard condition $R_{i\to j} = R_{i\to k} = R_{i\to jk} = 0$. Proposition 3.2.3 still holds after such a simple revision.

We now consider Scenario 2. We note that Proposition 3.2.1 holds only when the network is fully-connected. Actually, when the network is not fully-connected, the denominator of (3.5) may be zero and (3.5) becomes undefined. When the network is *not* fully-connected, it is an interesting open problem what the actual capacity region is going to be. Specifically, the outer bound (Proposition 3.1.1) still holds even when the network is not fully-connected. However, there are reasons to believe that the outer bound is not tight anymore. For example, suppose $p_{2\to 3\vee 1} = 0$, i.e., the PEC from node 2 is completely erasure, there is no dedicated control channel, and any feedback has to be sent through the forward channel, i.e., Scenario 2 but being not fully-connected. In this example, node 2 is completely "in the dark". Note that being in the dark does not mean that we cannot send messages to node 2. For example, we can use an MDS code to send messages from nodes 1 to node 2. When the MDS code rate is slightly lower than the success probability $p_{1\to 2}$, then node 2 can receive the correct messages with high probability without sending any ACK. However, when node 2 is in the dark, neither node 1 nor node 3 can be made aware of the reception status of node 2. Therefore, the classic network coding techniques in [48] do not apply in this scenario. How to characterize the Shannon capacity region when some node is in the dark is beyond the scope of this work and will be actively investigated in the future.

Remark: The above "asymmetric" feedback scenario is theoretically interesting. In practice, the PEC is usually used to model network communications, for which ACK is often required for any transmission and also necessary for the purpose of channel estimation. Therefore, if $p_{2\to 3\vee 1} = 0$ and node 2 is in the dark, then nodes 1 and 3 will give up communicating to node 2 immediately due to the lack of any ACK feedback. The aforementioned MDS code approach will not be used when node 2 cannot acknowledge the transmission in any way.

3.4 Sketch of The Proof of The Shannon Outer Bound

We now provide the sketch of the proof of Proposition 3.1.1. Given any reception probabilities and any $\epsilon > 0$, consider a joint network coding and scheduling scheme (2.7), (2.8), and (2.9) that can send 9 flows with rates $\vec{R}$ in n time slots with the overall error probability no larger than ϵ. Based on the given scheme, define $s^{(i)}$ as the normalized expected number of time slots for which node i is scheduled. That is,

$$s^{(i)} \triangleq \frac{1}{n}\mathbb{E}\left\{\sum_{t=1}^{n} 1_{\{\sigma(t)=i\}}\right\}, \tag{3.9}$$

where $1_{\{\cdot\}}$ is the indicator function. By the above definition, the computed scheduling frequencies $\{s^{(1)}, s^{(2)}, s^{(3)}\}$ must satisfy the time-sharing condition (3.1).

We will now prove (3.2) and (3.3) of Proposition 3.1.1, respectively. To that end, we assume that the logarithm of the mutual information and the entropy is of base

q, the order of the underlying finite field $\mathbb{F}_q$. For the case when the logarithm of the entropy is base-2, we will distinguish it by using $H_2(\cdot)$.

The inequality (3.2) can be proven by proving the following two inequalities separately:

$$\begin{aligned} & I(\mathbf{W}_{i*}\,;\,[\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^n \,|\, \mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^n) \\ & \quad \geq n\left(R_{i\to j}+R_{i\to k}+R_{i\to jk}-2\epsilon-\frac{H_2(2\epsilon)}{n\log_2 q}\right), \quad (3.2\text{A}) \\ & I(\mathbf{W}_{i*}\,;\,[\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^n \,|\, \mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^n) \leq ns^{(i)}p_{i\to j\vee k}. \quad (3.2\text{B}) \end{aligned}$$

Intuitively, (3.2A) follows from the Fano's inequality and (3.2B) follows from a simple cut condition. By choosing $\epsilon \to 0$, we have proven (3.2). The detailed derivation of (3.2A) and (3.2B) are relegated to Appendix D.

We now prove (3.3) by proving the following two inequalities:

$$\begin{aligned} & I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*},[\mathbf{Z}]_1^n) \\ & \quad \geq n\Bigg(R_{j\to i}+R_{k\to i}+R_{j\to ki}+R_{k\to ij}+\frac{p_{j\to i}}{p_{j\to k\vee i}}R_{j\to k} \\ & \qquad +\frac{p_{k\to i}}{p_{k\to i\vee j}}R_{k\to j}-6\epsilon-\frac{3H_2(\epsilon)}{n\log_2 q}\Bigg), \quad (3.3\text{A}) \\ & I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*},[\mathbf{Z}]_1^n) \leq n(s^{(j)}p_{j\to i}+s^{(k)}p_{k\to i}). \quad (3.3\text{B}) \end{aligned}$$

Intuitively, (3.3B) follows a simple cut condition. By choosing $\epsilon \to 0$, we have proven (3.3). The detailed derivation of (3.3A) and (3.3B) are relegated to Appendix D.

As discussed in Section 3.1, (3.3) is inspired by the *multiple-access channel* (MAC) cut-set bound. When considering the MAC, one usually focuses on all *incoming* traffic entering node i, i.e., $R_{j\to i}$, $R_{j\to ki}$, $R_{k\to i}$, and $R_{k\to ij}$, and thus might be interested in quantifying/bounding the following mutual information term:

$$I(\,\mathbf{W}_{j\to i},\mathbf{W}_{j\to ki},\mathbf{W}_{k\to i},\mathbf{W}_{k\to ij}\,;\,[\mathbf{Y}_{*i}]_1^n \mid \mathbf{W}_{i*},\mathbf{W}_{j\to k},\mathbf{W}_{k\to j},[\mathbf{Z}]_1^n). \quad (3.10)$$

Unfortunately, (3.10) does not take into the fact that node j has some private information that need to be delivered to node k (those $\mathbf{W}_{j\to k}$ packets) and vice versa. Due to such an observation, we quantify the mutual information term $I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n)$ instead of (3.10). Comparing $I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n)$ and (3.10), we can use the chain rule to show that

$$(3.10) = I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n) - I(\mathbf{W}_{j\to k}, \mathbf{W}_{k\to j}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n),$$

and the difference $I(\mathbf{W}_{j\to k}, \mathbf{W}_{k\to j}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n)$ can be viewed as the amount of the private information $\mathbf{W}_{j\to k}$ and $\mathbf{W}_{k\to j}$ that has been "leaked" to the other node i. In some broad sense, (3.3) (or equivalent (3.3A)) characterizes a new lower bound on the information leakage

$$I(\mathbf{W}_{j\to k}, \mathbf{W}_{k\to j}\,;\,[\mathbf{Y}_{*i}]_1^n \,|\, \mathbf{W}_{i*}, [\mathbf{Z}]_1^n) \geq \frac{p_{j\to i}}{p_{j\to k\vee i}} R_{j\to k} + \frac{p_{k\to i}}{p_{k\to i\vee j}} R_{k\to j}.$$

This is why in our discussion right after Proposition 3.1.1 we referred to the term $\frac{p_{j\to i}}{p_{j\to k\vee i}} R_{j\to k} + \frac{p_{k\to i}}{p_{k\to i\vee j}} R_{k\to j}$ as the penalty for sending those private-information. Note that similar information leakage arguments have been used in other channel models, e.g., the wireless deterministic channels [61].

3.5 Sketch of The Correctness Proof

We only provide the so-called *first-order analysis* for the achievability of a LNC solution.

We assume that all nodes know the channel reception probabilities, the total time budget n, and the rate vector $\vec{R}$ they want to achieve in the beginning of time 0. As a result, each node can compute the same 15 non-negative values $t_{[\mathrm{u}]}^{(i)}$ and $\{t_{[\mathrm{c},\,l]}^{(i)}\}_{l=1}^4$ for all $i \in \{1, 2, 3\}$ satisfying Proposition 3.2.2.

Our construction consists of 2 stages. Stage 1: Each node, say node i, has $n(R_{i\to j} + R_{i\to k} + R_{i\to jk})$ unicast and multicast packets (i.e., $\mathbf{W}_{i*}$) that need to be sent to other nodes j and k. Assume that those packets are grouped together and indexed as $l = 1$ to $n(R_{i\to j} + R_{i\to k} + R_{i\to jk})$. That is, the packet indices $l = 1$ to $nR_{i\to j}$ correspond to $\mathbf{W}_{i\to j}$ packets, the packet indices $l = nR_{i\to j} + 1$ to $n(R_{i\to j} + R_{i\to k})$ correspond to $\mathbf{W}_{i\to k}$ packets, and so forth. Then in the beginning of time 1, node 1 chooses the first packet (index 1) and repeatedly sends it uncodedly until at least one of nodes 2 and 3 receives it. Whether it is received or not can be known causally by network-wide feedbacks $\mathbf{Z}(t-1)$. Then node 1 picks the next indexed packet and repeat the same process until each of these $n(R_{1\to 2}+R_{1\to 3}+R_{1\to 23})$ packets is heard by at least one of nodes 2 and 3. By simple analysis, see [50], node 1 can finish the transmission in $nt^{(i)}_{[\mathrm{u}]}$ slots since (3.6).[2] We repeat this process for nodes 2 and 3, respectively.[3] Stage 1 can be finished in $n(\sum_i t^{(i)}_{[\mathrm{u}]})$ slots.

After Stage 1, the status of all packets is summarized as follows. Each of $\mathbf{W}_{i\to j}$ packets is heard by at least one of nodes j and k. Those that have already been heard by node j, the intended destination, is delivered successfully and thus will not be considered for future operations (Stage 2). We denote those $\mathbf{W}_{i\to j}$ packets that are overheard by node k only (not by node j) as $\mathbf{W}^{(k)}_{i\to j}$. In average, there are $nR_{i\to j}\frac{p_{i\to \overline{j}k}}{p_{i\to j\vee k}}$ number of $\mathbf{W}^{(k)}_{i\to j}$ packets. Since the causal feedback is available to all network nodes (not only node i), by letting all three nodes perform some simple bookkeeping, any one of the three network nodes (not only node i) is aware of the indices of all the $\mathbf{W}^{(k)}_{i\to j}$ packets. We denote the corresponding index set by $\mathbf{I}^{(k)}_{i\to j}$. Symmetrically, we also have $nR_{i\to k}\frac{p_{i\to j\overline{k}}}{p_{i\to j\vee k}}$ number of $\mathbf{W}^{(j)}_{i\to k}$ packets that was intended for node k but was overheard only by node j in Stage 1, and all three nodes can individually create the corresponding index set $\mathbf{I}^{(j)}_{i\to k}$.

Similarly for the common-information packets $\mathbf{W}_{i\to jk}$, each packet was heard by at least one of nodes j and k in Stage 1. Those that have been heard by both nodes j and

[2] By the law of large numbers, we can ignore the randomness of the events and treat them as deterministic when n is sufficiently large.

[3] Once node 1 has finished transmitting all its own packets $\mathbf{W}_{1*}$, node 2 can immediately take over and start transmitting its own packets $\mathbf{W}_{2*}$ because node 2 knows the value of $n(R_{1\to 2}+R_{1\to 3}+R_{1\to 23})$ and from the instant, error-free, network-wide feedback, node 2 can count in the end of each time slot how many packets node 1 finished transmission. By the same reason, node 3 can immediately take over after node 2 has finished.

k, is delivered successfully and thus will not be considered in Stage 2. We similarly denote those $\mathbf{W}_{i\to jk}$ packets that are heard by node k only (not by node j) as $\mathbf{W}^{(k)}_{i\to jk}$. In average, there are $nR_{i\to jk}\frac{p_{i\to \overline{j}k}}{p_{i\to j\vee k}}$ number of $\mathbf{W}^{(k)}_{i\to jk}$ packets. Symmetrically, we also have $nR_{i\to jk}\frac{p_{i\to j\overline{k}}}{p_{i\to j\vee k}}$ number of $\mathbf{W}^{(j)}_{i\to jk}$ packets that were heard only by node j in Stage 1. The corresponding index sets are denoted by $\mathbf{I}^{(k)}_{i\to jk}$ and $\mathbf{I}^{(j)}_{i\to jk}$, respectively, and they can be individually created by all three nodes through simple bookkeeping.

In sum, all three nodes individually know all 12 index sets $\{\mathbf{I}^{(k)}_{i\to j}, \mathbf{I}^{(k)}_{i\to jk}, \mathbf{I}^{(j)}_{i\to k}, \mathbf{I}^{(j)}_{i\to jk} : \forall(i,j,k)\}$ after Stage 1. In addition, each node i knows the content of its own packets $\mathbf{W}_{i\to j}$, $\mathbf{W}_{i\to k}$, and $\mathbf{W}_{i\to jk}$, and the content of what it has received from other nodes ($\mathbf{W}^{(i)}_{j\to k}$, $\mathbf{W}^{(i)}_{j\to ki}$, $\mathbf{W}^{(i)}_{k\to j}$, $\mathbf{W}^{(j)}_{k\to ij}$) during Stage 1.

Stage 2 is the LNC phase, in which each node i will send a linear combination of the overheard packets. That is, for each time t, node i sends a linear combination $X_i(t) = [\tilde{W}_j + \tilde{W}_k]$ with 4 possible ways of choosing the the constituent packets $\tilde{W}_j$ and $\tilde{W}_k$, which are detailed as follows.

$$
\begin{aligned}
&[\mathrm{c},1]: \quad \tilde{W}_j \in \mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk} \quad \text{and} \quad \tilde{W}_k \in \mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk},\\
&[\mathrm{c},2]: \quad \tilde{W}_j \in \mathbf{W}^{(i)}_{k\to j} \cup \mathbf{W}^{(i)}_{k\to ij} \quad \text{and} \quad \tilde{W}_k \in \mathbf{W}^{(i)}_{j\to k} \cup \mathbf{W}^{(i)}_{j\to ki},\\
&[\mathrm{c},3]: \quad \tilde{W}_j \in \mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk} \quad \text{and} \quad \tilde{W}_k \in \mathbf{W}^{(i)}_{j\to k} \cup \mathbf{W}^{(i)}_{j\to ki},\\
&[\mathrm{c},4]: \quad \tilde{W}_j \in \mathbf{W}^{(i)}_{k\to j} \cup \mathbf{W}^{(i)}_{k\to ij} \quad \text{and} \quad \tilde{W}_k \in \mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}.
\end{aligned}
$$

To explain the intuition behind the 4 coding choices $[\mathrm{c},1]$ to $[\mathrm{c},4]$, we observe that choice $[\mathrm{c},1]$ is the standard LNC operation for the 2-receiver broadcast channels [47] since node i sends a linear sum that benefits both nodes j and k simultaneously, i.e., the sum of two packets, each overheard by an undesired receiver. Choice $[\mathrm{c},2]$ is the standard LNC operation for the 2-way relay channels, since node i, as a relay for the 2-way traffic from $j\to k$ and from $k\to j$, respectively, mixes the packets from two opposite directions and sends their linear sum. Choices $[\mathrm{c},3]$ and $[\mathrm{c},4]$ are the new "hybrid" cases that are proposed in this work, for which we can mix part of the broadcast traffic and part of the 2-way traffic. We argue that transmitting such a linear mixture again benefits both nodes simultaneously. For example, suppose that coding choice $[\mathrm{c}, 3]$ is used, and the linear sum $[\tilde{W}_j + \tilde{W}_k]$ is received by node j. Since $\tilde{W}_k$ is a function of all packets originated from node j, node j can compute $\tilde{W}_k$ by itself and then subtract it from the linear sum and derive the desired packet $\tilde{W}_j$. Similarly, if node k receives the linear sum, since it has overheard all packets in $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$, it can subtract $\tilde{W}_j$ and decode its desired $\tilde{W}_k$. The argument for coding choice $[\mathrm{c}, 4]$ is symmetric.

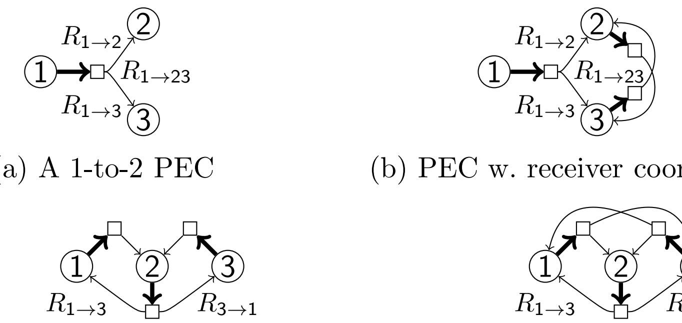
We now explain in details how to implement the above 4 coding choices for each of the time slots in Stage 2. The best way to explain the implementation is to temporarily view the overheard packets as being stored in a big queue. Namely, in the beginning of Stage 2, all the packets in $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ are put into a big queue. Similarly, all the packets in $\mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}$, $\mathbf{W}^{(i)}_{k\to j} \cup \mathbf{W}^{(i)}_{k\to ij}$, and $\mathbf{W}^{(i)}_{j\to k} \cup \mathbf{W}^{(i)}_{j\to ki}$ are put into 3 big queues as well, one queue for each set of packets respectively. Then coding choice $[\mathrm{c}, 1]$ means that node i takes the head-of-line packet from the queue of $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$, and combines it with the head-of-line packet from the queue of $\mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}$. Coding choices $[\mathrm{c}, 2]$ to $[\mathrm{c}, 4]$ can be interpreted similarly by combining the head-of-line packets from different queues.

Since each node i has 4 possible coding choices, we perform coding choice $[\mathrm{c}, l]$ for exactly $nt^{(i)}_{[\mathrm{c},l]}$ times sequentially for $l=1$ to 4. After sending the 4 coding choices for a combined total of $n\big(t^{(i)}_{[\mathrm{c},1]} + t^{(i)}_{[\mathrm{c},2]} + t^{(i)}_{[\mathrm{c},3]} + t^{(i)}_{[\mathrm{c},4]}\big)$ time slots for node i, we set $i = i+1$ and repeat the same process until all three nodes have finished transmission. Totally, Stage 2 takes $\sum_{i\in\{1,2,3\}} n\big(t^{(i)}_{[\mathrm{c},1]} + t^{(i)}_{[\mathrm{c},2]} + t^{(i)}_{[\mathrm{c},3]} + t^{(i)}_{[\mathrm{c},4]}\big)$ time slots. We now describe how to manage the "queues" within each node during transmission.

Suppose that node i is performing the coding choice $[\mathrm{c}, 1]$ and chooses two head-of-line packets $\tilde{W}_j \in \mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ and $\tilde{W}_k \in \mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}$ from the individual queues, respectively. If the linear combination $[\tilde{W}_j + \tilde{W}_k]$ is received by node j, then node j will decode the desired $\tilde{W}_j$ by subtracting the overheard packet $\tilde{W}_k$. As a result, we remove the successfully delivered packet $\tilde{W}_j$ from its queue. Similarly, if the combination $[\tilde{W}_j + \tilde{W}_k]$ is received by node k, then node k can decode the desired packet $\tilde{W}_k$ and we remove $\tilde{W}_k$ from the corresponding queue. If any one of the two queues is empty, say the queue corresponding to $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ is empty during coding choice $[\mathrm{c}, 1]$, then we simply set $\tilde{W}_j = 0$. Namely, in such a degenerate case we choose to send an uncoded packet $[0 + \tilde{W}_k]$ instead of a linear combination $[\tilde{W}_j + \tilde{W}_k]$. If both queues are empty, then we simply send a 0 packet. The same queue management is applied to coding choices $[\mathrm{c}, 2]$ to $[\mathrm{c}, 4]$ as well.

Note that the above process requires very detailed bookkeeping at each node. Namely, both nodes j and k needs to know the indices of the head-of-line packet $\tilde{W}_j$ and $\tilde{W}_k$ while node i is executing Stage 2. So that they can know which of the overheard packets it needs to subtract from the linear combination $[\tilde{W}_j + \tilde{W}_k]$ when received. This is possible since in the beginning of Stage 2, each node knows all 12 index sets: $\{\mathbf{I}^{(k)}_{i\to j}, \mathbf{I}^{(k)}_{i\to jk}, \mathbf{I}^{(j)}_{i\to k}, \mathbf{I}^{(j)}_{i\to jk} : \forall(i, j, k)\}$. Since the reception status $[\mathbf{Z}]_1^{t-1}$ is available to all nodes for free, through detailed bookkeeping, each node (not only node i but also nodes j and k) can successfully trace the status of the queues when node i is executing Stage 2. In this way each node maintains a synchronized view of the queue status of the other nodes and can thus know the indices of the head-of-line packets that constitute the linear combination.

Another important point worth emphasizing is that the queues cannot be replenished during Stage 2. Namely, if a packet is removed from the queue in one coding operation, then it will be removed from the synchronized queues at all three nodes and will not participate in any future coding operations. For example, the packets in $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ will participate in coding choice $[\mathrm{c}, 1]$ of node i, but they can also participate in coding choice $[\mathrm{c}, 3]$ of node i, and coding choices $[\mathrm{c}, 2]$ and $[\mathrm{c}, 3]$ of node k. If a packet in $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ is successfully delivered through coding choice $[\mathrm{c}, 1]$ of node i, then it will be removed from the queue and will not participate in any subsequent coding choices $[\mathrm{c}, 3]$ of node i, and coding choices $[\mathrm{c}, 2]$ and $[\mathrm{c}, 3]$ of node k in the future time slots. Again, this LNC design is possible since each node maintains a synchronized view of the queue status of the other nodes with the help of the causal feedback $[\mathbf{Z}]_1^{t-1}$.



(a) A 1-to-2 PEC (b) PEC w. receiver coordination

(c) A Two-way relay PEC (d) A Two-way relay PEC w. opportunistic routing

Fig. 3.1. Special examples of the 3-node Packet Erasure Network (PEN) considered in this work. The rectangle implies the broadcast packet erasure channel.

Since $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ participates in coding choices $[\mathrm{c}, 1]$ and $[\mathrm{c}, 3]$ of node i and coding choices $[\mathrm{c}, 2]$ and $[\mathrm{c}, 3]$ of node k, (3.7) guarantees that the queue of $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ will be empty in the end of Stage 2, which means that we can finish sending all $\mathbf{W}^{(k)}_{i\to j} \cup \mathbf{W}^{(k)}_{i\to jk}$ packets and they will all successfully arrive at node j, the intended destination.[4] Symmetrically, (3.8) guarantees that the queue of $\mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}$ will be empty, which means that we can finish sending all $\mathbf{W}^{(j)}_{i\to k} \cup \mathbf{W}^{(j)}_{i\to jk}$ packets to their intended destination node k in the end of Stage 2. Finally, (3.4) guarantees that we can finish Stages 1 and 2 in the allotted n time slots. The sketch of the proof is complete.

3.6 Special Examples and Numerical Evaluation

In the following, we apply Propositions 3.1.1 and 3.2.2 to the four special examples. We also numerically evaluate the 9-dimensional capacity region for some specific channel parameter values.

[4]Those $\mathbf{W}^{(k)}_{i\to jk}$ packets are the common-information packets that are intended for both nodes j and k. However, since our definition of $\mathbf{W}^{(k)}_{i\to jk}$ counts only those that have already been received by node k, we say herein their new intended destination is node j as instead.

The considered 3-node PEN contains many important practical and theoretically interesting scenarios as sub-cases. **Example 1:** If we set the broadcast PECs of nodes 2 and 3 to be always erasure (i.e., neither nodes can transmit anything), then Fig. 2.1(b) collapses to Fig. 3.1(a), the 2-receiver broadcast PEC scenario. The capacity region $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$ derived in our Scenario 1 is identical to the existing results in [47, 55]. **Example 2:** Instead of setting the PECs of nodes 2 and 3 to all erasure, we set $R_{2\to 1}$, $R_{2\to 3}$, $R_{3\to 1}$, $R_{3\to 2}$, $R_{2\to 31}$, $R_{3\to 12}$ to be zeros. Namely, we still allow nodes 2 and 3 to transmit but there is no information message emanating from nodes 2 and 3. In this case, node 2 can potentially be a relay that helps forwarding those node-1 packets destined for node 3 and node 3 can be a relay for flow $1 \to 2$, see Fig. 3.1(b). This work then characterizes the Shannon capacity[5] $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$ of a broadcast PEC with receiver coordination.

Example 3: If we set $R_{1\to 2}$, $R_{2\to 1}$, $R_{2\to 3}$, $R_{3\to 2}$, $R_{1\to 23}$, $R_{2\to 31}$, $R_{3\to 12}$ to be zeros and prohibit any direct communication between nodes 1 and 3, Fig. 2.1(b) now collapses to Fig. 3.1(c), in which node 2 is a two-way relay for unicast flows $1 \to 3$ and $3 \to 1$. The results in this work thus characterizes the Shannon capacity region $(R_{1\to 3}, R_{3\to 1})$ of this two-way relay network Fig. 3.1(c), which is identical to the existing result in [23]. **Example 4:** If we additionally allow direct communication between nodes 1 and 3, Fig. 2.1(b) now collapses to Fig. 3.1(d). Namely, when node 1 is sending packets to the relay node 2, the packets might be overheard directly by the destination node 3. If indeed node 3 overhears the communication, then node 1 could inform node 2 opportunistically that there is no need to forward that packet to node 3 anymore. Such a scheme is called *opportunistic routing* and testbed implementation [8] has shown that opportunistic routing can potentially improve the throughput by 20x. The results in this work thus characterize the Shannon capacity region $(R_{1\to 3}, R_{3\to 1})$ of Fig. 3.1(d), which allows for the possibility of both opportunistic routing and two-way-relay coding. The Shannon capacity region computed by this work again matches the existing result in [35].

[5] In [43], the LNC capacity of Fig. 3.1(b) was characterized, but the most general Shannon capacity region was unknown in [43].

3.6.1 Example 1: The Simplest 1-to-2 Broadcast PEC

Consider the simplest setting of a 1-to-2 broadcast PEC with 2 private-information flows of rates $R_{1\to 2}$ and $R_{1\to 3}$, and 1 common-information flow of rate $R_{1\to 23}$. See Fig. 3.1(a) for illustration. In this scenario, we assume that only node 1 can transmit and nodes 2 and 3 can only listen and send ACK/NACK feedback after each packet transmission. This simple 1-to-2 broadcast PEC can be viewed as a special example of the general problem by setting $p_{2\to 3\vee 1} = p_{3\to 1\vee 2} = 0$, and by hardwiring the unused rates $\{R_{2\to 1}, R_{2\to 3}, R_{3\to 1}, R_{3\to 2}\}$ and $\{R_{2\to 31}, R_{3\to 12}\}$ to zeros. One can thus use Proposition 3.1.1 to compute the 3-dimensional capacity region $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$ of the 1-to-2 broadcast PEC. More explicitly, by setting $s^{(1)} = 1$ and $s^{(2)} = s^{(3)} = 0$, (3.3) with $i = 2$ leads to the following (3.11) and (3.3) with $i = 3$ leads to the following (3.12):

$$R_{1\to 2} + R_{1\to 23} \leq p_{1\to 2} - \frac{p_{1\to 2}}{p_{1\to 2\vee 3}} R_{1\to 3}, \tag{3.11}$$

$$R_{1\to 3} + R_{1\to 23} \leq p_{1\to 3} - \frac{p_{1\to 3}}{p_{1\to 2\vee 3}} R_{1\to 2}. \tag{3.12}$$

As expected, the capacity region $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$ described by (3.11) and (3.12) is identical to the existing 1-to-2 broadcast PEC capacity results in [47].

3.6.2 Example 2: 1-to-2 Broadcast PEC With Receiver Coordination

Another special example is the 1-to-2 broadcast PEC with receiver coordination, see Fig. 3.1(b). In this scenario, node 1 still likes to communicate and send 3 flows to nodes 2 and 3 with rates $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$. However, we allow nodes 2 and 3 to communicate with each other with the constraint that whenever node 2 (or node 3) transmits, node 1 has to remain silent. The communication between nodes 2 and 3 can be used either to relay some overheard packets to the intended destination, or to send carefully designed coded packets that can further enhance the throughput.

Similar to the previous example, such a scenario is a special case of the general problem by setting $p_{2\to 1} = p_{3\to 1} = 0$, and by hardwiring $\{R_{2\to 1}, R_{2\to 3}, R_{3\to 1}, R_{3\to 2}\}$ and $\{R_{2\to 31}, R_{3\to 12}\}$ to zeros. We can again use Proposition 3.1.1 to compute the capacity region $(R_{1\to 2}, R_{1\to 3}, R_{1\to 23})$ of the 1-to-2 broadcast PEC with receiver coordination:

$$\sum_{\forall\, i\in\{1,2,3\}} s^{(i)} \leq 1, \tag{3.13}$$

$$R_{1\to 2} + R_{1\to 3} + R_{1\to 23} \leq s^{(1)} \cdot p_{1\to 2\vee 3}, \tag{3.14}$$

$$R_{1\to 2} + R_{1\to 23} + \frac{p_{1\to 2}}{p_{1\to 2\vee 3}} R_{1\to 3} \leq s^{(3)} \cdot p_{3\to 2} + s^{(1)} \cdot p_{1\to 2}, \tag{3.15}$$

$$R_{1\to 3} + R_{1\to 23} + \frac{p_{1\to 3}}{p_{1\to 2\vee 3}} R_{1\to 2} \leq s^{(1)} \cdot p_{1\to 3} + s^{(2)} \cdot p_{2\to 3}, \tag{3.16}$$

where (3.13) follows from (3.1); (3.14) follows from (3.2); and (3.15) and (3.16) follow from (3.3).

Compared to the existing work [43], our results have characterized the more general Shannon capacity region instead of linear capacity region while also considering the possibility of co-existing common-information rate $R_{1\to 23}$.

3.6.3 Example 3: Two-way Relay PEC

Another example is the two-way relay PEC as described in Fig. 3.1(c). Namely, nodes 1 and 3 want to communicate with each other with rates $(R_{1\to 3}, R_{3\to 1})$, respectively. The communication must be achieved via a relaying node 2. Such a scenario is a special case of the general problem by simply hardwiring $\{R_{1\to 2}, R_{2\to 1}, R_{2\to 3}, R_{3\to 2}\}$

and $\{R_{1\to 23}, R_{2\to 31}, R_{3\to 12}\}$ to zeros. We can again use Proposition 3.1.1 to compute the capacity region $(R_{1\to 3}, R_{3\to 1})$:

$$\sum_{\forall\, i\in\{1,2,3\}} s^{(i)} \leq 1, \tag{3.17}$$

$$R_{1\to 3} \leq s^{(1)} \cdot p_{1\to 2}, \qquad R_{3\to 1} \leq s^{(3)} \cdot p_{3\to 2}, \tag{3.18}$$

$$R_{1\to 3} \leq s^{(2)} \cdot p_{2\to 3}, \qquad R_{3\to 1} \leq s^{(2)} \cdot p_{2\to 1}, \tag{3.19}$$

where (3.17) and (3.18) follow from (3.1) and (3.2), respectively, and (3.19) follows from (3.3). One can easily verify that the capacity region described by (3.17) to (3.19) matches the existing results in [23].

3.6.4 Example 4: Two-way Relay PEC with Opportunistic Routing

For the same setting as in **Example 3** but allowing the direct communications between node 1 and node 3, see Fig. 3.1(d), we can also use Proposition 3.1.1 to compute the two-way relay PEC capacity region $(R_{1\to 3}, R_{3\to 1})$ with opportunistic routing:

$$\sum_{\forall\, i\in\{1,2,3\}} s^{(i)} \leq 1, \tag{3.20}$$

$$R_{1\to 3} \leq s^{(1)} p_{1\to 2\vee 3}, \quad R_{3\to 1} \leq s^{(3)} p_{3\to 1\vee 2}, \tag{3.21}$$

$$R_{1\to 3} \leq s^{(1)} p_{1\to 3} + s^{(2)} p_{2\to 3}, \tag{3.22}$$

$$R_{3\to 1} \leq s^{(2)} p_{2\to 1} + s^{(3)} p_{3\to 1}. \tag{3.23}$$

One can verify that the capacity region described by (3.20) to (3.23) matches the existing results in [35].

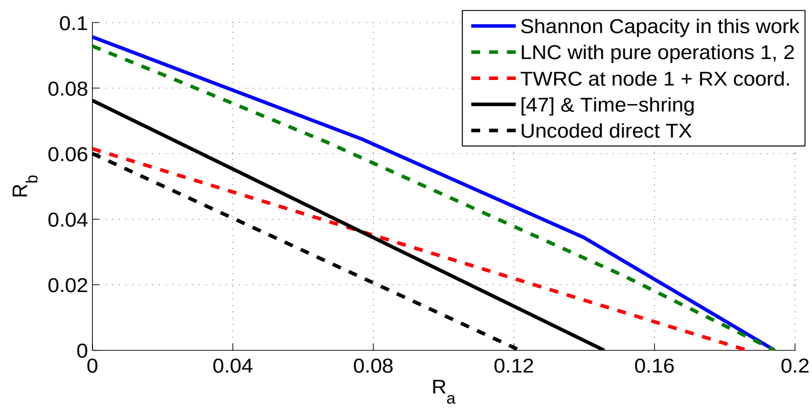


Fig. 3.2. Comparison of the capacity region with different achievable rates

3.6.5 Numerical Evaluation

Consider a 3-node network with marginal channel success probabilities $p_{1\to 2} = 0.35$, $p_{1\to 3} = 0.8$, $p_{2\to 1} = 0.6$, $p_{2\to 3} = 0.5$, $p_{3\to 1} = 0.3$, and $p_{3\to 2} = 0.75$, respectively, and we assume that all the erasure events are independent. That is, $p_{i\to j\vee k} = 1-(1-p_{i\to j})(1-p_{i\to k})$. To illustrate the 9-dimensional capacity region, we further assume that the following 3 flows are of the same rate $R_{1\to 2} = R_{1\to 3} = R_{1\to 23} = R_a$ and the other 6 flows are of rate $R_{2\to 1} = R_{2\to 3} = R_{3\to 1} = R_{3\to 2} = R_{2\to 31} = R_{3\to 12} = R_b$. We will use Proposition 3.1.1 to find the largest R_a and R_b value for this example scenario.

Fig. 3.2 compares the Shannon capacity region of (R_a, R_b) with different achievability schemes. The smallest rate region is achieved by simply performing uncoded direct transmission. The second achievability scheme combines the broadcast channel LNC in [47] with time-sharing among all three nodes. The third scheme performs two-way relay channel (TWRC) coding in node 1 for those $3 \to 2$ and $2 \to 3$ flows while allowing node 2 to relay the node 1's packets destined for node 3 and vice versa. The fourth scheme is derived from our achievability scheme in the proof of Proposition 3.2.2 except when we impose the restriction that the scheme can only use LNC choices that were known previously. Namely, we allow all three nodes to perform the broadcast-based LNC and/or TWRC-based LNC operations (coding choices $[\mathrm{c},1]$ and $[\mathrm{c},2]$ in Stage 2) but not the hybrid operations (coding choices $[\mathrm{c},3]$ and $[\mathrm{c},4]$) proposed in this work. One can see that the result is strictly suboptimal. It shows that the proposed hybrid operations are critical for achieving the Shannon capacity in Propositions 3.1.1 and 3.2.2. The detailed rate region description of each sub-optimal achievability scheme is described in Appendix E.

3.7 Chapter Summary

In this chapter, we discuss the capacity region of the 3-node network formulated in Section 2.2. In Sections 3.1 and 3.2, we propose the Shannon capacity outer bound and the simple LNC achievability scheme, respectively. In Section 3.3, we discuss the fully-connected assumption for Scenario 1 and Scenario 2, and identify some possible future work. In Section 3.4, we provide the proof sketch of the Shannon outer bound, where the full detailed derivations is relegated to Appendix D. In Section 3.5, we also provide the sketch of the correctness proof of our achievability scheme based on the first-order analysis. The full detailed proof invoking the law of large numbers can be found in Appendix B. In Section 3.6.5, we discuss the special examples of the 3-node network and use the numerical results to demonstrate that the proposed simple but capacity-achieving LNC scheme strictly outperforms existing results. The Space-based Framework and the LNC capacity region descriptions of the 3-node packet erasure network can also be found in Appendix A.

4. APPROACHING THE LNC CAPACITY OF THE SMART REPEATER PACKET ERASURE NETWORK

In Section 2.3, we formulated the problem of the wireless 2-flow smart repeater packet erasure network with feedback, linear encoding/decoding, and scheduling between the source s and the relay r. In this chapter, we investigate the LNC capacity region (R_1, R_2) of the smart repeater network. The outer bound is proposed by leveraging upon the algebraic structure of the underlying LNC problem. For the achievability scheme, we show that the classic butterfly-style is far from optimality and propose new LNC operations that lead to close-to-optimal performance. By numerical simulations, we demonstrate that the proposed outer/inner bounds are very close, thus effectively bracketing the LNC capacity of the smart repeater problem.

4.1 LNC Capacity Outer Bound

Recall that, since the coding vector $\mathbf{c}_t$ has $n(R_1 + R_2)$ number of coordinates, there are exponentially many ways of jointly designing the scheduling $\sigma(t)$ and the coding vector choices $\mathbf{c}_t$ over time when sufficiently large n and $\mathbb{F}_q$ are used. Therefore, we will first simplify the aforementioned design choices by comparing $\mathbf{c}_t$ to the knowledge spaces $S_h(t-1)$, $h \in \{d_1, d_2, r\}$. Such a simplification allows us to derive Proposition 4.1.1, which uses a linear programming (LP) solver to exhaustively search over the entire coding and scheduling choices and thus computes an LNC capacity outer bound. An LNC capacity inner bound will later be derived in Section 4.2 by proposing an elegant LNC solution and analyze its performance.

To that end, we use S_k as shorthand for $S_k(t-1)$, the destination d_k knowledge space in the end of time $t-1$. We first define the following 7 linear subspaces of Ω.

$$A_1(t) \triangleq S_1, \qquad A_2(t) \triangleq S_2, \tag{4.1}$$

$$A_3(t) \triangleq S_1 \oplus \Omega_1, \qquad A_4(t) \triangleq S_2 \oplus \Omega_2, \tag{4.2}$$

$$A_5(t) \triangleq S_1 \oplus S_2, \tag{4.3}$$

$$A_6(t) \triangleq S_1 \oplus S_2 \oplus \Omega_1, \qquad A_7(t) \triangleq S_1 \oplus S_2 \oplus \Omega_2, \tag{4.4}$$

where $A \oplus B \triangleq \mathsf{span}\{\mathbf{v} : \mathbf{v} \in A \cup B\}$ is the *sum space* of any $A, B \subseteq \Omega$. In addition to those seven subspaces $A_i(t)$, $i = 1, \cdots, 7$, we also define the following eight additional subspaces involving $S_r(t-1)$:

$$A_{i+7}(t) \triangleq A_i(t) \oplus S_r \quad \text{for all } i = 1, \cdots, 7, \tag{4.5}$$

$$A_{15}(t) \triangleq S_r, \tag{4.6}$$

where S_r is a shorthand notation for $S_r(t-1)$, the knowledge space of relay r in the end of time $t-1$.

In total, there are $7+8=15$ linear subspaces of Ω. We then partition the overall message space Ω into 2^{15} disjoint subsets by the *Venn diagram* generated by these 15 subspaces. That is, for any given coding vector $\mathbf{c}_t$, we can place it in exactly one of the 2^{15} disjoint subsets by testing whether it belongs to which A-subspaces. This is always true regardless of the time index t, i.e., any coding vector $\mathbf{c}_t$ transmitted by either source or relay always lies in one of the 2^{15} disjoint subsets while the regions of disjoint subsets may change over the course of time. In the following discussion, we often drop the input argument "(t)" when the time instant of interest is clear in the context.

We now use 15 bits to represent each disjoint subset of the overall message space Ω. For any 15-bit string $\mathbf{b} = b_1 b_2 \cdots b_{15}$, we define "the coding type-$\mathbf{b}$" by

$$\mathsf{TYPE}^{(s)}_{\mathbf{b}} \triangleq \Big(\bigcap_{l:b_l=1} A_l\Big) \setminus \Big(\bigcup_{l:b_l=0} A_l\Big). \tag{4.7}$$

The superscript "(s)" indicates that source s can send $\mathbf{c}_t$ from any of these 2^{15} types since source s knows all $\mathbf{W}_1$ and $\mathbf{W}_2$ packets to begin with. Note that not all 2^{15} disjoint subsets are feasible. For example, any $\mathsf{TYPE}^{(s)}_{\mathbf{b}}$ with $b_7 = 1$ but $b_{14} = 0$ is always empty because any coding vector that lies in $A_7 = S_1 \oplus S_2 \oplus \Omega_2$ cannot lie outside the larger $A_{14} = S_1 \oplus S_2 \oplus S_r \oplus \Omega_2$, see (4.4) and (4.5), respectively. We say those always empty subsets are *infeasible coding types* and the rest is called *feasible coding types* (FTs). By exhaustive computer search, we can prove that out of 2^{15}=32768 subsets, only 154 of them are feasible. Namely, the entire coding space Ω can be viewed as a union of 154 disjoint coding types. Source s can choose a coding vector $\mathbf{c}_t$ from one of these 154 types. See (2.12).

For coding vectors that relay r can choose, we can further reduce the number of possible placements of $\mathbf{c}_t$ in the following way. By (2.16), we know that when $\sigma(t) = r$, the $\mathbf{c}_t$ sent by relay must belong to its knowledge space $S_r(t-1)$. Hence, such $\mathbf{c}_t$ must always lie in $S_r(t-1)$, which is $A_{15}(t)$, see (4.6). As a result, any coding vector $\mathbf{c}_t$ sent by relay r must lie in those 154 subsets FTs that satisfy:

$$\mathsf{TYPE}^{(r)}_{\mathbf{b}} \triangleq \{\mathsf{TYPE}^{(s)}_{\mathbf{b}} : \mathbf{b} \in \mathsf{FTs} \text{ such that } b_{15} = 1\}. \tag{4.8}$$

Again by computer search, there are 18 such coding types out of 154 subsets FTs. We call those 18 subsets as *relay's feasible coding types* ($r\mathsf{FTs}$). Obviously, $r\mathsf{FTs} \subseteq \mathsf{FTs}$. See Appendix G for the enumeration of those FTs and $r\mathsf{FTs}$.

We can then derive the following upper bound.

Proposition 4.1.1. *A rate vector (R_1, R_2) is in the LNC capacity region only if there exists* 154 *non-negative variables $x^{(s)}_{\mathbf{b}}$ for all $\mathbf{b} \in s\mathsf{FTs}$,* 18 *non-negative variables $x^{(r)}_{\mathbf{b}}$*

for all $\mathbf{b} \in r\mathsf{FTs}$*, and* 14 *non-negative y-variables,* y_1 *to* y_{14}*, such that jointly they satisfy the following three groups of linear conditions:*

- *Group 1, termed the* time-sharing condition*, has* 1 *inequality:*

$$\left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs}} x_{\mathbf{b}}^{(s)}\right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs}} x_{\mathbf{b}}^{(r)}\right) \leq 1. \tag{4.9}$$

- *Group 2, termed the* rank-conversion conditions*, has* 14 *equalities:*

$$y_1 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_1=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_1=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_1) \right), \tag{4.10}$$

$$y_2 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_2=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_2) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_2=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_2) \right), \tag{4.11}$$

$$y_3 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_3=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_3=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_1) \right) + R_1, \tag{4.12}$$

$$y_4 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_4=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_2) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_4=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_2) \right) + R_2, \tag{4.13}$$

$$y_5 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_5=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_5=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_1, d_2) \right), \tag{4.14}$$

$$y_6 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_6=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_6=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_1, d_2) \right) + R_1, \tag{4.15}$$

$$y_7 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_7=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2) \right) + \left(\sum_{\forall \mathbf{b} \in r\mathsf{FTs} \textit{ s.t. } b_7=0} x_{\mathbf{b}}^{(r)} \cdot p_r(d_1, d_2) \right) + R_2, \tag{4.16}$$

$$y_8 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_8=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, r) \right), \qquad y_9 = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_9=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_2, r) \right), \tag{4.17}$$

$$y_{10} = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_{10}=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, r) \right) + R_1, \tag{4.18}$$

$$y_{11} = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_{11}=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_2, r) \right) + R_2, \tag{4.19}$$

$$y_{12} = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_{12}=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2, r) \right), \tag{4.20}$$

$$y_{13} = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_{13}=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2, r) \right) + R_1, \tag{4.21}$$

$$y_{14} = \left(\sum_{\forall \mathbf{b} \in s\mathsf{FTs} \textit{ s.t. } b_{14}=0} x_{\mathbf{b}}^{(s)} \cdot p_s(d_1, d_2, r) \right) + R_2, \tag{4.22}$$

- *Group 3, termed the* decodability conditions*, has* 5 *equalities:*

$$y_1 = y_3, \quad y_2 = y_4, \quad y_8 = y_{11}, \quad y_9 = y_{11}, \tag{4.23}$$

$$y_5 = y_6 = y_7 = y_{12} = y_{13} = y_{14} = (R_1 + R_2). \tag{4.24}$$

The intuition is as follows. Consider any achievable (R_1, R_2) and the associated LNC scheme. In the beginning of any time t, we can compute the knowledge spaces $S_1(t-1)$, $S_2(t-1)$, and $S_r(t-1)$ by (2.15) and use them to compute the A-subspaces in (4.1)–(4.6). Then suppose that for time t, the given scheme chooses source s to transmit a coding vector $\mathbf{c}_t$. By the previous discussions, we can classify which $\mathsf{TYPE}^{(s)}_{\mathbf{b}}$ this $\mathbf{c}_t$ belongs to, by comparing it to those 15 A-subspaces. After running the given scheme from time 1 to n, we can thus compute the variable $x^{(s)}_{\mathbf{b}} \triangleq \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} 1_{\{\mathbf{c}_t \in \mathsf{TYPE}^{(s)}_{\mathbf{b}}\}}\right]$ for each $\mathsf{TYPE}^{(s)}_{\mathbf{b}}$ as the *frequency* of scheduling source s with the chosen $\mathbf{c}_t$ happening to be in $\mathsf{TYPE}^{(s)}_{\mathbf{b}}$. Similarly for $\mathsf{TYPE}^{(r)}_{\mathbf{b}}$, we can compute the variable $x^{(r)}_{\mathbf{b}} \triangleq \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} 1_{\{\mathbf{c}_t \in \mathsf{TYPE}^{(r)}_{\mathbf{b}}\}}\right]$ for each $\mathsf{TYPE}^{(r)}_{\mathbf{b}}$ as the *frequency* of scheduling relay r with the chosen $\mathbf{c}_t$ happening to be in $\mathsf{TYPE}^{(r)}_{\mathbf{b}}$. Obviously, the computed $\{x^{(s)}_{\mathbf{b}}, x^{(r)}_{\mathbf{b}}\}$ satisfy the time-sharing inequality (4.9). We then compute the y-variables by

$$y_l \triangleq \frac{1}{n}\mathbb{E}\left[\mathsf{rank}\big(A_i(n)\big)\right], \ \forall l \in \{1, 2, \cdots, 14\}, \tag{4.25}$$

as normalized expected ranks of A-subspaces in the end of time n. We now claim that these variables satisfy (4.10) to (4.24). This claim implies that for any LNC-achievable (R_1, R_2), there exists $x^{(s)}_{\mathbf{b}}$, $x^{(r)}_{\mathbf{b}}$, and y-variables satisfying Proposition 4.1.1, which means that Proposition 4.1.1 constitutes an outer bound on the LNC capacity.

To prove that (4.10) to (4.22) are true,[1] consider an A-subspace, say $A_3(t) = S_1(t-1) \oplus \Omega_1 = RS_{d_1}(t-1) \oplus \Omega_1$ as defined in (4.2) and (2.15). In the beginning of

[1]For rigorous proofs, we need to invoke the law of large numbers and take care of the ϵ-error probability. For ease of discussion, the corresponding technical details are omitted when discussing the intuition of Proposition 4.1.1.

time 1, destination d_1 has not received any packet yet, i.e., $RS_{d_1}(0) = \{\mathbf{0}\}$. Thus the rank of $A_3(1)$ is $\mathsf{rank}(\Omega_1) = nR_1$.

The fact that $S_1(t-1)$ contributes to $A_3(t)$ implies that $\mathsf{rank}(A_3(t))$ will increase by one whenever destination d_1 receives a packet $\mathbf{c}_t\mathbf{W}^\top$ satisfying $\mathbf{c}_t \notin A_3(t)$. Whenever source s sends a $\mathbf{c}_t$ in $\mathsf{TYPE}_{\mathbf{b}}^{(s)}$ with $b_3 = 0$, such $\mathbf{c}_t$ is not in $A_3(t)$. Whenever destination d_1 receives it, $\mathsf{rank}(A_3(t))$ increases by 1. Moreover, whenever relay r sends a $\mathbf{c}_t$ in $\mathsf{TYPE}_{\mathbf{b}}^{(r)}$ with $b_3 = 0$ and destination d_1 receives it, $\mathsf{rank}(A_3(t))$ also increases by 1. Therefore, in the end of time n, we have

$$\begin{aligned}\mathsf{rank}(A_3(n)) = &\sum_{t=1}^{n} 1_{\left\{\substack{\text{source } s \text{ sends } \mathbf{c}_t \in \mathsf{TYPE}_{\mathbf{b}}^{(s)} \text{ with } b_3=0, \\ \text{and destination } d_1 \text{ receives it}}\right\}} \\ &+ \sum_{t=1}^{n} 1_{\left\{\substack{\text{relay } r \text{ sends } \mathbf{c}_t \in \mathsf{TYPE}_{\mathbf{b}}^{(r)} \text{ with } b_3=0, \\ \text{and destination } d_1 \text{ receives it}}\right\}} \\ &+ \mathsf{rank}(A_3(0)).\end{aligned} \tag{4.26}$$

Taking the normalized expectation of (4.26), we have proven (4.12). By similar *rank-conversion* arguments, (4.10) to (4.22) can be shown to be true.

In the end of time n, since both destination d_1 and d_2 can decode the desired packets $\mathbf{W}_1$ and $\mathbf{W}_2$, respectively, we thus have $S_1(n) \supseteq \Omega_1$ and $S_2(n) \supseteq \Omega_2$, or equivalently $S_k(n) = S_k(n) \oplus \Omega_k$ for all $k \in \{1, 2\}$. This implies that the ranks of $A_1(n)$ and $A_3(n)$, and the ranks of $A_2(n)$ and $A_4(n)$ are equal, respectively. Together with (4.25), we thus have the first two equalities in (4.23). Similarly, one can prove that the remaining equalities in (4.23) and (4.24) are satisfied as well. The claim is thus proven.

4.2 LNC Capacity Inner Bound

In the smart repeater problem of our interest, if the r-PEC is weaker than the s-PEC, then there is no need to do relaying since we can simply let s take over relay's operations. We thus assume

Definition 4.2.1. *The smart repeater network with two destinations* $\{d_1, d_2\}$ *is* strong-relaying *if*

$$p_r(d_1) > p_s(d_1),$$
$$p_r(d_1\overline{d_2}) > p_s(d_1\overline{d_2}),$$
$$p_r(d_2) > p_s(d_2),$$
$$p_r(\overline{d_1}d_2) > p_s(\overline{d_1}d_2),$$
$$\textit{and } p_r(d_1, d_2) > p_s(d_1, d_2),$$

i.e., the given r-PEC is stronger than the given s-PEC for all non-empty subsets of destinations.

We now describe our capacity-approaching achievability scheme.

Proposition 4.2.1. *A rate vector* (R_1, R_2) *is LNC-achievable if there exist* 2 *non-negative variables* t_s *and* t_r*,* $(6 \times 2 + 8)$ *non-negative s-variables:*

$$\left\{s^k_{\mathsf{UC}},\ s^k_{\mathsf{PM1}},\ s^k_{\mathsf{PM2}},\ s^k_{\mathsf{RC}},\ s^k_{\mathsf{DX}},\ s^{(k)}_{\mathsf{DX}}\ :\ \textit{for all}\ k \in \{1,2\}\right\},$$
$$\left\{s_{\mathsf{CX};l}\,(l\!=\!1,\cdots,8)\right\},$$

and $(3 \times 2 + 3)$ *non-negative r-variables:*

$$\left\{r^k_{\mathsf{UC}},\ r^{(k)}_{\mathsf{DT}},\ r^{[k]}_{\mathsf{DT}}\ :\ \textit{for all}\ k \in \{1,2\}\right\},\ \ \left\{r_{\mathsf{RC}},\ r_{\mathsf{XT}},\ r_{\mathsf{CX}}\right\},$$

such that jointly they satisfy the following five groups of linear conditions:

• Group 1, termed the *time-sharing conditions*, has 3 inequalities:

$$1 > t_s + t_r, \tag{4.27}$$

$$t_s \geq \sum_{k\in\{1,2\}} \Big(s^k_{\mathsf{UC}} + s^k_{\mathsf{PM1}} + s^k_{\mathsf{PM2}} + s^k_{\mathsf{RC}} + s^k_{\mathsf{DX}} + s^{(k)}_{\mathsf{DX}} \Big) + \sum_{l=1}^{8} s_{\mathsf{CX};l}, \tag{4.28}$$

$$t_r \geq \sum_{k\in\{1,2\}} \Big(r^k_{\mathsf{UC}} + r^{(k)}_{\mathsf{DT}} + r^{[k]}_{\mathsf{DT}} \Big) + r_{\mathsf{RC}} + r_{\mathsf{XT}} + r_{\mathsf{CX}}. \tag{4.29}$$

• Group 2, termed the *packets-originating condition*, has 2 inequalities: Consider any $i, j \in \{1, 2\}$ satisfying $i \neq j$. For each (i, j) pair (out of the two choices $(1, 2)$ and $(2, 1)$),

$$R_i \geq \big(s^i_{\mathsf{UC}} + s^i_{\mathsf{PM1}} \big) \cdot p_s(d_i, d_j, r), \tag{E}$$

• Group 3, termed the *packets-mixing condition*, has 4 inequalities: For each (i, j) pair,

$$\big(s^i_{\mathsf{UC}} + s^i_{\mathsf{PM1}} \big) \cdot p_{s\to\overline{d_i d_j} r} \geq (s^j_{\mathsf{PM1}} + s^i_{\mathsf{PM2}}) \cdot p_s(d_i, d_j) + r^i_{\mathsf{UC}} \cdot p_r(d_i, d_j), \tag{A}$$

$$s^i_{\mathsf{PM1}} \cdot p_{s\to\overline{d_i d_j}\,\overline{r}} \geq s^i_{\mathsf{RC}} \cdot p_s(d_i, d_j, r), \tag{B}$$

and the following one inequality:

$$\begin{aligned} &s^1_{\mathsf{PM1}} \cdot p_s(d_1, d_2 r) + s^2_{\mathsf{PM1}} \cdot p_s(d_2, d_1 r) + s^1_{\mathsf{PM2}} \cdot p_s(\overline{d_1} d_2) + \\ &s^2_{\mathsf{PM2}} \cdot p_s(d_1 \overline{d_2}) + \big(s^1_{\mathsf{RC}} + s^2_{\mathsf{RC}} \big) \cdot p_{s\to\overline{d_1 d_2} r} \geq r_{\mathsf{RC}} \cdot p_r(d_1, d_2). \end{aligned} \tag{M}$$

• Group 4, termed the *classic XOR condition by source only*, has 4 inequalities: For each (i,j) pair,

$$\begin{aligned}\left(s^i_{\mathsf{UC}}+s^i_{\mathsf{RC}}\right)p_{s\to\overline{d_i}d_j\overline{r}} &\geq \left(s^j_{\mathsf{PM2}}+s^i_{\mathsf{DX}}\right)\cdot p_s(d_i,r)+\\ &\quad(s_{\mathsf{CX};1}+s_{\mathsf{CX};1+i})\cdot\ p_s(d_i,r)+s_{\mathsf{CX};4+i}\cdot p_s(d_i,r),\end{aligned}\tag{S}$$

$$\begin{aligned}s^j_{\mathsf{RC}}\cdot p_{s\to\overline{d_i}d_j\overline{r}} &\geq s^{(i)}_{\mathsf{DX}}\cdot p_s(d_i,r)+r^{(i)}_{\mathsf{DT}}\cdot p_r(d_i,d_j)+\\ &\quad(s_{\mathsf{CX};1+j}+s_{\mathsf{CX};4})\cdot\ p_s(d_i,r)+s_{\mathsf{CX};6+i}\cdot p_s(d_i,r).\end{aligned}\tag{T}$$

• Group 5, termed the *XOR condition*, has 3 inequalities:

$$\sum_{l=1}^{4} s_{\mathsf{CX};l}\cdot p_{s\to\overline{d_1d_2}r}\geq r_{\mathsf{XT}}\cdot p_r(d_1,d_2),\tag{X0}$$

and for each (i,j) pair,

$$\begin{aligned}&s^j_{\mathsf{PM2}}\cdot p_s(d_id_j,\overline{d_i}r)+\left(s^i_{\mathsf{UC}}+s^i_{\mathsf{RC}}+s^j_{\mathsf{RC}}+\sum_{l=1}^{4}s_{\mathsf{CX};l}\right)\cdot p_{s\to\overline{d_i}d_jr}\\ &+\left(s_{\mathsf{CX};4+i}+s_{\mathsf{CX};6+i}+s^i_{\mathsf{DX}}+s^{(i)}_{\mathsf{DX}}\right)\cdot p_s(\overline{d_i}r)\\ &+\left(r^i_{\mathsf{UC}}+r_{\mathsf{RC}}+r^{(i)}_{\mathsf{DT}}+r_{\mathsf{XT}}\right)\cdot p_{r\to\overline{d_i}d_j}\\ &\geq(s_{\mathsf{CX};7-i}+s_{\mathsf{CX};9-i})\cdot p_s(d_i)+\left(r_{\mathsf{CX}}+r^{[i]}_{\mathsf{DT}}\right)\cdot p_r(d_i).\end{aligned}\tag{X}$$

• Group 6, termed the *decodability condition*, has 2 inequalities: For each (i,j) pair,

$$\begin{aligned}&\left(s^i_{\mathsf{UC}}+s^j_{\mathsf{PM2}}+\sum_{k\in\{1,2\}}s^k_{\mathsf{RC}}+\sum_{l=1}^{8}s_{\mathsf{CX};l}+s^i_{\mathsf{DX}}+s^{(i)}_{\mathsf{DX}}\right)\cdot p_s(d_i)+\\ &\left(r^i_{\mathsf{UC}}+r_{\mathsf{RC}}+r_{\mathsf{XT}}+r_{\mathsf{CX}}+r^{(i)}_{\mathsf{DT}}+r^{[i]}_{\mathsf{DT}}\right)\cdot p_r(d_i)\geq R_i.\end{aligned}\tag{D}$$

The intuition is as follows. The proposed LNC inner bound is derived based on the ideas of describing the packet movements in a queueing network, where movements are governed by LNC operations. Each variable (except t-variables for time-sharing) in Proposition 4.2.1 is associated with a specific LNC operation. Note that s-variables are associated with LNC operations performed by the source s, while r-variables are associated with LNC operations performed by the relay r. The inequalities (E) to (D) then describe the queueing process for packet movements, where the LHS and the RHS of each inequality implies the packet insertion and removal conditions, respectively, of the corresponding queue by the related LNC operations. For notational convenience, we define the following queue notations associated with these 14 inequalities (E) to (D):

Table 4.1: Queue denominations for the inequalities (E) to (D)

(E1): Q^1_ϕ	(B1): $Q^{m\|2}_{\{d_2\}\|\{r\}}$	(S1): $Q^1_{\{d_2\}}$	(X0): $Q^{m_{\mathsf{CX}}}_{\{r\}}$
(E2): Q^2_ϕ	(B2): $Q^{m\|1}_{\{d_1\}\|\{r\}}$	(T1): $Q^{(1)\|1}_{\{d_2\}\|\{r\}}$	(X1): $Q^{[1]}_{\{rd_2\}}$
(A1): $Q^1_{\{r\}}$	(M): Q_{mix}	(S2): $Q^2_{\{d_1\}}$	(X2): $Q^{[2]}_{\{rd_1\}}$
(A2): $Q^2_{\{r\}}$		(T2): $Q^{(2)\|2}_{\{d_1\}\|\{r\}}$	(D1): Q^1_{dec}
			(D2): Q^2_{dec}

where we use the index-after-reference to distinguish the session (i.e., flow) of focus of an inequality. For example, (E1) and (E2) are to denote the inequality (E) when $(i, j) = (1, 2)$ and $(i, j) = (2, 1)$, respectively.

For example, suppose that $\mathbf{W}_1 = (X_1, \cdots, X_{nR_1})$ packets and $\mathbf{W}_2 = (Y_1, \cdots, Y_{nR_2})$ packets are initially stored in queues Q^1_ϕ and Q^2_ϕ, respectively, at source s. The superscript $k \in \{1, 2\}$ indicates that the queue is for the packets intended to destination d_k. The subscript indicates that those packets have not been heard by any of $\{d_1, d_2, r\}$. The LNC operation corresponding to the variable s^1_{UC} (resp. s^2_{UC}) is to send a session-1 packet X_i (resp. a session-2 packet Y_j) uncodedly. Then the inequality (E1) (resp. (E2)) implies that whenever it is received by at least one of $\{d_1, d_2, r\}$, this packet is removed from the queue of Q^1_ϕ (resp. Q^2_ϕ).

Depending on the reception status, the packet will be either remained in the same queue or moved to another queue. For example, the use of s^1_{UC} (sending $X_i \in \mathbf{W}_1$ uncodedly from source) will take X_i from Q^1_ϕ and insert it into Q^1_{dec} when the reception status is $p_s(d_1)$, i.e., when the intended destination d_1 correctly receives it. Similarly, when the reception status is $p_{s \to \overline{d_1 d_2} r}$, this packet will be inserted to the queue $Q^1_{\{r\}}$

according to the packet movement rule of (A1); inserted to $Q^1_{\{d_2\}}$ when $p_{s\to\overline{d_1}d_2\overline{r}}$ by (S1); and inserted to $Q^{[1]}_{\{rd_2\}}$ when $p_{s\to\overline{d_1}d_2r}$ by (X1). Obviously when $p_{s\to\overline{d_1d_2r}}$, since any node in $\{d_1, d_2, r\}$ has not received at all, the packet X_i simply remains in Q^1_{ϕ}.

Fig. 4.1 illustrates the queueing network represented by Proposition 4.2.1. The detailed descriptions of the proposed LNC operations and the corresponding packet movement process following the inequalities in Proposition 4.2.1 are relegated to Appendix H.1.

4.2.1 The Properties of Queues and The Correctness Proof

Each queue in the queueing network, see Fig. 4.1, is carefully designed to store packets in a specific format such that the queue itself can represent a certain case to be beneficial. In this subsection, we highlight the properties of the queues, which will be later used to prove the correctness of our achievability scheme of Proposition 4.2.1.

To that end, we first describe the properties of Q^1_{ϕ}, Q^1_{dec}, $Q^1_{\{r\}}$, and $Q^1_{\{d_2\}}$ since their purpose is clear in the sense that the queue collects pure session-1 packets (indicated by the superscript), but heard only by the nodes (in the subscript $\{\cdot\}$) or correctly decoded by the desired destination d_1 (by the subscript dec). After that, we describe the property of Q_{mix}, and then explain $Q^{m|2}_{\{d_2\}|\{r\}}$, $Q^{(1)|1}_{\{d_2\}|\{r\}}$, and $Q^{[1]}_{\{rd_2\}}$ focusing on the queues related to the session-1 packets. For example, $Q^{m|2}_{\{d_2\}|\{r\}}$ implies the queue related to a session-1 packet that is mixed with a session-2 packet, where such mixture is known by d_2 but the session-2 packet is known by r as well. The properties of the queues related to the session-2 packets, i.e., Q^2_{ϕ}, Q^2_{dec}, $Q^2_{\{r\}}$, $Q^2_{\{d_1\}}$, $Q^{m|1}_{\{d_1\}|\{r\}}$, $Q^{(2)|2}_{\{d_1\}|\{r\}}$, and $Q^{[2]}_{\{rd_1\}}$, will be symmetrically explained by simultaneously swapping (a) session-1 and session-2 in the superscript; (b) X and Y; (c) i and j; and (d) d_1 and d_2, if applicable. The property of $Q^{m\mathsf{cx}}_{\{r\}}$ will be followed at last.

To help aid the explanations, we also define for each node in $\{d_1, d_2, r\}$, the *reception list* $\mathsf{RL}_{\{d_1\}}$, $\mathsf{RL}_{\{d_2\}}$, and $\mathsf{RL}_{\{r\}}$, respectively, that records how the received packet is constituted. The reception list is a binary matrix of its column size fixed to $n(R_1+R_2)$

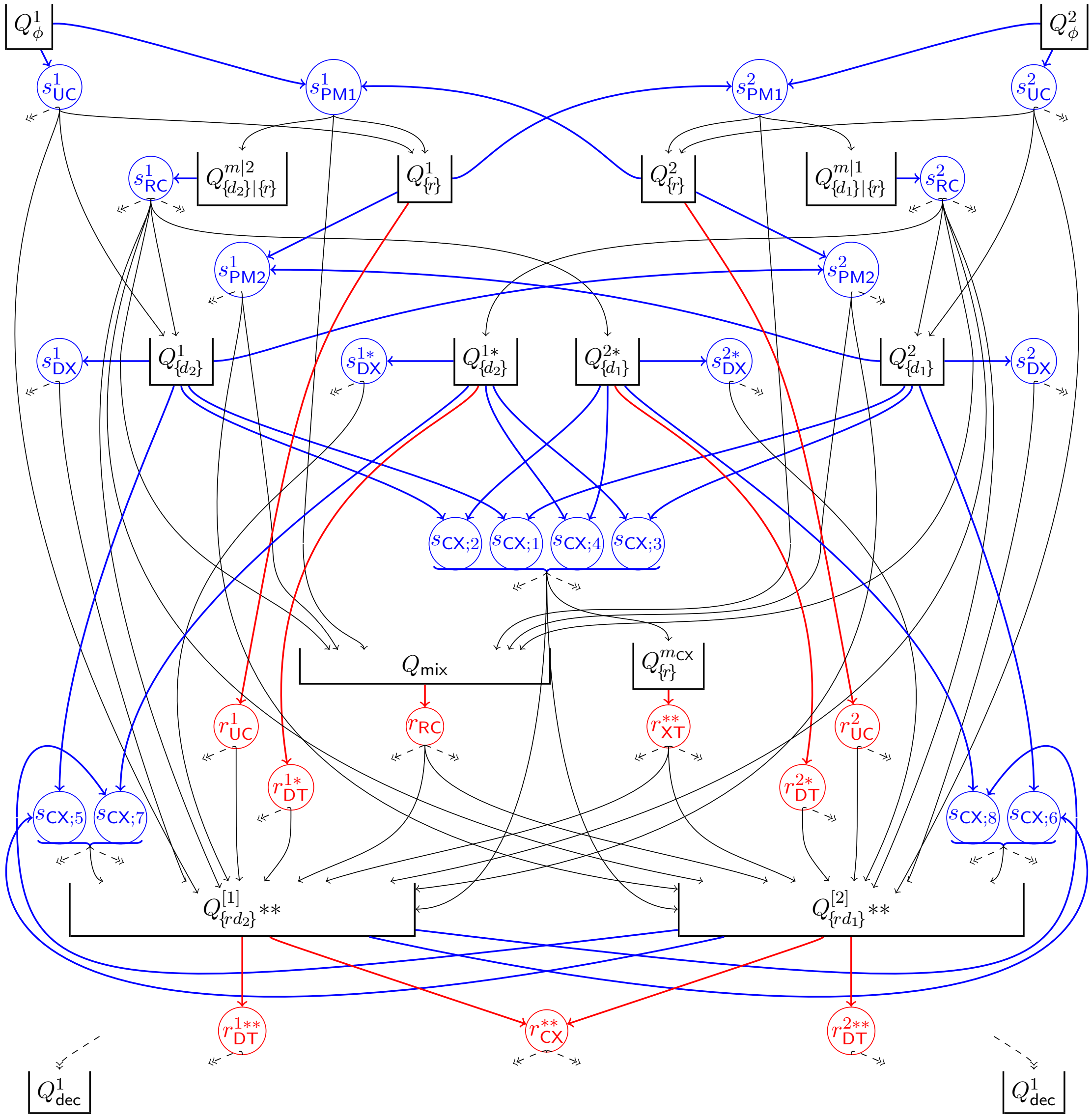


Fig. 4.1. Illustrations of The Queueing Network described by the inequalities (E) to (D) in Proposition 4.2.1. The upper-side-open rectangle represents the queue, and the circle represents LNC encoding operation, where the blue means the encoding by the source s and the red means the encoding by the relay r. The black outgoing arrows from a LNC operation (or from a set of LNC operations grouped by a curly brace) represent the packet movements process depending on the reception status, where the southwest and southeast dashed arrows are especially for into Q^1_{dec} and into Q^2_{dec}, respectively.

but its row size being the number of received packets and thus variable (increasing) over the course of total time slots. For example, suppose that d_1 has received a pure session-1 packet X_1, a self-mixture $[X_1 + X_2]$, and a cross-mixture $[X_3 + Y_1]$. Then $\mathsf{RL}_{\{d_1\}}$ will be

$$\begin{array}{cc} \overbrace{\begin{array}{l} 1\,0 \cdots \cdots \cdots \cdots \\ 1\,1\,0\,\cdots\cdots\cdots\cdots \\ 0\,0\,1\,0 \cdots \cdots \cdots \end{array}}^{nR_1} & \overbrace{\begin{array}{l} 0\,0 \cdots \cdots \cdots \cdots \\ 0\,0 \cdots \cdots \cdots \cdots \\ 1\,0 \cdots \cdots \cdots \cdots \end{array}}^{nR_2} \end{array}$$

such that the first row vector represents the pure X_1 received, the second row vector represents the mixture $[X_1 + X_2]$ received, and the third row vector represents the mixture $[X_3 + Y_1]$ received, all in a binary format. Namely, whenever a node receives a packet, whether such packet is pure or not, a new $n(R_1 + R_2)$-dimensional row vector is inserted into the reception list by marking the corresponding entries of X_i or Y_j as flagged ("1") or not flagged ("0") accordingly. From the previous example, $[X_1 + X_2]$ in the reception list $\mathsf{RL}_{\{d_1\}}$ means that the list contains a $n(R_1 + R_2)$-dimensional row vector of exactly $\{1, 1, 0, \cdots, 0\}$. We then say that a pure packet is *not flagged* in the reception list, if the column of the corresponding entry contains all zeros. From the previous example, the pure session-2 packet Y_2 is not flagged in $\mathsf{RL}_{\{d_1\}}$, meaning that d_1 has neither received Y_2 nor any mixture involving this Y_2. Note that "not flagged" is a stronger definition than "unknown". From the previous example, the pure session-1 packet X_3 is unknown to d_1 but still flagged in $\mathsf{RL}_{\{d_1\}}$ as d_1 has received the mixture $[X_3 + Y_1]$ involving this X_3. Another example is the pure X_2 that is flagged in $\mathsf{RL}_{\{d_1\}}$ but d_1 knows this X_2 as it can use the received X_1 and the mixture $[X_1 + X_2]$ to extract X_2. We sometimes abuse the reception list notation to denote the collective reception list by RL_T for some non-empty subset $T \subseteq \{d_1, d_2, r\}$. For example, $\mathsf{RL}_{\{d_1,d_2,r\}}$ implies the vertical concatenation of all $\mathsf{RL}_{\{d_1\}}$, $\mathsf{RL}_{\{d_2\}}$, and $\mathsf{RL}_{\{r\}}$.

We now describe the properties of the queues.

- Q^1_ϕ: Every packet in this queue is *of a pure session*-1 and *unknown* to any of $\{d_1, d_2, r\}$, even *not flagged* in $\mathsf{RL}_{\{d_1,d_2,r\}}$. Initially, this queue contains all the session-1 packets $\mathbf{W}_1$, and will be empty in the end.

• Q^1_{dec}: Every packet in this queue is *of a pure session*-1 and *known* to d_1. Initially, this queue is empty but will contain all the session-1 packets $\mathbf{W}_1$ in the end.

• $Q^1_{\{r\}}$: Every packet in this queue is *of a pure session*-1 and *known* by r but *unknown* to any of $\{d_1, d_2\}$, even *not flagged* in $\mathsf{RL}_{\{d_1,d_2\}}$.

• $Q^1_{\{d_2\}}$: Every packet in this queue is *of a pure session*-1 and *known* by d_2 but *unknown* to any of $\{d_1, r\}$, even *not flagged* in $\mathsf{RL}_{\{d_1,r\}}$.

• Q_{mix}: Every packet in this queue is *of a linear sum* $[X_i + Y_j]$ from a session-1 packet X_i and a session-2 packet Y_j such that at least one of the following conditions hold:

(a) $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_1\}}$; X_i is *unknown* to d_1; and Y_j is *known* by r but *unknown* to d_2.

(b) $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_2\}}$; X_i is *known* by r but *unknown* to d_1; and Y_j is *unknown* to d_2.

The detailed clarifications are as follows. For a NC designer, one important consideration is to generate as many "all-happy" scenarios as possible in an efficient manner so that single transmission benefits both destination simultaneously. One famous example is the *classic XOR* operation that a sender transmits a linear sum $[X_i + Y_j]$ when a session-1 packet X_i is not yet delivered to d_1 but overheard by d_2 and a session-2 packet Y_j is not yet delivered to d_2 but overheard by d_1. Namely, the source s can perform such classic butterfly-style operation of sending the linear mixture $[X_i + Y_j]$ whenever such pair of X_i and Y_j is available. Similarly, Q_{mix} represents such an "all-happy" scenario that the relay r can benefit both destinations simultaneously by sending either X_i or Y_j. For example, suppose that the source s has transmitted a packet mixture $[X_i + Y_j]$ and it is received by d_2 only. And assume that r already knows the individual X_i and Y_j but X_i is unknown to d_1, see Fig. 4.2(a). This example scenario falls into the second condition of Q_{mix} above. Then sending X_i from the relay r simultaneously enables d_1 to receive the desired X_i and d_2 to decode the desired Y_j by subtracting the received X_i from the known $[X_i + Y_j]$. Q_{mix} collects such all-happy mixtures $[X_i + Y_j]$ that has been received by either d_1 or d_2 or both. In the same scenario, however, notice that r cannot benefit both destinations simultaneously, if r sends Y_j, instead of X_i. As a result, we use the notation $[X_i + Y_j]\!:\!W$ to denote the specific packet W (known by r) that r can send to benefit both destinations. In this second condition scenario of Fig. 4.2(a), Q_{mix} is storing $[X_i + Y_j]\!:\!X_i$.

• $Q^{m|2}_{\{d_2\}|\{r\}}$: Every packet in this queue is *of a linear sum* $[X_i + Y_j]$ from a session-1 packet X_i and a session-2 packet Y_j such that they jointly satisfy the following conditions simultaneously.

(a) $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_2\}}$.

(b) X_i is *unknown* to any of $\{d_1, d_2, r\}$, even *not flagged* in $\mathsf{RL}_{\{d_1,r\}}$.

(c) Y_j is *known* by r but *unknown* to any of $\{d_1, d_2\}$, even *not flagged* in $\mathsf{RL}_{\{d_1\}}$.

The scenario is the same as in Fig. 4.2(a) when r not having X_i. In this scenario, we have observed that r cannot benefit both destinations by sending the known Y_j. $Q^{m|2}_{\{d_2\}|\{r\}}$ collects such unpromising $[X_i + Y_j]$ mixtures.

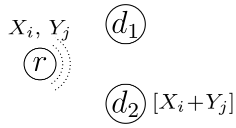
• $Q^{(1)|1}_{\{d_2\}|\{r\}}$: Every packet in this queue is *of a pure session*-2 packet Y_i such that there exists a pure session-1 packet X_i that Y_i is information equivalent to, and they jointly satisfy the following conditions simultaneously.

(a) $[X_i + Y_i]$ is in $\mathsf{RL}_{\{d_1\}}$.

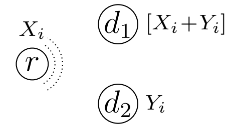
(b) X_i is *known* by r but *unknown* to any of $\{d_1, d_2\}$.

(c) Y_i is *known* by d_2 (i.e. already in Q^2_{dec}) but *unknown* to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{r\}}$.

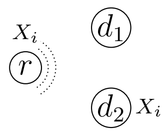
The concrete explanations are as follows. The main purpose of this queue is basically the same as $Q^1_{\{d_2\}}$, i.e., to store session-1 packet overheard by d_2, so as to be used by the source s for the classic XOR operation with the session-2 counterparts (e.g., any packet in $Q^2_{\{d_1\}}$). Notice that any $X_i \in Q^1_{\{d_2\}}$ is unknown to r and thus r cannot generate the corresponding linear mixture with the counterpart. However, because



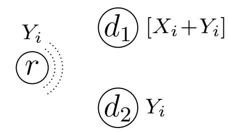
(a) Example scenario for Q_{mix}



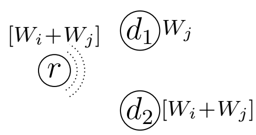
(b) Scenario for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$
It must be $Y_i \in Q^2_{\mathsf{dec}}$



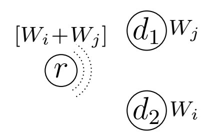
(c) Case 1: $X_i \in Q^{[1]}_{\{rd_2\}}$



(d) Case 2: $Y_i \in Q^{[1]}_{\{rd_2\}}$
It must be $Y_i \in Q^2_{\mathsf{dec}}$



(e) Case 3: $[W_i+W_j] \in Q^{[1]}_{\{rd_2\}}$



(f) Scenario for $[W_i+W_j] \in Q^{m_{\mathsf{CX}}}_{\{r\}}$

Fig. 4.2. Illustrations of Scenarios of the Queues.

X_i is unknown to the relay, r cannot even naively deliver X_i to the desired destination d_1. On the other hand, the queue $Q^{(1)|1}_{\{d_2\}|\{r\}}$ here not only allows s to perform the classic XOR operation but also admits naive delivery from r. To that end, consider the scenario in Fig. 4.2(b). Here, d_1 has received a linear sum $[X_i + Y_i]$. Whenever d_1 receives Y_i (session-2 packet), d_1 can use Y_i and the known $[X_i + Y_i]$ to decode the desired X_i. This Y_i is also known by d_2 (i.e., already in Q^2_{dec}), meaning that Y_i is no more different than a session-1 packet overheard by d_2 but not yet delivered to d_1. Namely, such Y_i can be treated as *information equivalent to* X_i. That is, using this session-2 packet Y_i for the sake of session-1 does not incur any information duplicity because Y_i is already received by the desired destination d_2.[2] For shorthand, we denote such Y_i as $Y_i \equiv X_i$. As a result, the source s can use this Y_i as for session-1

[2]This means that d_2 does not require Y_i any more, and thus s or r can freely use this Y_i in the network to represent not-yet-decoded X_i instead.

when performing the classic XOR operation with a session-2 counterpart. Moreover, r also knows the pure X_i and thus relay can perform naive delivery for d_1 as well.

- $Q^{[1]}_{\{rd_2\}}$: Every packet in this queue is *of either a pure or a mixed* packet $\overline{W}$ satisfying the following conditions simultaneously.

  (a) $\overline{W}$ is *known* by both r and d_2 but *unknown* to d_1.

  (b) d_1 can extract a desired session-1 packet when $\overline{W}$ is further received.

Specifically, there are three possible cases based on how the packet $\overline{W} \in Q^{[1]}_{\{rd_2\}}$ is constituted:

Case 1: $\overline{W}$ is a pure session-1 packet X_i. That is, X_i is known by both r and d_2 but unknown to d_1 as in Fig. 4.2(c). Obviously, d_1 acquires this new X_i when it is further delivered to d_1.

Case 2: $\overline{W}$ is a pure session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$. That is, Y_i is already received by d_2 and known by r as well but unknown to d_1. For such Y_i, as similar to the discussions of $Q^{(1)|1}_{\{d_2\}|\{r\}}$, there exists a session-1 packet X_i still unknown to d_1 where $X_i \equiv Y_i$, and their mixture $[X_i + Y_i]$ is in $\mathsf{RL}_{\{d_1\}}$, see Fig. 4.2(d). One can easily see that when d_1 further receives this Y_i, d_1 can use the received Y_i and the known $[X_i + Y_i]$ to decode the desired X_i.

Case 3: $\overline{W}$ is a mixed packet of the form $[W_i + W_j]$ where W_i and W_j are pure but generic that can be either a session-1 or a session-2 packet. That is, the linear sum $[W_i + W_j]$ is known by both r and d_2 but unknown to d_1. In this case, W_i is still unknown to d_1 but W_j is already received by d_1 so that whenever $[W_i + W_j]$ is delivered to d_1, W_i can further be decoded. See Fig. 4.2(e) for details. Specifically, there are two possible subcases depending on whether W_i is of a pure session-1 or of a pure session-2:

  – W_i is a session-1 packet X_i. As discussed above, X_i is unknown to d_1 and it is obvious that d_1 can decode the desired X_i whenever $[W_i + W_j]$ is delivered to d_1.

– W_i is a session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$. In this subcase, there exists a session-1 packet X_i (other than W_j in the above Case 3 discussions) still unknown to d_1 where $X_i \equiv Y_i$. Moreover, $[X_i + Y_i]$ is already in $\mathsf{RL}_{\{d_1\}}$. As a result, d_1 can decode the desired X_i whenever $[W_i + W_j]$ is delivered to d_1.

The concrete explanations are as follows. The main purpose of this queue is basically the same as $Q^{(1)|1}_{\{d_2\}|\{r\}}$ but the queue $Q^{[1]}_{\{rd_2\}}$ here allows not only the source s but also the relay r to perform the classic XOR operation. As elaborated above, we have three possible cases depending on the form of the packet $\overline{W} \in Q^{[1]}_{\{rd_2\}}$. Specifically, either a pure session-1 packet $X_i \notin Q^1_{\mathsf{dec}}$ (Case 1) or a pure session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$ (Case 2) or a mixture $[W_i + W_j]$ (Case 3) will be used when either s or r performs the classic XOR operation with a session-2 counterpart. For example, suppose that we have a packet $X \in Q^{[2]}_{\{rd_1\}}$ (Case 2) as a session-2 counterpart. Symmetrically following the Case 2 scenario of $Q^{[1]}_{\{rd_2\}}$ in Fig. 4.2(d), we know that X has been received by both r and d_1. There also exists a session-2 packet Y still unknown to d_2 where $Y \equiv X$, of which their mixture $[X + Y]$ is already in $\mathsf{RL}_{\{d_2\}}$. For this session-2 counterpart X, consider any packet $\overline{W}$ in $Q^{[1]}_{\{rd_2\}}$. Obviously, the relay r knows both $\overline{W}$ and X by assumption. As a result, either s or r can send their linear sum $[\overline{W} + X]$ as per the classic pairwise XOR operation. Since d_1 already knows X by assumption, such mixture $[\overline{W} + X]$, when received by d_1, can be used to decode $\overline{W}$ and further decode a desired session-1 packet as discussed above. Moreover, if d_2 receives $[\overline{W} + X]$, then d_2 can use the known $\overline{W}$ to extract X and further decode the desired Y since $[X + Y]$ is already in $\mathsf{RL}_{\{d_2\}}$ by assumption.

- $Q^{m\mathsf{cx}}_{\{r\}}$: Every packet in this queue is *of a linear sum* $[W_i + W_j]$ that satisfies the following conditions simultaneously.

(a) $[W_i + W_j]$ is in $\mathsf{RL}_{\{r\}}$.

(b) W_i is *known* by d_2 but *unknown* to any of $\{d_1, r\}$.

(c) W_j is *known* by d_1 but *unknown* to any of $\{d_2, r\}$.

where W_i and W_j are pure but generic that can be either a session-1 or a session-2 packet. Specifically, there are four possible cases based on the types of W_i and W_j packets:

Case 1: W_i is a pure session-1 packet X_i and W_j is a pure session-2 packet Y_j.

Case 2: W_i is a pure session-1 packet X_i and W_j is a pure session-1 packet $X_j \in Q^1_{\mathsf{dec}}$. For the latter X_j packet, as similar to the discussions of $Q^{(1)|1}_{\{d_2\}|\{r\}}$, there also exists a pure session-2 packet Y_j still unknown to d_2 where $Y_j \equiv X_j$ and their mixture $[X_j + Y_j]$ is already in $\mathsf{RL}_{\{d_2\}}$. As a result, later when d_2 decodes this X_j, d_2 can use X_j and the known $[X_j + Y_j]$ to decode the desired Y_j.

Case 3: W_i is a pure session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$ and W_j is a pure session-2 packet Y_j. For the former Y_i packet, there also exists a pure session-1 packet X_i still unknown to d_1 where $X_i \equiv Y_i$ and $[X_i + Y_i]$ is already in $\mathsf{RL}_{\{d_1\}}$. As a result, later when d_1 decodes this Y_i, d_1 can use Y_i and the known $[X_i + Y_i]$ to decode the desired X_i.

Case 3: W_i is a pure session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$ and W_j is a pure session-1 packet $X_j \in Q^1_{\mathsf{dec}}$. For the former Y_i and the latter X_j packets, the discussions follow the Case 3 and Case 2 above, respectively.

The concrete explanations are as follows. This queue represents the "all-happy" scenario as similar to the butterfly-style operation by the relay r, i.e., sending a linear mixture $[W_i + W_j]$ using W_i heard by d_2 and W_j heard by d_1. Originally, r must have known both individuals packets W_i and W_j to generate their linear sum. However, the sender in fact does not need to know both individuals to perform this classic XOR operation. The sender can still do the same operation even though it knows the linear sum $[W_i + W_j]$ only. This possibility only applies to the relay r as all the messages including both individual packets are originated from the source s. As a result, this queue represents such scenario that the relay r only knows the linear sum instead of individuals, as in Fig. 4.2(f). More precisely, Cases 1 to 4 happen when

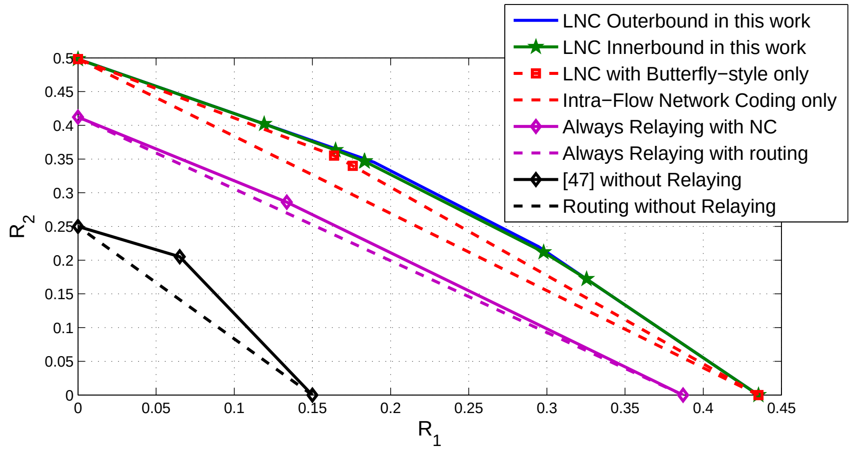


Fig. 4.3. Comparison of LNC regions with different achievable rates

the source s performed one of four classic XOR operations $s_{\mathsf{CX};1}$ to $s_{\mathsf{CX};4}$, respectively, and the corresponding linear sum is received only by r, see Appendix H.1 for details.

Based on the properties of queues, we now describe the correctness of Proposition 4.2.1, our LNC inner bound. To that end, we first investigate all the LNC operations involved in Proposition 4.2.1 and prove the "Queue Invariance", i.e., the queue properties explained above *remains invariant regardless of an LNC operation chosen*. Such long and tedious investigations are relegated to Appendix H.1. Then, the decodability condition (D), jointly with the Queue Invariance, imply that Q^1_{dec} and Q^2_{dec} will contain at least nR_1 and nR_2 number of pure session-1 and pure session-2 packets, respectively, in the end. This further means that, given a rate vector (R_1, R_2), any t-, s-, and r-variables that satisfy the inequalities (E) to (D) in Proposition 4.2.1 will be achievable. The correctness proof of Proposition 4.2.1 is thus complete.

For readability, we also describe for each queue, the associated LNC operations that moves packet into and takes packets out of, see the following Table 4.2.

4.3 Numerical Evaluation

Consider a smart repeater network with marginal channel success probabilities: (a) s-PEC: $p_s(d_1) = 0.15$, $p_s(d_2) = 0.25$, and $p_s(r) = 0.8$; and (b) r-PEC: $p_r(d_1) = 0.75$

Table 4.2: Summary of the associated LNC operations for queues in Fig. 4.2

LNC operations $\mapsto$	**Queue**	$\mapsto$ **LNC operations**
	Q^1_{ϕ}	$s^1_{\mathsf{UC}},\ s^1_{\mathsf{PM1}}$
$s^1_{\mathsf{UC}},\ s^1_{\mathsf{PM1}}$	$Q^1_{\{r\}}$	$s^2_{\mathsf{PM1}},\ s^1_{\mathsf{PM2}},\ r^1_{\mathsf{UC}}$
s^1_{PM1}	$Q^{m\|2}_{\{d_2\}\|\{r\}}$	s^1_{RC}
$s^1_{\mathsf{UC}},\ s^1_{\mathsf{RC}}$	$Q^1_{\{d_2\}}$	$s^2_{\mathsf{PM2}},\ s^1_{\mathsf{DX}}$ $s_{\mathsf{CX};1},\ s_{\mathsf{CX};2},\ s_{\mathsf{CX};5}$
s^2_{RC}	$Q^{(1)\|1}_{\{d_2\}\|\{r\}}$	$s^{(1)}_{\mathsf{DX}},\ s_{\mathsf{CX};3}$ $s_{\mathsf{CX};4},\ s_{\mathsf{CX};7},\ r^{(1)}_{\mathsf{DT}}$
$s^1_{\mathsf{UC}},\ s^2_{\mathsf{PM2}},\ s^1_{\mathsf{RC}},\ s^1_{\mathsf{DX}}$ $s_{\mathsf{CX};5},\ r^1_{\mathsf{UC}},\ r^{(1)}_{\mathsf{DT}},\ r_{\mathsf{RC}}$	$Q^{[1]}_{\{rd_2\}}$ (Case 1)	$s_{\mathsf{CX};6}, s_{\mathsf{CX};8}$ $r^{[1]}_{\mathsf{DT}}, r_{\mathsf{CX}}$
$s^2_{\mathsf{PM2}},\ s^2_{\mathsf{RC}},\ s^{(1)}_{\mathsf{DX}}$ $s_{\mathsf{CX};7},\ r_{\mathsf{RC}}$	$Q^{[1]}_{\{rd_2\}}$ (Case 2)	
$s_{\mathsf{CX};1},\ s_{\mathsf{CX};2}$ $s_{\mathsf{CX};3},\ s_{\mathsf{CX};4},\ r_{\mathsf{XT}}$	$Q^{[1]}_{\{rd_2\}}$ (Case 3)	
$s^1_{\mathsf{UC}},\ s^1_{\mathsf{PM2}},\ s^1_{\mathsf{RC}},\ s^2_{\mathsf{RC}}$ $s^1_{\mathsf{DX}},\ s^{(1)}_{\mathsf{DX}},\ \{s_{\mathsf{CX};1}$ to $s_{\mathsf{CX};8}\}$ $r^1_{\mathsf{UC}},\ r^{(1)}_{\mathsf{DT}},\ r^{[1]}_{\mathsf{DT}}$ $r_{\mathsf{RC}},\ r_{\mathsf{XT}},\ r_{\mathsf{CX}}$	Q^1_{dec}	
$s^1_{\mathsf{PM1}},\ s^2_{\mathsf{PM1}},\ s^1_{\mathsf{PM2}},\ s^2_{\mathsf{PM2}}$ $s^1_{\mathsf{RC}},\ s^2_{\mathsf{RC}}$	Q_{mix}	r_{RC}
$s_{\mathsf{CX};1},\ s_{\mathsf{CX};2},\ s_{\mathsf{CX};3},\ s_{\mathsf{CX};4}$	$Q^{m_{\mathsf{CX}}}_{\{r\}}$	r_{XT}
	Q^2_{ϕ}	$s^2_{\mathsf{UC}},\ s^2_{\mathsf{PM1}}$
$s^2_{\mathsf{UC}},\ s^2_{\mathsf{PM1}}$	$Q^2_{\{r\}}$	$s^1_{\mathsf{PM1}},\ s^2_{\mathsf{PM2}},\ r^2_{\mathsf{UC}}$
s^2_{PM1}	$Q^{m\|1}_{\{d_1\}\|\{r\}}$	s^2_{RC}
$s^2_{\mathsf{UC}},\ s^2_{\mathsf{RC}}$	$Q^2_{\{d_1\}}$	$s^1_{\mathsf{PM2}},\ s^2_{\mathsf{DX}}$ $s_{\mathsf{CX};1},\ s_{\mathsf{CX};3},\ s_{\mathsf{CX};6}$
s^1_{RC}	$Q^{(2)\|2}_{\{d_1\}\|\{r\}}$	$s^{(2)}_{\mathsf{DX}},\ s_{\mathsf{CX};2}$ $s_{\mathsf{CX};4},\ s_{\mathsf{CX};8},\ r^{(2)}_{\mathsf{DT}}$
$s^2_{\mathsf{UC}},\ s^1_{\mathsf{PM2}},\ s^2_{\mathsf{RC}},\ s^2_{\mathsf{DX}}$ $s_{\mathsf{CX};6},\ r^2_{\mathsf{UC}},\ r^{(2)}_{\mathsf{DT}},\ r_{\mathsf{RC}}$	$Q^{[2]}_{\{rd_1\}}$ (Case 1)	$s_{\mathsf{CX};5}, s_{\mathsf{CX};7}$ $r^{[2]}_{\mathsf{DT}}, r_{\mathsf{CX}}$
$s^1_{\mathsf{PM2}},\ s^1_{\mathsf{RC}},\ s^{(2)}_{\mathsf{DX}}$ $s_{\mathsf{CX};8},\ r_{\mathsf{RC}}$	$Q^{[2]}_{\{rd_1\}}$ (Case 2)	
$s_{\mathsf{CX};1},\ s_{\mathsf{CX};2}$ $s_{\mathsf{CX};3},\ s_{\mathsf{CX};4},\ r_{\mathsf{XT}}$	$Q^{[2]}_{\{rd_1\}}$ (Case 3)	
$s^2_{\mathsf{UC}},\ s^2_{\mathsf{PM2}},\ s^1_{\mathsf{RC}},\ s^2_{\mathsf{RC}}$ $s^2_{\mathsf{DX}},\ s^{(2)}_{\mathsf{DX}},\ \{s_{\mathsf{CX};1}$ to $s_{\mathsf{CX};8}\}$ $r^2_{\mathsf{UC}},\ r^{(2)}_{\mathsf{DT}},\ r^{[2]}_{\mathsf{DT}}$ $r_{\mathsf{RC}},\ r_{\mathsf{XT}},\ r_{\mathsf{CX}}$	Q^2_{dec}	

and $p_r(d_2) = 0.85$. And we assume that all the erasure events are independent. We will use the results in Propositions 4.1.1 and 4.2.1 to find the largest (R_1, R_2) value for this example scenario.

Fig. 4.3 compares the LNC capacity outer bound (Proposition 4.1.1) and the LNC inner bound (Proposition 4.2.1) with different achievability schemes. The smallest rate region is achieved by simply performing uncoded direct transmission without using the relay r. The second achievability scheme is the 2-receiver broadcast channel LNC from the source s in [47] while still not exploiting r at all. The third and fourth schemes always use r for any packet delivery. Namely, both schemes do not allow 2-hop delivery from s. Then r in the third scheme uses pure routing while r performs the 2-user broadcast channel LNC in the fourth scheme. The fifth scheme performs the time-shared transmission between s and r, while allowing only intra-flow network coding. The sixth scheme is derived from using only the classic butterfly-style LNCs corresponding to $s_{\mathsf{CX};l}\,(l=1,\cdots,8)$, r_{CX}, and r_{XT}. That is, we do not allow s to perform fancy operations such as s^k_{PM1}, s^k_{PM2}, s^k_{RC}, and r_{RC}. One can see that the result is strictly suboptimal.

In sum, one can see that our proposed LNC inner bound closely approaches the LNC capacity outer bound in all angles. This shows that the newly-identified LNC operations other than the classic butterfly-style LNCs are critical in approaching the LNC capacity. The detailed rate region description of each sub-optimal achievability scheme can be found in Appendix I.

Fig. 4.4 examines the relative gaps between the outer and inner bounds by choosing the channel parameters $p_s(\cdot)$ and $p_r(\cdot)$ uniformly randomly while obeying the strong-relaying condition in Definition 4.2.1. For any chosen parameter instance, we use a linear programming solver to find the largest sum rate R_Σ of the LNC outer and inner bounds of Propositions 4.1.1 and 4.2.1, which are denoted by $R_{\mathsf{sum.outer}}$ and $R_{\mathsf{sum.inner}}$, respectively. We then compute the relative gap per each experiment, $(R_{\mathsf{sum.outer}} - R_{\mathsf{sum.inner}})/R_{\mathsf{sum.outer}}$, and then repeat the experiment 10000 times, and plot the cumulative distribution function (cdf) in unit of percentage. We can see that

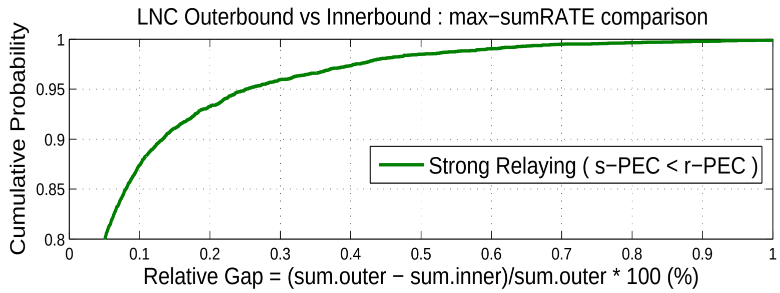


Fig. 4.4. The cumulative distribution of the relative gap between the outer and inner bounds.

with more than 85% of the experiments, the relative gap between the outer and inner bound is smaller than 0.08%.

4.4 Chapter Summary

In this chapter, we discuss the LNC capacity region of the smart repeater network formulated in Section 2.3. In Sections 4.1 and 4.2, we propose the LNC outer bound and the capacity-approaching LNC scheme with newly identified LNC operations other than the previously well-known classic butterfly-style operations. In Section 4.2.1, we provide the correctness proof of our LNC achievability scheme based on the invariance property of the queueing network analysis. In Section 4.3, we use the numerical results to describe the LNC capacity region, and demonstrate that the proposed LNC achievability scheme is close-to-optimal.

5. PRECODING-BASED FRAMEWORK FOR WIRELINE DIRECTED ACYCLIC NETWORK

In this chapter, we present and define the Precoding-based Framework. As discussed in Section 1.2, the LNC characterization problem in Wireline Networks is closely related to an underlying network topology and its corresponding algebraic solution. Thus we first start by defining some necessary graph-theoretic notations. The algebraic formulation of the proposed Precoding-based framework and its comparison to the classic LNC framework will follow in the subsequent sections. Based on our new framework, we will explain the recent wireless applications, 2-unicast Linear Deterministic Interference Channel (LDIC) [25, 26] and 3-unicast Asymptotic Network Alignment (ANA) [40, 41]. The motivation and contributions of our work for the second application, the 3-unicast ANA scheme, is further discussed. Finally, the main results of this chapter, i.e., the fundamental properties of the Precoding-based Framework, will follow in the subsequent section.

5.1 Graph-Theoretic Definitions

Consider a Directed Acyclic Integer-Capacity network (DAG) $G=(V,E)$ where V is the set of nodes and E is the set of directed edges. Each edge $e\in E$ is represented by $e=uv$, where $u=\mathsf{tail}(e)$ and $v=\mathsf{head}(e)$ are the tail and head of e, respectively. For any node $v\in V$, we use $\mathsf{In}(v)\subset E$ to denote the collection of its incoming edges $uv\in E$. Similarly, $\mathsf{Out}(v)\subset E$ contains all the outgoing edges $vw\in E$.

A path P is a series of adjacent edges $e_1e_2\cdots e_k$ where $\mathsf{head}(e_i)=\mathsf{tail}(e_{i+1})\ \forall\, i\in\{1,\cdots,k-1\}$. We say that e_1 and e_k are the starting and ending edges of P, respectively. For any path P, we use $e\in P$ to indicate that an edge e is used by P. For a given path P, xPy denotes the path segment of P from node x to node y. A path starting from node x and ending at node y is sometimes denoted by P_{xy}. By slightly abusing the notation, we sometimes substitute the nodes x and y by the edges e_1 and e_2 and use e_1Pe_2 to denote the path segment from $\mathsf{tail}(e_1)$ to $\mathsf{head}(e_2)$ along P. Similarly, $P_{e_1e_2}$ denotes a path from $\mathsf{tail}(e_1)$ to $\mathsf{head}(e_2)$. We say a node u is an *upstream* node of a node v (or v is a *downstream* node of u) if $u \neq v$ and there exists a path P_{uv}, and we denote it as $u \prec v$. If neither $u \prec v$ nor $u \succ v$, then we say that u and v are *not reachable* from each other. Similarly, e_1 is an upstream edge of e_2 if $\mathsf{head}(e_1) \preceq \mathsf{tail}(e_2)$ (where $\preceq$ means either $\mathsf{head}(e_1) \prec \mathsf{tail}(e_2)$ or $\mathsf{head}(e_1) = \mathsf{tail}(e_2)$), and we denote it by $e_1 \prec e_2$. Two distinct edges e_1 and e_2 are not reachable from each other, if neither $e_1 \prec e_2$ nor $e_1 \succ e_2$. Given any edge set E_1, we say an edge e is one of the most upstream edges in E_1 if (i) $e \in E_1$; and (ii) e is not reachable from any other edge $e' \in E_1 \backslash e$. One can easily see that the most upstream edge may not be unique. The collection of the most upstream edges of E_1 is denoted by $\mathsf{upstr}(E_1)$. A *k-edge cut* (sometimes just the "edge cut") separating node sets $U \subset V$ and $W \subset V$ is a collection of k edges such that any path from any $u \in U$ to any $w \in W$ must use at least one of those k edges. The value of an edge cut is the number of edges in the cut. (A k-edge cut has value k.) We denote the minimum value among all the edge cuts separating U and W as $\mathsf{EC}(U;W)$. By definition, we have $\mathsf{EC}(U;W) = 0$ when U and W are already disconnected. By convention, if $U \cap W \neq \emptyset$, we define $\mathsf{EC}(U;W) = \infty$. We also denote the collection of all distinct 1-edge cuts separating U and W as $\mathsf{1cut}(U;W)$.

5.2 Algebraic Formulation of The Precoding-based Framework

Given a network $G = (V, E)$, consider the multiple-unicast problem in which there are K coexisting source-destination pairs (s_k, d_k), $k = 1, \cdots, K$.[1] Let l_k denote the number of information symbols that s_k wants to transmit to d_k. Each information

[1]Since an arbitrary multi-session communication requirement can be equivalently converted to the corresponding multiple-unicast traffic demands, we formulate the Precoding-based Framework based on multiple unicasts without loss of generality.

symbol is chosen independently and uniformly from a finite field $\mathbb{F}_q$ with some sufficiently large q.

Following the widely-used instantaneous transmission model for DAGs [3], we assume that each edge is capable of transmitting one symbol in $\mathbb{F}_q$ in one time slot without delay. We consider *linear network coding* over the entire network, i.e., a symbol on an edge $e \in E$ is a linear combination of the symbols on its adjacent incoming edges $\mathsf{In}(\mathsf{tail}(e))$. The coefficients (also known as the network variables) used for such linear combinations are termed local encoding kernels. The collection of all local kernels $x_{e'e''} \in \mathbb{F}_q$ for all adjacent edge pairs (e', e'') is denoted by $\underline{\mathbf{x}} = \{x_{e'e''} : (e', e'') \in E^2 \text{ where } \mathsf{head}(e') = \mathsf{tail}(e'')\}$. See [3] for detailed discussion. Following this notation, the channel gain $m_{e_1;e_2}(\underline{\mathbf{x}})$ from an edge e_1 to an edge e_2 can be written as a polynomial with respect to $\underline{\mathbf{x}}$. More rigorously, $m_{e_1;e_2}(\underline{\mathbf{x}})$ can be rewritten as

$$m_{e_1;e_2}(\underline{\mathbf{x}}) = \sum_{\forall P_{e_1e_2} \in \mathbf{P}_{e_1e_2}} \left(\prod_{\forall\, e', e'' \in P_{e_1e_2} \text{ where } \mathsf{head}(e') = \mathsf{tail}(e'')} x_{e'e''} \right)$$

where $\mathbf{P}_{e_1e_2}$ denotes the collection of all distinct paths from e_1 to e_2.

By convention [3], we set $m_{e_1;e_2}(\underline{\mathbf{x}}) = 1$ when $e_1 = e_2$ and set $m_{e_1;e_2}(\underline{\mathbf{x}}) = 0$ when $e_1 \neq e_2$ and e_2 is not a downstream edge of e_1. The channel gain from a node u to a node v is defined by an $|\mathsf{In}(v)| \times |\mathsf{Out}(u)|$ polynomial matrix $\mathbf{M}_{u;v}(\underline{\mathbf{x}})$, where its (i, j)-th entry is the (edge-to-edge) channel gain from the j-th outgoing edge of u to the i-th incoming edge of v. When considering source s_i and destination d_j, we use $\mathbf{M}_{i;j}(\underline{\mathbf{x}})$ as shorthand for $\mathbf{M}_{s_i;d_j}(\underline{\mathbf{x}})$.

We allow the Precoding-based framework to code across τ number of time slots, which are termed the precoding frame and τ is the frame size. The network variables used in time slot t is denoted as $\underline{\mathbf{x}}^{(t)}$, and the corresponding channel gain from s_i to d_j becomes $\mathbf{M}_{i;j}(\underline{\mathbf{x}}^{(t)})$ for all $t = 1, \cdots, \tau$.

With these settings, let $\mathbf{z}_i \in \mathbb{F}_q^{l_i \times 1}$ be the set of to-be-sent information symbols from s_i. Then, for every time slot $t = 1, \cdots, \tau$, we can define the precoding matrix

$\mathbf{V}_i^{(t)} \in \mathbb{F}_q^{|\mathsf{Out}(s_i)| \times l_i}$ for each source s_i. Given the precoding matrices, each d_j receives an $|\mathsf{In}(d_j)|$-dimensional column vector $\mathbf{y}_j^{(t)}$ at time t:

$$\mathbf{y}_j^{(t)}(\underline{\mathbf{x}}^{(t)}) = \mathbf{M}_{j;j}(\underline{\mathbf{x}}^{(t)})\mathbf{V}_j^{(t)}\mathbf{z}_j + \sum_{\substack{i=1 \\ i \neq j}}^{K} \mathbf{M}_{i;j}(\underline{\mathbf{x}}^{(t)})\mathbf{V}_i^{(t)}\mathbf{z}_i.$$

where we use the input argument "$(\underline{\mathbf{x}}^{(t)})$" to emphasize that $\mathbf{M}_{j;j}$ and $\mathbf{y}_j^{(t)}$ are functions of the network variables $\underline{\mathbf{x}}^{(t)}$.

This system model can be equivalently expressed as

$$\overline{\mathbf{y}}_j = \overline{\mathbf{M}}_{j;j}\overline{\mathbf{V}}_j\,\mathbf{z}_j + \sum_{\substack{i=1 \\ i \neq j}}^{K} \overline{\mathbf{M}}_{i;j}\overline{\mathbf{V}}_i\,\mathbf{z}_i, \tag{5.1}$$

where $\overline{\mathbf{V}}_i$ is the overall precoding matrix for each source s_i by vertically concatenating $\{\mathbf{V}_i^{(t)}\}_{t=1}^{\tau}$, and $\overline{\mathbf{y}}_j$ is the vertical concatenation of $\{\mathbf{y}_j^{(t)}(\underline{\mathbf{x}}^{(t)})\}_{t=1}^{\tau}$. The overall channel matrix $\overline{\mathbf{M}}_{i;j}$ is a block-diagonal polynomial matrix with $\{\mathbf{M}_{i;j}(\underline{\mathbf{x}}^{(t)})\}_{t=1}^{\tau}$ as its diagonal blocks. Note that $\overline{\mathbf{M}}_{i;j}$ is a polynomial matrix with respect to the network variables $\{\underline{\mathbf{x}}^{(t)}\}_{t=1}^{\tau}$.

After receiving packets for τ time slots, each destination d_j applies the overall decoding matrix $\overline{\mathbf{U}}_j \in \mathbb{F}_q^{l_j \times (\tau \cdot |\mathsf{In}(d_j)|)}$. Then, the decoded message vector $\hat{\mathbf{z}}_j$ can be expressed as

$$\hat{\mathbf{z}}_j = \overline{\mathbf{U}}_j\overline{\mathbf{y}}_j = \overline{\mathbf{U}}_j\overline{\mathbf{M}}_{j;j}\overline{\mathbf{V}}_j\,\mathbf{z}_j + \sum_{\substack{i=1 \\ i \neq j}}^{K} \overline{\mathbf{U}}_j\overline{\mathbf{M}}_{i;j}\overline{\mathbf{V}}_i\,\mathbf{z}_i. \tag{5.2}$$

The combined effects of precoding, channel, and decoding from s_i to d_j is $\overline{\mathbf{U}}_j\overline{\mathbf{M}}_{i;j}\overline{\mathbf{V}}_i$, which is termed the *network transfer matrix* from s_i to d_j. We say that the Precoding-based NC problem is feasible if there exists a pair of precoding and decoding matrices $\{\overline{\mathbf{V}}_i, \forall\, i\}$ and $\{\overline{\mathbf{U}}_j, \forall\, j\}$ (which may be a function of $\{\underline{\mathbf{x}}^{(t)}\}_{t=1}^{\tau}$) such that when choosing each element of the collection of network variables $\{\underline{\mathbf{x}}^{(t)}\}_{t=1}^{\tau}$ independently and uniformly randomly from $\mathbb{F}_q$, with high probability,

$$\begin{aligned} \text{Satisfying the Demands:} \quad & \overline{\mathbf{U}}_j \overline{\mathbf{M}}_{i;j} \overline{\mathbf{V}}_i = \mathbf{I} \quad \text{(the identity matrix)} \quad \forall\, i = j, \\ \text{Interference-Free:} \quad & \overline{\mathbf{U}}_j \overline{\mathbf{M}}_{i;j} \overline{\mathbf{V}}_i = \mathbf{0} \quad \forall\, i \neq j. \end{aligned} \tag{5.3}$$

Remark 1: One can easily check by the cut-set bound that a necessary condition for the feasibility of a Precoding-based NC problem is for the frame size $\tau \geq \max_k \{l_k / \mathsf{EC}(s_k; d_k)\}$.

Remark 2: Depending on the time relationship of $\overline{\mathbf{V}}_i$ and $\overline{\mathbf{U}}_j$ with respect to the network variables $\{\underline{\mathbf{x}}^{(t)}\}_{t=1}^{\tau}$, a Precoding-based NC solution can be classified as causal vs. non-causal and time-varying vs. time-invariant schemes.

For convenience to the reader, we have summarized in Table 5.1 several key definitions used in the Precoding-based Framework.

5.2.1 Comparison to The Classic Algebraic Framework

The authors in [3] established the algebraic framework for linear network coding, which admits similar encoding and decoding equations as in (5.1) and (5.2) and the same algebraic feasibility equations as in (5.3). This original work focuses on a single time slot $\tau = 1$ while the corresponding results can be easily generalized for $\tau > 1$ as well. Note that $\tau > 1$ provides a greater degree of freedom when designing the coding matrices $\{\overline{\mathbf{V}}_i, \forall\, i\}$ and $\{\overline{\mathbf{U}}_j, \forall\, j\}$. Such *time extension* turns out to be especially critical in a Precoding-based NC design as it is generally much harder (sometimes impossible) to design $\{\overline{\mathbf{V}}_i, \forall\, i\}$ and $\{\overline{\mathbf{U}}_j, \forall\, j\}$ when $\tau = 1$. An example of this time extension will be discussed in Section 5.2.3.

The main difference between the Precoding-based framework and the classic framework is that the latter allows the NC designer to control the network variables $\underline{\mathbf{x}}$ while the former assumes that the entries of $\underline{\mathbf{x}}$ are chosen independently and uniformly randomly. One can thus view the Precoding-based NC as a distributed version of classic

Table 5.1: Key definitions of the Precoding-based Framework

Notations for the Precoding-based Framework	
K	The number of coexisting unicast sessions
l_i	The number of information symbols sent from s_i to d_i
$\underline{\mathbf{x}}$	The network variables / local encoding kernels
$m_{e_1;e_2}(\underline{\mathbf{x}})$	The channel gain from an edge e_1 to an edge e_2, which is a polynomial with respect to $\underline{\mathbf{x}}$
$\mathbf{M}_{u;v}(\underline{\mathbf{x}})$	The channel gain matrix from a node u to a node v where its (i,j)-th entry is the channel gain from j-th outgoing edge of u to i-th incoming edge of v
τ	The precoding frame size (number of time slot)
$\underline{\mathbf{x}}^{(t)}$	The network variables corresponding to time slot t
$\mathbf{V}_i^{(t)}$	The precoding matrix for s_i at time slot t
$\mathbf{M}_{i;j}(\underline{\mathbf{x}}^{(t)})$	The channel gain matrix from s_i to d_j at time slot t, shorthand for $\mathbf{M}_{s_i;d_j}(\underline{\mathbf{x}}^{(t)})$
$\mathbf{U}_j^{(t)}$	The decoding matrix for d_j at time slot t
$\overline{\mathbf{V}}_i$	The overall precoding matrix for s_i for the entire precoding frame $t = 1, \cdots, \tau$.
$\overline{\mathbf{M}}_{i;j}$	The overall channel gain matrix from s_i to d_j for the entire precoding frame $t = 1, \cdots, \tau$.
$\overline{\mathbf{U}}_j$	The overall decoding matrix for d_j for the entire precoding frame $t = 1, \cdots, \tau$.

NC schemes that trades off the ultimate achievable performance for more practical distributed implementation (not controlling the behavior in the interior of the network).

One challenge when using algebraic feasibility equations (5.3) is that given a network code, it is easy to verify whether or not (5.3) is satisfied, but it is difficult to decide whether there exists a NC solution satisfying (5.3), see [3, 38]. Only in some special scenarios can we convert those algebraic feasibility equations into some graph-theoretic conditions for which one can decide the existence of a feasible network code in polynomial time. For example, if there exists only a single session (s_1, d_1) in the network, then the existence of a NC solution satisfying (5.3) is equivalent to the time-averaged rate l_1/τ being no larger than $\mathsf{EC}(s_1; d_1)$. Moreover, if $(l_1/\tau) \leq \mathsf{EC}(s_1; d_1)$, then we can use random linear network coding [6] to construct the optimal network code. Another example is when there are only two sessions (s_1, d_1) and (s_2, d_2) with

$l_1 = l_2 = \tau = 1$. Then, the existence of a network code satisfying (5.3) is equivalent to the conditions that the 1-edge cuts in the network are properly placed in certain ways [24]. Except the scenarios taken as examples above, however, the algebraic conditions of many other scenarios are not interpreted as the graph-theoretic arguments. Note that checking the algebraic conditions can be computationally intractable. Motivated by the above observation, the main focus of this thesis is to develop a fundamental graph-theoretic properties of the Precoding-based NC, which can be utilized in characterizing the Precoding-based solutions. For the following subsections, we will introduce two special instances of the Precoding-based framework and present their corresponding algebraic conditions. We will demonstrate why such fundamental connection from the algebraic to the graph-theoretic is in need.

5.2.2 A Special Scenario : The 2-unicast Linear Deterministic Interference Channel (LDIC)

We now consider a special class of networks, called the 2-unicast LDIC network: A network G is a 2-unicast LDIC network if (i) there are 2 source-destination pairs, $(s_i, d_i), i = 1, 2$, where all source/destination nodes are distinct; (ii) $|\mathsf{In}(s_i)| = 0$ and $|\mathsf{Out}(s_i)| \geq 1 \ \forall\, i$; (iii) $|\mathsf{In}(d_j)| \geq 1$ and $|\mathsf{Out}(d_j)| = 0 \ \forall\, j$; and (iv) d_j can be reached from s_i for all (i, j) pairs (including those with $i = j$). We use the notation G_{2LDIC} to emphasize that we are focusing on this 2-unicast LDIC network.

The authors in [25] derived the capacity of the wireless two-user MIMO deterministic Interference Channel and applied this result to the above 2-unicast LDIC network. An independent work [26] has been done on the same 2-unicast LDIC network using the similar precoding and decoding techniques used in [25]. We present the result of [25] since it is a superset.

Let the rates (R_1, R_2) to be $(\frac{l_1}{\tau}, \frac{l_2}{\tau})$ and set $\tau = 1$. Since $\tau = 1$, we do not consider the time-extension of the Precoding-based framework and thus the overall channel matrix $\overline{\mathbf{M}}_{i;j}$ from s_i to d_j simply reduces to $\mathbf{M}_{i;j}(\underline{\mathbf{x}})$, where $\underline{\mathbf{x}}$ is the collection of variables in the given G_{2LDIC} of interest. The authors in [25] proves the following result.

Proposition 5.2.1 (page 7, [25])**.** *For a sufficiently large finite field $\mathbb{F}_q$, the 2-unicast LDIC scheme achieves the rate tuple (R_1, R_2) with close-to-one probability if the following conditions are satisfied:*

$$R_1 \leq \mathsf{EC}(s_1; d_1), \tag{5.4}$$

$$R_2 \leq \mathsf{EC}(s_2; d_2), \tag{5.5}$$

$$R_1 + R_2 \leq \mathsf{EC}(\{s_1, s_2\}; d_1) + \mathsf{EC}(s_2; \{d_1, d_2\}) - \mathsf{EC}(s_2; d_1), \tag{5.6}$$

$$R_1 + R_2 \leq \mathsf{EC}(\{s_1, s_2\}; d_2) + \mathsf{EC}(s_1; \{d_1, d_2\}) - \mathsf{EC}(s_1; d_2), \tag{5.7}$$

$$\begin{aligned} R_1 + R_2 \leq\ & \mathsf{rank}\left(\begin{bmatrix} \mathbf{M}_{1;1}(\underline{\mathbf{x}}) & \mathbf{M}_{1;2}(\underline{\mathbf{x}}) \\ \mathbf{M}_{2;1}(\underline{\mathbf{x}}) & \mathbf{0} \end{bmatrix}\right) + \mathsf{rank}\left(\begin{bmatrix} \mathbf{M}_{2;1}(\underline{\mathbf{x}}) & \mathbf{M}_{2;2}(\underline{\mathbf{x}}) \\ \mathbf{0} & \mathbf{M}_{1;2}(\underline{\mathbf{x}}) \end{bmatrix}\right) \\ & - \mathsf{EC}(s_1; d_2) - \mathsf{EC}(s_2; d_1), \end{aligned} \tag{5.8}$$

$$\begin{aligned} 2R_1 + R_2 \leq\ & \mathsf{EC}(\{s_1, s_2\}; d_1) + \mathsf{EC}(s_1; \{d_1, d_2\}) \\ & + \mathsf{rank}\left(\begin{bmatrix} \mathbf{M}_{2;1}(\underline{\mathbf{x}}) & \mathbf{M}_{2;2}(\underline{\mathbf{x}}) \\ \mathbf{0} & \mathbf{M}_{1;2}(\underline{\mathbf{x}}) \end{bmatrix}\right) - \mathsf{EC}(s_1; d_2) - \mathsf{EC}(s_2; d_1), \end{aligned} \tag{5.9}$$

$$\begin{aligned} R_1 + 2R_2 \leq\ & \mathsf{EC}(\{s_1, s_2\}; d_2) + \mathsf{EC}(s_2; \{d_1, d_2\}) \\ & + \mathsf{rank}\left(\begin{bmatrix} \mathbf{M}_{1;1}(\underline{\mathbf{x}}) & \mathbf{M}_{1;2}(\underline{\mathbf{x}}) \\ \mathbf{M}_{2;1}(\underline{\mathbf{x}}) & \mathbf{0} \end{bmatrix}\right) - \mathsf{EC}(s_1; d_2) - \mathsf{EC}(s_2; d_1), \end{aligned} \tag{5.10}$$

where $\mathsf{rank}(\mathbf{A})$ *denote the rank of a given matrix* $\mathbf{A}$.

We are not going to explain the network code construction to achieve a specific rates satisfying the above conditions (5.4) to (5.10).[2] But note that, given a G_{2LDIC}, the characterization problem of the corresponding 2-unicast LDIC scheme depends on some end-to-end edge-cut values and the ranks of two matrices of dimension $(|\mathsf{In}(s_1)| + |\mathsf{In}(s_2)|) \times (|\mathsf{Out}(d_1)| + |\mathsf{Out}(d_2)|)$, which appear in (5.8) to (5.10).

[2]The construction is based on the precoding and decoding at both ends using SVD technique, while choosing the network variables $\underline{\mathbf{x}}$ independently and uniformly randomly. See [25] for details.

Since the network variables are chosen independently and uniformly randomly, these ranks will have some fixed values with close-to-one probability given a network. And such ranks needs to be of full-rank to be operated in the maximum possible throughput. Since the edge-cut values (5.4) to (5.10) constitutes the capacity outer bounds in a given network, knowing when these channel polynomial matrices become full-rank or not will be of importance in revealing its relation to the currently-open arbitrary 2-unicast LNC capacity and in achieving the largest throughput in this 2-unicast LDIC application. Therefore, knowing the close relationship of these algebraic conditions to some graph-theoretic conditions is critical in multi-session LNC characterizations.

5.2.3 A Special Scenario : The 3-unicast Asymptotic Network Alignment (ANA)

Before proceeding, we introduce some algebraic definitions. We say that a set of polynomials $\mathbf{h}(\underline{\mathbf{x}}) = \{h_1(\underline{\mathbf{x}}), ..., h_N(\underline{\mathbf{x}})\}$ is linearly dependent if and only if $\sum_{k=1}^{N} \alpha_k h_k(\underline{\mathbf{x}}) = 0$ for some coefficients $\{\alpha_k\}_{k=1}^{N}$ that are not all zeros. By treating $\mathbf{h}(\underline{\mathbf{x}}^{(k)})$ as a polynomial row vector and vertically concatenating them together, we have an $M \times N$ polynomial matrix $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{M}$. We call this polynomial matrix a *row-invariant* matrix since each row is based on the same set of polynomials $\mathbf{h}(\underline{\mathbf{x}})$ but with different variables $\underline{\mathbf{x}}^{(k)}$ for each row k, respectively. We say that the row-invariant polynomial matrix $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{M}$ is generated from $\mathbf{h}(\underline{\mathbf{x}})$. For two polynomials $g(\underline{\mathbf{x}})$ and $h(\underline{\mathbf{x}})$, we say $g(\underline{\mathbf{x}})$ and $h(\underline{\mathbf{x}})$ are *equivalent*, denoted by $g(\underline{\mathbf{x}}) \equiv h(\underline{\mathbf{x}})$, if $g(\underline{\mathbf{x}}) = c \cdot h(\underline{\mathbf{x}})$ for some non-zero $c \in \mathbb{F}_q$. If not, we say that $g(\underline{\mathbf{x}})$ and $h(\underline{\mathbf{x}})$ are *not equivalent*, denoted by $g(\underline{\mathbf{x}}) \not\equiv h(\underline{\mathbf{x}})$. We use $\mathsf{GCD}(g(\underline{\mathbf{x}}), h(\underline{\mathbf{x}}))$ to denote the greatest common factor of the two polynomials.

We now consider a special class of networks, called the 3-unicast ANA network: A network G is a 3-unicast ANA network if (i) there are 3 source-destination pairs, $(s_i, d_i), i = 1, 2, 3$, where all source/destination nodes are distinct; (ii) $|\mathsf{In}(s_i)| = 0$ and

$|\mathsf{Out}(s_i)|=1\ \forall\, i$ (We denote the only outgoing edge of s_i as e_{s_i}, termed the s_i-source edge.); (iii) $|\mathsf{In}(d_j)|=1$ and $|\mathsf{Out}(d_j)|=0\ \forall\, j$ (We denote the only incoming edge of d_j as e_{d_j}, termed the d_j-destination edge.); and (iv) d_j can be reached from s_i for all (i,j) pairs (including those with $i=j$).[3] We use the notation $G_{3\text{ANA}}$ to emphasize that we are focusing on this 3-unicast ANA network. Note that by (ii) and (iii) the matrix $\mathbf{M}_{i;j}(\underline{\mathbf{x}})$ becomes a scalar, which we denote by $m_{ij}(\underline{\mathbf{x}})$ instead.

The authors in [40, 41] applied interference alignment to construct the precoding matrices $\{\overline{\mathbf{V}}_i, \forall\, i\}$ for the above 3-unicast ANA network. Namely, consider the following parameter values: $\tau=2n+1$, $l_1=n+1$, $l_2=n$, and $l_3=n$ for some positive integer n termed symbol extension parameter, and assume that all the network variables $\underline{\mathbf{x}}^{(1)}$ to $\underline{\mathbf{x}}^{(\tau)}$ are chosen independently and uniformly randomly from $\mathbb{F}_q$. The goal is to achieve the rate tuple $(\frac{n+1}{2n+1}, \frac{n}{2n+1}, \frac{n}{2n+1})$ in a 3-unicast ANA network by applying the following $\{\overline{\mathbf{V}}_i, \forall\, i\}$ construction method: Define $L(\underline{\mathbf{x}}) = m_{13}(\underline{\mathbf{x}})m_{32}(\underline{\mathbf{x}})m_{21}(\underline{\mathbf{x}})$ and $R(\underline{\mathbf{x}}) = m_{12}(\underline{\mathbf{x}})m_{23}(\underline{\mathbf{x}})m_{31}(\underline{\mathbf{x}})$, and consider the following 3 row vectors of dimensions $n+1$, n, and n, respectively. (Each entry of these row vectors is a polynomial with respect to $\underline{\mathbf{x}}$ but we drop the input argument $\underline{\mathbf{x}}$ for simplicity.)

$$\mathbf{v}_1^{(n)}(\underline{\mathbf{x}}) = m_{23}m_{32}\left[R^n,\ \ R^{n-1}L,\ \cdots,\ RL^{n-1},\ \ L^n\right], \tag{5.11}$$

$$\mathbf{v}_2^{(n)}(\underline{\mathbf{x}}) = m_{13}m_{32}\left[R^n,\ \ R^{n-1}L,\ \cdots,\ RL^{n-1}\right], \tag{5.12}$$

$$\mathbf{v}_3^{(n)}(\underline{\mathbf{x}}) = m_{12}m_{23}\left[R^{n-1}L,\ \cdots,\ RL^{n-1},\ \ L^n\right], \tag{5.13}$$

where the superscript "(n)" is to emphasize the value of the symbol extension parameter n used in the construction. The precoding matrix for each time slot t is designed to be $\mathbf{V}_i^{(t)}=\mathbf{v}_i^{(n)}(\underline{\mathbf{x}}^{(t)})$. The overall precoding matrix (the vertical concatenation of $\mathbf{V}_i^{(1)}$ to $\mathbf{V}_i^{(\tau)}$) is thus $\overline{\mathbf{V}}_i=[\mathbf{v}_i^{(n)}(\underline{\mathbf{x}}^{(t)})]_{t=1}^{2n+1}$.

[3]The above *fully interfered* setting is the worst case scenario. For the scenario in which there is some d_j who is not reachable from some s_i, one can devise an achievable solution by modifying the solution for the worst-case fully interfered 3-ANA networks [40].

The authors in [40, 41] prove that the above construction achieves the desired rates $(\frac{n+1}{2n+1}, \frac{n}{2n+1}, \frac{n}{2n+1})$ if the overall precoding matrices $\{\overline{\mathbf{V}}_i, \forall\, i\}$ satisfy the following six constraints:[4]

$$d_1 : \langle\, \overline{\mathbf{M}}_{3;1}\overline{\mathbf{V}}_3 \,\rangle = \langle\, \overline{\mathbf{M}}_{2;1}\overline{\mathbf{V}}_2 \,\rangle \tag{5.14}$$

$$\mathbf{S}_1^{(n)} \triangleq \left[\, \overline{\mathbf{M}}_{1;1}\overline{\mathbf{V}}_1 \;\; \overline{\mathbf{M}}_{2;1}\overline{\mathbf{V}}_2 \,\right], \text{ and } \mathsf{rank}(\mathbf{S}_1^{(n)}) = 2n+1 \tag{5.15}$$

$$d_2 : \langle\, \overline{\mathbf{M}}_{3;2}\overline{\mathbf{V}}_3 \,\rangle \subseteq \langle\, \overline{\mathbf{M}}_{1;2}\overline{\mathbf{V}}_1 \,\rangle \tag{5.16}$$

$$\mathbf{S}_2^{(n)} \triangleq \left[\, \overline{\mathbf{M}}_{2;2}\overline{\mathbf{V}}_2 \;\; \overline{\mathbf{M}}_{1;2}\overline{\mathbf{V}}_1 \,\right], \text{ and } \mathsf{rank}(\mathbf{S}_2^{(n)}) = 2n+1 \tag{5.17}$$

$$d_3 : \langle\, \overline{\mathbf{M}}_{2;3}\overline{\mathbf{V}}_2 \,\rangle \subseteq \langle\, \overline{\mathbf{M}}_{1;3}\overline{\mathbf{V}}_1 \,\rangle \tag{5.18}$$

$$\mathbf{S}_3^{(n)} \triangleq \left[\, \overline{\mathbf{M}}_{3;3}\overline{\mathbf{V}}_3 \;\; \overline{\mathbf{M}}_{1;3}\overline{\mathbf{V}}_1 \,\right], \text{ and } \mathsf{rank}(\mathbf{S}_3^{(n)}) = 2n+1 \tag{5.19}$$

with close-to-one probability, where $\langle\, \mathbf{A} \,\rangle$ and $\mathsf{rank}(\mathbf{A})$ denote the column vector space and the rank, respectively, of a given matrix $\mathbf{A}$. The overall channel matrix $\overline{\mathbf{M}}_{i;j}$ is a $(2n+1) \times (2n+1)$ diagonal matrix with the t-th diagonal element $m_{ij}(\underline{\mathbf{x}}^{(t)})$ due to the assumption of $|\mathsf{Out}(s_i)| = |\mathsf{In}(d_j)| = 1$. We also note that the construction in (5.15), (5.17), and (5.19) ensures that the square matrices $\{\mathbf{S}_i^{(n)}, \forall\, i\}$ are row-invariant.

The intuition behind (5.14) to (5.19) is straightforward. Whenever (5.14) is satisfied, the interference from s_2 and from s_3 are aligned from the perspective of d_1. Further, by simple linear algebra we must have $\mathsf{rank}(\overline{\mathbf{M}}_{2;1}\overline{\mathbf{V}}_2) \leq n$ and $\mathsf{rank}(\overline{\mathbf{M}}_{1;1}\overline{\mathbf{V}}_1) \leq n+1$. (5.15) thus guarantees that (i) the rank of $\left[\, \overline{\mathbf{M}}_{1;1}\overline{\mathbf{V}}_1 \;\; \overline{\mathbf{M}}_{2;1}\overline{\mathbf{V}}_2 \,\right]$ equals to $\mathsf{rank}(\overline{\mathbf{M}}_{1;1}\overline{\mathbf{V}}_1) + \mathsf{rank}(\overline{\mathbf{M}}_{2;1}\overline{\mathbf{V}}_2)$ and (ii) $\mathsf{rank}(\overline{\mathbf{M}}_{1;1}\overline{\mathbf{V}}_1) = n+1$. Jointly (i) and (ii) imply that d_1 can successfully remove the aligned interference while recovering all $l_1 = n+1$ information symbols intended for d_1. Similar arguments can be used to justify (5.16) to (5.19) from the perspectives of d_2 and d_3, respectively.

---

[4]Here the interference alignment is performed based on (s_1, d_1)-pair who achieves larger rate than others. Basically, any transmission pair can be chosen as an alignment-basis achieving $\frac{n+1}{2n+1}$, and the corresponding precoding matrices and six constraints can be constructed accordingly.

By noticing the special Vandermonde form of $\overline{\mathbf{V}}_i$, it is shown in [40,41] that (5.14), (5.16), and (5.18) always hold. The authors in [41] further prove that if

$$L(\underline{\mathbf{x}}) \not\equiv R(\underline{\mathbf{x}}) \tag{5.20}$$

and the following algebraic conditions are satisfied:

$$m_{11}m_{23}\sum_{i=0}^{n}\alpha_i\big(L/R\big)^i \neq m_{21}m_{13}\sum_{j=0}^{n-1}\beta_j\big(L/R\big)^j \tag{5.21}$$

$$m_{22}m_{13}\sum_{i=0}^{n-1}\alpha_i\big(L/R\big)^i \neq m_{12}m_{23}\sum_{j=0}^{n}\beta_j\big(L/R\big)^j \tag{5.22}$$

$$m_{33}m_{12}\sum_{i=1}^{n}\alpha_i\big(L/R\big)^i \neq m_{13}m_{32}\sum_{j=0}^{n}\beta_j\big(L/R\big)^j \tag{5.23}$$

for all $\alpha_i, \beta_j \in \mathbb{F}_q$ with at least one of α_i and at least one of β_j being non-zero, then the constraints (5.15), (5.17), and (5.19) hold with close-to-one probability (recalling that the network variables $\underline{\mathbf{x}}^{(1)}$ to $\underline{\mathbf{x}}^{(\tau)}$ are chosen independently and uniformly randomly).

In summary, [40, 41] proves the following result.

Proposition 5.2.2 (page 3, [41])**.** *For a sufficiently large finite field $\mathbb{F}_q$, the 3-unicast ANA scheme described in (5.11) to (5.13) achieves the rate tuple $(\frac{n+1}{2n+1}, \frac{n}{2n+1}, \frac{n}{2n+1})$ with close-to-one probability if (5.20), (5.21), (5.22), and (5.23) hold simultaneously.*

Therefore, whether we can use the 3-unicast ANA scheme depends on whether the given $G_{3\text{ANA}}$ satisfies the algebraic conditions (5.20), (5.21), (5.22), and (5.23) simultaneously.

However, it can be easily seen that directly verifying the above sufficient conditions is computationally intractable. Moreover, they heavily depend on the given $G_{3\text{ANA}}$ of interest. Note that in the setting of wireless interference channels, the individual channel gains are independently and continuously distributed, for which one can prove that the feasibility conditions (5.20), (5.15), (5.17), and (5.19) hold with probability one [39]. For a network setting here, the channel gain polynomials $m_{ij}(\underline{\mathbf{x}})$ are no

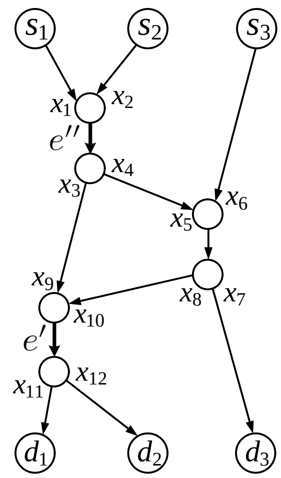


Fig. 5.1. Example $G_{3\text{ANA}}$ structure satisfying $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ with $\underline{\mathbf{x}} = \{x_1, x_2, ..., x_{12}\}$.

Table 5.2: Key definitions of the 3-unicast ANA scheme

Notations for the 3-unicast ANA network

$m_{ij}(\underline{\mathbf{x}})$	The channel gain polynomial from s_i to d_j
$L(\underline{\mathbf{x}})$	The product of three channel gains: $m_{13}(\underline{\mathbf{x}})m_{32}(\underline{\mathbf{x}})\ m_{21}(\underline{\mathbf{x}})$
$R(\underline{\mathbf{x}})$	The product of three channel gains: $m_{12}(\underline{\mathbf{x}})m_{23}(\underline{\mathbf{x}})\ m_{31}(\underline{\mathbf{x}})$

longer independently distributed for different (i, j) pairs and the correlation depends on the underlying network topology. For example, one can verify that the 3-unicast ANA network described in Fig. 5.1 always leads to $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ even when all network variables $\underline{\mathbf{x}}$ are chosen uniformly randomly from an arbitrarily large finite field $\mathbb{F}_q$.

For convenience to the reader, we have summarized in Table 5.2 several key definitions used in the 3-unicast ANA network.

5.2.4 A Critical Question

As discussed in the end of Sections 5.2.2 and 5.2.3, the channel relationship to the given network topology is important in characterizing these applications. Since the channel gains are finite field polynomials with respect to network variables, a more important question would be "How the polynomials over the network variables and graph theory are fundamentally related?" To answer these questions, we believe that a deeper understanding of the proposed Precoding-based Framework will play a key role. Along this investigation, we identify the several fundamental properties of the Precoding-based Framework which can bridge the gap between these two separate worlds. Moreover using these fundamental properties, we characterize graph-theoretically the algebraic feasibility conditions of one wireless application, the 3-unicast ANA scheme. More detailed discussions and contributions will follow in Section 5.3.

5.3 Motivation of Studying the Precoding-based 3-unicast ANA network and Detailed Summary of Contributions

As explained above, the classic algebraic framework [3] bridges between the satisfiability of a given network information flow and the solvability of the corresponding algebraic feasibility equations (5.3), both of which depend on the given network of interest. It is thus needless to say that the network structures and the existence of a network code satisfying traffic demands are closely related. From the perspective that the graph structures can be easily verifiable, the graph-theoretic characterization plays an not only important but also practical pivot in broadening the understandings of multi-session LNC problems.

The main challenge in the classic framework along this direction is that it is difficult to decide whether there exists a LNC solution satisfying the feasibility equations. In the single-session $(s, \{d_i\})$ where there are no interferences, we only need to solve non-zero-equations and thus the existence of a LNC solution can be characterized by each min-cut value $\mathsf{EC}(s; d_i)$ being larger than equal to the rate. In the multi-session, however, we also need to solve zero-equations to be interference-free. As a result, the corresponding graph-theoretic characterization also needs to provide the properly located special cuts that perform interference-removing along the network. This is the reason why we have the complete graph-theoretic characterization only for the simplest multi-session scenario of 2-unicast/multicast with single rates: by the existence of 1-edge cuts properly placed in certain ways [24, 29, 30]. The central control over the local encoding kernels inside the network intricates graph-theoretic implications.

However, such graph-theoretic implications become critical in the Precoding-based Framework that embraces the results of Wireless Interference Channels. Compared to the classic framework, this framework exploits the pure random linear network coding in the interior of the network while focusing on precoding and decoding designs for the balanced performance as in Wireless Interference Channels. Hence, the channels between sources and destinations determine the feasibility of such precoding-based NC design. Moreover, they are now high-order polynomials over the network variables and thus correlated to a given network. Therefore, knowing the relationship between the channel polynomials and the underlying network structures becomes critical in characterizing the feasibility of the precoding-based NC solutions over the network of interest. Especially for the wireless applications such as the interference alignment technique to 3-unicast, called the 3-unicast ANA scheme [40,41], such graph-theoretic implications are practically crucial because its feasibility conditions are computationally intractable to verify directly, see Proposition 5.2.2 for example. Considering the fact that such feasibility conditions typically hold in the original wireless interference channels with close-to-one probability (due to the continuously distributed wireless random channel gains), studying the precoding-based 3-unicast ANA network, and more fundamentally, the relationship between the network channel and the graph structure is of importance in broadening the understandings of multi-session LNC problems.

Our main contributions can be summarized as follows:

- The relationship between the network channel and the graph structures: We develop the several fundamental properties of the Precoding-based Framework which allows us to bridge the gap between the feasibility of the precoding-based NC solutions and the given network.

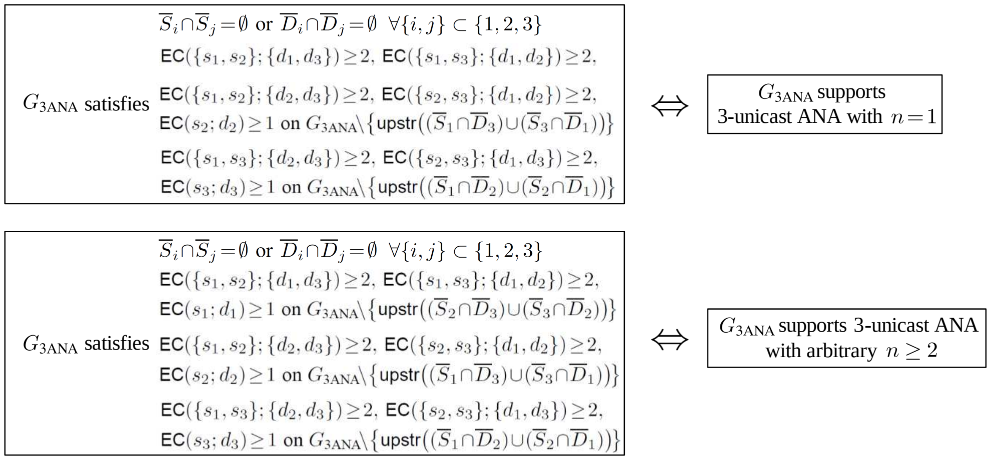


Fig. 5.2. The complete graph-theoretic characterization for the feasibility of the 3-unicast ANA scheme.

- The complete graph-theoretic characterization for the feasibility of the 3-unicast ANA scheme: Using the properties, we characterize the feasibility conditions of this interference alignment application by the existence of special edge-cuts or the min-cut values as shown in Fig. 5.2, which can be checked in polynomial time.

Note that our graph-theoretic characterization is bi-directions. Therefore, we can answer that the following conjecture is not true:

Conjecture (Page 3, [41]): For any n value used in the 3-unicast ANA scheme construction, if (5.20) and the following three conditions are satisfied simultaneously, then (5.21) to (5.23) must hold.

$$\mathsf{EC}(\{s_1,s_2\};\{d_1,d_3\}) \geq 2 \quad \text{and} \quad \mathsf{EC}(\{s_1,s_3\};\{d_1,d_2\}) \geq 2, \tag{5.24}$$

$$\mathsf{EC}(\{s_1,s_2\};\{d_2,d_3\}) \geq 2 \quad \text{and} \quad \mathsf{EC}(\{s_2,s_3\};\{d_1,d_2\}) \geq 2, \tag{5.25}$$

$$\mathsf{EC}(\{s_1,s_3\};\{d_2,d_3\}) \geq 2 \quad \text{and} \quad \mathsf{EC}(\{s_2,s_3\};\{d_1,d_3\}) \geq 2. \tag{5.26}$$

5.4 Fundamental Properties of The Precoding-based Framework

5.4.1 Properties of The Precoding-based Framework

In this section, we characterize a few fundamental relationships between the channel and the underlying DAG G, which bridge the gap between the algebraic feasibility of the precoding-based NC problem and the underlying network structure. These properties hold for any precoding-based schemes and can be of benefit to future development of any precoding-based solution. These newly discovered results will later be used to prove the graph-theoretic characterizations of the 3-unicast ANA scheme. In the following subsections, we state Propositions 5.4.1 to 5.4.3, respectively. In Section 5.4.2, we discuss how these results can be applied to the existing 3-unicast ANA scheme.

From Non-Zero Determinant to Linear Independence

Proposition 5.4.1. *Fix an arbitrary value of N. Consider any set of N polynomials $\mathbf{h}(\underline{\mathbf{x}}) = \{h_1(\underline{\mathbf{x}}), ..., h_N(\underline{\mathbf{x}})\}$ and the polynomial matrix $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{N}$ generated from $\mathbf{h}(\underline{\mathbf{x}})$. Then, assuming sufficiently large finite field size q, $\mathsf{det}([\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{N})$ is non-zero polynomial if and only if $\mathbf{h}(\underline{\mathbf{x}})$ is linearly independent.*

The proof of Proposition 5.4.1 is relegated to Appendix J.1.

Remark: Suppose a sufficiently large finite field $\mathbb{F}_q$ is used. If we choose the variables $\underline{\mathbf{x}}^{(1)}$ to $\underline{\mathbf{x}}^{(N)}$ independently and uniformly randomly from $\mathbb{F}_q$, by Schwartz-Zippel lemma, we have $\mathsf{det}([\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{N}) \neq 0$ with close-to-one probability if and only if $\mathbf{h}(\underline{\mathbf{x}})$ is linearly independent.

The implication of Proposition 5.4.1 is as follows. Similar to the seminal work [3], most algebraic characterization of the precoding-based framework involves checking whether or not a determinant is non-zero. For example, the first feasibility condition of (5.3) is equivalent to checking whether or not the determinant of the network transfer matrix is non-zero. Also, (5.15), (5.17), and (5.19) are equivalent to checking whether or not the determinant of the row-invariant matrix $\mathbf{S}_i^{(n)}$ is non-zero. Proposition 5.4.1 says that as long as we can formulate the corresponding matrix in a row-invariant form, then checking whether the determinant is non-zero is equivalent to checking whether the corresponding set of polynomials is linearly independent. As will be shown shortly after, the latter task admits more tractable analysis.

The Subgraph Property of the Precoding-Based Framework

Consider a DAG G and recall the definition of the channel gain $m_{e_1;e_2}(\underline{\mathbf{x}})$ from e_1 to e_2, see Definition 5.2. For a subgraph $G' \subseteq G$ containing e_1 and e_2, let $m_{e_1;e_2}(\underline{\mathbf{x}}')$ denote the channel gain from e_1 to e_2 in G'.

Proposition 5.4.2 (Subgraph Property)**.** *Given a DAG G, consider an arbitrary, but fixed, finite collection of edge pairs, $\{(e_i, e'_i) \in E^2 : i \in I\}$ where I is a finite index set, and consider two arbitrary polynomial functions $f : \mathbb{F}_q^{|I|} \mapsto \mathbb{F}_q$ and $g : \mathbb{F}_q^{|I|} \mapsto \mathbb{F}_q$. Then, $f(\{m_{e_i;e'_i}(\underline{\mathbf{x}}) : \forall i \in I\}) \equiv g(\{m_{e_i;e'_i}(\underline{\mathbf{x}}) : \forall i \in I\})$ if and only if for all subgraphs $G' \subseteq G$ containing all edges in $\{e_i, e'_i : \forall i \in I\}$, $f(\{m_{e_i;e'_i}(\underline{\mathbf{x}}') : \forall i \in I\}) \equiv g(\{m_{e_i;e'_i}(\underline{\mathbf{x}}') : \forall i \in I\})$.*

The proof of Proposition 5.4.2 is relegated to Appendix J.1.

Remark: Proposition 5.4.2 has a similar flavor to the classic results [3] and [6]. More specifically, for the single multicast setting from a source s to the destinations $\{d_j\}$, the transfer matrix $\mathbf{U}_{d_j}\mathbf{M}_{d_j;s}(\underline{\mathbf{x}})\mathbf{V}_s$ from s to d_j is of full rank (i.e., the polynomial $\mathsf{det}(\mathbf{U}_{d_j}\mathbf{M}_{d_j;s}(\underline{\mathbf{x}})\mathbf{V}_s)$ is non-zero in the original graph G) is equivalent to the existence of a subgraph G' (usually being chosen as the subgraph induced by a set of edge-disjoint paths from s to d_j) satisfying the polynomial $\mathsf{det}(\mathbf{U}_{d_j}\mathbf{M}_{d_j;s}(\underline{\mathbf{x}}')\mathbf{V}_s)$ being non-zero.

Compared to Proposition 5.4.1, Proposition 5.4.2 further connects the linear dependence of the polynomials to the subgraph properties of the underlying network. For example, to prove that a set of polynomials over a given arbitrary network is linearly independent, we only need to construct a (much smaller) subgraph and prove that the corresponding set of polynomials is linearly independent.

The Channel Gain Property

Both Propositions 5.4.1 and 5.4.2 have a similar flavor to the classic results of the LNC framework [3]. The following channel gain property, on the other hand, is unique to the precoding-based framework.

Proposition 5.4.3 (The Channel Gain Property)**.** *Consider a DAG G and two distinct edges e_s and e_d. For notational simplicity, we denote $\mathsf{head}(e_s)$ by s and denote $\mathsf{tail}(e_d)$ by d. Then, the following statements must hold (we drop the variables $\underline{\mathbf{x}}$ for shorthand):*

- *If $\mathsf{EC}(s; d) = 0$, then $m_{e_s;e_d} = 0$*
- *If $\mathsf{EC}(s; d) = 1$, then $m_{e_s;e_d}$ is reducible. Moreover, let $N \triangleq |\mathsf{1cut}(s; d)|$ denote the number of 1-edge cuts separating s and d, and we sort the 1-edge cuts by their topological order with e_1 being the most upstream and e_N being the most downstream. The channel gain $m_{e_s;e_d}$ can now be expressed as*

  $$m_{e_s;e_d} = m_{e_s;e_1}\left(\prod_{i=1}^{N-1} m_{e_i;e_{i+1}}\right) m_{e_N;e_d},$$

  and all the polynomial factors $m_{e_s;e_1}$, $\{m_{e_i;e_{i+1}}\}_{i=1}^{N-1}$, and $m_{e_N;e_d}$ are irreducible, and no two of them are equivalent.
- *If $\mathsf{EC}(s; d) \geq 2$ (including ∞), then $m_{e_s;e_d}$ is irreducible.*

The proof of Proposition 5.4.3 is relegated to Appendix J.3.

Remark: Proposition 5.4.3 only considers a channel gain between two distinct edges. If $e_s = e_d$, then by convention [3], we have $m_{e_s;e_d} = 1$.

Proposition 5.4.3 relates the factoring problem of the channel gain polynomial to the graph-theoretic edge cut property. As will be shown afterwards, this observation enables us to tightly connect the algebraic and graph-theoretic conditions for the precoding-based solutions.

5.4.2 Related Work: The 3-unicast ANA Scheme

In this section, we discuss how the properties of the precoding-based framework, Propositions 5.4.1 to 5.4.3, can benefit our understanding of the 3-unicast ANA scheme.

Application of The Properties of The Precoding-based Framework to The 3-unicast ANA Scheme

Proposition 5.4.1 enables us to simplify the feasibility characterization of the 3-unicast ANA scheme in the following way. From the construction in Section 5.2.3, the square matrix $\mathbf{S}_i^{(n)}$ can be written as a row-invariant matrix $\mathbf{S}_i^{(n)} = [\mathbf{h}_i^{(n)}(\underline{\mathbf{x}}^{(t)})]_{t=1}^{(2n+1)}$ for some set of polynomials $\mathbf{h}_i(\underline{\mathbf{x}})$. For example, by (5.11), (5.12), and (5.15) we have $\mathbf{S}_1^{(n)} = [\mathbf{h}_1^{(n)}(\underline{\mathbf{x}}^{(t)})]_{t=1}^{(2n+1)}$ where

$$\begin{aligned} \mathbf{h}_1^{(n)}(\underline{\mathbf{x}}) = \{\, & m_{11}m_{23}m_{32}R^n,\ m_{11}m_{23}m_{32}R^{n-1}L, \\ & \cdots,\ m_{11}m_{23}m_{32}L^n,\ m_{21}m_{13}m_{32}R^n, \\ & m_{21}m_{13}m_{32}R^{n-1}L,\ \cdots,\ m_{21}m_{13}m_{32}RL^{n-1}\,\}. \end{aligned} \tag{5.27}$$

Proposition 5.4.1 implies that (5.15) being true is equivalent to the set of polynomials $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent. Assuming the $G_{3\text{ANA}}$ of interest satisfies (5.20), $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ being linearly independent is equivalent to (5.21) being true. As a result, (5.21) is not only sufficient but also necessary for (5.15) to hold with close-to-one probability. By similar arguments (5.22) (resp. (5.23)) is both necessary and sufficient for (5.17) (resp. (5.19)) to hold with high probability.

Proposition 5.4.2 enables us to find the graph-theoretic equivalent counterparts of (5.21)–(5.23) of the *Conjecture* (p. 3, [41]).

Corollary 5.4.1 (First stated in [41]). *Consider a G_{3ANA} and four indices i_1, i_2, j_1, and j_2 satisfying $i_1 \neq i_2$ and $j_1 \neq j_2$. We have $\mathsf{EC}(\{s_{i_1}, s_{i_2}\}; \{d_{j_1}, d_{j_2}\}) = 1$ if and only if $m_{i_1j_1}m_{i_2j_2} \equiv m_{i_2j_1}m_{i_1j_2}$.*

The main intuition behind Corollary 5.4.1 can be stated as follows. When we have $\mathsf{EC}(\{s_{i_1}, s_{i_2}\}; \{d_{j_1}, d_{j_2}\}) = 1$, one can show that we must have $m_{i_1j_1}(\underline{\mathbf{x}})m_{i_2j_2}(\underline{\mathbf{x}}) = m_{i_2j_1}(\underline{\mathbf{x}})m_{i_1j_2}(\underline{\mathbf{x}})$ by analyzing the underlying graph structure. On the other hand, when we have $\mathsf{EC}(\{s_{i_1}, s_{i_2}\}; \{d_{j_1}, d_{j_2}\}) \neq 1$, we can construct a subgraph G' satisfying $m_{i_1j_1}(\underline{\mathbf{x}}')m_{i_2j_2}(\underline{\mathbf{x}}') \not\equiv m_{i_2j_1}(\underline{\mathbf{x}}')m_{i_1j_2}(\underline{\mathbf{x}}')$. Proposition 5.4.2 thus implies $m_{i_1j_1}(\underline{\mathbf{x}})$ $m_{i_2j_2}(\underline{\mathbf{x}}) \not\equiv m_{i_2j_1}(\underline{\mathbf{x}})m_{i_1j_2}(\underline{\mathbf{x}})$. A detailed proof of Corollary 5.4.1 is relegated to Appendix J.2.

Proposition 5.4.3 can be used to derive the following corollary, which studies the relationship of the channel polynomials m_{ij}.

Corollary 5.4.2. *Given a G_{3ANA}, consider a source s_i to destination d_j channel gain m_{ij}. Then, $\mathsf{GCD}(m_{i_1j_1}, m_{i_2j_2}) \equiv m_{i_2j_2}$ if and only if $(i_1, j_1) = (i_2, j_2)$. Intuitively, any channel gain $m_{i_1j_1}$ from source s_{i_1} to destination d_{j_1} cannot contain another source-destination channel gain $m_{i_2j_2}$ as its factor.*

The intuition behind Corollary 5.4.2 is as follows. For example, suppose we actually have $\mathsf{GCD}(m_{11}, m_{12}) \equiv m_{12}$ and assume that $\mathsf{EC}(\mathsf{head}(e_{s_1}); \mathsf{tail}(e_{d_2})) \geq 2$. Then we must have the d_2-destination edge e_{d_2} being an edge cut separating s_1 and d_1. The reason is that (i) Proposition 5.4.3 implies that any irreducible factor of the channel gain m_{11} corresponds to the channel gain between two consecutive 1-edge cuts separating s_1 and d_1; and (ii) The assumption $\mathsf{EC}(\mathsf{head}(e_{s_1}); \mathsf{tail}(e_{d_2})) \geq 2$ implies that m_{12} is irreducible. Thus (i), (ii), and $\mathsf{GCD}(m_{11}, m_{12}) \equiv m_{12}$ together imply that $e_{d_2} \in \mathsf{1cut}(s_1; d_1)$. This, however, contradicts the assumption of $|\mathsf{Out}(d_2)| = 0$ for any 3-unicast ANA network $G_{3\text{ANA}}$. The detailed proof of Corollary 5.4.2, which studies more general case in which $\mathsf{EC}(\mathsf{head}(e_{s_1}); \mathsf{tail}(e_{d_2})) = 1$, is relegated to Appendix J.2.

5.5 Chapter Summary

In this chapter, we define and discuss the proposed Precoding-based Framework. In Section 5.1, the related graph-theoretic notations are firstly defined. We then algebraically formulate the Precoding-based framework in Section 5.2. For the subsequent subsections, the comparison to the classic LNC framework is discussed, with the introductions of the recent wireless applications proposed by [25, 26, 40, 41]. The need for the deeper understandings between the network channel gain and the underlying graph structure is further motivated and our contributions are summarized in Section 5.3. The corresponding fundamental properties of the proposed Precoding-based Framework are provided in Section 5.4 including how they can benefit to understand the 3-unicast ANA problem.

6. GRAPH-THEORETIC CHARACTERIZATION OF THE 3-UNICAST ANA SCHEME

In Section 5.4, we investigated the basic relationships between the network channel gain polynomials and the underlying DAG G for arbitrary precoding-based solutions. In this chapter, we turn our attention to a specific precoding-based solution, the 3-unicast ANA scheme, and characterize graph-theoretically its feasibility conditions.

6.1 New Graph-Theoretic Notations and The Corresponding Properties

We begin by defining some new notations. Consider three indices i, j, and k in $\{1,2,3\}$ satisfying $j\neq k$ but i may or may not be equal to j (resp. k). Given a $G_{3\text{ANA}}$, define:

$$\overline{S}_{i;\{j,k\}} \triangleq \mathsf{1cut}(s_i;d_j)\cap \mathsf{1cut}(s_i;d_k)\backslash\{e_{s_i}\}$$
$$\overline{D}_{i;\{j,k\}} \triangleq \mathsf{1cut}(s_j;d_i)\cap \mathsf{1cut}(s_k;d_i)\backslash\{e_{d_i}\}$$

as the 1-edge cuts separating s_i and $\{d_j,d_k\}$ minus the s_i-source edge e_{s_i} and the 1-edge cuts separating $\{s_j,s_k\}$ and d_i minus the d_i-destination edge e_{d_i}. When the values of indices are all distinct, we use $\overline{S}_i$ (resp. $\overline{D}_i$) as shorthand for $\overline{S}_{i;\{j,k\}}$ (resp. $\overline{D}_{i;\{j,k\}}$). The following lemmas prove some topological relationships between the edge sets $\overline{S}_i$ and $\overline{D}_j$ and the corresponding proofs are relegated to Appendix K.

Lemma 6.1.1. *For all $i\neq j$, $e'\in\overline{S}_i$, and $e''\in\overline{D}_j$, one of the following three statements is true: $e'\prec e''$, $e'\succ e''$, or $e'=e''$.*

Lemma 6.1.2. *For any distinct i, j, and k in $\{1,2,3\}$, we have $(\overline{D}_i\cap\overline{D}_j)\subset\overline{S}_k$.*

Lemma 6.1.3. *For all $i\neq j$, $e'\in\overline{S}_i\backslash\overline{D}_j$, and $e''\in\overline{D}_j$, we have $e'\prec e''$.*

Lemma 6.1.4. *For any distinct i, j, and k in $\{1,2,3\}$, $\overline{D}_j \cap \overline{D}_k \neq \emptyset$ if and only if both $\overline{S}_i \cap \overline{D}_j \neq \emptyset$ and $\overline{S}_i \cap \overline{D}_k \neq \emptyset$.*

Lemma 6.1.5. *For all $i \neq j$ and $e'' \in \overline{D}_i \cap \overline{D}_j$, if $\overline{S}_i \cap \overline{S}_j \neq \emptyset$, then there exists $e' \in \overline{S}_i \cap \overline{S}_j$ such that $e' \preceq e''$.*

Lemma 6.1.6. *Consider four indices i, j_1, j_2, and j_3 taking values in $\{1,2,3\}$ for which the values of j_1, j_2 and j_3 must be distinct and i is equal to one of j_1, j_2 and j_3. If $\overline{S}_{i;\{j_1,j_2\}} \neq \emptyset$ and $\overline{S}_{i;\{j_1,j_3\}} \neq \emptyset$, then the following three statements are true: (i) $\overline{S}_{i;\{j_1,j_2\}} \cap \overline{S}_{i;\{j_1,j_3\}} \neq \emptyset$; (ii) $\overline{S}_{i;\{j_2,j_3\}} \neq \emptyset$; and (iii) $\overline{S}_i \neq \emptyset$.*

Remark: All the above lemmas are purely graph-theoretic. If we swap the roles of sources and destinations, then we can also derive the (s,d)-*symmetric version* of these lemmas. For example, the (s,d)-symmetric version of Lemma 6.1.2 becomes $(\overline{S}_i \cap \overline{S}_j) \subseteq \overline{D}_k$. The (s,d)-symmetric version of Lemma 6.1.5 is: For all $i \neq j$ and $e'' \in \overline{S}_i \cap \overline{S}_j$, if $\overline{D}_i \cap \overline{D}_j \neq \emptyset$, then there exists $e' \in \overline{D}_i \cap \overline{D}_j$ such that $e' \succeq e''$.

Lemmas 6.1.1 to 6.1.6 discuss the topological relationship between the edge sets $\overline{S}_i$ and $\overline{D}_j$. The following lemma establishes the relationship between $\overline{S}_i$ (resp. $\overline{D}_j$) and the channel gains.

Lemma 6.1.7. *Given a G_{3ANA}, consider the corresponding channel gains as defined in Section II-D. Consider three indices i, j_1, and j_2 taking values in $\{1,2,3\}$ for which the values of j_1 and j_2 must be distinct. Then, $\mathsf{GCD}(m_{ij_1}, m_{ij_2}) \equiv 1$ if and only if $\overline{S}_{i;\{j_1,j_2\}} = \emptyset$. Symmetrically, $\mathsf{GCD}(m_{j_1 i}, m_{j_2 i}) \equiv 1$ if and only if $\overline{D}_{i;\{j_1,j_2\}} = \emptyset$.*

The proof of Lemma 6.1.7 is relegated to Appendix K.

6.2 The Graph-Theoretic Characterization of $L(\underline{\mathbf{x}}) \not\equiv R(\underline{\mathbf{x}})$

A critical condition of the 3-unicast ANA scheme [40, 41] is the assumption that $L(\underline{\mathbf{x}}) \not\equiv R(\underline{\mathbf{x}})$, which is the fundamental reason why the Vandermonde precoding matrix $\overline{\mathbf{V}}_i$ is of full (column) rank. However, for some networks we may have $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$, for which the 3-unicast ANA scheme does not work (see Fig. 5.1). Next, we prove the following graph-theoretic condition that fully characterizes whether $L(\underline{\mathbf{x}})\equiv R(\underline{\mathbf{x}})$.

Proposition 6.2.1. *For a given G_{3ANA}, we have $L(\underline{\mathbf{x}})\equiv R(\underline{\mathbf{x}})$ if and only if there exists a pair of distinct indices $i,j\in\{1,2,3\}$ satisfying both $\overline{S}_i\cap\overline{S}_j\neq\emptyset$ and $\overline{D}_i\cap\overline{D}_j\neq\emptyset$.*

Proof of the "$\Leftarrow$" direction of Proposition 6.2.1: Without loss of generality, suppose $\overline{S}_1\cap\overline{S}_2\neq\emptyset$ and $\overline{D}_1\cap\overline{D}_2\neq\emptyset$ (i.e., $i=1$ and $j=2$). By Lemma 6.1.5, we can find two edges $e'\in\overline{S}_1\cap\overline{S}_2$ and $e''\in\overline{D}_1\cap\overline{D}_2$ such that $e'\preceq e''$. Also note that Lemma 6.1.2 and its (s,d)-symmetric version imply that $e'\in\overline{D}_3$ and $e''\in\overline{S}_3$. Then by Proposition 5.4.3, the channel gains $m_{ij}(\underline{\mathbf{x}})$ for all $i\neq j$ can be expressed by (we omit the variables $\underline{\mathbf{x}}$ for simplicity):

$$\begin{aligned} &m_{13}=m_{e_{s_1};e'}\,m_{e';e_{d_3}} && m_{12}=m_{e_{s_1};e'}\,m_{e';e''}\,m_{e'';e_{d_2}}\\ &m_{32}=m_{e_{s_3};e''}\,m_{e'';e_{d_2}} && m_{23}=m_{e_{s_2};e'}\,m_{e';e_{d_3}}\\ &m_{21}=m_{e_{s_2};e'}\,m_{e';e''}\,m_{e'';e_{d_1}} && m_{31}=m_{e_{s_3};e''}\,m_{e'';e_{d_1}} \end{aligned}$$

where the expressions of m_{12} and m_{21} are derived based on the facts that $e'\preceq e''$ and $\{e',e''\}\subset\mathsf{1cut}(s_1;d_2)\cap\mathsf{1cut}(s_2;d_1)$. By plugging in the above 6 equalities to the definitions of $L=m_{13}m_{32}m_{21}$ and $R=m_{12}m_{23}m_{31}$, we can easily verify that $L\equiv R$. The proof of this direction is complete. ■

Remark: In the example of Fig. 5.1, one can easily see that $e'\in\overline{S}_1\cap\overline{S}_2$ and $e''\in\overline{D}_1\cap\overline{D}_2$. Hence, the above proof shows that the example network in Fig. 5.1 satisfies $L(\underline{\mathbf{x}})\equiv R(\underline{\mathbf{x}})$ without actually computing the polynomials $L(\underline{\mathbf{x}})$ and $R(\underline{\mathbf{x}})$.

We will now focus on proving the necessity. Before proceeding, we state and prove the following lemma.

Lemma 6.2.1. *If the G_{3ANA} of interest satisfies $L(\underline{\mathbf{x}})\equiv R(\underline{\mathbf{x}})$, then $\overline{S}_i\neq\emptyset$ and $\overline{D}_j\neq\emptyset$ for all i and j, respectively.*

Proof of Lemma 6.2.1: We prove this by contradiction. Suppose $\overline{S}_1=\emptyset$. Denote the most upstream 1-edge cut separating $\mathsf{head}(e_{s_1})$ and d_2 by e_{12} (we have at least the d_2-destination edge e_{d_2}). Also denote the most upstream 1-edge cut separating $\mathsf{head}(e_{s_1})$ and d_3 by e_{13} (we have at least the d_3-destination edge e_{d_3}). Since $\overline{S}_1{=}\emptyset$ and by the definition of the 3-unicast ANA network, it is obvious that $e_{12} \neq e_{13}$. Moreover, both of the two polynomials $m_{e_{s_1};e_{12}}$ (a factor of m_{12}) and $m_{e_{s_1};e_{13}}$ (a factor of m_{13}) are irreducible and non-equivalent to each other. Therefore, these two polynomials are coprime. If we plug in the two polynomials into $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$, then it means that one of the following three cases must be true: (i) $m_{e_{13};e_{d_3}}$ contains $m_{e_{s_1};e_{12}}$ as a factor; (ii) m_{32} contains $m_{e_{s_1};e_{12}}$ as a factor; or (iii) m_{21} contains $m_{e_{s_1};e_{12}}$ as a factor. However, (i), (ii), and (iii) cannot be true as $|\mathsf{In}(s_1)| = 0$ and by Proposition 5.4.3. The proof is thus complete by applying symmetry. □

Proof of the "⇒" direction of Proposition 6.2.1: Suppose the $G_{3\text{ANA}}$ of interest satisfies $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$. By Lemma 6.2.1, we know that $\overline{S}_i {\neq} \emptyset$ and $\overline{D}_j {\neq} \emptyset$ for all i and j. Then it is obvious that $\mathsf{EC}(\mathsf{head}(e_{s_i}); \mathsf{tail}(e_{d_j})) = 1$ for all $i \neq j$ because if (for example) $\mathsf{EC}(\mathsf{head}(e_{s_1}); \mathsf{tail}(e_{d_2})) \geq 2$ then both $\overline{S}_1$ and $\overline{D}_2$ will be empty by definition. Thus by Proposition 5.4.3, we can express each channel gain m_{ij} $(i \neq j)$ as a product of irreducibles, each corresponding to the channel gain between two consecutive 1-edge cuts (including e_{s_i} and e_{d_j}) separating s_i and d_j. We now consider two cases.

Case 1: $\overline{S}_i \cap \overline{D}_j = \emptyset$ for some $i \neq j$. Assume without loss of generality that $\overline{S}_2 \cap \overline{D}_1 {=} \emptyset$ (i.e., $i{=}2$ and $j{=}1$). Let e_2^* denote the most downstream edge in $\overline{S}_2$ and let e_1^* denote the most upstream edge in $\overline{D}_1$. Since $\overline{S}_2 \cap \overline{D}_1 {=} \emptyset$, the edge e_2^* must not be in $\overline{D}_1$. By Lemma 6.1.3, we have $e_2^* \prec e_1^*$.

For the following, we will prove $\{e_2^*, e_1^*\} \subset \mathsf{1cut}(s_1; d_2)$. We first notice that by definition, $e_2^* \in \overline{S}_2 \subset \mathsf{1cut}(s_2; d_1)$ and $e_1^* \in \overline{D}_1 \subset \mathsf{1cut}(s_2; d_1)$. Hence by Proposition 5.4.3, we can express m_{21} as $m_{21} = m_{e_{s_2};e_2^*} m_{e_2^*;e_1^*} m_{e_1^*;e_{d_1}}$. Note that by our construction $e_2^* \prec e_1^*$ we have $m_{e_2^*;e_1^*} \not\equiv 1$.

We now claim $\mathsf{GCD}(\,m_{e_2^*;e_1^*},\, m_{23}m_{31}) \equiv 1$, i.e., $m_{23}m_{31}$ cannot contain any factor of $m_{e_2^*;e_1^*}$. We will prove this claim by contradiction. Suppose $\mathsf{GCD}(\,m_{e_2^*;e_1^*},\, m_{23}) \not\equiv 1$, i.e., m_{23} contains an irreducible factor of $m_{e_2^*;e_1^*}$. Since that factor is also a factor of m_{21}, by Proposition 5.4.3, there must exist at least one edge e satisfying (i) $e_2^* \prec e \preceq e_1^*$;

and (ii) $e \in \mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_2; d_3)$. These jointly implies that we have an $\overline{S}_2$ edge in the downstream of e_2^*. This, however, contradicts the assumption that e_2^* is the most downstream edge of $\overline{S}_2$. By a symmetric argument, we can also show that m_{31} must not contain any irreducible factor of $m_{e_2^*;e_1^*}$. The proof of the claim $\mathsf{GCD}(\,m_{e_2^*;e_1^*},\, m_{23}m_{31}) \equiv 1$ is complete. Since the assumption $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ implies that $\mathsf{GCD}(\,m_{e_2^*;e_1^*},\, R) = m_{e_2^*;e_1^*}$, we must have $\mathsf{GCD}(\,m_{e_2^*;e_1^*},\, m_{12}) = m_{e_2^*;e_1^*}$. This implies by Proposition 5.4.3 that $\{e_2^*, e_1^*\} \subset \mathsf{1cut}(s_1; d_2)$.

For the following, we will prove that $e_2^* \in \mathsf{1cut}(s_1; d_3)$. To that end, we consider the factor $m_{e_2^*;e_{d_3}}$ of the channel gain m_{23}. This is possible by Proposition 5.4.3 because $e_2^* \in \overline{S}_2 \subset \mathsf{1cut}(s_2; d_3)$. Then similarly following the above discussion, we must have $\mathsf{GCD}(\,m_{21},\, m_{e_2^*;e_{d_3}}) \equiv 1$ otherwise there will be an $\overline{S}_2$ edge in the downstream of e_2^*. Since the assumption $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ means that $\mathsf{GCD}(\,L,\, m_{e_2^*;e_{d_3}}) = m_{e_2^*;e_{d_3}}$, this further implies that $\mathsf{GCD}(\,m_{13}m_{32},\, m_{e_2^*;e_{d_3}}) = m_{e_2^*;e_{d_3}}$.

Now consider the most upstream $\mathsf{1cut}(s_2; d_3)$ edge that is in the downstream of e_2^*, and denote it as e_u (we have at least the d_3-destination edge e_{d_3}). Obviously, $e_2^* \prec e_u \preceq e_{d_3}$ and $m_{e_2^*;e_u}$ is an irreducible factor of $m_{e_2^*;e_{d_3}}$. Then we must have $\mathsf{GCD}(\,m_{32},\, m_{e_2^*;e_u}) \equiv 1$ and the reason is as follows. If not, then by $m_{e_2^*;e_u}$ being irreducible we have $e_2^* \in \mathsf{1cut}(s_3; d_2)$. Then every path from s_3 to $\mathsf{tail}(e_1^*)$ must use e_2^*, otherwise s_3 can reach e_1^* without using e_2^* and finally arrive at d_2 since e_1^* can reach d_2 (we showed in the above discussion that $e_1^* \in \mathsf{1cut}(s_1; d_2)$). This contradicts the previously constructed $e_2^* \in \mathsf{1cut}(s_3; d_2)$. Therefore, we must have $e_2^* \in \mathsf{1cut}(s_3; \mathsf{tail}(e_1^*))$. Since $e_1^* \in \overline{D}_1 \subset \mathsf{1cut}(s_3; d_1)$, this in turn implies that e_2^* is also an 1-edge cut separating s_3 and d_1. However, note by the assumption that $e_2^* \in \overline{S}_2 \subset \mathsf{1cut}(s_2; d_1)$. Thus, e_2^* will belong to $\overline{D}_1$, which contradicts the assumption that e_1^* is the most upstream $\overline{D}_1$ edge. We thus have proven $\mathsf{GCD}(\,m_{32},\, m_{e_2^*;e_u}) \equiv 1$. Since we showed that $\mathsf{GCD}(\,m_{13}m_{32},\, m_{e_2^*;e_{d_3}}) = m_{e_2^*;e_{d_3}}$, this further implies that the irreducible factor $m_{e_2^*;e_u}$ of $m_{e_2^*;e_{d_3}}$ must be contained by m_{13} as a factor. Therefore, we have proven that $e_2^* \in \mathsf{1cut}(s_1; d_3)$. Symmetrically applying the above argument using the factor $m_{e_{s_3};e_1^*}$ of the channel gain m_{31}, we can also prove that $e_1^* \in \mathsf{1cut}(s_3; d_2)$.

Thus far, we have proven that $e_2^* \in \mathsf{1cut}(s_1;d_2)$ and $e_2^* \in \mathsf{1cut}(s_1;d_3)$. However, $e_2^* = e_{s_1}$ is not possible since e_2^*, by our construction, is a downstream edge of e_{s_2} but e_{s_1} is not (since $|\mathsf{In}(s_1)|=0$). As a result, we have proven $e_2^* \in \overline{S}_1$. Recall that e_2^* was chosen as one edge in $\overline{S}_2$. Therefore, $\overline{S}_1 \cap \overline{S}_2 \neq \emptyset$. Similarly, we can also prove that $e_1^* \in \overline{D}_1 \cap \overline{D}_2$ and thus $\overline{D}_1 \cap \overline{D}_2 \neq \emptyset$. The proof of **Case 1** is complete.

Case 2: $\overline{S}_i \cap \overline{D}_j \neq \emptyset$ for all $i \neq j$. By Lemma 6.1.4 and its (s,d)-symmetric version, we must have $\overline{S}_i \cap \overline{S}_j \neq \emptyset$ and $\overline{D}_i \cap \overline{D}_j \neq \emptyset \; \forall\, i \neq j$. The proof of **Case 2** is complete. ■

6.3 The Graph-Theoretic Conditions of the Feasibility of the 3-unicast ANA Scheme

Proposition 6.2.1 provides the graph-theoretic condition that characterizes whether or not the $G_{3\text{ANA}}$ of interest satisfies the algebraic condition of (5.20), which implies that (5.14), (5.16), and (5.18) hold simultaneously with close-to-one probability. However, to further ensure the feasibility of the 3-unicast ANA scheme, $\mathsf{det}(\mathbf{S}_i^{(n)})$ must be non-zero polynomial (see (5.15), (5.17), and (5.19)) for all $i \in \{1,2,3\}$. As a result, we need to prove the graph-theoretic characterization for the inequalities $\mathsf{det}(\mathbf{S}_i^{(n)}) \neq 0$. Note by Proposition 5.4.1 that the condition $\mathsf{det}(\mathbf{S}_i^{(n)}) \neq 0$ is equivalent to for all $i \in \{1,2,3\}$ the set of polynomials $\mathbf{h}_i^{(n)}(\underline{\mathbf{x}})$ is linearly independent, where $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is defined in (5.27) and $\mathbf{h}_2^{(n)}(\underline{\mathbf{x}})$ and $\mathbf{h}_3^{(n)}(\underline{\mathbf{x}})$ are defined as follows:

$$\begin{aligned}\mathbf{h}_2^{(n)}(\underline{\mathbf{x}}) = \{\, & m_{22}m_{13}m_{32}R^n,\; m_{22}m_{13}m_{32}R^{n-1}L, \\ & \cdots, m_{22}m_{13}m_{32}RL^{n-1},\; m_{12}m_{23}m_{32}R^n, \\ & m_{12}m_{23}m_{32}R^{n-1}L,\; \cdots,\; m_{12}m_{23}m_{32}L^n \,\},\end{aligned} \tag{6.1}$$

$$\begin{aligned}\mathbf{h}_3^{(n)}(\underline{\mathbf{x}}) = \{\, & m_{33}m_{12}m_{23}R^{n-1}L,\; \cdots, \\ & m_{33}m_{12}m_{23}RL^{n-1},\; m_{33}m_{12}m_{23}L^n, \\ & m_{13}m_{23}m_{32}R^n,\; m_{13}m_{23}m_{32}R^{n-1}L, \\ & \cdots,\; m_{13}m_{23}m_{32}L^n \,\}.\end{aligned} \tag{6.2}$$

Thus in this subsection, we prove a graph-theoretic condition that characterizes the linear independence of $\mathbf{h}_i^{(n)}(\underline{\mathbf{x}})$ for all $i\in\{1,2,3\}$ when $n=1$ and $n\geq 2$, respectively. Consider the following graph-theoretic conditions:

$$\overline{S}_i \cap \overline{S}_j = \emptyset \text{ or } \overline{D}_i \cap \overline{D}_j = \emptyset \;\; \forall\, i,j \in \{1,2,3\},\, i\neq j, \tag{6.3}$$

$$\mathsf{EC}(\{s_1,s_2\};\{d_1,d_3\})\geq 2,\ \mathsf{EC}(\{s_1,s_3\};\{d_1,d_2\})\geq 2, \tag{6.4}$$

$$\mathsf{EC}(s_1;d_1)\geq 1 \text{ on } G_{3\text{ANA}}\backslash\big\{\mathsf{upstr}\big((\overline{S}_2\cap\overline{D}_3)\cup(\overline{S}_3\cap\overline{D}_2)\big)\big\}, \tag{6.5}$$

$$\mathsf{EC}(\{s_1,s_2\};\{d_2,d_3\})\geq 2,\ \mathsf{EC}(\{s_2,s_3\};\{d_1,d_2\})\geq 2, \tag{6.6}$$

$$\mathsf{EC}(s_2;d_2)\geq 1 \text{ on } G_{3\text{ANA}}\backslash\big\{\mathsf{upstr}\big((\overline{S}_1\cap\overline{D}_3)\cup(\overline{S}_3\cap\overline{D}_1)\big)\big\}, \tag{6.7}$$

$$\mathsf{EC}(\{s_1,s_3\};\{d_2,d_3\})\geq 2,\ \mathsf{EC}(\{s_2,s_3\};\{d_1,d_3\})\geq 2, \tag{6.8}$$

$$\mathsf{EC}(s_3;d_3)\geq 1 \text{ on } G_{3\text{ANA}}\backslash\big\{\mathsf{upstr}\big((\overline{S}_1\cap\overline{D}_2)\cup(\overline{S}_2\cap\overline{D}_1)\big)\big\}. \tag{6.9}$$

Note that (i) (6.3) is equivalent to $L(\underline{\mathbf{x}})\not\equiv R(\underline{\mathbf{x}})$ by Proposition 6.2.1; (ii) (6.4), (6.6), and (6.8) are equivalent to (5.24) to (5.26) by Corollary 5.4.1; and (iii) (6.5), (6.7), and (6.9) are the new conditions that help characterize (5.21) to (5.23).

To further simplify the analysis, we consider the following set of polynomials:

$$\begin{aligned}\mathbf{k}_1^{(n)}(\underline{\mathbf{x}}) = \{\, & m_{11}m_{23}m_{31}L^n,\ m_{11}m_{23}m_{31}L^{n-1}R, \\ & \cdots, m_{11}m_{23}m_{31}LR^{n-1},\ m_{21}m_{13}m_{31}L^n, \\ & m_{21}m_{13}m_{31}L^{n-1}R,\ \cdots,\ m_{21}m_{13}m_{31}R^n\,\},\end{aligned} \tag{6.10}$$

where $\mathbf{k}_1^{(n)}(\underline{\mathbf{x}})$ is obtained by swapping the roles of s_1 and s_2 (resp. s_3), and the roles of d_1 and d_2 (resp. d_3) to the expression of $\mathbf{h}_2^{(n)}(\underline{\mathbf{x}})$ in (6.1) (resp. $\mathbf{h}_3^{(n)}(\underline{\mathbf{x}})$ in (6.2)). Note that $R=m_{12}m_{23}m_{31}$ becomes $L=m_{13}m_{32}m_{21}$ and vice versa by such swap operation. Once we characterize the graph-theoretic conditions for the linear independence of $\mathbf{k}_1^{(n)}(\underline{\mathbf{x}})$, then the characterization for $\mathbf{h}_2^{(n)}(\underline{\mathbf{x}})$ and $\mathbf{h}_3^{(n)}(\underline{\mathbf{x}})$ being linearly independent will be followed symmetrically.[1]

Proposition 6.3.1. *For a given G_{3ANA}, when $n=1$, we have*

[1] In Section 5.2.3, (s_1,d_1)-pair was chosen to achieve larger rate than other pairs when aligning the interference. Thus the feasibility characterization for the other transmission pairs, (s_2,d_2) and (s_3,d_3) who achieve the same rate, becomes symmetric.

(H1) $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$ *is linearly independent if and only if* G_{3ANA} *satisfies (6.3) and (6.4).*

(K1) $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ *is linearly independent if and only if* G_{3ANA} *satisfies (6.3), (6.4), and (6.5).*

Moreover when $n \geq 2$*, we have*

(H2) $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ *is linearly independent if and only if* G_{3ANA} *satisfies (6.3), (6.4), and (6.5).*

(K2) $\mathbf{k}_1^{(n)}(\underline{\mathbf{x}})$ *is linearly independent if and only if* G_{3ANA} *satisfies (6.3), (6.4), and (6.5).*

Remark: Proposition 6.3.1 proves that the conjecture in [41] holds only for the linearly independent $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$. In general, it is no longer true for the case of $n \geq 2$ and even for $n = 1$. This coincides with the recent results [62], which show that for the case of $n \geq 2$, the conjecture in [41] no longer holds.

Proof of Proposition 6.3.1: Similar to most graph-theoretic proofs, the proofs of (H1), (K1), (H2), and (K2) involve detailed discussion of several subcases. To structure our proof, we first define the following logic statements. Each statement could be true or false. We will later use these statements to complete the proof.

- **H1:** $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent for $n=1$.
- **K1:** $\mathbf{k}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent for $n=1$.
- **H2:** $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent for some $n \geq 2$.
- **K2:** $\mathbf{k}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent for some $n \geq 2$.
- **LNR:** $L(\underline{\mathbf{x}}) \not\equiv R(\underline{\mathbf{x}})$.
- **G1:** $m_{11}m_{23} \not\equiv m_{21}m_{13}$ and $m_{11}m_{32} \not\equiv m_{31}m_{12}$.
- **G2:** $\mathsf{EC}(s_1; d_1) \geq 1$ on $G_{\text{3ANA}} \backslash \left\{\mathsf{upstr}\left((\overline{S}_2 \cap \overline{D}_3) \cup (\overline{S}_3 \cap \overline{D}_2)\right)\right\}$.

One can clearly see that proving Statement (H1) is equivalent to proving "**LNR** $\wedge$ **G1** $\Leftrightarrow$ **H1**" where "$\wedge$" is the AND operator. Similarly, proving Statements (K1), (H2), and (K2) is equivalent to proving "**LNR** $\wedge$ **G1** $\wedge$ **G2** $\Leftrightarrow$ **K1**", "**LNR** $\wedge$ **G1** $\wedge$ **G2** $\Leftrightarrow$ **H2**", and "**LNR** $\wedge$ **G1** $\wedge$ **G2** $\Leftrightarrow$ **K2**", respectively.

The reason why we use the notation of "logic statements" (e.g., **H1**, **LNR**, etc.) is that it enables us to break down the overall proof into proving several smaller "logic relationships" (e.g, "$\mathbf{LNR} \wedge \mathbf{G1} \Leftrightarrow \mathbf{H1}$", etc.) and later assemble all the logic relationships to derive the final results. The interested readers can thus separate the verification of the proof of each individual logic relationship from the examination of the overall structure of the proof of the main results. The proof of each logic relationship is kept no longer than one page and is independent from the proof of any other logic relationship. This allows the readers to set their own pace when going through the proofs.

To give an insight how the proof works, here we provide the proof of "$\mathbf{LNR} \wedge \mathbf{G1} \Leftarrow \mathbf{H1}$" at the bottom. All the other proofs are relegated to the appendices. Specifically, we provide the general structured proofs for the necessity direction "$\Leftarrow$" in Appendix M. Applying this result, the proofs of "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Leftarrow \mathbf{H2}, \mathbf{K1}, \mathbf{K2}$" are provided in Appendix M.3. Similarly, the general structured proofs for the sufficiency direction "$\Rightarrow$" is provided in Appendix N. The proofs of "$\mathbf{LNR} \wedge \mathbf{G1} \Rightarrow \mathbf{H1}$" and "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{K1}, \mathbf{H2}, \mathbf{K2}$" are provided in Appendix N.4.

The proof of "$\mathbf{LNR} \wedge \mathbf{G1} \Leftarrow \mathbf{H1}$*":* We prove the following statement instead: $(\neg\,\mathbf{LNR}) \vee (\neg\,\mathbf{G1}) \Rightarrow (\neg\,\mathbf{H1})$ where $\neg$ is the NOT logic operator and "$\vee$" is the OR operator. From the expression of $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ in (5.27), consider $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$ which contains 3 polynomials:

$$\mathbf{h}_1^{(1)}(\underline{\mathbf{x}}) = \{\, m_{11}m_{23}m_{32}R,\ m_{11}m_{23}m_{32}L,\ m_{21}m_{13}m_{32}R \,\}. \tag{6.11}$$

Suppose $G_{3\text{ANA}}$ satisfies $(\neg\,\mathbf{LNR}) \vee (\neg\,\mathbf{G1})$, which means $G_{3\text{ANA}}$ satisfies either $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ or $m_{11}m_{23} \equiv m_{21}m_{13}$ or $m_{11}m_{32} \equiv m_{31}m_{12}$. If $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$, then we notice that $m_{11}m_{23}m_{32}R \equiv m_{11}m_{23}m_{32}L$ and $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$, defined in (6.11), is thus linearly dependent. If $m_{11}m_{23} \equiv m_{21}m_{13}$, then we notice that $m_{11}m_{23}m_{32}R \equiv m_{21}m_{13}m_{32}R$. Similarly if $m_{11}m_{32} \equiv m_{31}m_{12}$, then we have $m_{11}m_{23}m_{32}L \equiv m_{21}m_{13}m_{32}R$. The proof is thus complete. ■

6.4 Chapter Summary

In this chapter, we characterize the graph-theoretic conditions for the feasibility of the 3-unicast ANA scheme. In Section 6.1, we first define the new graph-theoretic notations that are useful for the characterization. In Section 6.2, we then characterize the first algebraic feasibility condition of the 3-unicast ANA scheme in a graph-theoretic sense. The full graph-theoretic characterization of all the remaining algebraic feasibility conditions are fully described in Section 6.3, where the main proofs can be found in Appendices L to N.

7. CONCLUSION AND FUTURE WORK

In this thesis, we first studied the 3-node wireless packet erasure network that incorporates feedback, NC encoding/decoding descriptions, and scheduling decisions all together. In this model, we considered the most general traffic setting: six private-information flows and three common-information flows in total, and characterized the corresponding 9-dimensional Shannon capacity region within a gap that is inversely proportional to the packet size. The gap can be attributed to exchanging reception status (ACK/NACK) between three nodes. When the causal feedback can be communicated for free, we further proved that the proposed simple LNC inner bound achieves the capacity. In the second part, we studied the smart repeater packet erasure network and effectively bracketed the LNC capacity region by proposing the outer and inner bounds. The outer bound was developed based following the principles of the proposed Space-based Framework, which can jointly optimize the LNC operations and scheduling decisions simultaneously for the best possible LNC throughput. For an inner bound, we have identified a new way of encoding packet mixtures that is critical to approach the LNC capacity in a close-to-optimal sense. In the third part, we studied the general class of precoding-based LNC schemes in wireline directed acyclic integer-capacity networks. The Precoding-based Framework focuses on designing the precoding and decoding mappings at the sources and destinations while using randomly generated local encoding kernels within the network. One example of the precoding-based structure is the 3-unicast ANA scheme, originally proposed in [40, 41]. In this thesis, we have identified new graph-theoretic relationships for the precoding-based NC solutions, and based on the findings on the general precoding-based NC, we have further characterized the graph-theoretic feasibility conditions of the 3-unicast ANA scheme.

In the 3-node wireless network setting, when the casual ACK/NACK feedback exchanges must be through the forward erasure channel (Scenario 2), we have observed in Section 3.3 that the fully-connected assumption is critical to operate the capacity-achieving LNC scheme. In other words, when the network is not fully-connected, the proposed LNC strategy might not be in a right play. Therefore, it would be an interesting extension to study how the actual capacity region is going to be when the network admits such asymmetric feedback scenario. For the smart repeater network setting, we have described the corresponding LNC capacity region. However, the true capacity outer bound based on information-theoretic arguments is still open to be described. Considering the fact that "Linearity" was shown not sufficient to achieve the multi-session capacity in general [36], it would be an interesting future work to see whether the regions described by LNC operations and by the information-theoretic arguments can be matched or not.

REFERENCES

REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] S.-Y. Li, R. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, February 2003.

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

[4] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41st Annual Allerton Conf. on Comm., Contr., and Computing.*, Monticello, Illinois, USA, October 2003.

[5] P. Sanders, S. Egner, and L. Tolhuizen, "Polynomial time algorithms for network information flow," in *Proc. 15th Annual ACM Symp. Parallel Algorithms and Architectures (SPAA'03)*, San Diego, California, USA, June 2003.

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

[7] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "Xors in the air: Practical wireless network coding," in *Proc. ACM SIGCOMM*, Pisa, Italy, September 2006, pp. 243–254.

[8] D. Koutsonikolas, C.-C. Wang, and Y. Hu, "Efficient network coding based opportunistic routing through cumulative coded acknowledgment," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1368–1381, 2011.

[9] C. Ning and R. W. Yeung, "Secure network coding," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Lausanne, Switzerland, July 2002, p. 323.

[10] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, and M. Medard, "Resilient network coding in the presence of byzantine adversaries," in *Proc. 26th IEEE Conf. Comput. Commun. (INFOCOM)*, Barcelona, Spain, May 2007, pp. 616–624.

[11] M. Luby, "Lt codes," in *Proc. IEEE Symposium on Foundations of Computer Science*, Vancouver, Canada, November 2002, pp. 271–280.

[12] C. Hausl and P. Dupraz, "Joint network-channel coding for the multiple-access relay channel," in *Proc. IEEE Sensor and Ad Hoc Communications and Networks (SECON)*, Reston, VA, USA, September 2006, pp. 817–822.

[13] D. Silva and F. R. Kschischang, "Using rank-metric codes for error correction in random network coding," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Nice, France, June 2007, pp. 796–800.

[14] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3579–3591, August 2008.

[15] M. Shao, X. Wu, and N. Sarshar, "Rainbow network flow with network coding," in *Proc. 4th Workshop on Network Coding, Theory and Applications (NetCod)*, Hong Kong, China, January 2008, pp. 1–6.

[16] M. Kim, D. Lucani, X. Shi, F. Zhao, and M. Medard, "Network coding for multi-resolution multicast," in *Proc. 29th IEEE Conf. Comput. Commun. (INFOCOM)*, San Diego, California, USA, March 2009, pp. 1–9.

[17] A. Dimakis, P. Godfrey, M. Wainwright, and K. Ramchandran, "Netowrk coding for distributed storage systems," in *Proc. 26th IEEE Conf. Comput. Commun. (INFOCOM)*, Anchorage, Alaska, USA, May 2007, pp. 2000d–2008.

[18] Y. Wu, "Existence and construction of capacity-achieving network codes for distributed storage," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Seoul, Korea, July 2009, pp. 1150–1154.

[19] S. E. Rouayheb, A. Sprintson, and C. Georghiades, "On the index coding problem and its relation to network coding and matroid theory," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3187–3195, July 2010.

[20] C.-C. Wang, "On the capacity of wireless 1-hop intersession network coding - a broadcast packet erasure channel approach," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 957–988, February 2012.

[21] N. Harvey, R. Kleinberg, and A. Lehman, "On the capacity of information network," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2345–2364, June 2006.

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

[23] S. M. S. T. Yazdi, S. A. Savari, and G. Kramer, "Network coding in node-constrained line and star networks," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4452–4468, July 2011.

[24] C.-C. Wang and N. B. Shroff, "Pairwise intersession network coding on directed networks," *IEEE Transactions on Information Theory*, vol. 56, no. 8, pp. 3879–3900, August 2010.

[25] X. Xu, Y. Zeng, Y. L. Guan, and T. Ho, "On the capacity region of two-user linear deterministic interference channel and its application to multi-session network coding," *IEEE Transactions on Information Theory*, submitted for publication. [Online]. Available: http://arxiv.org/abs/1207.1986.

[26] S. Huang and A. Ramamoorthy, "An achievable region for the double unicast problem based on a minimum cut analysis," *IEEE Transactions on Communications*, vol. 61, no. 7, pp. 2890–2899, July 2013.

[27] X. Yan, R. W. Yeung, and Z. Zhang, "The capacity region for multi-source multi-sink network coding," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Nice, France, June 2007, pp. 116–120.

[28] G. Kramer, S. T. Yazdi, and S. A. Savari, "Refined edge-cut bounds for network coding," in *Proc. Int'l Zurich Seminar on Comm. (IZS)*, Zurich, Switzerland, March 2010, p. 12.

[29] C.-C. Wang and N. B. 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, pp. 121–125.

[30] ——, "Intersession network coding for two simple multicast sessions," in *Proc. 45th Annual Allerton Conf. on Comm., Contr., and Computing.*, Monticello, Illinois, USA, September 2007, pp. 682–689.

[31] D. Traskov, N. Ratnakar, D. S. Lun, R. Koetter, and M. Medard, "Network coding for multiple unicasts: An approach based on linear optimization," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Seattle, Washington, USA, July 2009, pp. 1758–1762.

[32] T. Ho, Y. Chang, and K. J. Han, "On constructive network coding for multiple unicasts," in *Proc. 44th Annual Allerton Conf. on Comm., Contr., and Computing.*, Monticello, Illinois, USA, September 2006.

[33] A. Khreishah, C.-C. Wang, and N. Shroff, "Cross-layer optimization for wireless multihop networks with pairwise intersession network coding," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 606–621, June 2009.

[34] ——, "Rate control with pairwise inter-session network coding," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 816–829, June 2010.

[35] W.-C. Kuo and C.-C. Wang, "Two-flow capacity region of the cope principle for two-hop wireless erasure networks," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7553–7575, November 2013.

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

[37] C. C. Wang and N. B. Shroff, "Random linear intersession network coding with selective canceling," in *Proc. IEEE Information Theory Workshop.*, Taormina, Sicily, Italy, October 2009, pp. 559–563.

[38] A. Lehman, "Network coding," Ph.D. dissertation, MIT, Cambridge, MA, 2005.

[39] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the k-user interference channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, August 2008.

[40] A. Das, S. Vishwanath, S. Jafar, and A. Markopoulou, "Network coding for multiple unicasts: An interference alignement approach," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Austin, Texas, USA, June 2010, pp. 1878–1882.

[41] A. Ramakrishnan, A. Dasz, H. Maleki, A. Markopoulou, S. Jafar, and S. Vishwanath, "Network coding for three unicast sessions: Interference alignment approaches," in *Proc. 48th Annual Allerton Conf. on Comm., Contr., and Computing.*, Monticello, Illinois, USA, September 2010, pp. 1054–1061.

[42] C. C. Wang, "Capacity region of two symmetric nearby erasure channels with channel state feedback," in *Proc. IEEE Information Theory Workshop.*, Lausanne, Switzerland, September 2012, pp. 352–356.

[43] C.-C. Wang, "Linear network coding capacity for broadcast erasure channels with feedback, receiver coordination, and arbitrary security requirement," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Istanbul, Turkey, July 2013, pp. 2900–2904.

[44] G. Kramer, "Capacity results for the discrete memoryless network," *IEEE Transactions on Information Theory*, vol. 49, no. 1, pp. 4–21, January 2003.

[45] S. H. Lim, Y.-H. Kim, A. E. Gamal, and S.-Y. Chung, "Noisy network coding," *IEEE Transactions on Information Theory*, vol. 57, no. 5, pp. 3132–3152, May 2011.

[46] E. C. V. der Meulen, "Three-terminal communication channels," in *Adv. Appl. Probab.,*, vol. 3, 1971, pp. 120–154.

[47] L. Georgiadis and L. Tassiulas, "Broadcast erasure channel with feedback - capacity and algorithms," in *Proc. 5th Workshop on Network Coding, Theory, and Applications (NetCod)*, Lausanne, Switzerland, June 2009, pp. 54–61.

[48] C.-C. Wang, "On the capacity of 1-to-K broadcast packet erasure channels with channel output feedback," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 931–956, February 2012.

[49] W.-C. Kuo and C.-C. Wang, "On the capacity of 2-user 1-hop relay erasure networks - the union of feedback, scheduling, opportunistic routing, and network coding," in *Proc. IEEE Int'l Symp. Inform. Theory.*, St. Peterburg, Russia, July 2011, pp. 1337–1341.

[50] C.-C. Wang and J. Han, "The capacity region of 2-receiver multiple-input broadcast packet erasure channels with channel output feedback," *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5597–5626, Sept. 2014.

[51] A. Dana and B. Hassibi, "The capacity region of multiple input erasure broadcast channels," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Adelaide, Australia, September 2005, pp. 2315–2319.

[52] L. Czap, V. M. Prabhakaran, S. Diggavi, and C. Fragouli, "Broadcasting private messages securely," in *Proc. IEEE Int'l Symp. Inform. Theory.*, Cambridge, MA, USA, July 2012, pp. 428–432.

[53] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Multiuser broadcast erasure channel with feedback - capacity and algorithms," *IEEE Transactions on Information Theory*, vol. 59, no. 9, pp. 5779–5804, September 2013.

[54] A. Papadopoulos and L. Georgiadis, "Multiuser broadcast erasure channel with feedback and side information and related index coding results," in *Proc. 52st Annual Allerton Conf. on Comm., Contr., and Computing.*, Monticello, Illinois, USA, October 2014.

[55] C.-C. Wang and D. J. Love, “Linear network coding capacity region of 2-receiver mimo broadcast packet erasure channels with feedback,” in *Proc. IEEE Int'l Symp. Inform. Theory.*, Boston, MA, USA, July 2012, pp. 2062–2066.

[56] W.-C. Kuo and C.-C. Wang, “Robust and optimal opportunistic scheduling for downlink 2-flow inter-session network coding,” in *Proc. 33rd IEEE Conf. Comput. Commun. (INFOCOM)*, Toronto, Canada, May 2014, pp. 655–663.

[57] J. Han and C.-C. Wang, “General capacity region for the fully-connected 3-node packet erasure network,” in *Proc. IEEE Int'l Symp. Inform. Theory.*, Hong Kong, June 2015, pp. 2648–2652.

[58] V. Anantharam and S. Verdu, “Bits through queues,” *IEEE Transactions on Information Theory*, vol. 42, no. 1, pp. 4–18, January 1996.

[59] J. Han and C.-C. Wang, “Optimal linear network coding when 3 nodes communicate over broadcast erasure channels with ack,” in *Technical Report*, Purdue University, June 2014. [Online]. Available: http://docs.lib.purdue.edu/ecetr/459

[60] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons Inc., 2006.

[61] A. Vahid, M. A. Maddah-Ali, and A. S. Avestimehr, “Binary fading interference channel with no csit,” in *Proc. IEEE Int'l Symp. Inform. Theory.*, Honolulu, HI, USA, Jun. 29-Jul. 4 2014, pp. 666–670.

[62] C. Meng, A. Ramakrishnan, A. Markopoulou, and S. A. Jafar, “On the feasibility of precoding-based network alignment for three unicast sessions,” in *Proc. IEEE Int'l Symp. Inform. Theory.*, Cambridge, MA, USA, July 2012, pp. 1907–1911.

APPENDICES

A. LNC CAPACITY REGION OF THE 3-NODE PACKET ERASURE NETWORK

In this appendix, we describe the LNC capacity region of the 3-node PEN. To that end, we first re-formulate the problem definition in Section 2.2 into the linear NC version. Namely, the encoding/decoding descriptions and the capacity definition in Section 2.2 will be re-formulated to the LNC equivalents as in the smart repeater problem formulation of Section 2.3. Then the LNC outer bound will be constructed based on the proposed Space-based Framework. To highlight the central idea of the Space-based Framework, here we only consider the 6-dimensional private information rates $(R_{1\to2}, R_{1\to3}, R_{2\to1}, R_{2\to3}, R_{3\to1}, R_{3\to2})$ and ignore the 3-dimensional common information rates $(R_{1\to23}, R_{2\to31}, R_{3\to12})$. The total 9-dimensional LNC capacity outer bound construction can be followed similarly. Moreover, here we focus only on Scenario 1 such that the casual ACK/NACK can be communicated for free. As similar to Proposition 3.2.3, we further show that the constructed LNC outer bound matches with the simple LNC achievability scheme of Proposition 3.2.2 for all possible channel parameters.

A.1 The Space-based Formulation of Linear NC

Let $\mathbf{W}$ be an nR_Σ-dimensional row vector defined by

$$\mathbf{W} \triangleq (\mathbf{W}_{1\to2}, \mathbf{W}_{1\to3}, \mathbf{W}_{2\to1}, \mathbf{W}_{2\to3}, \mathbf{W}_{3\to1}, \mathbf{W}_{3\to2}). \tag{A.1}$$

That is, $\mathbf{W}$ is the collection of all the information packets for the 6-dimensional traffic $\vec{R}$. Define $\Omega \triangleq (\mathbb{F}_q)^{nR_\Sigma}$ as the *overall message/coding space*. Then, a network code is called *linear* if (2.3) can be rewritten as

$$\text{If } \sigma(t) = i, \text{ then } X_i(t) = \mathbf{c}_t \mathbf{W}^\top \text{ for some } \mathbf{c}_t \in \Omega, \tag{A.2}$$

where $\mathbf{c}_t$ is a row coding vector in Ω. We assume that $\mathbf{c}_t$ is known causally to the entire network. [1]

We now define two important concepts: The *individual message subspace* and the *reception subspace*. To that end, we first define $\mathbf{e}_l$ as an nR_Σ-dimensional elementary row vector with its l-th coordinate being one and all the other coordinates being zero. Recall that the nR_Σ coordinates of a vector in Ω can be divided into 6 consecutive "intervals", each of them corresponds to the information packets $\mathbf{W}_{i\to h}$ for the unicast flow from node i to node $h \neq i$. For example, from (A.1), the third interval corresponds to the packets $\mathbf{W}_{2\to 1}$. We then define the *individual message subspace* $\Omega_{i\to j}$:

$$\Omega_{i\to j} \triangleq \mathsf{span}\{\mathbf{e}_l : l \in \text{“interval” associated to } \mathbf{W}_{i\to j}\}, \tag{A.3}$$

That is, $\Omega_{i\to j}$ is a linear subspace corresponding to any linear combination of $\mathbf{W}_{i\to j}$ packets. By (A.3), each $\Omega_{i\to j}$ is a linear subspace of Ω and $\mathsf{rank}(\Omega_{i\to j}) = nR_{i\to j}$.

For each node $i \in \{1, 2, 3\}$, the *reception subspace* in the end of time t is defined by

$$\begin{aligned} RS_i(t) \triangleq \mathsf{span}\{\mathbf{c}_\tau : \forall \tau \leq t \text{ s.t. } \sigma\tau \neq i,\ Z_{\sigma\tau\to i}(\tau) = 1, \\ \text{and } Y_{\sigma\tau\to i}(\tau) = X_{\sigma\tau}(\tau) = \mathbf{c}_\tau \mathbf{W}^\top\}. \end{aligned} \tag{A.4}$$

[1]Coding vector $\mathbf{c}_t$ can either be appended in the header or be computed by the network-wide causal CSI feedback $\mathbf{Z}(t)$.

That is, $RS_i(t)$ is the linear subspace spanned by the coding vectors $\mathbf{c}_\tau$ corresponding to the packets that are sent by node $\sigma\tau \neq i$ and have successfully arrived at node i by the end of time t. We now define the *knowledge space* $S_i(t)$ by

$$S_i(t) \triangleq \Omega_{i\to j} \oplus \Omega_{i\to k} \oplus RS_i(t), \tag{A.5}$$

where $A \oplus B \triangleq \mathsf{span}\{\mathbf{v} : \mathbf{v} \in A \cup B\}$ is the *sum space* of any $A, B \subseteq \Omega$. Basically, $S_i(t)$ represents the "overall knowledge" available at node i, which contains those that are originated from node i, i.e., $\Omega_{i\to j} \oplus \Omega_{i\to k}$, and those overheard by node i until time t, i.e., $RS_i(t)$. By the above definitions, we quickly have that node i can decode the desired packets $\hat{\mathbf{W}}_{h\to i}$, $h \neq i$, as long as $S_i(n) \supseteq \Omega_{h\to i}$. That is, when the knowledge space in the end of time n contains the desired message space.

Note that each node can only send a linear mixture of the packets that it currently "knows." Therefore, we can further strengthen the encoding part (A.2) by the following statement:

$$\text{If } \sigma(t) = i, \text{ then } X_i(t) = \mathbf{c}_t \mathbf{W}^\top \text{ for some } \mathbf{c}_t \in S_i(t-1). \tag{A.6}$$

We can now define the LNC capacity region.

Definition A.1.1. *Fix the distribution of $\mathbf{Z}(t)$ and finite field $\mathbb{F}_q$. A 6-dimensional rate vector $\vec{R}$ is achievable by LNC if for any $\epsilon > 0$ there exists a joint scheduling and LNC scheme with sufficiently large n such that $\mathsf{Prob}(\hat{\mathbf{W}}_{i\to h} \neq \mathbf{W}_{i\to h}) < \epsilon$ for all $i \in \{1, 2, 3\}$ and $h \neq i$. The LNC capacity region is the closure of all LNC-achievable $\vec{R}$.*

A.2 The LNC Capacity outer bound

Since the coding vector $\mathbf{c}_t$ has nR_Σ number of coordinates, there are exponentially many ways of jointly designing the scheduling $\sigma(t)$ and the coding vector choices $\mathbf{c}_t$ over time when sufficiently large n and $\mathbb{F}_q$ are used. We will first simplify the aforementioned design choices by comparing $\mathbf{c}_t$ to the knowledge spaces $S_i(t-1)$ described previously. Such a simplification allows us to derive Proposition A.2.1, which uses a linear programming (LP) solver to exhaustively search over the entire coding and scheduling choices and thus computes an LNC capacity outer bound.

Recall that $(i,j,k)\in\{(1,2,3),(2,3,1),(3,1,2)\}$, the cyclically shifted node indices. For example, if $i=2$, then $j=3$ and $k=1$. We also use S_i as shorthand for $S_i(t-1)$, the node-i knowledge space in the end of time $t-1$. For all $i\in\{1,2,3\}$, define the following seven linear subspaces of Ω:

$$A_1^{(i)}(t) \triangleq S_i, \qquad A_2^{(i)}(t) \triangleq S_i \oplus \Omega_{j\to i}, \tag{A.7}$$

$$A_3^{(i)}(t) \triangleq S_i \oplus \Omega_{k\to i}, \qquad A_4^{(i)}(t) \triangleq S_i \oplus \Omega_{j\to i} \oplus \Omega_{k\to i}, \tag{A.8}$$

$$A_1^{(i,j)}(t) \triangleq S_i \oplus S_j, \qquad A_2^{(i,j)}(t) \triangleq S_i \oplus S_j \oplus \Omega_{k\to i}, \tag{A.9}$$

$$A_3^{(i,j)}(t) \triangleq S_i \oplus S_j \oplus \Omega_{k\to j}. \tag{A.10}$$

Since the knowledge spaces S_i evolves over time, see (A.5), the above "A-subspaces" also evolves over time.

There are in total $7\times 3 = 21$ linear subspaces of Ω. We often drop the input argument "(t)" when the time instant of interest is clear in the context. We then partition the overall message space Ω into 2^{21} disjoint subsets by the *Venn diagram* generated by these 21 subspaces. That is, for any given coding vector $\mathbf{c}_t$, we can place it in exactly one of the 2^{21} disjoint subsets by testing whether it belongs to which A-subspaces.

We can further reduce the possible placement of $\mathbf{c}_t$ in the following way. By (A.6), we know that when $\sigma(t)=i$, node i selects $\mathbf{c}_t$ from its knowledge space $S_i(t-1)$. Hence, such $\mathbf{c}_t$ must always lie in any A-subspace that S_i appears in the definition. There are 10 such A-subspaces: $A_1^{(i)}$ to $A_4^{(i)}$; $A_1^{(i,j)}$ to $A_3^{(i,j)}$; and $A_1^{(k,i)}$ to $A_3^{(k,i)}$. As a result, for any coding vector $\mathbf{c}_t$ sent by node i, we only needs to check whether $\mathbf{c}_t$ belongs to which of the following 11 remaining A-subspaces:

$$\begin{aligned}
&\ddot{\mathsf{A}}_1^{(i)} \triangleq A_1^{(j)}, && \ddot{\mathsf{A}}_2^{(i)} \triangleq A_2^{(j)}, && \ddot{\mathsf{A}}_3^{(i)} \triangleq A_3^{(j)}, && \ddot{\mathsf{A}}_4^{(i)} \triangleq A_4^{(j)}, \\
&\ddot{\mathsf{A}}_5^{(i)} \triangleq A_1^{(k)}, && \ddot{\mathsf{A}}_6^{(i)} \triangleq A_2^{(k)}, && \ddot{\mathsf{A}}_7^{(i)} \triangleq A_3^{(k)}, && \ddot{\mathsf{A}}_8^{(i)} \triangleq A_4^{(k)}, \\
&\ddot{\mathsf{A}}_9^{(i)} \triangleq A_1^{(j,k)}, && \ddot{\mathsf{A}}_{10}^{(i)} \triangleq A_2^{(j,k)}, && \ddot{\mathsf{A}}_{11}^{(i)} \triangleq A_3^{(j,k)}. &&
\end{aligned} \tag{A.11}$$

In (A.11), we rename those 11 remaining A-subspace by $\ddot{\mathsf{A}}_1^{(i)}$ to $\ddot{\mathsf{A}}_{11}^{(i)}$ for easier future reference. For example when $i = \mathsf{3}$, such 11 subspaces $\ddot{\mathsf{A}}_1^{(\mathsf{3})}$ to $\ddot{\mathsf{A}}_{11}^{(\mathsf{3})}$ are $A_1^{(\mathsf{1})}$ to $A_4^{(\mathsf{1})}$; $A_1^{(\mathsf{2})}$ to $A_4^{(\mathsf{2})}$; and $A_1^{(\mathsf{1,2})}$ to $A_3^{(\mathsf{1,2})}$, respectively. For any 11-bitstring $\mathbf{b} = b_1 b_2 \cdots b_{11}$, we define "the coding type-$\mathbf{b}$ of node i" by

$$\mathsf{TYPE}_{\mathbf{b}}^{(i)} \triangleq S_i \cap \left(\bigcap_{l: b_l = 1} \ddot{\mathsf{A}}_l^{(i)} \right) \setminus \left(\bigcup_{l: b_l = 0} \ddot{\mathsf{A}}_l^{(i)} \right). \tag{A.12}$$

Namely, the $S_i(t-1)$ that node i can choose $\mathbf{c}_t$ from at time t is now further divided into $2^{11} = 2048$ disjoint subsets, depending on whether $\mathbf{c}_t$ belongs to $\ddot{\mathsf{A}}_l^{(i)}$ or not for $l = 1$ to 11. For example, $\mathsf{TYPE}_{169}^{(\mathsf{1})}$ (i.e., type-00010101001 of node $\mathsf{1}$) contains the $\mathbf{c}_t$ in S_{1} that is in the intersection of $\{\ddot{\mathsf{A}}_4^{(\mathsf{1})}, \ddot{\mathsf{A}}_6^{(\mathsf{1})}, \ddot{\mathsf{A}}_8^{(\mathsf{1})}, \ddot{\mathsf{A}}_{11}^{(\mathsf{1})}\}$ but not in the union of $\{\ddot{\mathsf{A}}_1^{(\mathsf{1})}, \ddot{\mathsf{A}}_2^{(\mathsf{1})}, \ddot{\mathsf{A}}_3^{(\mathsf{1})}, \ddot{\mathsf{A}}_5^{(\mathsf{1})}, \ddot{\mathsf{A}}_7^{(\mathsf{1})}, \ddot{\mathsf{A}}_9^{(\mathsf{1})}, \ddot{\mathsf{A}}_{10}^{(\mathsf{1})}\}$. By (A.11) and (A.12), we can write

$$\begin{aligned}
\mathsf{TYPE}_{169}^{(\mathsf{1})} \triangleq S_{\mathsf{1}} \cap & \left(A_4^{(\mathsf{2})} \cap A_2^{(\mathsf{3})} \cap A_4^{(\mathsf{3})} \cap A_3^{(\mathsf{2,3})} \right) \\
& \setminus \left(A_1^{(\mathsf{2})} \cup A_2^{(\mathsf{2})} \cup A_3^{(\mathsf{2})} \cup A_1^{(\mathsf{3})} \cup A_3^{(\mathsf{3})} \cup A_1^{(\mathsf{2,3})} \cup A_2^{(\mathsf{2,3})} \right).
\end{aligned}$$

In sum, any $\mathbf{c}_t$ chosen by node i must fall into one of the $2^{11} = 2048$ subsets $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$ defined by (A.11) and (A.12).

We can further strengthen the above observation by proving that 1996 (out of 2048) subsets are empty. For example, $\mathsf{TYPE}_{1024}^{(i)}$ (i.e., type-10000000000) is always empty since there is no such vector that can be inside $\ddot{\mathsf{A}}_1^{(i)} \triangleq A_1^{(j)}$ but not in $\ddot{\mathsf{A}}_2^{(i)} \triangleq A_2^{(j)}$ because we clearly have $A_2^{(j)} \supset A_1^{(j)}$ by definition (A.7). By eliminating all the empty subsets, $\mathbf{c}_t$ chosen by node i can only be in one of 52 (out of 2048) subsets. We call those 52 subsets the *Feasible Coding Types* (FTs) and they are enumerated as follows.

$$\begin{aligned}\mathsf{FTs} \triangleq \{&0, 1, 2, 3, 7, 9, 11, 15, 31, 41, 43, 47, 63, 127, 130, 131, 135, 139, 143, 159, 171, 175,\\ &191, 255, 386, 387, 391, 395, 399, 415, 427, 431, 447, 511, 647, 655, 671, 687, 703,\\ &767, 903, 911, 927, 943, 959, 1023, 1927, 1935, 1951, 1967, 1983, 2047\}.\end{aligned} \tag{A.13}$$

Since the coding choices are finite (52 per node and totally 3 nodes), we can derive the following upper bound using those $52 \times 3 = 156$ feasible types that fully cover Ω at any time t.

Proposition A.2.1. *A 6-dimensional rate vector $\vec{R}$ is in the LNC capacity region only if there exists 52×3 non-negative variables $x_{\mathbf{b}}^{(i)}$ for all $\mathbf{b} \in \mathsf{FTs}$ and $i \in \{1, 2, 3\}$ and 7×3 non-negative y-variables, $y_1^{(i)}$ to $y_4^{(i)}$, $y_1^{(i,j)}$ to $y_3^{(i,j)}$ for all $i \in \{1, 2, 3\}$, such that jointly they satisfy the following three groups of linear conditions:*

- *Group 1, termed the* time-sharing condition, *has 1 inequality:*

$$\left(\sum_{\forall \mathbf{b} \in \mathsf{FTs}} x_{\mathbf{b}}^{(1)}\right) + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs}} x_{\mathbf{b}}^{(2)}\right) + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs}} x_{\mathbf{b}}^{(3)}\right) \leq 1. \tag{A.14}$$

- *Group 2, termed the* rank-conversion conditions*, has 21 equalities: For all* $i \in \{1,2,3\}$*, and distinct indices* j *and* k *in* $\{1,2,3\}\backslash i$ *by circular-shifted way,*

$$y_1^{(i)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_5=0} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_1=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i} + R_{i\to j} + R_{i\to k}, \tag{A.15}$$

$$y_2^{(i)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_6=0} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_2=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i} + R_{i\to j} + R_{i\to k} + R_{j\to i}, \tag{A.16}$$

$$y_3^{(i)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_7=0} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_3=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i} + R_{i\to j} + R_{i\to k} + R_{k\to i}, \tag{A.17}$$

$$y_4^{(i)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_8=0} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_4=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i} + R_{i\to j} + R_{i\to k} + R_{j\to i} + R_{k\to i}, \tag{A.18}$$

$$y_1^{(i,j)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_9=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i\vee j} + R_{i\to j} + R_{i\to k} + R_{j\to i} + R_{j\to k}, \tag{A.19}$$

$$y_2^{(i,j)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_{10}=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i\vee j} + R_{i\to j} + R_{i\to k} + R_{j\to i} + R_{j\to k} + R_{k\to i}, \tag{A.20}$$

$$y_3^{(i,j)} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_{11}=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i\vee j} + R_{i\to j} + R_{i\to k} + R_{j\to i} + R_{j\to k} + R_{k\to j}. \tag{A.21}$$

- *Group 3, termed the* decodability conditions*, has* 6 *equalities:*

$$\forall\, i \in \{1,2,3\}, \quad y_1^{(i)} = y_2^{(i)} = y_3^{(i)} = y_4^{(i)}, \tag{A.22}$$

$$\forall\, i \in \{1,2,3\}, \quad y_1^{(i,j)} = y_2^{(i,j)} = y_3^{(i,j)} = R_\Sigma. \tag{A.23}$$

The intuition is as follows. Consider any achievable $\vec{R}$ and the associated LNC scheme. For any time t, suppose the given scheme chooses node i to transmit a coding vector $\mathbf{c}_t$. By the previous discussions, we can examine this $\mathbf{c}_t$ to see which $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$ it belongs to by looking at the corresponding A-subspaces in the end of $t-1$. Then after running the given scheme from time 1 to n, we can compute the variable

$x_{\mathbf{b}}^{(i)} \triangleq \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} 1_{\{\mathbf{c}_t \in \mathsf{TYPE}_{\mathbf{b}}^{(i)}\}}\right]$ for each $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$ as the *frequency* of scheduling node i with the chosen $\mathbf{c}_t$ happening to be in $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$. Since each $\mathbf{c}_t$ belongs to exactly one of the $52 \times 3 = 156$ feasible coding types, the time-sharing condition (A.14) holds naturally. We then compute the y-variables by

$$y_l^{(i)} \triangleq \frac{1}{n}\mathbb{E}\left[\mathsf{rank}\big(A_l^{(i)}(n)\big)\right],\ \forall l \in \{1,2,3,4\}, \tag{A.24}$$
$$y_l^{(i,j)} \triangleq \frac{1}{n}\mathbb{E}\left[\mathsf{rank}\big(A_l^{(i,j)}(n)\big)\right],\ \forall l \in \{1,2,3\},$$

as normalized expected ranks of A-subspaces in the end of time n. We now claim that these variables satisfy (A.15) to (A.23). This claim implies that for any LNC-achievable $\vec{R}$, there exists $x_{\mathbf{b}}^{(i)}$ and y-variables satisfying Proposition A.2.1, which means that Proposition A.2.1 constitutes an outer bound on the LNC capacity.

To prove that (A.15)–(A.21) are true,[2] consider an A-subspace, say $A_3^{(1)}(t) = S_1(t-1) \oplus \Omega_{3\to1} = RS_1(t-1) \oplus \Omega_{1\to2} \oplus \Omega_{1\to3} \oplus \Omega_{3\to1}$ as defined in (A.8) and (A.5) when $(i,j,k) = (1,2,3)$. In the beginning of time 1, node 1 has not received any packet yet, i.e., $RS_1(0) = \{\mathbf{0}\}$. Thus the rank of $A_3^{(1)}(1)$ is $\mathsf{rank}(\Omega_{1\to2} \oplus \Omega_{1\to3} \oplus \Omega_{3\to1}) = nR_{1\to2} + nR_{1\to3} + nR_{3\to1}$.

The fact that $S_1(t-1)$ contributes to $A_3^{(1)}(t)$ implies that $\mathsf{rank}(A_3^{(1)}(t))$ will increase by one whenever node 1 receives a packet $\mathbf{c}_t\mathbf{W}^\top$ satisfying $\mathbf{c}_t \notin A_3^{(1)}(t)$. Since $A_3^{(1)}(t)$ is labeled as $\ddot{\mathsf{A}}_7^{(2)}$, see (A.11) with $(i,j,k) = (2,3,1)$, whenever node 2 sends a $\mathbf{c}_t$ in $\mathsf{TYPE}_{\mathbf{b}}^{(2)}$ with $b_7 = 0$, such $\mathbf{c}_t$ is not in $A_3^{(1)}(t)$. Whenever node 1 receives it, $\mathsf{rank}(A_3^{(1)}(t))$ increases by 1. On the other hand, $A_3^{(1)}(t)$ is also labeled as $\ddot{\mathsf{A}}_3^{(3)}$, see (A.11) with $(i,j,k) = (3,1,2)$. Hence, whenever node 3 sends a $\mathbf{c}_t$ in $\mathsf{TYPE}_{\mathbf{b}}^{(3)}$ with

[2]For rigorous proofs, we need to invoke the law of large numbers and take care of the ϵ-error probability. For ease of discussion, the corresponding technical details are omitted when discussing the intuition of Proposition A.2.1.

$b_3 = 0$ and node 1 receives it, $\mathsf{rank}(A_3^{(1)}(t))$ also increases by 1. Therefore, in the end of time n, we have

$$\begin{aligned} \mathsf{rank}(A_3^{(1)}(n)) = &\sum_{t=1}^{n} 1_{\left\{\substack{\text{node 2 sends } \mathbf{c}_t \in \mathsf{TYPE}_{\mathbf{b}}^{(2)} \text{ with } b_7=0, \\ \text{and node 1 receives it}}\right\}} \\ &+ \sum_{t=1}^{n} 1_{\left\{\substack{\text{node 3 sends } \mathbf{c}_t \in \mathsf{TYPE}_{\mathbf{b}}^{(3)} \text{ with } b_3=0, \\ \text{and node 1 receives it}}\right\}} \\ &+ \mathsf{rank}(A_3^{(1)}(0)). \end{aligned} \tag{A.25}$$

Taking the normalized expectation of (A.25), we have proven (A.17) for $i = 1$. By similar *rank-conversion* arguments, (A.15)–(A.21) are true for all $i \in \{1, 2, 3\}$.

In the end of time n, since every node $i \in \{1, 2, 3\}$ can decode the desired packets $\mathbf{W}_{j\to i}$ and $\mathbf{W}_{k\to i}$, we thus have $S_i(n) \supseteq \Omega_{j\to i}$ and $S_i(n) \supseteq \Omega_{k\to i}$, or equivalently $S_i(n) = S_i(n) \oplus \Omega_{j\to i} \oplus \Omega_{k\to i}$. This implies that the ranks of $A_1^{(i)}(n)$ to $A_4^{(i)}(n)$ in (A.7) and (A.8) are all equal. Together with (A.24), we thus have (A.22). Similarly, one can prove that (A.23) is satisfied as well. The claim is thus proven.

A.3 The Match Proof

We now prove that both the constructed LNC outer bound of Proposition A.2.1 and the simple LNC achievability scheme of Proposition 3.2.2 in Scenario 1 meets regardless of channel parameters.

Proposition A.3.1. *The outer and inner bounds in Propositions A.2.1 and 3.2.2 match for all channel parameters and they thus describe the* 6*-dimensional LNC capacity region.*

Remark: One important implication is that for the 3-node 6-flow setting, we do not need to resort to any "exotic" LNC operation. Instead, 4 simple coding choices described in Section 3.5 are sufficient to achieve the optimal LNC capacity under *any channel parameters.*

A.3.1 Proof of Proposition A.3.1

For the readability, we rewrite the original 52 *Feasible Types* (FTs) defined in (A.13) that each node $i \in \{1,2,3\}$ can transmit:

$$\begin{aligned}\mathsf{FTs} \triangleq \{&\mathtt{000}, \mathtt{001}, \mathtt{002}, \mathtt{003}, \mathtt{007}, \mathtt{011}, \mathtt{013}, \mathtt{017}, \mathtt{037}, \mathtt{051},\\ &\mathtt{053}, \mathtt{057}, \mathtt{077}, \mathtt{0F7}, \mathtt{102}, \mathtt{103}, \mathtt{107}, \mathtt{113}, \mathtt{117}, \mathtt{137},\\ &\mathtt{153}, \mathtt{157}, \mathtt{177}, \mathtt{1F7}, \mathtt{302}, \mathtt{303}, \mathtt{307}, \mathtt{313}, \mathtt{317}, \mathtt{337},\\ &\mathtt{353}, \mathtt{357}, \mathtt{377}, \mathtt{3F7}, \mathtt{507}, \mathtt{517}, \mathtt{537}, \mathtt{557}, \mathtt{577}, \mathtt{5F7},\\ &\mathtt{707}, \mathtt{717}, \mathtt{737}, \mathtt{757}, \mathtt{777}, \mathtt{7F7}, \mathtt{F07}, \mathtt{F17}, \mathtt{F37}, \mathtt{F57},\\ &\mathtt{F77}, \mathtt{FF7}\}, \end{aligned} \tag{A.26}$$

where each 3-digit index $\overline{\mathbf{b}}_1\overline{\mathbf{b}}_2\overline{\mathbf{b}}_3$ represent a 11-bitstring $\mathbf{b}$ of which $\overline{\mathbf{b}}_1$ is a hexadecimal of first four bits, $\overline{\mathbf{b}}_2$ is a hexadecimal of the next four bits, and $\overline{\mathbf{b}}_3$ is octal of the last three bits. It should be clear from the context whether we are representing $\mathbf{b}$ as a decimal index, e.g., $\mathsf{TYPE}^{(1)}_{169}$, or as a 3-digit index based on hexadecimal/octal, e.g., $\mathsf{TYPE}^{(1)}_{\mathsf{FF7}}$.

For the notational convenience, we often use $\mathsf{FTs}(\cdot,\cdot,\cdot)$ to denote some collection of coding types in FTs. For example, $\mathsf{FTs}(\mathtt{F},\cdot,\cdot) \triangleq \{\,\mathbf{b} \in \mathsf{FTs} \text{ with } \overline{\mathbf{b}}_1 = \mathtt{F}\,\}$, corresponding to the collection of coding types in FTs with $b_1 = b_2 = b_3 = b_4 = 1$.

Without loss of generality, we also assume that $p_{i\to j} > 0$ and $p_{i\to k} > 0$ for all $(i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}$ since the case that any one of them is zero can be viewed as a limiting scenario and the polytope of the LP problem in Proposition A.2.1 is continuous with respect to the channel success probability parameters.

We now introduce the following three lemmas.

Lemma A.3.1. *Given any rate vector* $\vec{R}$ *and the associated* $\{x_{\mathbf{b}}^{(i)}\}$*-variables satisfying Proposition A.2.1, the following equalities, (A.27) to (A.36), always hold for all* $(i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}$.

$$R_{k\to i} + R_{k\to j} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_9=0} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i\vee j}, \tag{A.27}$$

$$R_{k\to j} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_{10}=0} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i\vee j}, \tag{A.28}$$

$$R_{k\to i} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_{11}=0} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i\vee j}, \tag{A.29}$$

$$\left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}(\cdot,\cdot,0)} x_{\mathbf{b}}^{(k)}\right) = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}(\cdot,\cdot,3)} x_{\mathbf{b}}^{(k)}\right). \tag{A.30}$$

$$R_{j\to i} + R_{k\to i} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_5=0, b_8=1} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_1=0, b_4=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i}, \tag{A.31}$$

$$R_{j\to i} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_5=0, b_6=1} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_1=0, b_2=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i}, \tag{A.32}$$

$$R_{k\to i} = \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_5=0, b_7=1} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in \mathsf{FTs}\ w.\ b_1=0, b_3=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i}, \tag{A.33}$$

$$\begin{aligned}\left(\sum_{\mathbf{b}\in \mathsf{FTs}(\cdot,7,\cdot)} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + &\left(\sum_{\mathbf{b}\in \mathsf{FTs}(7,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i} \\ &= \left(\sum_{\mathbf{b}\in \mathsf{FTs}(\cdot,1,\cdot)} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\mathbf{b}\in \mathsf{FTs}(1,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i}.\end{aligned} \tag{A.34}$$

$$\left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_{10}=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i\vee j} = \left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_5=0,b_6=1} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to j} + \left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_1=0,b_2=1} x_{\mathbf{b}}^{(i)}\right)\cdot p_{i\to j}, \tag{A.35}$$

$$\left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_{11}=0} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i\vee j} = \left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_5=0,b_7=1} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b}\in\mathsf{FTs}\ w.\ b_1=0,b_3=1} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i}. \tag{A.36}$$

The proof is relegated to Appendix A.3.2.

The following Lemma A.3.2 implies that we can impose special structure on the $\{x_{\mathbf{b}}^{(i)}\}$-variables satisfying Proposition A.2.1. For that, let us denote

$$\overline{\mathsf{FTs}} \triangleq \{\mathsf{051}, \mathsf{302}, \mathsf{337}, \mathsf{357}, \mathsf{3F7}, \mathsf{537}, \mathsf{557}, \mathsf{5F7}, \mathsf{F37}, \mathsf{F57}\}, \tag{A.37}$$

of which contains only 10 types out of 52 feasible coding types of the original FTs.

Lemma A.3.2. *Given any $\vec{R}$ and the associated 156 non-negative values $\{x_{\mathbf{b}}^{(i)}\}$ satisfying Proposition A.2.1, we can always find another set of 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly also satisfy Proposition A.2.1 and*

$$\ddot{x}_{\mathbf{b}}^{(i)} = 0 \text{ for all } \mathbf{b} \in \mathsf{FTs}\backslash\overline{\mathsf{FTs}}. \tag{A.38}$$

That is, without loss of generality, we can assume only those $\{x_{\mathbf{b}}^{(i)}\}$ with $\mathbf{b} \in \overline{\mathsf{FTs}}$ may have non-zero values. The proof of this lemma is relegated to Appendix A.3.3.

Lemma A.3.3. *Given any $\vec{R}$ and the associated 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ that satisfy Proposition A.2.1 and (A.38), we can always find 15 non-negative values $t_{[u]}^{(i)}$ and $\{t_{[c,l]}^{(i)}\}_{l=1}^{4}$ for all $i \in \{1,2,3\}$ such that jointly satisfy three groups of linear conditions in Proposition 3.2.2 (when replacing all strict inequality $<$ by $\leq$).*

The proof of this lemma is relegated to Appendix A.3.4.

One can clearly see that Lemmas A.3.2 and A.3.3 jointly imply that the outer bound in Proposition A.2.1 matches the closure of the inner bound in Proposition 3.2.2. The proof of Proposition A.3.1 is thus complete.

A.3.2 Proof of Lemma A.3.1

We prove the equalities (A.27) to (A.30) as follows.

Proof. These equalities can be derived by using (A.19)–(A.21) and (A.23) in Proposition A.2.1. Since $y_1^{(i,j)} = y_2^{(i,j)} = y_3^{(i,j)} = R_\Sigma$ by (A.23), substituting R_Σ to the left-hand side of (A.19)–(A.21), respectively, we have

$$\begin{aligned} R_{k\to i} + R_{k\to j} &= \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_9=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j}, \\ R_{k\to j} &= \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_{10}=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j}, \\ R_{k\to i} &= \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_{11}=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j}, \end{aligned}$$

which are equivalent to (A.27), (A.28), and (A.29), respectively.

We now prove the relationship (A.30). Substituting (A.28) and (A.29) to the left-hand side of (A.27), we then have

$$\left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_{10}=0} x_{\mathbf{b}}^{(k)} + \sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_{11}=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_9=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j}. \tag{A.39}$$

Note that for any type-$\mathbf{b}$, whenever $b_{10} = 0$ (resp. $b_{11} = 0$), b_9 is also zero. This is because $\ddot{\mathsf{A}}_9^{(i)} \subset \ddot{\mathsf{A}}_{10}^{(i)}$ (resp. $\ddot{\mathsf{A}}_9^{(i)} \subset \ddot{\mathsf{A}}_{11}^{(i)}$) regardless of node index i, see (A.11). Therefore, (A.39) can be further reduced to

$$\left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_9=0, b_{10}=0, b_{11}=0} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_9=0, b_{10}=1, b_{11}=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i\vee j}. \tag{A.40}$$

Dividing $p_{k\to i\vee j}$ on both sides of (A.40), we finally have (A.30). The proof is thus complete. ■

We prove the equalities (A.31) to (A.34) as follows.

Proof. These equalities can be derived by using the decodability equality (A.22) in Proposition A.2.1, i.e., $y_1^{(i)} = y_2^{(i)} = y_3^{(i)} = y_4^{(i)}$. First from $y_1^{(i)} = y_4^{(i)}$ and by (A.15) and (A.18), one can easily see that we have

$$R_{j\to i} + R_{k\to i} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_8=1} x_{\mathbf{b}}^{(j)} \right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_4=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i},$$

which is equivalent to (A.31). This is because for any type-$\mathbf{b}$, if $b_8 = 0$ (resp. $b_4 = 0$), then b_5 (resp. b_1) must be zero as well due to the fact that $\ddot{\mathsf{A}}_5^{(i)} \subset \ddot{\mathsf{A}}_8^{(i)}$ (resp. $\ddot{\mathsf{A}}_1^{(i)} \subset \ddot{\mathsf{A}}_4^{(i)}$) regardless of node index, see (A.11). Similarly from the facts that $\ddot{\mathsf{A}}_5^{(i)} \subset \ddot{\mathsf{A}}_6^{(i)}$, $\ddot{\mathsf{A}}_1^{(i)} \subset \ddot{\mathsf{A}}_2^{(i)}$, and by (A.15) and (A.16), $y_1^{(i)} = y_2^{(i)}$ implies

$$R_{j\to i} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_6=1} x_{\mathbf{b}}^{(j)} \right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_2=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i},$$

which is equivalent to (A.32).

Moreover, from the facts that $\ddot{\mathsf{A}}_5^{(i)} \subset \ddot{\mathsf{A}}_7^{(i)}$, $\ddot{\mathsf{A}}_1^{(i)} \subset \ddot{\mathsf{A}}_3^{(i)}$), and by (A.15) and (A.17), $y_1^{(i)} = y_3^{(i)}$ implies

$$R_{k\to i} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_7=1} x_{\mathbf{b}}^{(j)} \right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_3=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i}, \tag{A.41}$$

which is equivalent to (A.33).

We now prove the relationship (A.34). Substituting (A.32) and (A.33) to the left-hand side of (A.31), we thus have

$$\begin{aligned} &\left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_6=1} x_{\mathbf{b}}^{(j)} + \sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_7=1} x_{\mathbf{b}}^{(j)} \right) \cdot p_{j\to i} \\ &+ \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_2=1} x_{\mathbf{b}}^{(k)} + \sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_3=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i} \\ &\quad = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0, b_8=1} x_{\mathbf{b}}^{(j)} \right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0, b_4=1} x_{\mathbf{b}}^{(k)} \right) \cdot p_{k\to i}. \end{aligned}$$

Note that for any type-$\mathbf{b}$, whenever $b_6 = 1$ (resp. $b_7 = 1$), b_8 must be one due to the fact that $\ddot{\mathsf{A}}_6^{(i)} \subset \ddot{\mathsf{A}}_8^{(i)}$ (resp. $\ddot{\mathsf{A}}_7^{(i)} \subset \ddot{\mathsf{A}}_8^{(i)}$). The same argument holds such that for any type-$\mathbf{b}$, whenever $b_2 = 1$ (resp. $b_3 = 1$), we have $b_4 = 1$. Then the above equality further reduces to

$$\left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0,b_6=1,b_7=1,b_8=1} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0,b_2=1,b_3=1,b_4=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i}$$
$$= \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0,b_6=0,b_7=0,b_8=1} x_{\mathbf{b}}^{(j)}\right) \cdot p_{j\to i} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0,b_2=0,b_3=0,b_4=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to i},$$

which is equivalent to (A.34). The proof is thus compelte. ■

We prove the equalities (A.35) and (A.36) as follows.

Proof. By cyclic symmetry, we can rewrite (A.32) as follows.

$$R_{k\to j} = \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_5=0,b_6=1} x_{\mathbf{b}}^{(k)}\right) \cdot p_{k\to j} + \left(\sum_{\forall \mathbf{b} \in \mathsf{FTs} \text{ w. } b_1=0,b_2=1} x_{\mathbf{b}}^{(i)}\right) \cdot p_{i\to j}. \tag{A.42}$$

Then, (A.35) is a direct result of (A.28) and (A.42). Similarly, (A.36) is a direct result of (A.29) and (A.33). The proof is thus complete. ■

A.3.3 Proof of Lemma A.3.2

Before proving this lemma, we introduce the following "weight-movement" operator.

1. For any 2 non-negative values a and b, the operator $a \to b$ implies that we keep decreasing a and increasing b by the same amount until $a = 0$. Namely, after the operator, the new a and b values are

$$a_{\text{new}} = 0, \qquad b_{\text{new}} = b + a.$$

2. For any 3 non-negative values a, b, and c, the operator $\{a, b\} \to c$ implies that we keep decreasing a and b simultaneously and keep increasing c by the same amount until at least one of a and b being 0. Namely, after the operator, the new a, b, and c values are

$$a_{\text{new}} = a - \min\{a, b\}, \quad b_{\text{new}} = b - \min\{a, b\},$$
$$c_{\text{new}} = c + \min\{a, b\}.$$

3. For any 4 non-negative values a, b, c, and d, the operator $\{a, b\} \to \{c, d\}$ implies that we keep decreasing a and b simultaneously and keep increasing c and d simultaneously by the same amount until at least one of a and b being 0. Namely, after the operator, we have

$$a_{\text{new}} = a - \min\{a, b\}, \quad b_{\text{new}} = b - \min\{a, b\},$$
$$c_{\text{new}} = c + \min\{a, b\}, \quad d_{\text{new}} = d + \min\{a, b\}.$$

4. We can also concatenate the operators. For example, for any three non-negative values a, b, and c, the operator $a \to b \to c$ implies that

$$a_{\text{new}} = 0, \quad b_{\text{new}} = 0, \quad c_{\text{new}} = c + (a + b).$$

5. Sometimes, we do not want to "move the weight to the largest possible degree" as was defined previously. To that end, we define the operator $a \xrightarrow{\Delta} b$:

$$a_{\text{new}} = a - \Delta, \qquad b_{\text{new}} = b + \Delta.$$

   where Δ ($\leq a$) is the amount of weight being moved from a to b.

6. Finally, $a \to \emptyset$ means $a_{\text{new}} = 0$ and $a \xrightarrow{\Delta} \emptyset$ means $a_{\text{new}} = a - \Delta$.

We now prove Lemma A.3.2. Given $\vec{R}$ and $\{x_{\mathbf{b}}^{(i)}\}$-values satisfying Proposition A.2.1, let us denote the corresponding values of y-variables in the rank-conversion conditions (A.15)–(A.21) as $\{y\}$.

Recall that each coding type $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$ of node i corresponds to a specific subset of its knowledge space S_i, governed by 11 A-subspaces $\ddot{\mathsf{A}}_1^{(i)}$ to $\ddot{\mathsf{A}}_{11}^{(i)}$, see (A.11). As a result, by the rank conversion equalities (A.15)–(A.21), the bitstring $\mathbf{b}$ of each $\mathsf{TYPE}_{\mathbf{b}}^{(i)}$ will determine the contribution from the value $x_{\mathbf{b}}^{(i)}$ to the associated 11 y-values: $y_1^{(j)}$ to $y_4^{(j)}$; $y_1^{(k)}$ to $y_4^{(k)}$; and $y_1^{(j,k)}$ to $y_3^{(j,k)}$. For example, any vector $\mathbf{c}_t$ of $\mathsf{TYPE}_{\mathtt{7F7}}^{(i)}$ (i.e., type-01111111111 of node i), does not belong to $\ddot{\mathsf{A}}_1^{(i)}$. By (A.11) and (A.7)–(A.10), we know that $\ddot{\mathsf{A}}_1^{(i)} = A_1^{(j)}(t) = S_j(t-1)$. As a result, whenever a $\mathsf{TYPE}_{\mathtt{7F7}}^{(i)}$ coding vector, sent by node i at time t, is succesfully received by node j, the rank of $S_j(t-1)$ will increase by 1. Therefore, the value $x_{\mathtt{7F7}}^{(i)}$ (the frequency of using type-$\mathtt{7F7}$ of node i) contributes to $y_1^{(j)}$ (the normalized expected rank of $A_1^{(j)}(n)$ in the end of time n) by $x_{\mathtt{7F7}}^{(i)} \cdot p_{i\to j}$. Any change of the value $x_{\mathtt{7F7}}^{(i)}$ will thus change the corresponding value $y_1^{(j)}$ accordingly as described in the rank conversion equalities (A.15)–(A.21) in Proposition A.2.1.

The above intuition/explanation turns out to be very helpful when discussing the LP problem. Also, since all $\{y\}$-values can always be calculated from the given $\{x_{\mathbf{b}}^{(i)}\}$-values by (A.15)–(A.21), all our discussion can be focused on the given $\{x_{\mathbf{b}}^{(i)}\}$-values, and all $\{y\}$-values can be automatically computed. The proof of Lemma A.3.2 is done by proving the following intermediate claims.

Intermediate Claim 1*:* For any $\vec{R}$ and the corresponding 156 non-negative values $\{x_{\mathbf{b}}^{(i)}\}$ satisfying Proposition A.2.1, we can always find another set of 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly satisfy Proposition A.2.1 and

$$\ddot{x}_{\mathbf{b}}^{(i)} = 0, \quad \forall\, i \in \{1,2,3\} \text{ and } \forall\, \mathbf{b} \in \{\mathtt{FF7}, \mathtt{F07}, \mathtt{0F7}, \mathtt{007}\}. \tag{A.43}$$

Proof of Intermediate Claim 1: The proof is done by explicit construction. We sequentially perform the following weight movement operations for all $i \in \{1,2,3\}$:

$x^{(i)}_{\texttt{FF7}} \rightarrow \emptyset$; $x^{(i)}_{\texttt{F07}} \rightarrow \emptyset$; $x^{(i)}_{\texttt{0F7}} \rightarrow \emptyset$; and $x^{(i)}_{\texttt{007}} \rightarrow \emptyset$. After the weight movement, (A.43) is obviously true for the new values of $\{x^{(i)}_{\mathbf{b}}\}$. What remains to prove that the time-sharing condition (A.14) and the decodability conditions (A.22)–(A.23) still hold (when computing the new $\{y\}$-values using the new $\{x^{(i)}_{\mathbf{b}}\}$-values) after the weight movement.

To that end, we prove that (A.14), (A.22), and (A.23) hold after each of the weight movement operations. We first observe that $x^{(i)}_{\texttt{FF7}} \rightarrow \emptyset$ does not change any y-value because the coding type-11111111111 does not participate in the rank conversion process. As a result, after $x^{(i)}_{\texttt{FF7}} \rightarrow \emptyset$, the decodability conditions (A.22)–(A.23) still hold. Since $x^{(i)}_{\texttt{FF7}} \rightarrow \emptyset$ reduces the value of $x^{(i)}_{\texttt{FF7}}$, the time sharing condition (A.14) still holds.

We now consider $x^{(i)}_{\texttt{F07}} \rightarrow \emptyset$. Since $\texttt{F07} = 11110000111$ in 11-bitstring, it means that $x^{(i)}_{\texttt{F07}}$ contributes to the ranks of $\ddot{\mathsf{A}}^{(i)}_5$ to $\ddot{\mathsf{A}}^{(i)}_8$. By (A.11), $x^{(i)}_{\texttt{F07}}$ contributes[3] to the values of $y^{(k)}_1$ to $y^{(k)}_4$, the ranks of $A^{(k)}_1$ to $A^{(k)}_4$ in the end of time n, respectively. By (A.15)–(A.18), the operation $x^{(i)}_{\texttt{F07}} \rightarrow \emptyset$ will decrease each of $y^{(k)}_1$ to $y^{(k)}_4$ by the same amount $(x^{(i)}_{\texttt{F07}} \cdot p_{i\rightarrow k})$. Therefore, after $x^{(i)}_{\texttt{F07}} \rightarrow \emptyset$, the new values of $y^{(k)}_1$ to $y^{(k)}_4$ still satisfy the decodability equality (A.22). Note that $x^{(i)}_{\texttt{F07}}$ does not contribute to any of $y^{(j,k)}_1$ to $y^{(j,k)}_3$ and therefore (A.23) still holds after $x^{(i)}_{\texttt{F07}} \rightarrow \emptyset$.

By similar arguments, the operation $x^{(i)}_{\texttt{0F7}} \rightarrow \emptyset$ will decrease $y^{(j)}_1$ to $y^{(j)}_4$ by the same amount $(x^{(i)}_{\texttt{0F7}} \cdot p_{i\rightarrow j})$ while keeping all $y^{(k)}_1$ to $y^{(k)}_4$ and $y^{(j,k)}_1$ to $y^{(j,k)}_3$ unchanged. Therefore the decodability condition (A.22) still holds. By similar arguments, the operation $x^{(i)}_{\texttt{007}} \rightarrow \emptyset$ will decrease $y^{(j)}_1$ to $y^{(j)}_4$ by the same amount of $(x^{(i)}_{\texttt{007}} \cdot p_{i\rightarrow j})$ and decrease $y^{(k)}_1$ to $y^{(k)}_4$ by the same amount $(x^{(i)}_{\texttt{007}} \cdot p_{i\rightarrow k})$ while keeping all $y^{(j,k)}_1$ to $y^{(j,k)}_3$ unchanged. Therefore the decodability conditions (A.22) and (A.23) still hold. Intermediate Claim 1 is thus proven. ■

Intermediate Claim 2: For any $\vec{R}$ vector and the 156 corresponding non-negative $\{x^{(i)}_{\mathbf{b}}\}$-values satisfying Proposition A.2.1 and (A.43), we can always find another set of

[3] This argument can also be made by directly examining equalities (A.15)–(A.21). In (A.15)–(A.21), we can see that only in (A.15)–(A.18) we use the b_5 to b_8 values to determine the contribution of $\{x^{(i)}_{\mathbf{b}}, x^{(j)}_{\mathbf{b}}, x^{(k)}_{\mathbf{b}}\}$. Since $y^{(i)}_1$ to $y^{(i)}_4$ are contributed by $x^{(j)}_{\texttt{F07}}$, we thus know that only $y^{(k)}_1$ to $y^{(k)}_4$ are contributed by $x^{(i)}_{\texttt{F07}}$.

156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly satisfy Proposition A.2.1 and (A.43), plus

$$\ddot{x}_{\mathbf{b}}^{(i)} = 0, \quad \forall\, i \in \{1,2,3\} \text{ and } \forall\, \mathbf{b} \in \left\{ \begin{array}{c} 000, 003, 013, 053, 103, \\ 113, 153, 303, 313, 353 \end{array} \right\}. \tag{A.44}$$

Proof of Intermediate Claim 2: Consider any $\{x_{\mathbf{b}}^{(i)}\}$-values satisfying Proposition A.2.1 and (A.43). Since Proposition A.2.1 holds, Lemma A.3.1 implies that (A.30) holds as well. When we count the non-zero coding types in (A.30) (those not in (A.43)), we immediately have

$$x_{000}^{(i)} = x_{003}^{(i)} + x_{013}^{(i)} + x_{053}^{(i)} + x_{103}^{(i)} + x_{113}^{(i)} + x_{153}^{(i)} + x_{303}^{(i)} + x_{313}^{(i)} + x_{353}^{(i)}. \tag{A.45}$$

Then, we sequentially perform the following operations:

$$\begin{aligned}
&\{x_{003}^{(i)}, x_{000}^{(i)}\} \to \{x_{001}^{(i)}, x_{002}^{(i)}\}, \\
&\{x_{013}^{(i)}, x_{000}^{(i)}\} \to \{x_{002}^{(i)}, x_{011}^{(i)}\}, \\
&\{x_{053}^{(i)}, x_{000}^{(i)}\} \to \{x_{002}^{(i)}, x_{051}^{(i)}\}, \\
&\{x_{103}^{(i)}, x_{000}^{(i)}\} \to \{x_{001}^{(i)}, x_{102}^{(i)}\}, \\
&\{x_{113}^{(i)}, x_{000}^{(i)}\} \to \{x_{011}^{(i)}, x_{102}^{(i)}\}, \\
&\{x_{153}^{(i)}, x_{000}^{(i)}\} \to \{x_{051}^{(i)}, x_{102}^{(i)}\}, \\
&\{x_{303}^{(i)}, x_{000}^{(i)}\} \to \{x_{001}^{(i)}, x_{302}^{(i)}\}, \\
&\{x_{313}^{(i)}, x_{000}^{(i)}\} \to \{x_{011}^{(i)}, x_{302}^{(i)}\}, \\
&\{x_{353}^{(i)}, x_{000}^{(i)}\} \to \{x_{051}^{(i)}, x_{302}^{(i)}\}.
\end{aligned}$$

By (A.45), one can easily verify that after the above operations, we have (A.44). Thus it is left to show that after these operations the linear conditions of Proposition A.2.1 are still satisfied.

First notice that the time-sharing condition (A.14) is still satisfied since weight-moving operation decreases weights of two entries and increases the weights of another two entries by the same amount. We now argue that after each of the totally 9 weight-moving operations, the associated y-values remain unchanged. Take the last weight-moving operation $\{x^{(i)}_{353}, x^{(i)}_{000}\} \to \{x^{(i)}_{051}, x^{(i)}_{302}\}$ for example. The corresponding coding types are

$$\mathsf{TYPE}^{(i)}_{353} \text{ in 11-bitstring} = 0011\,0101\,011,$$
$$\mathsf{TYPE}^{(i)}_{000} \text{ in 11-bitstring} = 0000\,0000\,000,$$
$$\mathsf{TYPE}^{(i)}_{051} \text{ in 11-bitstring} = 0000\,0101\,001,$$
$$\mathsf{TYPE}^{(i)}_{302} \text{ in 11-bitstring} = 0011\,0000\,010.$$

Let $b_l(\mathsf{353})$ denote the l-th bit of the 11-bitstring $\mathsf{353} = \mathsf{00110101011}$, and similarly $b_l(\mathsf{000})$, $b_l(\mathsf{051})$, and $b_l(\mathsf{302})$ denote the l-th bit of 11-bitstrings $\mathsf{000}$, $\mathsf{051}$, and $\mathsf{302}$, respectively. One can see that for any l, the set $\{b_l(\mathsf{353}), b_l(\mathsf{000})\}$ is identical, as a set, to the set $\{b_l(\mathsf{051}), b_l(\mathsf{302})\}$ for all $l = 1$ to 11. Namely, when performing $\{x^{(i)}_{353}, x^{(i)}_{000}\} \to \{x^{(i)}_{051}, x^{(i)}_{302}\}$, for all $l = 1$ to 11, the impact on the rank of $\ddot{\mathsf{A}}^{(i)}_l$ by decreasing simultaneously the two entries $\{x^{(i)}_{353}, x^{(i)}_{000}\}$ is offset completely by increasing simultaneously the two entries $\{x^{(i)}_{051}, x^{(i)}_{302}\}$. For example, bit b_1 (when $l = 1$) corresponds to $\ddot{\mathsf{A}}^{(i)}_1 = A^{(j)}_1$ and we have $b_1(\mathsf{353}) = 0$ and $b_1(\mathsf{000}) = 0$. Therefore, if we separate the weight-moving operation $\{x^{(i)}_{353}, x^{(i)}_{000}\} \to \{x^{(i)}_{051}, x^{(i)}_{302}\}$ into the decreasing half and the increasing half, then during the decreasing half, the $y^{(j)}_1$-value will decrease by $\min\{x^{(i)}_{353}, x^{(i)}_{000}\} \cdot p_{i\to j}$ due to the decrease of $x^{(i)}_{353}$ and then decrease by another $\min\{x^{(i)}_{353}, x^{(i)}_{000}\} \cdot p_{i\to j}$ due to the decrease of $x^{(i)}_{000}$. On the other hand, during the increasing half, the $y^{(j)}_1$ value will increase by $\min\{x^{(i)}_{353}, x^{(i)}_{000}\} \cdot p_{i\to j}$ due to the increase of $x^{(i)}_{051}$ and then increase by another $\min\{x^{(i)}_{353}, x^{(i)}_{000}\} \cdot p_{i\to j}$ due to the increase of $x^{(i)}_{302}$. The amounts of increase and decrease perfectly offset each other since $\{b_1(\mathsf{353}), b_1(\mathsf{000})\} = \{0, 0\} = \{b_1(\mathsf{051}), b_1(\mathsf{302})\}$.

In sum, by similar reasoning, all the y-values will remain the same after each of the above 9 weight-moving operations. The proof is thus complete. ■

Intermediate Claim 3*:* For any $\vec{R}$ vector and the 156 corresponding non-negative $\{x_{\mathbf{b}}^{(i)}\}$-values satisfying Proposition A.2.1 and (A.43) to (A.44), we can always find another set of 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly satisfy Proposition A.2.1 and (A.43) to (A.44), plus for all $i \in \{1, 2, 3\}$,

$$\begin{aligned}\left(\sum_{\mathbf{b}\in\mathsf{FTs}(\cdot,7,\cdot)} x_{\mathbf{b}}^{(i)}\right) &= \left(\sum_{\mathbf{b}\in\mathsf{FTs}(\cdot,1,\cdot)} x_{\mathbf{b}}^{(i)}\right),\\ \left(\sum_{\mathbf{b}\in\mathsf{FTs}(7,\cdot,\cdot)} x_{\mathbf{b}}^{(i)}\right) &= \left(\sum_{\mathbf{b}\in\mathsf{FTs}(1,\cdot,\cdot)} x_{\mathbf{b}}^{(i)}\right).\end{aligned} \tag{A.46}$$

Proof of Intermediate Claim 3*:* Since the node indices are cyclically decided, we will prove the following equivalent forms:

$$\left(\sum_{\mathbf{b}\in\mathsf{FTs}(\cdot,7,\cdot)} x_{\mathbf{b}}^{(j)}\right) = \left(\sum_{\mathbf{b}\in\mathsf{FTs}(\cdot,1,\cdot)} x_{\mathbf{b}}^{(j)}\right), \tag{A.47}$$

$$\left(\sum_{\mathbf{b}\in\mathsf{FTs}(7,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right) = \left(\sum_{\mathbf{b}\in\mathsf{FTs}(1,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right), \tag{A.48}$$

based on the equality (A.34) of Lemma A.3.1. For shorthand, define the following 4 non-negative terms of (A.34) as follows:

$$\mathsf{term}_1 \triangleq \left(\sum_{\mathbf{b}\in \mathsf{FTs}(\cdot,7,\cdot)} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i},$$

$$\mathsf{term}_2 \triangleq \left(\sum_{\mathbf{b}\in \mathsf{FTs}(7,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i},$$

$$\mathsf{term}_3 \triangleq \left(\sum_{\mathbf{b}\in \mathsf{FTs}(\cdot,1,\cdot)} x_{\mathbf{b}}^{(j)}\right)\cdot p_{j\to i},$$

$$\mathsf{term}_4 \triangleq \left(\sum_{\mathbf{b}\in \mathsf{FTs}(1,\cdot,\cdot)} x_{\mathbf{b}}^{(k)}\right)\cdot p_{k\to i}.$$

Using the above 4 terms, (A.34) can be rewritten by

$$\mathsf{term}_1 + \mathsf{term}_2 = \mathsf{term}_3 + \mathsf{term}_4. \tag{A.49}$$

Recall that we assume both $p_{j\to i} > 0$ and $p_{k\to i} > 0$. Consider the following three cases depending on the values of term_1 and term_3.

Case 1: $\mathsf{term}_1 = \mathsf{term}_3$. By (A.49), we also have $\mathsf{term}_2 = \mathsf{term}_4$. By the definitions of term_1 to term_4, both (A.47) and (A.48) hold automatically.

Case 2: $\mathsf{term}_1 < \mathsf{term}_3$. Since each term is strictly non-negative, we thus have $\mathsf{term}_3 > 0$. Also by (A.49), we must also have $\mathsf{term}_2 > \mathsf{term}_4$ and thus $\mathsf{term}_2 > 0$. In the following, we will describe a set of weight-moving operations such that after moving the weights among $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$, the new $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$ satisfy Proposition A.2.1, (A.43), and (A.44); and the gap $\mathsf{term}_3 - \mathsf{term}_1$ computed using the new $\{x_{\mathbf{b}}^{(j)}\}$ is strictly smaller than the gap computed by the old $\{x_{\mathbf{b}}^{(j)}\}$ while $\mathsf{term}_3 \geq \mathsf{term}_1$. We can thus iteratively perform the weight movements until $\mathsf{term}_1 = \mathsf{term}_3$. The final $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$ then satisfy (A.46) now.

The desired weight-moving operations are described as follows. Since $\mathsf{term}_3 > 0$, we can find an 11-bitstring $\mathbf{b}^{\mathsf{term}_3} \in \mathsf{FTs}(\cdot, 1, \cdot)$ such that $x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}} > 0$. Similarly, since $\mathsf{term}_2 > 0$, we can find a $\mathbf{b}^{\mathsf{term}_2} \in \mathsf{FTs}(7, \cdot, \cdot)$ such that $x^{(k)}_{\mathbf{b}^{\mathsf{term}_2}} > 0$. We then define

$$\Delta = \min\left\{ x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}} \cdot p_{j\to i},\ \ x^{(k)}_{\mathbf{b}^{\mathsf{term}_2}} \cdot p_{k\to i},\ \ \mathsf{term}_3 - \mathsf{term}_1 \right\}.$$

Obviously, we have $\Delta > 0$ since we assume $p_{j\to i} > 0$ and $p_{k\to i} > 0$ for all (i, j, k). We then compute $\Delta^{\mathsf{term}_3} = \Delta / p_{j\to i}$ and $\Delta^{\mathsf{term}_2} = \Delta / p_{k\to i}$. By the definition of Δ, we have $0 < \Delta^{\mathsf{term}_3} \leq x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}}$ and $0 < \Delta^{\mathsf{term}_2} \leq x^{(k)}_{\mathbf{b}^{\mathsf{term}_2}}$.

Then, we perform the following weight-moving operations:

$$x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}} \xrightarrow{\Delta^{\mathsf{term}_3}} x^{(j)}_{\mathbf{b}^{\mathsf{term}_3} \oplus \mathsf{040}}, \tag{OP1}$$

$$x^{(k)}_{\mathbf{b}^{\mathsf{term}_2}} \xrightarrow{\Delta^{\mathsf{term}_2}} x^{(k)}_{\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400}}, \tag{OP2}$$

where $\oplus$ is bit-wise exclusive or. For example, if $\mathbf{b}^{\mathsf{term}_3} = \mathsf{117}$ which belongs to $\mathsf{FTs}(\cdot, 1, \cdot)$, then $\mathbf{b}^{\mathsf{term}_3} \oplus \mathsf{040} = \mathsf{157}$ which now belongs to $\mathsf{FTs}(\cdot, 5, \cdot)$ instead. Similarly, if $\mathbf{b}^{\mathsf{term}_2} = \mathsf{737}$, then $\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400} = \mathsf{337}$, which now belongs to $\mathsf{FTs}(3, \cdot, \cdot)$.

We now argue that after moving the weights among $\{x^{(j)}_{\mathbf{b}}, x^{(k)}_{\mathbf{b}}\}$, the new $\{x^{(j)}_{\mathbf{b}}, x^{(k)}_{\mathbf{b}}\}$ satisfy Proposition A.2.1, (A.43), and (A.44); and the gap $\mathsf{term}_3 - \mathsf{term}_1$ computed using the new $\{x^{(j)}_{\mathbf{b}}\}$ is strictly smaller than the gap computed by the old $\{x^{(j)}_{\mathbf{b}}\}$ while $\mathsf{term}_3 \geq \mathsf{term}_1$. To that end, we first argue that after the above weight movements, both (A.43) and (A.44) still hold. The reason is that since $\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400} \in \mathsf{FTs}(3, \cdot, \cdot)$ and $\mathbf{b}^{\mathsf{term}_3} \oplus \mathsf{040} \in \mathsf{FTs}(\cdot, 5, \cdot)$, we never move any weight to the frequencies $\{x^{(j)}_{\mathbf{b}}, x^{(k)}_{\mathbf{b}}\}$ satisfying (A.43). As a result, (A.43) still holds after the above weight movements. Since $\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400} \in \mathsf{FTs}(3, \cdot, \cdot)$, it may look possible that we can increase the weight of $x^{(k)}_{\mathsf{303}}$, $x^{(k)}_{\mathsf{313}}$, and $x^{(k)}_{\mathsf{353}}$ in (A.44) by the weight-moving operation (OP2). However, to increase the weight of $x^{(k)}_{\mathsf{303}}$, $x^{(k)}_{\mathsf{313}}$, and $x^{(k)}_{\mathsf{353}}$, it means that we must have $\mathbf{b}^{\mathsf{term}_2} \in \{\mathsf{703}, \mathsf{713}, \mathsf{753}\}$ to begin with. However, they are not in the feasible coding types FTs, see (A.26). As a result, after (OP2) movement, (A.44) still holds. Since $x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}} \oplus \mathsf{040} \in \mathsf{FTs}(\cdot, 5, \cdot)$, it may look possible that we can increase the weight of $x^{(j)}_{\mathsf{053}}$, $x^{(j)}_{\mathsf{153}}$, and $x^{(j)}_{\mathsf{353}}$

in (A.44) by the weight-moving operation (OP1). However, to increase the weight of $x^{(j)}_{\mathsf{053}}$, $x^{(j)}_{\mathsf{153}}$, and $x^{(j)}_{\mathsf{353}}$, it means that we must have $\mathbf{b}^{\mathsf{term}_3} \in \{\mathsf{013}, \mathsf{113}, \mathsf{313}\}$ to begin with. However, since we choose $\mathbf{b}^{\mathsf{term}_3}$ such that $x^{(j)}_{\mathbf{b}^{\mathsf{term}_3}} > 0$, and the original $\{x^{(j)}_{\mathbf{b}}\}$-values satisfy (A.44), it is impossible to have $\mathbf{b}^{\mathsf{term}_3} \in \{\mathsf{013}, \mathsf{113}, \mathsf{313}\}$. As a result, after (OP1) movement, (A.44) still holds.

We now consider the conditions in Proposition A.2.1. We first notice that it is clear that after moving the weights, the time-sharing condition of Proposition A.2.1 still holds because at every iteration we only "move" the weights on the frequencies $\{x^{(j)}_{\mathbf{b}}, x^{(k)}_{\mathbf{b}}\}$ without changing the overall sum. We now examine whether other conditions of Proposition A.2.1 are still satisfied after the above modification process. For that, we argue that the above process keeps all the y-values unchanged. To see that, suppose $(i, j, k) = (\mathsf{1}, \mathsf{2}, \mathsf{3})$ without loss of generality. Since the 11-bitstring $\mathsf{040}$ has only 6-th bit being 1 and all the other bits being 0, the (OP1) operation will change only the rank of $\ddot{\mathsf{A}}^{(j)}_6$, i.e., $\ddot{\mathsf{A}}^{(2)}_6$ when $(i, j, k) = (\mathsf{1}, \mathsf{2}, \mathsf{3})$. By (A.11), $\ddot{\mathsf{A}}^{(2)}_6 = A^{(1)}_2$ and thus only $y^{(1)}_2$ will be affected by this operation. Since we are moving the weight of Δ^{term_3} from $x^{(2)}_{\mathbf{b}^{\mathsf{term}_3}}$ (the 6-th bit of $\mathbf{b}^{\mathsf{term}_3}$ is 0 since $\mathbf{b}^{\mathsf{term}_3} \in \mathsf{FTs}(\cdot, \mathsf{1}, \cdot)$) to $x^{(2)}_{\mathbf{b}^{\mathsf{term}_3} \oplus \mathsf{040}}$ (the 6-th bit of $\mathbf{b}^{\mathsf{term}_3} \oplus \mathsf{040}$ is 1), $y^{(1)}_2$ will be decreased by $(\Delta^{\mathsf{term}_3} \cdot p_{2\to 1})$, which is equal to Δ. On the other hand since the 11-bitstring $\mathsf{400}$ has only the 2nd bit being 1 and all the other bits being 0, the (OP2) operation will change only the rank of $\ddot{\mathsf{A}}^{(k)}_2$, i.e., $\ddot{\mathsf{A}}^{(3)}_2$ when $(i, j, k) = (\mathsf{1}, \mathsf{2}, \mathsf{3})$. By (A.11), $\ddot{\mathsf{A}}^{(3)}_2 = A^{(1)}_2$ and thus again only $y^{(1)}_2$ will be affected by this operation. Since we are moving the weight of Δ^{term_2} from $x^{(3)}_{\mathbf{b}^{\mathsf{term}_2}}$ (the 2nd bit of $\mathbf{b}^{\mathsf{term}_2}$ is 1 since $\mathbf{b}^{\mathsf{term}_2} \in \mathsf{FTs}(\mathsf{7}, \cdot, \cdot)$) to $x^{(3)}_{\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400}}$ (the 2nd bit of $\mathbf{b}^{\mathsf{term}_2} \oplus \mathsf{400}$ is 0), $y^{(1)}_2$ will be increased by $(\Delta^{\mathsf{term}_2} \cdot p_{3\to 1})$, which is equal to Δ. The impacts of the two weight-moving operations (OP1) and (OP2) on $y^{(1)}_2$ perfectly offset each other. As a result, any of y-values are unchanged.

In the following, we will prove that (OP1) will decrease the value of term_3 by Δ while keeping the values of term_1, term_2, and term_4 unchanged; and (OP2) will decrease the value of term_2 by Δ while keeping the values of term_1, term_3, and term_4 unchanged. Thus after performing (OP1) and (OP2), the gap $\mathsf{term}_3 - \mathsf{term}_1$ computed by the new $\{x_{\mathbf{b}}^{(j)}\}$-values decreases by Δ and we still have $\mathsf{term}_3 \geq \mathsf{term}_1$ by the definition of Δ while satisfying (A.49). We first observe that (OP1) manipulates only $\{x_{\mathbf{b}}^{(j)}\}$, thus term_2 and term_4 will not be affected since both are based on $\{x_{\mathbf{b}}^{(k)}\}$ of another node index. Also notice that $\mathbf{b}^{\mathsf{term}_3} \in \mathsf{FTs}(\cdot, \mathtt{1}, \cdot)$ if and only if $\mathbf{b}^{\mathsf{term}_3} \oplus \mathtt{040} \in \mathsf{FTs}(\cdot, \mathtt{5}, \cdot)$. Therefore, the weight movement (OP1) does not change the value of term_1 since term_1 involves only those frequencies with $\mathbf{b} \in \mathsf{FTs}(\cdot, \mathtt{7}, \cdot)$. Finally, since $\mathbf{b}^{\mathsf{term}_3} \in \mathsf{FTs}(\cdot, \mathtt{1}, \cdot)$ and $\mathbf{b}^{\mathsf{term}_3} \oplus \mathtt{040} \in \mathsf{FTs}(\cdot, \mathtt{5}, \cdot)$, the (OP1) movement will decrease the value of term_3 and the decrease amount will be $\Delta^{\mathsf{term}_3} \cdot p_{j\to i} = \Delta$. The statement that (OP2) decreases the value of term_2 by Δ while keeping the values of term_1, term_3, and term_4 unchanged can be proved similarly. The proof of Case 2 is thus complete.

Case 3: $\mathsf{term}_1 > \mathsf{term}_3$. Since each term is strictly non-negative, we thus have $\mathsf{term}_1 > 0$ and by (A.49), we must also have $\mathsf{term}_4 > 0$. Again, we will describe a set of weight-moving operations such that after moving the weights among $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$, the new $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$ satisfy Proposition A.2.1, (A.43), and (A.44); and the gap $\mathsf{term}_1 - \mathsf{term}_3$ computed using the new $\{x_{\mathbf{b}}^{(j)}\}$ is strictly smaller than the gap computed by the old $\{x_{\mathbf{b}}^{(j)}\}$ while satisfying (A.49) and $\mathsf{term}_1 \geq \mathsf{term}_3$. We can thus iteratively perform the weight movements until $\mathsf{term}_1 = \mathsf{term}_3$. The final $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$ thus satisfy (A.46).

The desired weight-moving operations are described as follows. Since $\mathsf{term}_1 > 0$, we can find an 11-bitstring $\mathbf{b}^{\mathsf{term}_1} \in \mathsf{FTs}(\cdot, \mathtt{7}, \cdot)$ such that $x_{\mathbf{b}^{\mathsf{term}_1}}^{(j)} > 0$. Similarly, since $\mathsf{term}_4 > 0$, we can find a $\mathbf{b}^{\mathsf{term}_4} \in \mathsf{FTs}(\mathtt{1}, \cdot, \cdot)$ such that $x_{\mathbf{b}^{\mathsf{term}_4}}^{(k)} > 0$. We then define

$$\Delta = \min\left\{ x_{\mathbf{b}^{\mathsf{term}_1}}^{(j)} \cdot p_{j\to i}, \;\; x_{\mathbf{b}^{\mathsf{term}_4}}^{(k)} \cdot p_{k\to i}, \;\; \mathsf{term}_1 - \mathsf{term}_3 \right\}.$$

We then compute $\Delta^{\mathsf{term}_1} = \Delta / p_{j\to i}$ and $\Delta^{\mathsf{term}_4} = \Delta / p_{k\to i}$. Then, we perform the following weight-moving operations:

$$x_{\mathbf{b}^{\mathsf{term}_1}}^{(j)} \xrightarrow{\Delta^{\mathsf{term}_1}} x_{\mathbf{b}^{\mathsf{term}_1} \oplus \mathtt{040}}^{(j)}, \qquad x_{\mathbf{b}^{\mathsf{term}_4}}^{(k)} \xrightarrow{\Delta^{\mathsf{term}_4}} x_{\mathbf{b}^{\mathsf{term}_4} \oplus \mathtt{400}}^{(k)}.$$

By almost identical reasonings as in the discussion of Case 2, we can prove that after the above modification process, we have that the new $\{x_{\mathbf{b}}^{(j)}, x_{\mathbf{b}}^{(k)}\}$ satisfy Proposition A.2.1, (A.43), and (A.44); and the gap $\mathsf{term}_1 - \mathsf{term}_3$ computed using the new $\{x_{\mathbf{b}}^{(j)}\}$ is strictly smaller than the gap computed by the old $\{x_{\mathbf{b}}^{(j)}\}$ while satisfying (A.49) and $\mathsf{term}_1 \geq \mathsf{term}_3$. The proof of Case 3 is thus complete. ■

Intermediate Claim 4*:* For any $\vec{R}$ vector and the 156 corresponding non-negative $\{x_{\mathbf{b}}^{(i)}\}$-values satisfying Proposition A.2.1 and (A.43) to (A.46), we can always find another set of 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly satisfy Proposition A.2.1 and (A.43) to (A.46), plus

$$\ddot{x}_{\mathbf{b}}^{(i)} = 0,\ \forall\, i \in \{1,2,3\} \text{ and } \forall\, \mathbf{b} \in \left\{ \begin{array}{l} \mathtt{011}, \mathtt{017}, \mathtt{037}, \mathtt{057}, \mathtt{077}, \mathtt{102}, \mathtt{107}, \\ \mathtt{117}, \mathtt{137}, \mathtt{157}, \mathtt{177}, \mathtt{1F7}, \mathtt{307}, \mathtt{317}, \\ \mathtt{377}, \mathtt{507}, \mathtt{517}, \mathtt{577}, \mathtt{707}, \mathtt{717}, \mathtt{737}, \\ \mathtt{757}, \mathtt{777}, \mathtt{7F7}, \mathtt{F17}, \mathtt{F77} \end{array} \right\}. \tag{A.50}$$

Proof of Intermediate Claim 4*:* We simultaneously perform the weight-moving operations in the first column of Table A.1 for all nodes $i \in \{1,2,3\}$. For each operation, we also present how the associated y-values are affected after each operation. As described in the proof of *Intermediate Claim* 1, one can verify the variations of y-values by each operation in Table A.1. For example, the first operation $x_{\mathtt{011}}^{(i)} \to x_{\mathtt{051}}^{(i)}$ moves all the weight from $x_{\mathtt{011}}^{(i)}$ to $x_{\mathtt{051}}^{(i)}$. Since

$$\mathsf{TYPE}_{\mathtt{011}}^{(i)} \text{ in 11-bitstring} = 0000\,0001\,001,$$
$$\mathsf{TYPE}_{\mathtt{051}}^{(i)} \text{ in 11-bitstring} = 0000\,0101\,001,$$

one can easily see that only the rank of $\ddot{\mathsf{A}}_6^{(i)}$ will be affected since the only different bit between $\mathtt{011}$ and $\mathtt{051}$ is the 6-th bit. By (A.11), $\ddot{\mathsf{A}}_6^{(i)} = A_2^{(k)}$ and thus only $y_2^{(k)}$ will be affected by $x_{\mathtt{011}}^{(i)} \to x_{\mathtt{051}}^{(i)}$ operation. We observe that $\mathsf{TYPE}_{\mathtt{011}}^{(i)}$ participates in the increase of $y_2^{(k)}$ (the 6-th bit of $\mathtt{011}$ being 0) but $\mathsf{TYPE}_{\mathtt{051}}^{(i)}$ (the 6-th bit of $\mathtt{051}$ being 1) does not. Thus after the weight movement, $y_2^{(k)}$ will be decreased by the amount of $(x_{\mathtt{011}}^{(i)} \cdot p_{i\to k})$ as indicated in Table A.1. The rest of Table A.1 is populated by examining all 10 weight-moving operations (the 10 rows) and their corresponding impact on the y-values.

One can easily see from Table A.1 that after completing all 10 weight-moving operations, for each node i, 26 coding types (enumerated in (A.50)) of the new values $\{x^{(i)}_{\mathbf{b}}\}$ will be set to zeros.

Table A.1: The weight-moving operations and the corresponding variations of the associated y-values for *Intermediate Claim* 4.

The underlying y-values are associated to 11-bitstring of node i's coding type-$\mathbf{b}$. See (A.11) for conversion. For shorthand, we define $p \triangleq p_{i\to j}$ and $q \triangleq p_{i\to k}$.

	$y^{(j)}_1$	$y^{(j)}_2$	$y^{(j)}_3$	$y^{(j)}_4$	$y^{(k)}_1$	$y^{(k)}_2$	$y^{(k)}_3$	$y^{(k)}_4$
$x^{(i)}_{\mathtt{011}} \to x^{(i)}_{\mathtt{051}}$						$-x^{(i)}_{\mathtt{011}}\cdot q$		
$x^{(i)}_{\mathtt{102}} \to x^{(i)}_{\mathtt{302}}$			$-x^{(i)}_{\mathtt{102}}\cdot p$					
$x^{(i)}_{\mathtt{137}} \to x^{(i)}_{\mathtt{177}} \to x^{(i)}_{\mathtt{377}} \to x^{(i)}_{\mathtt{337}}$			$-x^{(i)}_{\mathtt{137}}\cdot p$ $-x^{(i)}_{\mathtt{177}}\cdot p$			$+x^{(i)}_{\mathtt{177}}\cdot q$ $+x^{(i)}_{\mathtt{377}}\cdot q$		
$x^{(i)}_{\mathtt{117}} \to x^{(i)}_{\mathtt{157}} \to x^{(i)}_{\mathtt{317}} \to x^{(i)}_{\mathtt{357}}$			$-x^{(i)}_{\mathtt{117}}\cdot p$ $-x^{(i)}_{\mathtt{157}}\cdot p$			$-x^{(i)}_{\mathtt{117}}\cdot q$ $-x^{(i)}_{\mathtt{317}}\cdot q$		
$x^{(i)}_{\mathtt{107}} \to x^{(i)}_{\mathtt{307}} \to x^{(i)}_{\mathtt{1F7}} \to x^{(i)}_{\mathtt{3F7}}$			$-x^{(i)}_{\mathtt{107}}\cdot p$ $-x^{(i)}_{\mathtt{1F7}}\cdot p$		$-x^{(i)}_{\mathtt{107}}\cdot q$ $-x^{(i)}_{\mathtt{307}}\cdot q$	$-x^{(i)}_{\mathtt{107}}\cdot q$ $-x^{(i)}_{\mathtt{307}}\cdot q$	$-x^{(i)}_{\mathtt{107}}\cdot q$ $-x^{(i)}_{\mathtt{307}}\cdot q$	$-x^{(i)}_{\mathtt{107}}\cdot q$ $-x^{(i)}_{\mathtt{307}}\cdot q$
$x^{(i)}_{\mathtt{577}} \to x^{(i)}_{\mathtt{737}} \to x^{(i)}_{\mathtt{777}} \to x^{(i)}_{\mathtt{537}}$			$+x^{(i)}_{\mathtt{737}}\cdot p$ $+x^{(i)}_{\mathtt{777}}\cdot p$			$+x^{(i)}_{\mathtt{577}}\cdot q$ $+x^{(i)}_{\mathtt{777}}\cdot q$		
$x^{(i)}_{\mathtt{517}} \to x^{(i)}_{\mathtt{717}} \to x^{(i)}_{\mathtt{757}} \to x^{(i)}_{\mathtt{557}}$			$+x^{(i)}_{\mathtt{717}}\cdot p$ $+x^{(i)}_{\mathtt{757}}\cdot p$			$-x^{(i)}_{\mathtt{517}}\cdot q$ $-x^{(i)}_{\mathtt{717}}\cdot q$		
$x^{(i)}_{\mathtt{507}} \to x^{(i)}_{\mathtt{707}} \to x^{(i)}_{\mathtt{7F7}} \to x^{(i)}_{\mathtt{5F7}}$			$+x^{(i)}_{\mathtt{707}}\cdot p$ $+x^{(i)}_{\mathtt{7F7}}\cdot p$		$-x^{(i)}_{\mathtt{507}}\cdot q$ $-x^{(i)}_{\mathtt{707}}\cdot q$	$-x^{(i)}_{\mathtt{507}}\cdot q$ $-x^{(i)}_{\mathtt{707}}\cdot q$	$-x^{(i)}_{\mathtt{507}}\cdot q$ $-x^{(i)}_{\mathtt{707}}\cdot q$	$-x^{(i)}_{\mathtt{507}}\cdot q$ $-x^{(i)}_{\mathtt{707}}\cdot q$
$x^{(i)}_{\mathtt{037}} \to x^{(i)}_{\mathtt{077}} \to x^{(i)}_{\mathtt{F77}} \to x^{(i)}_{\mathtt{F37}}$	$-x^{(i)}_{\mathtt{037}}\cdot p$ $-x^{(i)}_{\mathtt{077}}\cdot p$	$-x^{(i)}_{\mathtt{037}}\cdot p$ $-x^{(i)}_{\mathtt{077}}\cdot p$	$-x^{(i)}_{\mathtt{037}}\cdot p$ $-x^{(i)}_{\mathtt{077}}\cdot p$	$-x^{(i)}_{\mathtt{037}}\cdot p$ $-x^{(i)}_{\mathtt{077}}\cdot p$		$+x^{(i)}_{\mathtt{077}}\cdot q$ $+x^{(i)}_{\mathtt{F77}}\cdot q$		
$x^{(i)}_{\mathtt{017}} \to x^{(i)}_{\mathtt{057}} \to x^{(i)}_{\mathtt{F17}} \to x^{(i)}_{\mathtt{F57}}$	$-x^{(i)}_{\mathtt{017}}\cdot p$ $-x^{(i)}_{\mathtt{057}}\cdot p$	$-x^{(i)}_{\mathtt{017}}\cdot p$ $-x^{(i)}_{\mathtt{057}}\cdot p$	$-x^{(i)}_{\mathtt{017}}\cdot p$ $-x^{(i)}_{\mathtt{057}}\cdot p$	$-x^{(i)}_{\mathtt{017}}\cdot p$ $-x^{(i)}_{\mathtt{057}}\cdot p$		$-x^{(i)}_{\mathtt{017}}\cdot q$ $-x^{(i)}_{\mathtt{F17}}\cdot q$		

We now argue that after completing all 10 operations, the linear conditions of Proposition A.2.1 plus (A.43) to (A.46) are still satisfied. To that end, we first notice that only those $\{x^{(i)}_{\mathbf{b}}\}$ with $\mathbf{b} \in \{\mathtt{051}, \mathtt{302}, \mathtt{337}, \mathtt{357}, \mathtt{3F7}, \mathtt{537}, \mathtt{557}, \mathtt{5F7}, \mathtt{F37}, \mathtt{F57}\}$ will increase after the weight movements. Since those coding types do not participate in any of the terms in (A.43) to (A.46), the conditions (A.43) to (A.46) still hold after the weight movements.

We now observe that the time-sharing conditions (A.14) are still satisfied since we only "move" the weights. We now argue that after completing all 10 operations, all $y_1^{(j)}$ to $y_4^{(j)}$ will decrease by the same amount $(x_{\mathtt{037}}^{(i)}+x_{\mathtt{077}}^{(i)}+x_{\mathtt{017}}^{(i)}+x_{\mathtt{057}}^{(i)})\cdot p_{i\to j}$. The fact that $y_1^{(j)}$, $y_2^{(j)}$ and $y_4^{(j)}$ all decrease by the same amount $(x_{\mathtt{037}}^{(i)}+x_{\mathtt{077}}^{(i)}+x_{\mathtt{017}}^{(i)}+x_{\mathtt{057}}^{(i)})\cdot p_{i\to j}$ can be easily verified by summing up the "impact" of the 10 weight movement operations over each column of Table A.1, for columns 1, 2, and 4, respectively. To prove that $y_3^{(j)}$ also decreases by the same amount, we need to prove that

$$\begin{aligned}&\left(x_{\mathtt{737}}^{(i)}+x_{\mathtt{777}}^{(i)}+x_{\mathtt{717}}^{(i)}+x_{\mathtt{757}}^{(i)}+x_{\mathtt{707}}^{(i)}+x_{\mathtt{7F7}}^{(i)}\right)\cdot p_{i\to j}\\&=\left(x_{\mathtt{102}}^{(i)}+x_{\mathtt{137}}^{(i)}+x_{\mathtt{177}}^{(i)}+x_{\mathtt{117}}^{(i)}+x_{\mathtt{157}}^{(i)}+x_{\mathtt{107}}^{(i)}+x_{\mathtt{1F7}}^{(i)}\right)\cdot p_{i\to j}.\end{aligned} \tag{A.51}$$

We can prove that (A.51) holds by noticing that (A.51) is equivalent to the second equality in (A.46) when removing the zero terms specified in (A.43) and (A.44).

We now argue that after completing all 10 operations, all $y_1^{(k)}$ to $y_4^{(k)}$ will decrease by the same amount $(x_{\mathtt{107}}^{(i)}+x_{\mathtt{307}}^{(i)}+x_{\mathtt{507}}^{(i)}+x_{\mathtt{707}}^{(i)})\cdot p_{i\to k}$. The fact that $y_1^{(k)}$, $y_3^{(k)}$ and $y_4^{(k)}$ all increase by the same amount $(x_{\mathtt{107}}^{(i)}+x_{\mathtt{307}}^{(i)}+x_{\mathtt{507}}^{(i)}+x_{\mathtt{707}}^{(i)})\cdot p_{i\to k}$ can be easily verified by summing up the "impact" of the 10 weight movement operations over each column, for columns 5, 7, and 8, respectively. To prove that $y_2^{(k)}$ also increases by the same amount, we need to prove that

$$\begin{aligned}&\left(x_{\mathtt{177}}^{(i)}+x_{\mathtt{377}}^{(i)}+x_{\mathtt{577}}^{(i)}+x_{\mathtt{777}}^{(i)}+x_{\mathtt{077}}^{(i)}+x_{\mathtt{F77}}^{(i)}\right)\cdot p_{i\to k}\\&=\left(x_{\mathtt{011}}^{(i)}+x_{\mathtt{117}}^{(i)}+x_{\mathtt{317}}^{(i)}+x_{\mathtt{517}}^{(i)}+x_{\mathtt{717}}^{(i)}+x_{\mathtt{017}}^{(i)}+x_{\mathtt{F17}}^{(i)}\right)\cdot p_{i\to k}.\end{aligned} \tag{A.52}$$

We can prove that (A.52) holds by noticing that (A.52) is equivalent to the first equality in (A.46) when removing the zero terms specified in (A.43) and (A.44).

One can also prove that $y_1^{(j,k)}$ to $y_3^{(j,k)}$ remain unchanged since the 10 weight movement operations have no impact on these three y-values. Since $y_1^{(j)}$ to $y_4^{(j)}$ all decrease by the same amount; $y_1^{(k)}$ to $y_4^{(k)}$ all decrease by the same amount; and $y_1^{(j,k)}$ to $y_3^{(j,k)}$ all remain the same, then the decodability conditions (A.22) and (A.23) must hold after the 10 weight movement operations. The proof of Intermediate Case 4 is thus complete. ■

Intermediate Claim 5*:* For any $\vec{R}$ vector and the 156 corresponding non-negative $\{x_{\mathbf{b}}^{(i)}\}$-values satisfying Proposition A.2.1 and (A.43) to (A.50), we can always find another set of 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that $\vec{R}$ and $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ jointly satisfy Proposition A.2.1 and (A.43) to (A.50), plus for all $i \in \{1, 2, 3\}$,

$$\ddot{x}_{\mathbf{b}}^{(i)} = 0, \ \forall\, \mathbf{b} \in \{001, 002\}\,. \tag{A.53}$$

Proof of Intermediate Claim 5*:* We now provide an explicit weight movement such that after the weight-moving process, Proposition A.2.1 and (A.43) to (A.50) hold, and additionally (A.53) holds for the case when $i = 1$, i.e., $(i, j, k) = (1, 2, 3)$. Then by applying the cyclically symmetric weight-moving process to the cases of $(i, j, k) = (2, 3, 1)$ and $(i, j, k) = (3, 1, 2)$, we can construct the new values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ that satisfy Proposition A.2.1, (A.43) to (A.50), and (A.53) for all i.

The weight movements for the case of $(i, j, k) = (1, 2, 3)$ consist of two steps: Firstly, we make $x_{001}^{(1)} = 0$, and then secondly, we make $x_{002}^{(1)} = 0$. For the first step, we assume $x^{(1)}_{\mathsf{001}} > 0$. Otherwise, we can skip to the second step directly. We now perform the following six operations:

$$\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{357}}\} \to \{x^{(1)}_{\mathsf{051}}, x^{(1)}_{\mathsf{3F7}}\}, \tag{OP3}$$

$$\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{557}}\} \to \{x^{(1)}_{\mathsf{051}}, x^{(1)}_{\mathsf{5F7}}\}, \tag{OP4}$$

$$\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{F57}}\} \to x^{(1)}_{\mathsf{051}}, \tag{OP5}$$

$$x^{(1)}_{\mathsf{001}} \xrightarrow{\Delta/p_{1\to 3}} x^{(1)}_{\mathsf{051}} \quad \text{and} \quad x^{(2)}_{\mathsf{537}} \xrightarrow{\Delta/p_{2\to 3}} x^{(2)}_{\mathsf{F37}} \quad \text{where } \Delta = \mathsf{min}\{x^{(1)}_{\mathsf{001}} \cdot p_{1\to 3},\ x^{(2)}_{\mathsf{537}} \cdot p_{2\to 3}\}, \tag{OP6}$$

$$x^{(1)}_{\mathsf{001}} \xrightarrow{\Delta/p_{1\to 3}} x^{(1)}_{\mathsf{051}} \quad \text{and} \quad x^{(2)}_{\mathsf{557}} \xrightarrow{\Delta/p_{2\to 3}} x^{(2)}_{\mathsf{F57}} \quad \text{where } \Delta = \mathsf{min}\{x^{(1)}_{\mathsf{001}} \cdot p_{1\to 3},\ x^{(2)}_{\mathsf{557}} \cdot p_{2\to 3}\}, \tag{OP7}$$

$$x^{(1)}_{\mathsf{001}} \xrightarrow{\Delta/p_{1\to 3}} x^{(1)}_{\mathsf{051}} \quad \text{and} \quad x^{(2)}_{\mathsf{5F7}} \xrightarrow{\Delta/p_{2\to 3}} \emptyset \quad \text{where } \Delta = \mathsf{min}\{x^{(1)}_{\mathsf{001}} \cdot p_{1\to 3},\ x^{(2)}_{\mathsf{5F7}} \cdot p_{2\to 3}\}. \tag{OP8}$$

We now argue that after these operations, (i) Proposition A.2.1 and (A.43) to (A.50) still hold; and (ii) the new value of $x^{(1)}_{\mathsf{001}}$ is zero. To prove (i), we note that after the above weight movements, the time-sharing condition (A.14) of Proposition A.2.1 still holds because except for the operations (OP5) and (OP8), we only "move" the weight between different frequencies while keeping the overall sum. And both (OP5) and (OP8) decrease the total sum. As a result, the time-sharing condition still holds. Moreover, since none of the coding types involved in (OP3) to (OP8) participate in any of the terms in (A.43) to (A.50), the conditions (A.43) to (A.50) still hold after these operations.

In the following, we prove that the decodability conditions (A.22) and (A.23) of Proposition A.2.1 still hold after performing any one of the above 6 weight-moving operations. For example, we claim that the decodability conditions still hold after (OP3). For that, we first notice that

$$\begin{aligned}
\mathsf{TYPE}^{(1)}_{\mathsf{001}} \text{ in 11-bitstring} &= 0000\,0000\,001,\\
\mathsf{TYPE}^{(1)}_{\mathsf{357}} \text{ in 11-bitstring} &= 0011\,0101\,111,\\
\mathsf{TYPE}^{(1)}_{\mathsf{051}} \text{ in 11-bitstring} &= 0000\,0101\,001,\\
\mathsf{TYPE}^{(1)}_{\mathsf{3F7}} \text{ in 11-bitstring} &= 0011\,1111\,111,
\end{aligned}$$

where each bit is associated to one y-value and the associated 11 y-values are $y^{(2)}_1$ to $y^{(2)}_4$, $y^{(3)}_1$ to $y^{(3)}_4$, and $y^{(2,3)}_1$ to $y^{(2,3)}_3$ in the order of 11-bitstring, see (A.11). For shorthand, we denote the collection of these y-values corresponding to the first four bits, the second four bits, and the last three bits as $\vec{y}^{(2)}_{1-4}$, $\vec{y}^{(3)}_{1-4}$, and $\vec{y}^{(2,3)}_{1-3}$, respectively. Then by the same arguments as used in the proof of *Intermediate Claim* 2, one can easily prove that the 7 different y-values: $\vec{y}^{(2)}_{1-4}$ and $\vec{y}^{(2,3)}_{1-3}$, remain unchanged after (OP3). If we apply the same arguments as used in the proof of *Intermediate Claim* 2, we can also prove that all y-values in the collection $\vec{y}^{(3)}_{1-4}$ (the second four) decrease by the same amount of $\left(\mathsf{min}\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{357}}\} \cdot p_{1\to 3}\right)$. Since other y-values were intact, the decodability equalities (A.22) and (A.23) are still satisfied after (OP3).

For the weight movement (OP4), we can prove by similar arguments that after (OP4), all $\vec{y}^{(2)}_{1-4}$ and $\vec{y}^{(2,3)}_{1-3}$ remain the same and all $\vec{y}^{(3)}_{1-4}$ decrease by the same amount of $\left(\mathsf{min}\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{557}}\} \cdot p_{1\to 3}\right)$. Similarly, after the weight movement (OP5), all $\vec{y}^{(2)}_{1-4}$ and $\vec{y}^{(2,3)}_{1-3}$ remain the same and all $\vec{y}^{(3)}_{1-4}$ decrease by the same amount of $\left(\mathsf{min}\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{F57}}\} \cdot p_{1\to 3}\right)$. Since other y-values were intact, the decodability equalities (A.22) and (A.23) still hold after these operations.

We now prove that after (OP6), the decodability conditions in Proposition A.2.1 still hold. Since (OP6) involves the frequencies of different node indices $\{x^{(1)}_{\mathsf{001}}, x^{(1)}_{\mathsf{051}}, x^{(2)}_{\mathsf{537}}, x^{(2)}_{\mathsf{F37}}\}$, we first provide the following table that summarizes the contributions of these frequencies to the y-values:

Table A.2: The contributions of $x^{(1)}_{\texttt{001}}$, $x^{(2)}_{\texttt{537}}$, $x^{(1)}_{\texttt{051}}$, and $x^{(2)}_{\texttt{F37}}$ to the y-values.

	$\vec{y}^{(1)}_{1-4}$	$\vec{y}^{(2)}_{1-4}$	$\vec{y}^{(3)}_{1-4}$	$\vec{y}^{(1,2)}_{1-3}$	$\vec{y}^{(2,3)}_{1-3}$	$\vec{y}^{(3,1)}_{1-3}$
$x^{(1)}_{\texttt{001}}$		0000	0000		001	
$x^{(2)}_{\texttt{537}}$	0011		0101			111
$x^{(1)}_{\texttt{051}}$		0000	0101		001	
$x^{(2)}_{\texttt{F37}}$	0011		1111			111

For example, since $\texttt{537} = 01010011111$ in 11-bitstring and $x^{(2)}_{\texttt{537}}$ contributes to $\{\vec{y}^{(3)}_{1-4}, \vec{y}^{(1)}_{1-4}, \vec{y}^{(3,1)}_{1-3}\}$, we can thus list the contribution of $x^{(2)}_{\texttt{537}}$ to all the y-values as in the second row of Table A.2. The first, third, and fourth rows of Table A.2 can be populated similarly. If we compare the first and the third rows of Table A.2, we can see that the operation of $x^{(1)}_{\texttt{001}} \xrightarrow{\Delta/p_{1\to 3}} x^{(1)}_{\texttt{051}}$ in (OP6) will decrease both $y^{(3)}_2$ and $y^{(3)}_4$ by the same amount Δ while all the other 19 y-values remain the same. If we compare the second and the fourth rows of Table A.2, we can see that the operation of $x^{(2)}_{\texttt{537}} \xrightarrow{\Delta/p_{2\to 3}} x^{(2)}_{\texttt{F37}}$ will decrease both $y^{(3)}_1$ and $y^{(3)}_3$ by the same amount Δ while all the other 19 y-values remain the same. Since (OP6) performs both $x^{(1)}_{\texttt{001}} \xrightarrow{\Delta/p_{1\to 3}} x^{(1)}_{\texttt{051}}$ and $x^{(2)}_{\texttt{537}} \xrightarrow{\Delta/p_{2\to 3}} x^{(2)}_{\texttt{F37}}$ simultaneously, in the end we will have all four values of $\vec{y}^{(3)}_{1-4}$ decrease by the same amount of Δ while the rest 17 y-values remain the same. As a result, the decodability equalities (A.22) and (A.23) of Proposition A.2.1 are still satisfied after (OP6). Similar arguments can be used to prove that after (OP7) and (OP8), the decodability equalities of Proposition A.2.1 still hold.

To prove (ii), we notice that after the above 6 weight movements (OP3) to (OP8), the final $\{x^{(i)}_{\mathbf{b}}\}$-values satisfy Proposition A.2.1. Then Lemma A.3.1 implies that (A.27) to (A.36) must hold. Since (A.43), (A.44), and (A.50) are true, if we only count the coding types that may have non-zero value, then (A.35) can be written as follows.

$$\begin{aligned}\left(x^{(i)}_{\mathsf{001}}+x^{(i)}_{\mathsf{051}}\right)\cdot p_{i\to j\vee k}=&\left(x^{(i)}_{\mathsf{051}}+x^{(i)}_{\mathsf{357}}+x^{(i)}_{\mathsf{557}}+x^{(i)}_{\mathsf{F57}}\right)\cdot p_{i\to k}\\&+\left(x^{(j)}_{\mathsf{537}}+x^{(j)}_{\mathsf{557}}+x^{(j)}_{\mathsf{5F7}}\right)\cdot p_{j\to k},\end{aligned}\tag{A.54}$$

Eq. (A.54) further implies the following inequality:

$$\begin{aligned}x^{(i)}_{\mathsf{001}}\cdot p_{i\to j\vee k}\leq&\left(x^{(i)}_{\mathsf{357}}+x^{(i)}_{\mathsf{557}}+x^{(i)}_{\mathsf{F57}}\right)\cdot p_{i\to k}\\&+\left(x^{(j)}_{\mathsf{537}}+x^{(j)}_{\mathsf{557}}+x^{(j)}_{\mathsf{5F7}}\right)\cdot p_{j\to k},\end{aligned}\tag{A.55}$$

because we always have $x^{(i)}_{\mathsf{051}}\cdot p_{i\to j\vee k}\geq x^{(i)}_{\mathsf{051}}\cdot p_{i\to k}$.

Then notice that after performing (OP3) to (OP8), we will have either $x^{(1)}_{\mathsf{001}}=0$ or the total sum $x^{(1)}_{\mathsf{357}}+x^{(1)}_{\mathsf{557}}+x^{(1)}_{\mathsf{F57}}+x^{(2)}_{\mathsf{537}}+x^{(2)}_{\mathsf{557}}+x^{(2)}_{\mathsf{5F7}}=0$. Note that whenever the latter sum is zero, by (A.55) when $(i,j,k)=(1,2,3)$, we also have $x^{(1)}_{\mathsf{001}}=0$. As a result, we must have $x^{(1)}_{\mathsf{001}}=0$ after the above 6 weight movements.

We now present the second step, which makes $x^{(1)}_{\mathsf{002}}=0$. To that end, we perform the following six operations:

$$\{x^{(1)}_{\mathsf{002}},x^{(1)}_{\mathsf{337}}\}\to\{x^{(1)}_{\mathsf{302}},x^{(1)}_{\mathsf{F37}}\},\tag{OP9}$$

$$\{x^{(1)}_{\mathsf{002}},x^{(1)}_{\mathsf{357}}\}\to\{x^{(1)}_{\mathsf{302}},x^{(1)}_{\mathsf{F57}}\},\tag{OP10}$$

$$\{x^{(1)}_{\mathsf{002}},x^{(1)}_{\mathsf{3F7}}\}\to x^{(1)}_{\mathsf{302}},\tag{OP11}$$

$$\begin{aligned}&x^{(1)}_{\mathsf{002}}\xrightarrow{\Delta/p_{1\to2}}x^{(1)}_{\mathsf{302}}\quad\text{and}\quad x^{(3)}_{\mathsf{337}}\xrightarrow{\Delta/p_{3\to2}}x^{(3)}_{\mathsf{3F7}}\\&\text{where }\Delta=\mathsf{min}\{x^{(1)}_{\mathsf{002}}\cdot p_{1\to2},\ x^{(3)}_{\mathsf{337}}\cdot p_{3\to2}\},\end{aligned}\tag{OP12}$$

$$\begin{aligned}&x^{(1)}_{\mathsf{002}}\xrightarrow{\Delta/p_{1\to2}}x^{(1)}_{\mathsf{302}}\quad\text{and}\quad x^{(3)}_{\mathsf{537}}\xrightarrow{\Delta/p_{3\to2}}x^{(3)}_{\mathsf{5F7}}\\&\text{where }\Delta=\mathsf{min}\{x^{(1)}_{\mathsf{002}}\cdot p_{1\to2},\ x^{(3)}_{\mathsf{537}}\cdot p_{3\to2}\},\end{aligned}\tag{OP13}$$

$$\begin{aligned}&x^{(1)}_{\mathsf{002}}\xrightarrow{\Delta/p_{1\to2}}x^{(1)}_{\mathsf{302}}\quad\text{and}\quad x^{(3)}_{\mathsf{F37}}\xrightarrow{\Delta/p_{3\to2}}\emptyset\\&\text{where }\Delta=\mathsf{min}\{x^{(1)}_{\mathsf{002}}\cdot p_{1\to2},\ x^{(3)}_{\mathsf{F37}}\cdot p_{3\to2}\}.\end{aligned}\tag{OP14}$$

Again, we will prove that after these 6 weight movements, (i) Proposition A.2.1 and (A.43) to (A.50) hold; and (ii) the new value of $x^{(1)}_{\mathsf{002}}$ is zero. The proof of (i) is almost identical to that of the first step and we thus omit the detailed derivations. To prove (ii), we notice that after these weight-moving operations, the final $\{x^{(i)}_{\mathbf{b}}\}$-values still satisfy Proposition A.2.1. Then Lemma A.3.1 implies that (A.27) to (A.36) must hold. Since (A.43), (A.44), and (A.50) are true, if we only count the coding types that may have non-zero value, then (A.36) can be written as follows.

$$\left(x^{(i)}_{\mathsf{002}} + x^{(i)}_{\mathsf{302}}\right) \cdot p_{i\to j\vee k} = \left(x^{(i)}_{\mathsf{302}} + x^{(i)}_{\mathsf{337}} + x^{(i)}_{\mathsf{357}} + x^{(i)}_{\mathsf{3F7}}\right) \cdot p_{i\to j} + \left(x^{(k)}_{\mathsf{337}} + x^{(k)}_{\mathsf{537}} + x^{(k)}_{\mathsf{F37}}\right) \cdot p_{k\to j},$$

which in turn implies when $(i,j,k) = (\mathsf{1},\mathsf{2},\mathsf{3})$,

$$x^{(\mathsf{1})}_{\mathsf{002}} \cdot p_{\mathsf{1}\to\mathsf{2}\vee\mathsf{3}} \leq \left(x^{(\mathsf{1})}_{\mathsf{337}} + x^{(\mathsf{1})}_{\mathsf{357}} + x^{(\mathsf{1})}_{\mathsf{3F7}}\right) \cdot p_{\mathsf{1}\to\mathsf{2}} + \left(x^{(\mathsf{3})}_{\mathsf{337}} + x^{(\mathsf{3})}_{\mathsf{537}} + x^{(\mathsf{3})}_{\mathsf{F37}}\right) \cdot p_{\mathsf{3}\to\mathsf{2}}. \tag{A.56}$$

We then observe that after the above 6 operations (OP9) to (OP14), we will have either $x^{(1)}_{\mathsf{002}} = 0$ or $x^{(1)}_{\mathsf{337}} + x^{(1)}_{\mathsf{357}} + x^{(1)}_{\mathsf{3F7}} + x^{(3)}_{\mathsf{337}} + x^{(3)}_{\mathsf{537}} + x^{(3)}_{\mathsf{F37}} = 0$. The by (A.56), we must have $x^{(1)}_{\mathsf{002}} = 0$ after the above 6 weight-moving process.

Thus far, we have proven (A.53) for the case of $i = \mathsf{1}$ while satisfying the linear conditions of Proposition A.2.1 and (A.43) to (A.50). Note that in our weight movements (OP3)–(OP8) and (OP9)–(OP14), we never increase $x^{(2)}_{\mathsf{001}}$, $x^{(2)}_{\mathsf{002}}$, $x^{(3)}_{\mathsf{001}}$, and $x^{(3)}_{\mathsf{002}}$. Therefore, we can simply apply the above 2-step procedure to the cases of $(i,j,k) = (\mathsf{2},\mathsf{3},\mathsf{1})$ and $(i,j,k) = (\mathsf{3},\mathsf{1},\mathsf{2})$, sequentially. In the end, the final $\{x^{(i)}_{\mathbf{b}}\}$-values satisfy Proposition A.2.1 and the conditions (A.43) to (A.53). The proof is thus complete. ∎

A.3.4 Proof of Lemma A.3.3

Given $\vec{R}$ and the reception probabilities, consider 156 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ such that jointly they satisfy Proposition A.2.1 and (A.38). Since by (A.38) all the $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$-values with $\mathbf{b} \in \mathsf{FTs}\backslash\overline{\mathsf{FTs}}$ are zeros, we only consider the 30 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ with $\mathbf{b} \in \overline{\mathsf{FTs}}$ for the ongoing discussions.

For the proof of Lemma A.3.3, we first prove the following claim.

Claim: The above 30 non-negative values $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$ for all $\mathbf{b} \in \overline{\mathsf{FTs}}$ jointly satisfy the following equalities: for all $(i, j, k) \in \{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$,

$$R_{i\to j} + R_{i\to k} = \left(\ddot{x}_{\mathsf{051}}^{(i)} + \ddot{x}_{\mathsf{302}}^{(i)}\right) p_{i\to j\vee k}, \tag{A.57}$$

$$R_{i\to j}\frac{p_{i\to \bar{j}k}}{p_{i\to j\vee k}} = \left(\ddot{x}_{\mathsf{337}}^{(i)} + \ddot{x}_{\mathsf{357}}^{(i)} + \ddot{x}_{\mathsf{3F7}}^{(i)}\right) \cdot p_{i\to j} + \left(\ddot{x}_{\mathsf{337}}^{(k)} + \ddot{x}_{\mathsf{537}}^{(k)} + \ddot{x}_{\mathsf{F37}}^{(k)}\right) \cdot p_{k\to j}, \tag{A.58}$$

$$R_{i\to k}\frac{p_{i\to j\bar{k}}}{p_{i\to j\vee k}} = \left(\ddot{x}_{\mathsf{357}}^{(i)} + \ddot{x}_{\mathsf{557}}^{(i)} + \ddot{x}_{\mathsf{F57}}^{(i)}\right) \cdot p_{i\to k} + \left(\ddot{x}_{\mathsf{537}}^{(j)} + \ddot{x}_{\mathsf{557}}^{(j)} + \ddot{x}_{\mathsf{5F7}}^{(j)}\right) \cdot p_{j\to k}. \tag{A.59}$$

Proof of Claim. Since node indices are cyclically decided, we prove (A.57)–(A.59) only for the case when $(i, j, k) = (1, 2, 3)$. That is,

$$R_{1\to 2} + R_{1\to 3} = \left(\ddot{x}_{\mathsf{051}}^{(1)} + \ddot{x}_{\mathsf{302}}^{(1)}\right) \cdot p_{1\to 2\vee 3}, \tag{A.60}$$

$$R_{1\to 2}\frac{p_{1\to \bar{2}3}}{p_{1\to 2\vee 3}} = \left(\ddot{x}_{\mathsf{337}}^{(1)} + \ddot{x}_{\mathsf{357}}^{(1)} + \ddot{x}_{\mathsf{3F7}}^{(1)}\right) \cdot p_{1\to 2} + \left(\ddot{x}_{\mathsf{337}}^{(3)} + \ddot{x}_{\mathsf{537}}^{(3)} + \ddot{x}_{\mathsf{F37}}^{(3)}\right) \cdot p_{3\to 2}, \tag{A.61}$$

$$R_{1\to 3}\frac{p_{1\to 2\bar{3}}}{p_{1\to 2\vee 3}} = \left(\ddot{x}_{\mathsf{357}}^{(1)} + \ddot{x}_{\mathsf{557}}^{(1)} + \ddot{x}_{\mathsf{F57}}^{(1)}\right) \cdot p_{1\to 3} + \left(\ddot{x}_{\mathsf{537}}^{(2)} + \ddot{x}_{\mathsf{557}}^{(2)} + \ddot{x}_{\mathsf{5F7}}^{(2)}\right) \cdot p_{2\to 3}. \tag{A.62}$$

We now make the following observations. Since the above $\{\ddot{x}_{\mathbf{b}}^{(i)} : \forall\, i \in \{1, 2, 3\}$ and $\mathbf{b} \in \overline{\mathsf{FTs}}\}$ satisfy Proposition A.2.1, Lemma A.3.1 implies that they satisfies (A.27) as well. We then note that (A.60) is a direct result of the equality (A.27) of Lemma A.3.1 when $(i, j, k) = (2, 3, 1)$.

We now use the equalities (A.28) and (A.29) when $(i, j, k) = (2, 3, 1)$. Since type-051 (resp. type-302) is the only coding type in $\overline{\mathsf{FT}}\mathsf{s}$ with $b_{10} = 0$ (resp. $b_{11} = 0$), we thus have, respectively,

$$R_{1\to 3} = \ddot{x}^{(1)}_{\mathsf{051}} \cdot p_{1\to 2\vee 3}, \tag{A.63}$$

$$R_{1\to 2} = \ddot{x}^{(1)}_{\mathsf{302}} \cdot p_{1\to 2\vee 3}. \tag{A.64}$$

Then, (A.61) can be derived as follows. From the equality (A.35) when $(i, j, k) = (2, 3, 1)$, we have

$$\ddot{x}^{(1)}_{\mathsf{302}} \cdot p_{1\to 2\vee 3} = \left(\ddot{x}^{(3)}_{\mathsf{337}} + \ddot{x}^{(3)}_{\mathsf{537}} + \ddot{x}^{(3)}_{\mathsf{F37}}\right) \cdot p_{3\to 2} + \left(\ddot{x}^{(1)}_{\mathsf{302}} + \ddot{x}^{(1)}_{\mathsf{337}} + \ddot{x}^{(1)}_{\mathsf{357}} + \ddot{x}^{(1)}_{\mathsf{3F7}}\right) \cdot p_{1\to 2}.$$

By simple probability manipulation, the above equality is equivalent to

$$\ddot{x}^{(1)}_{\mathsf{302}} \cdot p_{1\to \overline{2}3} = \left(\ddot{x}^{(1)}_{\mathsf{337}} + \ddot{x}^{(1)}_{\mathsf{357}} + \ddot{x}^{(1)}_{\mathsf{3F7}}\right) \cdot p_{1\to 2} + \left(\ddot{x}^{(3)}_{\mathsf{337}} + \ddot{x}^{(3)}_{\mathsf{537}} + \ddot{x}^{(3)}_{\mathsf{F37}}\right) \cdot p_{3\to 2}. \tag{A.65}$$

Then (A.61) is derived by substituting $\ddot{x}^{(1)}_{\mathsf{302}} = R_{1\to 2}/p_{1\to 2\vee 3}$ (see (A.64) again) on the LHS of (A.65).

Similarly, one can derive (A.62) by using (A.63) and the equality (A.36) when $(i, j, k) = (2, 3, 1)$. The claim is thus proven.. ■

Using the above claim, we will prove Lemma A.3.3 by explicitly constructing $t^{(i)}_{[\mathrm{u}]}$ and $t^{(i)}_{[\mathrm{c},\,1]}$ to $t^{(i)}_{[\mathrm{c},\,4]}$ values as follows.

$$t^{(i)}_{[\mathrm{u}]} = \ddot{x}^{(i)}_{\mathsf{051}} + \ddot{x}^{(i)}_{\mathsf{302}}, \tag{A.66}$$

$$t^{(i)}_{[\mathrm{c},\,1]} = \ddot{x}^{(i)}_{\mathsf{357}} + \ddot{x}^{(i)}_{\mathsf{3F7}}, \tag{A.67}$$

$$t^{(i)}_{[\mathrm{c},\,2]} = \ddot{x}^{(i)}_{\mathsf{537}} + \ddot{x}^{(i)}_{\mathsf{5F7}}, \tag{A.68}$$

$$t^{(i)}_{[\mathrm{c},\,3]} = \ddot{x}^{(i)}_{\mathsf{337}} + \ddot{x}^{(i)}_{\mathsf{F37}}, \tag{A.69}$$

$$t^{(i)}_{[\mathrm{c},\,4]} = \ddot{x}^{(i)}_{\mathsf{557}} + \ddot{x}^{(i)}_{\mathsf{F57}}. \tag{A.70}$$

In the following, we prove that the above $\{t_i\}$-values satisfy the linear conditions of Proposition 3.2.2 (when $<$ being replaced by $\leq$).

Since the $\{\ddot{x}_{\mathbf{b}}^{(i)}\}$-values satisfy the time-sharing condition (A.14) of Proposition A.2.1, the $\{t_i\}$-values in the above construction also satisfy the time-sharing condition (3.4).

By (A.57) and (A.66), we have

$$R_{i\to j} + R_{i\to k} = t_{[\mathrm{u}]}^{(i)} \cdot p_{i\to j\vee k},$$

which implies (3.6).

We now show that our construction also satisfies (3.7) and (3.8). By our construction (A.67)–(A.70), the followings are always true: for all $i \in \{1, 2, 3\}$,

$$\begin{aligned}
\left(\ddot{x}_{337}^{(i)} + \ddot{x}_{357}^{(i)} + \ddot{x}_{3F7}^{(i)}\right) &\leq \left(t_{[\mathrm{c},1]}^{(i)} + t_{[\mathrm{c},3]}^{(i)}\right), \\
\left(\ddot{x}_{337}^{(i)} + \ddot{x}_{537}^{(i)} + \ddot{x}_{F37}^{(i)}\right) &\leq \left(t_{[\mathrm{c},2]}^{(i)} + t_{[\mathrm{c},3]}^{(i)}\right), \\
\left(\ddot{x}_{357}^{(i)} + \ddot{x}_{557}^{(i)} + \ddot{x}_{F57}^{(i)}\right) &\leq \left(t_{[\mathrm{c},1]}^{(i)} + t_{[\mathrm{c},4]}^{(i)}\right), \\
\left(\ddot{x}_{537}^{(j)} + \ddot{x}_{557}^{(j)} + \ddot{x}_{5F7}^{(j)}\right) &\leq \left(t_{[\mathrm{c},2]}^{(i)} + t_{[\mathrm{c},4]}^{(i)}\right).
\end{aligned}$$

Since we have already shown that (A.58) and (A.59) are true, one can easily verify by direct substitutions that (3.7) and (3.8) are satisfied as well. The proof of Lemma A.3.3 is thus complete.

B. DETAILED CONSTRUCTION FOR THE 3-NODE ACHIEVABILITY SCHEME IN SCENARIO 2

We provide the first-order analysis for the achievability scheme in Proposition 3.2.1.

Suppose that all the network nodes share the same parameters before initiating, see the discussion in Section 3.5. Also assume that a common random seed is available to all nodes in advance.

We similarly follow the 2-stage scheme used in the proof of Proposition 3.2.2. The difference is that we need to revise the scheme in Scenario 1 to take into account the assumption of Scenario 2 that any feedback information needs to be sent over the regular channels as well. Our scheme closely mimics the scheme in Section 3.5 but now uses some form of random linear network coding (RLNC), which allows us to circumvent the need of instant causal feedback (after each transmission) and can thus use "batch feedback" that reports the reception status with delay. With a common random seed available to all three nodes, the RLNC operations of one node can be "simulated" in the other nodes as well. This allows the same kind of "bookkeeping" as used in the proof of Proposition 3.2.2. Since bookkeeping may be computationally expensive, in practice, network code designers can place the coding vectors used by the RLNC in the header of the packets, which circumvents the need of bookkeeping. However, putting the coding vectors in the header reduces the data rate. As a result, to minimize the loss of capacity, we opt to use bookkeeping instead of the traditional practice of putting the coding vectors in the header of the packet.

We now explain the main RLNC process for each stage. In each stage, we assume that nodes will sequentially transmit following the order of the node indices $\{1, 2, 3\}$.

We now define the following three constants for each node $i \in \{1, 2, 3\}$ that can facilitate our discussion:

$$\eta_1^{(i)} \triangleq \frac{R_{i\to j}}{R_{i\to j} + R_{i\to k} + R_{i\to jk}}, \tag{B.1}$$

$$\eta_2^{(i)} \triangleq \frac{R_{i\to k}}{R_{i\to j} + R_{i\to k} + R_{i\to jk}}, \tag{B.2}$$

$$\eta_3^{(i)} \triangleq \frac{R_{i\to jk}}{R_{i\to j} + R_{i\to k} + R_{i\to jk}}. \tag{B.3}$$

Obviously, $\eta_1^{(i)} + \eta_2^{(i)} + \eta_3^{(i)} = 1$ for any $i \in \{1, 2, 3\}$. Totally, there are 9 such constants. Note that each of the network node can compute all 9 constants since $\vec{R}$ is available to all nodes. Without loss of generality, we can also assume

$$t_{[\mathrm{u}]}^{(i)} \left(\eta_1^{(i)} + \eta_3^{(i)}\right) \cdot p_{i\to \overline{j}k} < \left(t_{[\mathrm{c},\,1]}^{(i)} + t_{[\mathrm{c},\,3]}^{(i)}\right) \cdot p_{i\to j} + \left(t_{[\mathrm{c},\,2]}^{(k)} + t_{[\mathrm{c},\,3]}^{(k)}\right) \cdot p_{k\to j}, \tag{B.4}$$

$$t_{[\mathrm{u}]}^{(i)} \left(\eta_2^{(i)} + \eta_3^{(i)}\right) \cdot p_{i\to j\overline{k}} < \left(t_{[\mathrm{c},\,1]}^{(i)} + t_{[\mathrm{c},\,4]}^{(i)}\right) \cdot p_{i\to k} + \left(t_{[\mathrm{c},\,2]}^{(j)} + t_{[\mathrm{c},\,4]}^{(j)}\right) \cdot p_{j\to k}. \tag{B.5}$$

The reason is that we can always set $t_{[\mathrm{u}]}^{(i)}$ to be arbitrarily close to $\frac{R_{i\to j}+R_{i\to k}+R_{i\to jk}}{p_{i\to j\vee k}}$ but still larger than $\frac{R_{i\to j}+R_{i\to k}+R_{i\to jk}}{p_{i\to j\vee k}}$ without violating any of the inequalities (3.4) and (3.6). As a result, $t_{[\mathrm{u}]}^{(i)}\eta_1^{(i)}$ can be made arbitrarily close to $\frac{R_{i\to j}}{p_{i\to j\vee k}}$ and $t_{[\mathrm{u}]}^{(i)}\eta_3^{(i)}$ arbitrarily close to $\frac{R_{i\to jk}}{p_{i\to j\vee k}}$. By (3.7), we thus have (B.4). Similarly, since $t_{[\mathrm{u}]}^{(i)}\eta_2^{(i)}$ and $t_{[\mathrm{u}]}^{(i)}\eta_3^{(i)}$ can be made arbitrarily close to $\frac{R_{i\to k}}{p_{i\to j\vee k}}$ and $\frac{R_{i\to jk}}{p_{i\to j\vee k}}$, respectively, (3.8) implies (B.5).

Description of Stage 1: Each node i performs the following RLNC operations exactly for $nt_{[\mathrm{u}]}^{(i)}$ number of time slots. Specifically, consider the first $\eta_1^{(i)}$ portion of the allotted $nt_{[\mathrm{u}]}^{(i)}$ times. In each of those $nt_{[\mathrm{u}]}^{(i)}\eta_1^{(i)}$ time slots, node i chooses a $1 \times (nR_{i\to j})$ random encoding row vector $\mathbf{c}_t \in \mathbb{F}_q^{nR_{i\to j}}$ independently and uniformly randomly and transmits $X_i(t)$ by $X_i(t) = \mathbf{c}_t\mathbf{W}_{i\to j}^{\top}$. We now consider the the second $\eta_2^{(i)}$ portion of the $nt_{[\mathrm{u}]}^{(i)}$ time slots. In each of those $nt_{[\mathrm{u}]}^{(i)}\eta_2^{(i)}$ time slots, node i chooses a $1 \times (nR_{i\to k})$ random coding vector $\mathbf{c}_t$ independently and uniformly randomly and transmits $X_i(t) = \mathbf{c}_t\mathbf{W}_{i\to k}^{\top}$. Finally, consider the last $\eta_3^{(i)}$ portion of the $nt_{[\mathrm{u}]}^{(i)}$ time slots. In each of those $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_3$ time slots, node i chooses a $1 \times (nR_{i\to jk})$ vector $\mathbf{c}_t$ independently and uniformly randomly and transmits $X_i(t) = \mathbf{c}_t\mathbf{W}^{\top}_{i\to jk}$. Namely for the allotted $nt^{(i)}_{[\mathrm{u}]}$ time slots, node i sequentially transmits some random mixture of the packets $\mathbf{W}_{i\to j}$, $\mathbf{W}_{i\to k}$, and $\mathbf{W}_{i\to jk}$ over the fixed fractions $\eta^{(i)}_1$, $\eta^{(i)}_2$, and $\eta^{(i)}_3$ of times, respectively, and does not care whether the transmitted packet is correctly received or not. Stage 1 can be finished in exactly $n(\sum_i t^{(i)}_{[\mathrm{u}]})$ slots.

Note that when node i computes the coding vectors $\mathbf{c}_t$, the other nodes j and k can also "simulate" the computation and thus know the $\mathbf{c}_t$ vector used by node i. As a result, if node j receives a coded packet $X_i(t) = \mathbf{c}_t\mathbf{W}^{\top}_{i\to jk}$ during the third fraction of node i's transmission, node j knows the $\mathbf{c}_t$ vector used for encoding.

New Packet Regrouping After Stage 1: After Stage 1, we put some of those $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}$ packets that were sent during the first fraction of Stage 1, totally there are $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_1$ such packets, into two disjoint groups. Specifically, we use $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}_{\overline{j}k}$ to denote those packets $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}$ that are heard only by node k and not by node j; and we use $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}_j$ to denote those packets that are heard by node j (may or may not be heard by node k). In average, there are $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_1 p_{i\to\overline{j}k}$ number of $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}_{\overline{j}k}$ packets and $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_1 p_{i\to j}$ number of $\{\mathbf{c}_t\mathbf{W}_{i\to j}\}_j$ packets.

Symmetrically, we put some of those $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_2$ packets sent during the second fraction of Stage 1 into two disjoint groups. That is, $\{\mathbf{c}_t\mathbf{W}_{i\to k}\}_{j\overline{k}}$ and $\{\mathbf{c}_t\mathbf{W}_{i\to k}\}_k$ denote those packets that are heard by node j only, and by node k (may or may not be heard by node j), respectively. The size of each group, in average, is $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_2 p_{i\to j\overline{k}}$ and $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_2 p_{i\to k}$, respectively.

Finally, among the $nt^{(i)}_{[\mathrm{u}]}\eta^{(i)}_3$ number of the packets $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}$ sent in the third fraction $\eta^{(i)}_3$, we place them into 4 different groups but this time the groups are not necessarily disjoint. Specifically, we use $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_{\overline{j}k}$ and $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_j$ to denote, respectively, the packets that are received by node k only (not by node j) and by node j (regardless whether node k receives them). We use $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_{j\overline{k}}$ and $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_k$ to denote, respectively, those packets that are heard by node j only (not by node k) and by node k (regardless whether node j receives them). The first two groups of packets are disjoint and the last two groups of packets are disjoint. But there may be overlap between $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_j$ and $\{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_k$. In average, the sizes of these four groups are $nt^{(i)}_{[u]}\eta^{(i)}_3 p_{i\to\bar{j}k}$, $nt^{(i)}_{[u]}\eta^{(i)}_3 p_{i\to j}$, $nt^{(i)}_{[u]}\eta^{(i)}_3 p_{i\to j\bar{k}}$, and $nt^{(i)}_{[u]}\eta^{(i)}_3 p_{i\to k}$, respectively.

For ease of description, we further put some of the groups of the packets into super groups. Specifically, for all $(i, j, k) \in \{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$,

$$\tilde{\mathbf{W}}^{(k)}_{i\to j} \triangleq \{\mathbf{c}_t\mathbf{W}_{i\to j}\}_{\bar{j}k} \cup \{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_{\bar{j}k}, \tag{B.6}$$

$$\tilde{\mathbf{W}}^{(j)}_{i\to k} \triangleq \{\mathbf{c}_t\mathbf{W}_{i\to k}\}_{j\bar{k}} \cup \{\mathbf{c}_t\mathbf{W}_{i\to jk}\}_{j\bar{k}}. \tag{B.7}$$

In total, there are 6 such $\tilde{\mathbf{W}}$-groups by definition and their sizes, in average, are

$$|\tilde{\mathbf{W}}^{(k)}_{i\to j}| = nt^{(i)}_{[u]}\left(\eta^{(i)}_1 + \eta^{(i)}_3\right) p_{i\to\bar{j}k}, \tag{B.8}$$

$$|\tilde{\mathbf{W}}^{(j)}_{i\to k}| = nt^{(i)}_{[u]}\left(\eta^{(i)}_2 + \eta^{(i)}_3\right) p_{i\to j\bar{k}}. \tag{B.9}$$

Description of The Feedback Stage: Thus far, the above re-grouping of the packets can be made only when one has the full knowledge of the reception status. However, right after Stage 1, no node has received any feedback yet and it is thus impossible to perform the packet regrouping as described previously. After Stage 1, we thus perform the following feedback stage so that after the feedback stage, all nodes can share a synchronized view about which packets are in which groups. Again, we emphasize that the goal of the feedback stage is to convey the reception status ACK/NACK. We never send any actual coded/uncoded messages (the payload) during the feedback stage.

Specifically, during Stage 1, each node i has been on the listening side for a total duration of $n\left(t^{(j)}_{[u]} + t^{(k)}_{[u]}\right)$ number of time slots. As a result, each node i can record whether it received a packet or not during those time slots and generate a single file of $n\left(t^{(j)}_{[u]} + t^{(k)}_{[u]}\right)$ bits. Then node i would like to deliver this file to both nodes j and k. It can be achieved by the following two-step approach. Step 1: Node i converts the file into $\left\lceil \frac{n\left(t^{(j)}_{[\mathrm{u}]}+t^{(k)}_{[\mathrm{u}]}\right)}{\log_2 q} \right\rceil$ number of packets. Then it uses an MDS code or a rate-less code to *broadcast* those packets for totally

$$\frac{n\left(t^{(j)}_{[\mathrm{u}]}+t^{(k)}_{[\mathrm{u}]}\right)}{\log_2(q)\cdot \mathsf{nzmin}\{p_{i\to j}, p_{i\to k}\}} \tag{B.10}$$

number of time slots. As a result, if $\min(p_{i\to j}, p_{i\to k}) > 0$, then $\mathsf{nzmin}\{p_{i\to j}, p_{i\to k}\} = \min(p_{i\to j}, p_{i\to k})$ and both nodes j and k can recover the file. The feedback transmission for node i is thus complete.

However, it is possible that $p_{i\to j} = 0$ (or $p_{i\to k} = 0$). In this case, $\mathsf{nzmin}\{p_{i\to j}, p_{i\to k}\} = p_{i\to k}$ and only node k can recover the file of node i. In this case, we let node k help relay the file to node j, which will take additionally

$$\frac{n\left(t^{(j)}_{[\mathrm{u}]}+t^{(k)}_{[\mathrm{u}]}\right)}{\log_2(q)\cdot p_{k\to j}} \tag{B.11}$$

number[1] of time slots. The feedback stage of node i finishes after node k helps relay the file of node i. Note that the number of time slots used for node i during the feedback stage can be upper bounded by

$$\frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{i\to j}, p_{i\to k}\}} + \frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{j\to k}, p_{j\to i}\}} + \frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{k\to i}, p_{k\to j}\}},$$

where the first term $\frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{i\to j}, p_{i\to k}\}}$ upper bounds (B.10) since $t^{(j)}_{[\mathrm{u}]}+t^{(k)}_{[\mathrm{u}]} \leq 1$; and the summation $\frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{j\to k}, p_{j\to i}\}} + \frac{n}{\log_2(q)\cdot \mathsf{nzmin}\{p_{k\to i}, p_{k\to j}\}}$ upper bounds (B.11) regardless whether $p_{i\to j} = 0$ or $p_{i\to k} = 0$.

Since the feedback stage has to be executed for all three nodes, the total number of time slots of the feedback stage is upper bounded by nt_{FB}, as defined in (3.5).

After the feedback stage, every node will know the reception status of all other nodes during Stage 1. All three nodes can thus share a synchronized view of the

[1]If $p_{i\to j} = 0$, then by our fully-connectedness assumption, $p_{k\to j} > 0$.

packet reception status and the packet regrouping, as discussed in (B.6) and (B.7). In particular, each node i exactly knows

- The contents and size of the RLNC packet groups $(\tilde{\mathbf{W}}^{(k)}_{i\to j}, \tilde{\mathbf{W}}^{(j)}_{i\to k})$. The content of the packets in each group is known since those are the messages originated from node i.
- The contents and size of the RLNC packet groups $(\tilde{\mathbf{W}}^{(i)}_{j\to k}, \tilde{\mathbf{W}}^{(i)}_{k\to j})$. The content of the packets in each group is known since those are the packets overheard by node i.
- The sizes of $|\tilde{\mathbf{W}}^{(k)}_{j\to i}|$ and $|\tilde{\mathbf{W}}^{(j)}_{k\to i}|$, which are obtained by the feedback it has received from nodes j and k.
- The content of all packets in $(\{\mathbf{c}_t\mathbf{W}_{j\to i}\}_i, \{\mathbf{c}_t\mathbf{W}_{j\to ki}\}_i, \{\mathbf{c}_t\mathbf{W}_{k\to i}\}_i, \{\mathbf{c}_t\mathbf{W}_{k\to ij}\}_i)$ are known by node i since it has received those packets during Stage 1. Note that these are the packets that have already been delivered to their target destination, which is node i. In comparison, the $(\tilde{\mathbf{W}}^{(i)}_{j\to k}, \tilde{\mathbf{W}}^{(i)}_{k\to j})$ in the second bullet are those packets destined for either node j or k but is *overheard* by i.
- The random coding vectors $\{\mathbf{c}_t\}$ for all RLNC packets sent during Stage 1. This is due to that all three nodes compute the coding vectors based on a common random seed.

Description of Stage 2: We describe the LNC operations of node i only and the operations for other nodes follow symmetrically. Similar to Stage 2 of Proposition 3.2.2, each node i will perform 4 different types of LNC operations and each operation will last for $nt^{(i)}_{[\mathrm{c},1]}$ to $nt^{(i)}_{[\mathrm{c},4]}$, respectively. For each time slot of the first coding operations (out of totally $nt^{(i)}_{[\mathrm{c},1]}$ time slots), we let node i choose two coding vectors $\mathbf{c}_{t;j}$ and $\mathbf{c}_{t;k}$ independently and uniformly randomly, where $\mathbf{c}_{t;j}$ is a $1\times|\tilde{\mathbf{W}}^{(k)}_{i\to j}|$ random row vector and $\mathbf{c}_{t;k}$ is a $1\times|\tilde{\mathbf{W}}^{(j)}_{i\to k}|$ random row vector. Then we let node i send a linear combination

$$[\mathrm{c},1]:\quad X_i(t) = \tilde{\mathbf{W}}^{(k)}_{i\to j}\mathbf{c}^{\top}_{t;j} + \tilde{\mathbf{W}}^{(j)}_{i\to k}\mathbf{c}^{\top}_{t;k}. \tag{B.12}$$

For the next time slot, another pair of $\mathbf{c}_{t;j}$ and $\mathbf{c}_{t;k}$ coding vectors are randomly chosen and used to encode $X_i(t)$ according to (B.12). Repeat the above operations until the time-budget $nt^{(i)}_{[\mathrm{c},1]}$ is used up. Then we move on and encode the next coding type $[\mathrm{c}, l], l = 2, 3, 4$

$$[\mathrm{c},2]: \quad X_i(t) = \tilde{\mathbf{W}}^{(i)}_{k\to j}\mathbf{c}^\top_{t;j} + \tilde{\mathbf{W}}^{(i)}_{j\to k}\mathbf{c}^\top_{t;k}, \tag{B.13}$$

$$[\mathrm{c},3]: \quad X_i(t) = \tilde{\mathbf{W}}^{(k)}_{i\to j}\mathbf{c}^\top_{t;j} + \tilde{\mathbf{W}}^{(i)}_{j\to k}\mathbf{c}^\top_{t;k}, \tag{B.14}$$

$$[\mathrm{c},4]: \quad X_i(t) = \tilde{\mathbf{W}}^{(i)}_{k\to j}\mathbf{c}^\top_{t;j} + \tilde{\mathbf{W}}^{(j)}_{i\to k}\mathbf{c}^\top_{t;k}. \tag{B.15}$$

Each coding type $[\mathrm{c}, l]$ will last for $nt^{(i)}_{[\mathrm{c},l]}$ time slots and the coding vectors $\mathbf{c}_{t;j}$ and $\mathbf{c}_{t;k}$ are chosen independently and uniformly randomly with the properly selected dimension. For example of the coding choice $[\mathrm{c}, 3]$, the randomly chosen $\mathbf{c}_{t;j}$ is a $1 \times |\tilde{\mathbf{W}}^{(k)}_{i\to j}|$ row vector and the randomly chosen $\mathbf{c}_{t;k}$ is a $1 \times |\tilde{\mathbf{W}}^{(i)}_{j\to k}|$ row vector.

Stage 2 is completed after all three nodes have finished sending their corresponding 4 coding types. The description of the proposed scheme is complete. (There is no need to have the second feedback stage.)

Analysis of the scheme: The total amount of time to finish the transmission is upper bounded by

$$n\left(\left(\sum_{\forall i\in\{1,2,3\}} t^{(i)}_{[\mathrm{u}]}\right) + t_{\mathsf{FB}} + \left(\sum_{\forall i\in\{1,2,3\}} t^{(i)}_{[\mathrm{c},1]} + t^{(i)}_{[\mathrm{c},2]} + t^{(i)}_{[\mathrm{c},3]} + t^{(i)}_{[\mathrm{c},4]}\right)\right).$$

By (3.4), we can thus finish all the transmissions within the total time budget of n time slots.

We now argue that after finishing transmission, all nodes can decode their desired packets. To that end, we focus only on node $\mathsf{1}$. The discussions of nodes $\mathsf{2}$ and $\mathsf{3}$ can be made by symmetry.

During Stage 2, consider the transmission of node $\mathsf{3}$. Node $\mathsf{3}$ has 4 possible coding choices. In each coding choices, it randomly mixes from two groups of packets. For example, in coding choice $[\mathrm{c}, 1]$, node $\mathsf{3}$ mixes $\tilde{\mathbf{W}}^{(\mathsf{2})}_{\mathsf{3}\to\mathsf{1}}$ and $\tilde{\mathbf{W}}^{(\mathsf{1})}_{\mathsf{3}\to\mathsf{2}}$, see (B.12) when

$(i, j, k) = (3, 1, 2)$. Since the content of any packets in $\tilde{\mathbf{W}}^{(1)}_{3\to 2}$ is known to node 1, see the discussion in the end of the feedback stage, node 1, upon the reception of any [c, 1] packet transmitted by node 3, can subtract the term $\tilde{\mathbf{W}}^{(1)}_{3\to 2}\mathbf{c}^{\top}_{t;2}$ from the received packet. Therefore, it is as if node 1 has received a packet of the form

$$\tilde{\mathbf{W}}^{(2)}_{3\to 1}\mathbf{c}^{\top}_{t;1} \tag{B.16}$$

without the corruption term $\tilde{\mathbf{W}}^{(1)}_{3\to 2}\mathbf{c}^{\top}_{t;2}$. Similarly, when node 3 performs coding choice [c, 3], again, node 1 will receive coded packets of the form (B.16) after subtracting those $\tilde{\mathbf{W}}^{(3)}_{1\to 2}\mathbf{c}^{\top}_{t;2}$ packets of its own, see (B.14) when $(i, j, k) = (3, 1, 2)$. Also, during node 2 performing coding choices [c, 2] and [c, 3], node 1 can again receive coded packets of the form (B.16) after subtracting those known packets (either of the form $\tilde{\mathbf{W}}^{(2)}_{1\to 3}\mathbf{c}^{\top}_{t;3}$ or of the form $\tilde{\mathbf{W}}^{(1)}_{2\to 3}\mathbf{c}^{\top}_{t;3}$), see (B.13) and (B.14) when $(i, j, k) = (2, 3, 1)$.

Since $\tilde{\mathbf{W}}^{(2)}_{3\to 1}$ participates in coding choices [c, 1] and [c, 3] of node 3 and coding choices [c, 2] and [c, 3] of node 2, node 1 will receive $n\left(t^{(3)}_{[\mathrm{c},1]} + t^{(3)}_{[\mathrm{c},3]}\right) \cdot p_{3\to 1} + n\left(t^{(2)}_{[\mathrm{c},2]} + t^{(2)}_{[\mathrm{c},3]}\right) \cdot p_{2\to 1}$ number of packets of the form (B.16). Note that the number of $\tilde{\mathbf{W}}^{(2)}_{3\to 1}$ packets, in average, has been computed in (B.8). By (B.4), the number of linear combinations (B.16) received by node 1 is larger than the number of $\tilde{\mathbf{W}}^{(2)}_{3\to 1}$ packets to be begin with. As a result, node 1 is guaranteed to decode $\tilde{\mathbf{W}}^{(2)}_{3\to 1}$ correctly with close-to-one probability when the finite-field size q is sufficiently large enough.

Recall that by definition (B.6), $\tilde{\mathbf{W}}^{(2)}_{3\to 1} = \{\mathbf{c}_t\mathbf{W}_{3\to 1}\}_{\overline{1}2} \cup \{\mathbf{c}_t\mathbf{W}_{3\to 12}\}_{\overline{1}2}$. We now observe that node 1 has also received all the RLNC packets of $(\{\mathbf{c}_t\mathbf{W}_{3\to 1}\}_1, \{\mathbf{c}_t\mathbf{W}_{3\to 12}\}_1)$ during Stage 1. As a result, in the end of Stage 2, node 1 has correctly received $nt^{(3)}_{[\mathrm{u}]}\eta^{(3)}_1 p_{3\to 1\vee 2}$ number of packets of the form $\{\mathbf{c}_t\mathbf{W}_{3\to 1}\}$ and $nt^{(3)}_{[\mathrm{u}]}\eta^{(3)}_3 p_{3\to 1\vee 2}$ number of packets of the form $\{\mathbf{c}_t\mathbf{W}_{3\to 12}\}$. Note that we only have $nR_{3\to 1}$ of $\mathbf{W}_{3\to 1}$ packets and $nR_{3\to 12}$ of $\mathbf{W}_{3\to 12}$ packets to begin with. Since by definition $t^{(3)}_{[\mathrm{u}]}$ is strictly larger than $\frac{R_{3\to 1}+R_{3\to 2}+R_{3\to 12}}{p_{3\to 1\vee 2}}$, and also by definitions (B.1) and (B.3), the number of linear combinations received by node 1 is larger than the number of uncoded message symbols $\mathbf{W}_{3\to 1}$ and $\mathbf{W}_{3\to 12}$. As a result, node 1 is guaranteed to decode $\mathbf{W}_{3\to 1}$ and

$\mathbf{W}_{3\to12}$ correctly with close-to-one probability when the finite field size q is sufficiently large enough.

By symmetric arguments, with close-to-one probability node 1 can also decode $\tilde{\mathbf{W}}^{(3)}_{2\to1}$ in the end of Stage 2 and later combines $\tilde{\mathbf{W}}^{(3)}_{2\to1}$ with the packets ($\{\mathbf{c}_t\mathbf{W}_{2\to1}\}_1$, $\{\mathbf{c}_t\mathbf{W}_{2\to31}\}_1$) it has received in Stage 1 to decode message symbols $\mathbf{W}_{2\to1}$ and $\mathbf{W}_{2\to31}$. Symmetric arguments can be used to shown that nodes 2 and 3 can also decode their desired messages. The proof of Proposition 3.2.1 is thus complete.

Remark: The arguments of letting the finite field size approach infinity is to ensure that the simple RLNC construction leads to legitimate MDS codes. When the finite field size is fixed to, say $q = 2$, we can use the fact that for any fixed $\mathbb{F}_q$ we can always construct an (n, k) code that is *nearly MDS* in the sense that as long as we receive $k + O(\sqrt{k})$ number of encoded packets we can reconstruct the original file. Since we focus only on the normalized throughput, such a near-MDS code is sufficient for our achievability construction.

C. THE MATCH PROOF OF PROPOSITION 3.2.3.

Without loss of generality, we assume that $p_{i\to j} > 0$ and $p_{i\to k} > 0$ for all $(i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}$ since the case that any one of them is zero can be viewed as a limiting scenario and the polytope of the capacity outer bound in Proposition 3.1.1 is continuous with respect to the channel success probability parameters.

We first introduce the following Lemma.

Lemma C.0.4. *Given any $\vec{R}$ and the associated 3 non-negative values $\{s^{(i)}\}$ that satisfy Proposition 3.1.1, we can always find 15 non-negative values $t^{(i)}_{[u]}$ and $\{t^{(i)}_{[c,l]}\}_{l=1}^{4}$ for all $i \in \{1,2,3\}$ such that jointly they satisfy the groups of linear conditions in Proposition 3.2.2 (when replacing all strict inequality $<$ by $\leq$).*

One can clearly see that Lemma C.0.4 imply that the capacity outer bound in Proposition 3.1.1 matches the closure of the inner bound in Proposition 3.2.2. The proof of Proposition 3.2.3 is thus complete.

The proof of Lemma C.0.4: Given $\vec{R}$ and the reception probabilities, consider 3 non-negative values $\{s^{(i)}\}$ that jointly satisfy the linear conditions of Proposition 3.1.1.

We first choose $t^{(i)}_{[\mathrm{u}]} \triangleq \frac{R_{i\to j}+R_{i\to k}+R_{i\to jk}}{p_{i\to j\vee k}}$ which is non-negative by definition. Then define $\tilde{s}^{(i)} \triangleq s^{(i)} - t^{(i)}_{[\mathrm{u}]}$ for all $i \in \{1,2,3\}$. By (3.2) in Proposition 3.1.1, the newly constructed values $\{\tilde{s}^{(i)}\}$ must be non-negative. Then, we can rewrite (3.3) in Proposition 3.1.1 as follows: For all $(i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}$, we have

$$\Big(R_{j\to i} + R_{j\to ki}\Big)\frac{p_{j\to k\bar{i}}}{p_{j\to k\vee i}} + \Big(R_{k\to i} + R_{k\to ij}\Big)\frac{p_{k\to \bar{i}j}}{p_{k\to i\vee j}} \leq \tilde{s}^{(j)} \cdot p_{j\to i} + \tilde{s}^{(k)} \cdot p_{k\to i}.$$

For each tuple (i, j, k), define a constant α_{ijk} as follows:

$$\alpha_{ijk} = \frac{\Big(R_{j\to i} + R_{j\to ki}\Big)\frac{p_{j\to k\bar{i}}}{p_{j\to k\vee i}}}{\Big(R_{j\to i} + R_{j\to ki}\Big)\frac{p_{j\to k\bar{i}}}{p_{j\to k\vee i}} + \Big(R_{k\to i} + R_{k\to ij}\Big)\frac{p_{k\to \bar{i}j}}{p_{k\to i\vee j}}}.$$

For each tuple (i, j, k), we will use α_{ijk}, $\tilde{s}^{(j)}$ and $\tilde{s}^{(k)}$ to define/compute 4 more variables.

$$\begin{aligned}
\tilde{s}^{(j)}_{ijk,+} &= \alpha_{ijk} \cdot \tilde{s}^{(j)}, \\
\tilde{s}^{(k)}_{ijk,+} &= \alpha_{ijk} \cdot \tilde{s}^{(k)}, \\
\tilde{s}^{(j)}_{ijk,-} &= (1 - \alpha_{ijk}) \cdot \tilde{s}^{(j)}, \\
\tilde{s}^{(k)}_{ijk,-} &= (1 - \alpha_{ijk}) \cdot \tilde{s}^{(k)}.
\end{aligned}$$

By the above construction, we quickly have

$$\tilde{s}^{(j)}_{ijk,+} + \tilde{s}^{(j)}_{ijk,-} = \tilde{s}^{(j)}, \tag{C.1}$$

$$\tilde{s}^{(k)}_{ijk,+} + \tilde{s}^{(k)}_{ijk,-} = \tilde{s}^{(k)}, \tag{C.2}$$

and

$$\Big(R_{j\to i} + R_{j\to ki}\Big)\frac{p_{j\to k\bar{i}}}{p_{j\to k\vee i}} \leq \tilde{s}^{(j)}_{ijk,+} \cdot p_{j\to i} + \tilde{s}^{(k)}_{ijk,+} \cdot p_{k\to i}, \tag{C.3}$$

$$\Big(R_{k\to i} + R_{k\to ij}\Big)\frac{p_{k\to \bar{i}j}}{p_{k\to i\vee j}} \leq \tilde{s}^{(j)}_{ijk,-} \cdot p_{j\to i} + \tilde{s}^{(k)}_{ijk,-} \cdot p_{k\to i}, \tag{C.4}$$

for every cyclically shifted (i, j, k) tuple. Totally, we have 3 variables of the form $\tilde{s}^{(i)}$ and 12 variables of the forms $\tilde{s}^{(j)}_{ijk,+}$, $\tilde{s}^{(j)}_{ijk,-}$, $\tilde{s}^{(k)}_{ijk,+}$, and $\tilde{s}^{(k)}_{ijk,-}$. Since each $\tilde{s}^{(i)}$ may participate in more than one "splitting operations (C.1) and (C.2)", we thus have that for all $(i, j, k) \in \{(1, 2, 3), (2, 3, 1), (3, 1, 2)\}$,

$$\tilde{s}^{(i)}_{jki,+} + \tilde{s}^{(i)}_{jki,-} = \tilde{s}^{(i)}_{kij,+} + \tilde{s}^{(i)}_{kij,-} = \tilde{s}^{(i)}. \tag{C.5}$$

The following claim allows us to convert the $\tilde{s}^{(i)}_{jki,+}$, $\tilde{s}^{(i)}_{jki,-}$, $\tilde{s}^{(i)}_{kij,+}$, and $\tilde{s}^{(i)}_{kij,-}$ values to the targeted $t^{(i)}_{[\mathrm{c},1]}$ to $t^{(i)}_{[\mathrm{c},4]}$ values.

Claim C.0.1. *For any cyclically shifted (i,j,k) tuple, given the above four values of $\{\tilde{s}^{(i)}_{jki,+}, \tilde{s}^{(i)}_{jki,-}, \tilde{s}^{(i)}_{kij,+}, \tilde{s}^{(i)}_{kij,-}\}$ and the value of $\tilde{s}^{(i)}$, we can always find another four non-negative values $t^{(i)}_{[c,1]}$, $t^{(i)}_{[c,2]}$, $t^{(i)}_{[c,3]}$, and $t^{(i)}_{[c,4]}$ such that*

$$t^{(i)}_{[c,2]} + t^{(i)}_{[c,4]} = \tilde{s}^{(i)}_{jki,+}, \tag{C.6}$$

$$t^{(i)}_{[c,1]} + t^{(i)}_{[c,3]} = \tilde{s}^{(i)}_{jki,-}, \tag{C.7}$$

$$t^{(i)}_{[c,1]} + t^{(i)}_{[c,4]} = \tilde{s}^{(i)}_{kij,+}, \tag{C.8}$$

$$t^{(i)}_{[c,2]} + t^{(i)}_{[c,3]} = \tilde{s}^{(i)}_{kij,-}, \tag{C.9}$$

$$\text{and } t^{(i)}_{[c,1]} + t^{(i)}_{[c,2]} + t^{(i)}_{[c,3]} + t^{(i)}_{[c,4]} = \tilde{s}^{(i)}. \tag{C.10}$$

Proof of Claim C.0.1: Since the given values $\{\tilde{s}^{(i)}_{jki,+}, \tilde{s}^{(i)}_{jki,-}, \tilde{s}^{(i)}_{kij,+}, \tilde{s}^{(i)}_{kij,-}\}$ satisfy (C.5), consider the following two cases depending on the order of the two values $\tilde{s}^{(i)}_{jki,-}$ and $\tilde{s}^{(i)}_{kij,+}$.

Case 1: $\tilde{s}^{(i)}_{jki,-} \geq \tilde{s}^{(i)}_{kij,+}$. We then construct four values $t^{(i)}_{[\mathrm{c},1]}$, $t^{(i)}_{[\mathrm{c},2]}$, $t^{(i)}_{[\mathrm{c},3]}$, and $t^{(i)}_{[\mathrm{c},4]}$ in the following way:

$$\begin{aligned} t^{(i)}_{[\mathrm{c},1]} &= \tilde{s}^{(i)}_{kij,+}, \\ t^{(i)}_{[\mathrm{c},2]} &= \tilde{s}^{(i)}_{jki,+}, \\ t^{(i)}_{[\mathrm{c},3]} &= \tilde{s}^{(i)}_{kij,-} - \tilde{s}^{(i)}_{jki,+}, \\ t^{(i)}_{[\mathrm{c},4]} &= 0. \end{aligned}$$

The above construction clearly gives non-negative $t^{(i)}_{[\mathrm{c},1]}$ to $t^{(i)}_{[\mathrm{c},4]}$ values. One can easily verify that the above construction satisfies all the equalities (C.6) to (C.10). For example, by our construction $t^{(i)}_{[\mathrm{c},2]} + t^{(i)}_{[\mathrm{c},3]} = \tilde{s}^{(i)}_{jki,+} + \tilde{s}^{(i)}_{kij,-} - \tilde{s}^{(i)}_{jki,+} = \tilde{s}^{(i)}_{kij,-}$, which satisfies (C.9).

Case 2: $\tilde{s}^{(i)}_{jki,-} < \tilde{s}^{(i)}_{kij,+}$. We then construct four non-negative values $t^{(i)}_{[c,1]}$, $t^{(i)}_{[c,2]}$, $t^{(i)}_{[c,3]}$, and $t^{(i)}_{[c,4]}$ in the following way:

$$\begin{aligned} t^{(i)}_{[c,1]} &= \tilde{s}^{(i)}_{jki,-}, \\ t^{(i)}_{[c,2]} &= \tilde{s}^{(i)}_{kij,-}, \\ t^{(i)}_{[c,3]} &= 0, \\ t^{(i)}_{[c,4]} &= \tilde{s}^{(i)}_{jki,+} - \tilde{s}^{(i)}_{kij,-}. \end{aligned}$$

Again, the above construction leads to non-negative $t^{(i)}_{[c,1]}$ to $t^{(i)}_{[c,4]}$ values that satisfy (C.6) to (C.10). Since the above two cases cover all possible scenarios, the claim is thus proven. □

Using the above claim, we now prove that the constructed values $\{t^{(i)}_{[c,1]}, t^{(i)}_{[c,2]}, t^{(i)}_{[c,3]}, t^{(i)}_{[c,4]}\}$ for all $i \in \{1,2,3\}$ together with the previously chosen $t^{(i)}_{[u]} \triangleq \frac{R_{i\to j}+R_{i\to k}+R_{i\to jk}}{p_{i\to j\vee k}}$ satisfy the linear conditions of Proposition 3.2.2 (when $<$ being replaced by $\leq$).

To that end, we first notice that

$$\begin{aligned} t^{(i)}_{[u]} + t^{(i)}_{[c,1]} + t^{(i)}_{[c,2]} + t^{(i)}_{[c,3]} + t^{(i)}_{[c,4]} &= \frac{R_{i\to j} + R_{i\to k} + R_{i\to jk}}{p_{i\to j\vee k}} + \tilde{s}^{(i)} \\ &= s^{(i)}, \end{aligned}$$

where the first equality follows from the definition of $t^{(i)}_{[u]}$ and (C.10); and the second equality follows from the definition of $\tilde{s}^{(i)}$. Since the given values $s^{(i)}$ for all $i \in \{1,2,3\}$ satisfy the time-sharing condition (3.1) of Proposition 3.1.1, the time-sharing condition (3.4) of Proposition 3.2.2 must hold as well.

Moreover, the second condition (3.6) of Proposition 3.2.2 obviously holds by the definition of $t^{(i)}_{[u]}$. In the following, we prove (3.7) and (3.8) for the case when $(i, j, k) =$

$(1,2,3)$ and other cases can be proven symmetrically. In other words, we will prove the following equalities:

$$\left(R_{1\to 2} + R_{1\to 23}\right)\frac{p_{1\to\overline{2}3}}{p_{1\to 2\vee 3}} \leq \left(t^{(1)}_{[c,1]} + t^{(1)}_{[c,3]}\right)\cdot p_{1\to 2} + \left(t^{(3)}_{[c,2]} + t^{(3)}_{[c,3]}\right)\cdot p_{3\to 2}, \tag{C.11}$$

$$\left(R_{1\to 3} + R_{1\to 23}\right)\frac{p_{1\to 2\overline{3}}}{p_{1\to 2\vee 3}} \leq \left(t^{(1)}_{[c,1]} + t^{(1)}_{[c,4]}\right)\cdot p_{1\to 3} + \left(t^{(2)}_{[c,2]} + t^{(2)}_{[c,4]}\right)\cdot p_{2\to 3}. \tag{C.12}$$

By (C.7) and (C.9), we have

$$\begin{aligned}\left(t^{(1)}_{[c,1]} + t^{(1)}_{[c,3]}\right)\cdot p_{1\to 2} + \left(t^{(3)}_{[c,2]} + t^{(3)}_{[c,3]}\right)\cdot p_{3\to 2}& \\ = \tilde{s}^{(1)}_{231,-}\cdot p_{1\to 2} + \tilde{s}^{(3)}_{231,-}\cdot p_{3\to 2}&.\end{aligned}$$

As a result, by (C.4) with the (i,j,k) substituted by $(2,3,1)$, we have proven (C.11). Similarly, by (C.8) and (C.6), we have

$$\begin{aligned}\left(t^{(1)}_{[c,1]} + t^{(1)}_{[c,4]}\right)\cdot p_{1\to 3} + \left(t^{(2)}_{[c,2]} + t^{(2)}_{[c,4]}\right)\cdot p_{2\to 3}& \\ = \tilde{s}^{(1)}_{312,+}\cdot p_{1\to 3} + \tilde{s}^{(2)}_{312,+}\cdot p_{2\to 3}&.\end{aligned}$$

As a result, by (C.3) with (i,j,k) substituted by $(3,1,2)$, we have proven (C.12).

In sum, from the given values $\{s^{(i)}\}$ for all $i\in\{1,2,3\}$ satisfying the linear conditions of Proposition 3.1.1, we have constructed 15 non-negative values $\{t^{(i)}_{[u]}, t^{(i)}_{[c,1]}, t^{(i)}_{[c,2]}, t^{(i)}_{[c,3]}, t^{(i)}_{[c,4]}\}$ for all $i\in\{1,2,3\}$ such that they jointly satisfy the linear inequalities of Proposition 3.2.2. The proof of Lemma C.0.4 is thus complete. ■

D. DETAILED PROOFS OF THE SHANNON OUTER BOUND

We provide the proofs of (3.2A) and (3.2B) for the broadcasting cut-set condition (3.2) in Proposition 3.1.1.

Firstly, (3.2A) can be derived as follows:

$$
\begin{aligned}
&I(\mathbf{W}_{i*}\,;\,[\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^n\,|\,\mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^n) \\
&= I(\mathbf{W}_{i*}\,;\,\mathbf{W}_{\{j,k\}*},[\mathbf{Y}_{*j},\mathbf{Y}_{*k},\mathbf{Z}]_1^n) && \text{(D.1)} \\
&\geq n(R_{i\to j}+R_{i\to k}+R_{i\to jk})(1-2\epsilon)-\frac{H_2(2\epsilon)}{\log_2 q}, && \text{(D.2)}
\end{aligned}
$$

where (D.1) follows from the definition of mutual information and the fact that $\mathbf{W}_{i*}$, $\mathbf{W}_{\{j,k\}*}$, and $[\mathbf{Z}]_1^n$ are independent of each other. To derive (D.2), we observe that the messages $\mathbf{W}_{i*}$ can be decoded from $[\mathbf{Y}_{*j},\mathbf{Y}_{*k},\mathbf{Z}]_1^n$ and $\mathbf{W}_{\{j,k\}*}\triangleq\mathbf{W}_{j*}\cup\mathbf{W}_{k*}$, see (2.8) for nodes j and k, with error probability being at most 2ϵ by the union bound. As a result, by Fano's inequality, we have (D.2).

Secondly, (3.2B) can be derived as follows:

$$
\begin{aligned}
& I(\mathbf{W}_{i*}\,;\,[\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^n \,|\, \mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^n) \\
& \leq H([\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^n \,|\, \mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^n) && \text{(D.3)} \\
& = \sum_{t=1}^{n} H(\mathbf{Y}_{*j}(t),\mathbf{Y}_{*k}(t) \,|\, [\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^{t-1},\mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^t) && \text{(D.4)} \\
& = \sum_{t=1}^{n} H(\mathbf{Y}_{*j}(t),\mathbf{Y}_{*k}(t) \,|\, [\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^{t-1},\mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^t, X_j(t), X_k(t)) && \text{(D.5)} \\
& = \sum_{t=1}^{n} H(Y_{i\to j}(t),Y_{i\to k}(t) \,|\, [\mathbf{Y}_{*j},\mathbf{Y}_{*k}]_1^{t-1},\mathbf{W}_{\{j,k\}*},[\mathbf{Z}]_1^t, X_j(t), X_k(t)) && \text{(D.6)} \\
& \leq \sum_{t=1}^{n} \mathbb{E}\Big\{1_{\{\sigma(t)=i\}} \circ 1_{\{Z_{i\to j}(t)=1 \text{ or } Z_{i\to k}(t)=1\}}\Big\} && \text{(D.7)} \\
& = \sum_{t=1}^{n} \mathbb{E}\Big\{1_{\{\sigma(t)=i\}}\Big\}\mathbb{E}\Big\{1_{\{Z_{i\to j}(t)=1 \text{ or } Z_{i\to k}(t)=1\}}\Big\} && \text{(D.8)} \\
& = p_{i\to j\vee k}\mathbb{E}\left\{\sum_{t=1}^{n} 1_{\{\sigma(t)=i\}}\right\} = n s^{(i)} p_{i\to j\vee k}, && \text{(D.9)}
\end{aligned}
$$

where (D.3) follows from the definition of mutual information; (D.4) follows from the chain rule and from the fact that the future channel outputs $[\mathbf{Z}]_{t+1}^n$ are independent of $\mathbf{Y}_{*j}(t), \mathbf{Y}_{*k}(t)$; (D.5) follows from the fact that the transmitted symbol $X_j(t)$ (resp. $X_k(t)$) is a function of the past received symbols $[\mathbf{Y}_{*j}]_1^{t-1}$ (resp. $[\mathbf{Y}_{*k}]_1^{t-1}$), the information messages $\mathbf{W}_{j*}$ (resp. $\mathbf{W}_{k*}$), and the past channel outputs $[\mathbf{Z}]_1^{t-1}$, see (2.7); (D.6) follows from the fact that the received symbol $Y_{k\to j}(t)$ in $\mathbf{Y}_{*j}(t)$ (resp. $Y_{j\to k}(t)$ in $\mathbf{Y}_{*k}(t)$) can be uniquely computed from the values of the current input $X_k(t)$ (resp. $X_j(t)$), the current channel output $\mathbf{Z}(t)$, and the current scheduling decision $\sigma(t)$, which depends only on the past channel outputs $[\mathbf{Z}]_1^{t-1}$, see (2.9); (D.7) follows from that only when $\sigma(t) = i$ with $Z_{i\to j}(t) = 1$ or $Z_{i\to k}(t) = 1$, we will have a non-zero value of the entropy and it is upper bounded by 1 since the base of the logarithm is q; (D.8) follows from the fact that since the scheduling decision $\sigma(t)$ depends only on the past channel outputs $[\mathbf{Z}]_1^{t-1}$, see (2.9), the random variables $\sigma(t)$ and $\mathbf{Z}(t)$ are independent; and (D.9) follows from the definition (3.9).

We provide the proofs of (3.3A) and (3.3B) for the 3-way multiple-access cut-set condition (3.3) in Proposition 3.1.1.

The inequality (3.3B) can be derived in a similar way as (3.2B). Specifically, we have

$$I(\mathbf{W}_{\{j,k\}*}\,;\,[\mathbf{Y}_{*i}]_1^n\,|\,\mathbf{W}_{i*},[\mathbf{Z}]_1^n) \leq H([\mathbf{Y}_{*i}]_1^n\,|\,\mathbf{W}_{i*},[\mathbf{Z}]_1^n) \tag{D.10}$$

$$= \sum_{t=1}^{n} H(Y_{j\to i}(t), Y_{k\to i}(t)\,|\,[\mathbf{Y}_{*i}]_1^{t-1}, \mathbf{W}_{i*}, [\mathbf{Z}]_1^t), \tag{D.11}$$

$$\leq n(s^{(j)}p_{j\to i} + s^{(k)}p_{k\to i}), \tag{D.12}$$

where (D.10) follows from the definition of mutual information; (D.11) follows from the chain rule and the fact that the future channel outputs $[\mathbf{Z}]_{t+1}^n$ are independent of $Y_{j\to i}(t), Y_{k\to i}(t)$; and (D.12) follows from similar arguments as used in (D.7) to (D.9).

We now prove (3.3A). For the ease of exposition, we only prove for the case when the node indices are fixed to $(i,j,k) = (1,2,3)$. Then (3.3A) becomes

$$I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}]_1^n\,|\,\mathbf{W}_{1*},[\mathbf{Z}]_1^n)$$
$$\geq n\left(R_{2\to1} + R_{3\to1} + R_{2\to31} + R_{3\to12} + \frac{p_{2\to1}}{p_{2\to3\vee1}}R_{2\to3} + \frac{p_{3\to1}}{p_{3\to1\vee2}}R_{3\to2} - 6\epsilon - \frac{3H_2(\epsilon)}{n\log_2 q}\right).$$

The cases of other node indices $(i,j,k) \in \{(2,3,1),\,(3,1,2)\}$ can be proven by symmetry.

Consider the following lemmas and claims, of which their proofs are relegated to Appendix F.

Lemma D.0.5. *Consider Scenario 1 and any fixed $t \in \{1,\cdots,n\}$. Then, knowing all the messages $\mathbf{W}_{\{1,2,3\}*}$ and the past channel outputs $[\mathbf{Z}]_1^{t-1}$ can uniquely decide $[X_1,X_2,X_3]_1^t$ and $[\mathbf{Y}_{1*},\mathbf{Y}_{2*},\mathbf{Y}_{3*}]_1^{t-1}$. Namely, $[X_1,X_2,X_3]_1^t$ and $[\mathbf{Y}_{1*},\mathbf{Y}_{2*},\mathbf{Y}_{3*}]_1^{t-1}$ are functions of the random variables $\{\mathbf{W}_{\{1,2,3\}*},[\mathbf{Z}]_1^{t-1}\}$ for any time $t \in \{1,\cdots,n\}$.*

Lemma D.0.6. *Consider Scenario 1 and any fixed time slot $t \in \{1,\cdots,n\}$. Then, knowing the messages $\mathbf{W}_{\{1,3\}*}$, the received symbols $[\mathbf{Y}_{2*}]_1^{t-1}$, and the past channel* *outputs* $[\mathbf{Z}]_1^{t-1}$ *can uniquely decide* $[X_1, X_3]_1^t$. *Namely,* $[X_1, X_3]_1^t$ *is a function of the random variables* $\{\mathbf{W}_{\{1,3\}*}, [\mathbf{Y}_{2*}]_1^{t-1}, [\mathbf{Z}]_1^{t-1}\}$ *for any time* $t \in \{1, \cdots, n\}$.

Claim D.0.2. *The following is true:*

$$\begin{aligned} &I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}]_1^n \,|\, \mathbf{W}_{1*}, [\mathbf{Z}]_1^n) \\ &= \sum_{t=1}^{n} I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{1*}, \mathbf{Z}(t)). \end{aligned} \tag{D.13}$$

Claim D.0.3. *Define*

$$\mathbf{W}_{\overline{2\to3}} \triangleq \mathbf{W}_{\{1,3\}*} \cup \mathbf{W}_{2\to1} \cup \mathbf{W}_{2\to31}, \tag{D.14}$$

That is, $\mathbf{W}_{\overline{2\to3}}$ *is the collection of all the* 9*-flow information messages except* $\mathbf{W}_{2\to3}$. *This is why we use the overline in the subscript. Symmetrically, define*

$$\mathbf{W}_{\overline{3\to2}} \triangleq \mathbf{W}_{\{1,2\}*} \cup \mathbf{W}_{3\to1} \cup \mathbf{W}_{3\to12}. \tag{D.15}$$

Then, the following is true: $\forall\, t \in \{1, \cdots, n\}$,

$$\begin{aligned} &I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{1*}, \mathbf{Z}(t)) \\ &\geq I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(t) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{1*}, \mathbf{Z}(t)) \\ &\quad + \frac{p_{2\to1}}{p_{2\to3\vee1}} I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t) \,|\, [\mathbf{Y}_{2*}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{\overline{2\to3}}, \mathbf{Z}(t)) \\ &\quad + \frac{p_{3\to1}}{p_{3\to1\vee2}} I(\mathbf{W}_{3\to2}\,;\,\mathbf{Y}_{3*}(t) \,|\, [\mathbf{Y}_{3*}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{\overline{3\to2}}, \mathbf{Z}(t)). \end{aligned} \tag{D.16}$$

Claim D.0.4. *The followings are true:*

$$\sum_{t=1}^{n} I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1},\mathbf{Z}]_1^{t-1},\mathbf{W}_{1*},\mathbf{Z}(t)) = I(\mathbf{W}_{*1}\,;\,\mathbf{W}_{1*},[\mathbf{Y}_{*1},\mathbf{Z}]_1^{n}), \tag{D.17}$$

$$\sum_{t=1}^{n} I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*},\mathbf{Z}]_1^{t-1},\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)) \geq I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{3*},[\mathbf{Y}_{*3},\mathbf{Z}]_1^{n}), \tag{D.18}$$

$$\sum_{t=1}^{n} I(\mathbf{W}_{3\to2}\,;\,\mathbf{Y}_{3*}(t)\,|\,[\mathbf{Y}_{3*},\mathbf{Z}]_1^{t-1},\mathbf{W}_{\overline{3\to2}},\mathbf{Z}(t)) \geq I(\mathbf{W}_{3\to2}\,;\,\mathbf{W}_{2*},[\mathbf{Y}_{*2},\mathbf{Z}]_1^{n}). \tag{D.19}$$

By the above Claims D.0.2 to D.0.4 we have

$$\begin{aligned}
&I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}]_1^n\,|\,\mathbf{W}_{1*},[\mathbf{Z}]_1^n) \\
&\geq I(\mathbf{W}_{*1}\,;\,\mathbf{W}_{1*},[\mathbf{Y}_{*1},\mathbf{Z}]_1^n) \\
&\quad + \frac{p_{2\to1}}{p_{2\to3\vee1}} I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{3*},[\mathbf{Y}_{*3},\mathbf{Z}]_1^n) + \frac{p_{3\to1}}{p_{3\to1\vee2}} I(\mathbf{W}_{3\to2}\,;\,\mathbf{W}_{2*},[\mathbf{Y}_{*2},\mathbf{Z}]_1^n), &\text{(D.20)}\\
&\geq n(R_{2\to1}+R_{3\to1}+R_{2\to31}+R_{3\to12})(1-\epsilon) - \frac{H_2(\epsilon)}{\log_2 q} \\
&\quad + \frac{p_{2\to1}}{p_{2\to3\vee1}}\left(nR_{2\to3}(1-\epsilon) - \frac{H_2(\epsilon)}{\log_2 q}\right) + \frac{p_{3\to1}}{p_{3\to1\vee2}}\left(nR_{3\to2}(1-\epsilon) - \frac{H_2(\epsilon)}{\log_2 q}\right) &\text{(D.21)}
\end{aligned}$$

where (D.20) follows from jointly combining (D.13) to (D.19); and (D.21) follows from applying Fano's inequality to each individual term. Since we can choose ϵ arbitrarily, by letting $\epsilon \to 0$, we have proven (3.3A).

E. DETAILED DESCRIPTION OF ACHIEVABILITY SCHEMES IN FIG. 3.2.

In the following, we describe the 9-dimensional rate regions of each suboptimal achievability scheme used for the numerical evaluation in Section 3.6.5.

• **LNC with pure operations 1, 2:** The rate regions can be described by Proposition 3.2.2 with the variables $t^{(i)}_{[\mathrm{c},\,3]}$ and $t^{(i)}_{[\mathrm{c},\,4]}$ hardwired to 0 for all $i \in \{1,2,3\}$.

• **TWRC at node 1 and RX coord.:** This scheme performs two-way relay channel (TWRC) coding only at node 1 for those $3 \to 2$ and $2 \to 3$ flows while allowing node 2 to relay the node 1's packets destined for node 3 (i.e., $\mathbf{W}_{1\to 3}$ and $\mathbf{W}_{1\to 23}$) and vice versa. The corresponding rate regions can be described as follows:

$$\sum_{\forall\, i\in\{1,2,3\}} t^{(i)}_{[\mathrm{u}]} + t^{(i)}_{[\mathrm{c}]} \leq 1, \tag{E.1}$$

$$\frac{R_{1\to 2} + R_{1\to 3} + R_{1\to 23}}{p_{1\to 2\vee 3}} < t^{(1)}_{[\mathrm{u}]}, \tag{E.2}$$

$$\frac{R_{2\to 1}}{p_{2\to 1}} + \frac{R_{2\to 3}}{p_{2\to 3\vee 1}} + \frac{R_{2\to 31}}{p_{2\to 1}} + \frac{R_{2\to 31}}{p_{2\to 3}} < t^{(2)}_{[\mathrm{u}]}, \tag{E.3}$$

$$\frac{R_{3\to 1}}{p_{3\to 1}} + \frac{R_{3\to 2}}{p_{3\to 1\vee 2}} + \frac{R_{3\to 12}}{p_{3\to 1}} + \frac{R_{3\to 12}}{p_{3\to 2}} < t^{(3)}_{[\mathrm{u}]}, \tag{E.4}$$

$$R_{2\to 3}\frac{p_{2\to \overline{3}1}}{p_{2\to 3\vee 1}} < t^{(1)}_{[\mathrm{c}]} \cdot p_{1\to 3}, \tag{E.5}$$

$$R_{3\to 2}\frac{p_{3\to 1\overline{2}}}{p_{3\to 1\vee 2}} < t^{(1)}_{[\mathrm{c}]} \cdot p_{1\to 2}, \tag{E.6}$$

$$\Big(R_{1\to 3} + R_{1\to 23}\Big)\frac{p_{1\to 2\overline{3}}}{p_{1\to 2\vee 3}} < t^{(2)}_{[\mathrm{c}]} \cdot p_{2\to 3}, \tag{E.7}$$

$$\Big(R_{1\to 2} + R_{1\to 23}\Big)\frac{p_{1\to \overline{2}3}}{p_{1\to 2\vee 3}} < t^{(3)}_{[\mathrm{c}]} \cdot p_{3\to 2}. \tag{E.8}$$

Namely, each node i has two variables $t^{(i)}_{[\mathrm{u}]}$ and $t^{(i)}_{[\mathrm{c}]}$ for the respective stages, see (E.1). During Stage 1, node 1 repeatedly transmits its packets uncodedly until at least one of nodes 2 and 3 receives it. This stage can be finished within $nt^{(1)}_{[\mathrm{u}]}$ time slots, see (E.2). For node 2, we send all $\mathbf{W}_{2\to 1}$ and all $\mathbf{W}_{2\to 31}$ messages directly to node 1 and send all $\mathbf{W}_{2\to 31}$ directly to node 3; but we send all $\mathbf{W}_{2\to 3}$ messages uncodedly until at least one of the nodes 1 and 3 receives it. Such an *uncoded stage* can be finished in $nt^{(2)}_{[\mathrm{u}]}$ time slots, see (E.3). Node 3's uncoded stage is symmetric to that of node 2.

Eq. (E.5) to (E.6) allow node 1 to perform Two-Way-Relay coding over the $3 \to 2$ and $2 \to 3$ packets overheard at node 1. (E.7) allows node 2 to relay those packets it has overheard from node 1 to the desired destination node 3. (E.8) is symmetric to (E.7).

• **[47] & Time-sharing:** The rate regions can be described by Proposition 3.2.2 with the variables $t^{(i)}_{[\mathrm{c},\,2]}$, $t^{(i)}_{[\mathrm{c},\,3]}$, and $t^{(i)}_{[\mathrm{c},\,4]} = 0$ hardwired to 0 for all $i \in \{1, 2, 3\}$. Namely, we only allow, as in [47], the broadcast channel LNC of coding choice $[\mathrm{c}, 1]$ during the Stage 2.

• **Uncoded direct TX:** This scheme does not perform any coding operation when transmitting, and just uncodedly transmits packets one by one until the desired receivers receive it. The rate region of this primitive scheme can be described by

$$\sum_{\forall\, i \in \{1,2,3\}} \frac{R_{i\to j}}{p_{i\to j}} + \frac{R_{i\to k}}{p_{i\to k}} + \frac{R_{i\to jk}}{p_{i\to j}} + \frac{R_{i\to jk}}{p_{i\to k}} \leq 1.$$

F. PROOFS OF THREE LEMMAS AND THREE CLAIMS

Proof of Lemma 2.2.1: We prove this by induction. When $t = 1$, then (2.6) and (2.10) are equivalent by definition. Suppose (2.6) and (2.10) are equivalent for $t = 1$ to $t_0 - 1$. We now consider $t = t_0$. By Lemma D.0.5, $[\mathbf{Y}_{*i}]_1^{t_0-1}$ can be uniquely computed by the values of $\mathbf{W}_{\{1,2,3\}*}$ and $[\mathbf{Z}]_1^{t_0-1}$. As a result, we can rewrite (2.6) by

$$\sigma_i(t_0) = \overline{f}^{(t_0)}_{\mathsf{SCH},\,i}(\mathbf{W}_{\{1,2,3\}*}, [\mathbf{Z}]_1^{t_0-1}). \tag{F.1}$$

Then due to the information equality (2.5), there is no dependence between $\sigma_i(t_0)$ and $\mathbf{W}_{\{1,2,3\}*}$. As a result, we can further remove $\mathbf{W}_{\{1,2,3\}*}$ from the input arguments in (F.1), which leads to (2.10). By induction, the proof of Lemma 2.2.1 is thus complete. ■

Proof of Lemma D.0.5: The proof follows from the induction on time t. When $t = 1$, each node i encodes the input symbol $X_i(1)$ purely based on its information messages $\mathbf{W}_{i*}$, see (2.7). As a result, $\{X_1(1), X_2(1), X_3(1)\}$ can be uniquely determined by $\mathbf{W}_{\{1,2,3\}*}$. Lemma D.0.5 thus holds for $t = 1$.

Suppose that the statement of Lemma D.0.5 is true until time $t = t_0 - 1$. Consider $t = t_0$. By induction, $[X_1, X_2, X_3]_1^{t_0-1}$ can be uniquely decided by $\mathbf{W}_{\{1,2,3\}*}$ and $[\mathbf{Z}]_1^{t_0-2}$. Since $[\mathbf{Y}_{1*}, \mathbf{Y}_{2*}, \mathbf{Y}_{3*}]_1^{t_0-1}$ is a function of $[X_1, X_2, X_3]_1^{t_0-1}$ and $[\mathbf{Z}]_1^{t_0-1}$, we know that $[\mathbf{Y}_{1*}, \mathbf{Y}_{2*}, \mathbf{Y}_{3*}]_1^{t_0-1}$ can be uniquely decided by $\mathbf{W}_{\{1,2,3\}*}$ and $[\mathbf{Z}]_1^{t_0-1}$. Then by the encoding functions in (2.7), the input symbols $\{X_1(t_0), X_2(t_0), X_3(t_0)\}$ at time $t = t_0$ can be uniquely determined as well. The proof of Lemma D.0.5 is thus complete. ■

Proof of Lemma D.0.6: Similar to Lemma D.0.5, the proof follows from induction on time t. When $t = 1$, in the beginning of time slot 1, $X_1(1)$ (resp. $X_3(1)$) is encoded purely based on the message $\mathbf{W}_{1*}$ (resp. $\mathbf{W}_{3*}$), see (2.7). As a result, $\{X_1(1), X_3(1)\}$ can be uniquely determined by $\mathbf{W}_{\{1,3\}*}$.

Assume that the statement of Lemma D.0.6 is true until time $t = t_0 - 1$. By induction, $[X_1, X_3]_1^{t_0-1}$ can be uniquely determined by $\{\mathbf{W}_{\{1,3\}*}, [\mathbf{Y}_{2*}]_1^{t_0-2}, [\mathbf{Z}]_1^{t_0-2}\}$. Now consider time $t = t_0$. Compared to time $t = t_0 - 1$, we know additionally $Y_{2\to1}(t_0-1)$, $Y_{2\to3}(t_0-1)$, and $\mathbf{Z}(t_0-1)$. Since we already knew $[X_3]_1^{t_0-1}$, the received symbols $[Y_{3\to1}]_1^{t_0-1}$ can be uniquely determined from the given $[\mathbf{Z}]_1^{t_0-1}$. Jointly with the known messages $\mathbf{W}_{1*}$, the received symbols $[Y_{2\to1}]_1^{t_0-1}$, and $[\mathbf{Z}]_1^{t_0-1}$, we can also uniquely determine $X_1(t_0)$, see the encoding function of node 1 in (2.7). The proof regarding to $X_3(t_0)$ can be done by symmetry. The proof of Lemma D.0.6 is thus complete. ■

Proof of Claim D.0.2. The equality (D.13) in Claim D.0.2 can be proven as follows. Notice that

$$\begin{aligned}
&I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}]_1^n \,|\, \mathbf{W}_{1*}, [\mathbf{Z}]_1^n) \\
&= I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^n \,|\, \mathbf{W}_{1*}) - I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Z}]_1^n \,|\, \mathbf{W}_{1*}) && \text{(F.2)} \\
&= I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^n \,|\, \mathbf{W}_{1*}) && \text{(F.3)} \\
&= I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{1*}) \\
&\quad + I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Z}(n) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*}) \\
&\quad + I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(n) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*}, \mathbf{Z}(n)) && \text{(F.4)} \\
&= I(\mathbf{W}_{\{2,3\}*}\,;\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{1*}) \\
&\quad + I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(n) \,|\, [\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*}, \mathbf{Z}(n)), && \text{(F.5)}
\end{aligned}$$

where (F.2) follows from the chain rule; (F.3) follows from the fact that $\mathbf{W}_{\{2,3\}*}$, $\mathbf{W}_{1*}$ and $[\mathbf{Z}]_1^n$ are independent with each other; (F.4) follows from the chain rule; and (F.5) can be obtained by showing that the second term of (F.4) is zero. The reason is that by our problem formulation, $\mathbf{Z}(n)$ is independent of $\mathbf{W}_{\{2,3\}*}$, $[\mathbf{Y}_{*1}, \mathbf{Z}]_1^{n-1}$, and $\mathbf{W}_{1*}$.

By iteratively applying the equalities (F.4) to (F.5), we have proven Claim D.0.2.

■

Proof of Claim D.0.3. For any fixed deterministic channel realization $[\mathbf{z}]_1^{t-1}$, we will consider the mutual information terms in (D.16), conditioning on the event $[\mathbf{Z}]_1^{t-1} = [\mathbf{z}]_1^{t-1}$. For notational simplicity, we use $\vec{\mathbf{z}}$ to denote the deterministic channel realization $[\mathbf{z}]_1^{t-1}$ of interest and use $\langle\vec{\mathbf{z}}\rangle \triangleq \{[\mathbf{Z}]_1^{t-1} = [\mathbf{z}]_1^{t-1}\}$ to denote the corresponding event.

For any fixed deterministic $\vec{\mathbf{z}}$ and fixed time instant t, we define

$$\mathsf{term}_0^{[\vec{\mathbf{z}}]} \triangleq I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{1*}, \mathbf{Z}(t)), \tag{F.6}$$

$$\mathsf{term}_1^{[\vec{\mathbf{z}}]} \triangleq I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{1*}, \mathbf{Z}(t)), \tag{F.7}$$

$$\mathsf{term}_2^{[\vec{\mathbf{z}}]} \triangleq I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to3}}, \mathbf{Z}(t)), \tag{F.8}$$

$$\mathsf{term}_3^{[\vec{\mathbf{z}}]} \triangleq I(\mathbf{W}_{3\to2}\,;\,\mathbf{Y}_{3*}(t)\,|\,[\mathbf{Y}_{3*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{3\to2}}, \mathbf{Z}(t)). \tag{F.9}$$

By the definition of mutual information, we have

$$I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{1*}, \mathbf{Z}(t)) = \sum_{\forall\,\vec{\mathbf{z}}} \mathsf{Prob}([\mathbf{Z}]_1^{t-1} = \vec{\mathbf{z}}) \cdot \mathsf{term}_0^{[\vec{\mathbf{z}}]}, \tag{F.10}$$

$$I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{1*}, \mathbf{Z}(t)) = \sum_{\forall\,\vec{\mathbf{z}}} \mathsf{Prob}([\mathbf{Z}]_1^{t-1} = \vec{\mathbf{z}}) \cdot \mathsf{term}_1^{[\vec{\mathbf{z}}]}, \tag{F.11}$$

$$I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{\overline{2\to3}}, \mathbf{Z}(t)) = \sum_{\forall\,\vec{\mathbf{z}}} \mathsf{Prob}([\mathbf{Z}]_1^{t-1} = \vec{\mathbf{z}}) \cdot \mathsf{term}_2^{[\vec{\mathbf{z}}]}, \tag{F.12}$$

$$I(\mathbf{W}_{3\to2}\,;\,\mathbf{Y}_{3*}(t)\,|\,[\mathbf{Y}_{3*}, \mathbf{Z}]_1^{t-1}, \mathbf{W}_{\overline{3\to2}}, \mathbf{Z}(t)) = \sum_{\forall\,\vec{\mathbf{z}}} \mathsf{Prob}([\mathbf{Z}]_1^{t-1} = \vec{\mathbf{z}}) \cdot \mathsf{term}_3^{[\vec{\mathbf{z}}]}. \tag{F.13}$$

Comparing (D.16) and equalities (F.10) to (F.13), it is clear that we only need to prove that for all $\vec{\mathbf{z}}$, the following inequality holds:

$$\mathsf{term}_0^{[\vec{\mathbf{z}}]} \geq \mathsf{term}_1^{[\vec{\mathbf{z}}]} + \frac{p_{2\to 1}}{p_{2\to 3\vee 1}}\mathsf{term}_2^{[\vec{\mathbf{z}}]} + \frac{p_{3\to 1}}{p_{3\to 1\vee 2}}\mathsf{term}_3^{[\vec{\mathbf{z}}]}. \quad \text{(F.14)}$$

To prove (F.14), we first partition all the past channel status realizations $\vec{\mathbf{z}}$ into three disjoint sets, depending on the value of the scheduling decision $\sigma(t)$, see (2.9). That is, for all $i \in \{1, 2, 3\}$,

$$\mathcal{Z}_i \triangleq \{\,\forall\,\vec{\mathbf{z}} : \sigma(t) = i\}.$$

This partition can be done uniquely since the scheduling decision $\sigma(t)$ is a function of the past channel status $[\mathbf{Z}]_1^{t-1}$.

We now prove (F.14) depending on to which $\mathcal{Z}_i$ the realization vector $\vec{\mathbf{z}}$ belong. Specifically, we will prove the following:

- For all $\vec{\mathbf{z}} \in \mathcal{Z}_1$, we have

$$\mathsf{term}_0^{[\vec{\mathbf{z}}]} = 0, \quad \text{(F.15)}$$

$$\mathsf{term}_1^{[\vec{\mathbf{z}}]} = 0, \quad \text{(F.16)}$$

$$\mathsf{term}_2^{[\vec{\mathbf{z}}]} = 0, \quad \text{(F.17)}$$

$$\mathsf{term}_3^{[\vec{\mathbf{z}}]} = 0. \quad \text{(F.18)}$$

- For all $\vec{\mathbf{z}} \in \mathcal{Z}_2$, we have

$$\mathsf{term}_0^{[\vec{\mathbf{z}}]} \geq \mathsf{term}_1^{[\vec{\mathbf{z}}]} + \frac{p_{2\to 1}}{p_{2\to 3\vee 1}}\mathsf{term}_2^{[\vec{\mathbf{z}}]}, \quad \text{(F.19)}$$

$$\mathsf{term}_3^{[\vec{\mathbf{z}}]} = 0. \quad \text{(F.20)}$$

- For all $\vec{\mathbf{z}} \in \mathcal{Z}_3$, we have

$$\mathsf{term}_0^{[\vec{\mathbf{z}}]} \geq \mathsf{term}_1^{[\vec{\mathbf{z}}]} + \frac{p_{3\to 1}}{p_{3\to 1\vee 2}}\mathsf{term}_3^{[\vec{\mathbf{z}}]}, \tag{F.21}$$

$$\mathsf{term}_2^{[\vec{\mathbf{z}}]} = 0. \tag{F.22}$$

Then, one can see that (F.15) to (F.22) jointly imply that (F.14) holds for all the past channel output realizations $\vec{\mathbf{z}}$.

Consider the first case in which $\vec{\mathbf{z}} \in \mathcal{Z}_1$. (F.15) is true because

$$\begin{aligned}
\mathsf{term}_0^{[\vec{\mathbf{z}}]} &\triangleq I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{1*}, \mathbf{Z}(t)) \\
&\leq H(\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle) && \text{(F.23)} \\
&= 0, && \text{(F.24)}
\end{aligned}$$

where (F.23) follows from the definition of mutual information, non-negativity of entropy, and the fact that conditioning reduces entropy; and (F.24) follows from that, when the scheduling decision is $\sigma(t) = 1$, the received symbols at node 1, i.e., $\mathbf{Y}_{*1}(t)$, are always erasure.

Similarly applying the above arguments, one can prove that (F.16) to (F.18) are true as well when $\vec{\mathbf{z}} \in \mathcal{Z}_1$. The first case is thus proven.

Consider the second case in which $\vec{\mathbf{z}} \in \mathcal{Z}_2$. By the same argument as used in proving (F.15) to (F.18), we can easily prove (F.20). We now prove (F.19). Then notice that

$$\begin{aligned}
\mathsf{term}_0^{[\vec{\mathbf{z}}]} &\triangleq I(\mathbf{W}_{\{2,3\}*}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{1*},\mathbf{Z}(t)) \\
&= I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{1*},\mathbf{Z}(t)) \\
&\quad + I(\mathbf{W}_{3\to2}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{1*},\mathbf{W}_{*1},\mathbf{Z}(t)) \\
&\quad + I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)) \qquad \text{(F.25)} \\
&\geq \mathsf{term}_1^{[\vec{\mathbf{z}}]} + I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{*1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)) \qquad \text{(F.26)} \\
&= \mathsf{term}_1^{[\vec{\mathbf{z}}]} + I(\mathbf{W}_{2\to3}\,;\,Y_{2\to1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)), \qquad \text{(F.27)}
\end{aligned}$$

where (F.25) follows from the chain rule and the fact that $\mathbf{W}_{1*}\cup\mathbf{W}_{*1}\cup\mathbf{W}_{3\to2}$ contains all 9-flow messages except for $\mathbf{W}_{2\to3}$, which, by definition (D.14), equals $\mathbf{W}_{\overline{2\to3}}$. (F.26) follows from the definition (F.7) and the non-negativity of mutual information. (F.27) follows from that when $\vec{\mathbf{z}} \in \mathcal{Z}_2$, the received symbol $Y_{3\to1}(t) \subset \mathbf{Y}_{*1}(t)$ is always erasure.

The second term in the RHS of (F.27) satisfies

$$\begin{aligned}
&I(\mathbf{W}_{2\to3}\,;\,Y_{2\to1}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)) \\
&\quad = \frac{p_{2\to1}}{p_{2\to3\vee1}} I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1},\langle\vec{\mathbf{z}}\rangle,\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(t)). \qquad \text{(F.28)}
\end{aligned}$$

Proof of (F.28): For the ease of exposition, let us denote $\mathbf{V} \triangleq \{[\mathbf{Y}_{*1}]_1^{t-1},\mathbf{W}_{\overline{2\to3}}\}$. Rewriting (F.28), we thus need to prove

$$I(\mathbf{W}_{2\to3}\,;\,Y_{2\to1}(t)\,|\,\mathbf{V},\langle\vec{\mathbf{z}}\rangle,\mathbf{Z}(t)) = \frac{p_{2\to1}}{p_{2\to3\vee1}} I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(t)\,|\,\mathbf{V},\langle\vec{\mathbf{z}}\rangle,\mathbf{Z}(t)). \qquad \text{(F.29)}$$

Since $\vec{\mathbf{z}} \in \mathcal{Z}_2$, we have $Y_{2\to1}(t) = X_2(t) \circ Z_{2\to1}(t)$. Since $Z_{2\to1}(t)$ is independent of $\mathbf{W}_{2\to3}$, $X_2(t)$, $\mathbf{V}$, and the random event $\langle\vec{\mathbf{z}}\rangle$, we thus have

$$\begin{aligned} & I(\mathbf{W}_{2\to3}\,;\, Y_{2\to1}(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle, \mathbf{Z}(t)) \\ & = \mathsf{Prob}(Z_{2\to1}(t) = 1) \cdot I(\mathbf{W}_{2\to3}\,;\, X_2(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle) \\ & = p_{2\to1} \cdot I(\mathbf{W}_{2\to3}\,;\, X_2(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle). \end{aligned} \tag{F.30}$$

By similar arguments, we can also prove that

$$\begin{aligned} & I(\mathbf{W}_{2\to3}\,;\, \mathbf{Y}_{2*}(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle, \mathbf{Z}(t)) \\ & = \mathsf{Prob}(\{Z_{2\to1}(t) = 1\} \cup \{Z_{2\to3}(t) = 1\}) \cdot I(\mathbf{W}_{2\to3}\,;\, X_2(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle) \\ & = p_{2\to3\vee1} \cdot I(\mathbf{W}_{2\to3}\,;\, X_2(t) \,|\, \mathbf{V}, \langle\vec{\mathbf{z}}\rangle). \end{aligned} \tag{F.31}$$

Equalities (F.30) and (F.31) jointly imply (F.29), which completes the proof of (F.28). $\square$

Then we observe that the mutual information term on the RHS of (F.28) also satisfies

$$
\begin{aligned}
& I(\mathbf{W}_{2\to 3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) \\
&= H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) \\
&\quad - H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{W}_{2\to 3}, \mathbf{Z}(t)) && \text{(F.32)} \\
&= H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{*1}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) \\
&\quad - H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{W}_{2\to 3}, \mathbf{Z}(t)) && \text{(F.33)} \\
&\geq H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}, \mathbf{Y}_{3*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) \\
&\quad - H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{W}_{2\to 3}, \mathbf{Z}(t)) && \text{(F.34)} \\
&= H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) \\
&\quad - H(\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{W}_{2\to 3}, \mathbf{Z}(t)) && \text{(F.35)} \\
&= I(\mathbf{W}_{2\to 3}\,;\,\mathbf{Y}_{2*}(t)\,|\,[\mathbf{Y}_{2*}]_1^{t-1}, \langle\vec{\mathbf{z}}\rangle, \mathbf{W}_{\overline{2\to 3}}, \mathbf{Z}(t)) && \text{(F.36)} \\
&= \mathsf{term}_2^{[\vec{\mathbf{z}}]}, && \text{(F.37)}
\end{aligned}
$$

where (F.32) follows from the definition of mutual information; (F.33) follows from that (i) $\mathbf{W}_{\overline{2\to 3}} \cup \mathbf{W}_{2\to 3}$ contains all the 9-flow information messages $\mathbf{W}_{\{1,2,3\}*}$, and (ii) by Lemma D.0.5, both $[\mathbf{Y}_{*1}]_1^{t-1}$ and $[\mathbf{Y}_{2*}]_1^{t-1}$ can be uniquely computed once we know all the messages $\mathbf{W}_{\{1,2,3\}*} = \mathbf{W}_{\overline{2\to 3}} \cup \mathbf{W}_{2\to 3}$ and the past channel realizations $\vec{\mathbf{z}} = [\mathbf{z}]_1^{t-1}$. Therefore, the conditional entropy remains identical even when we substitute $[\mathbf{Y}_{*1}]_1^{t-1}$ by $[\mathbf{Y}_{2*}]_1^{t-1}$; (F.34) follows from the fact that conditioning reduces entropy; (F.35) follows from Lemma D.0.6 that knowing the messages $\{\mathbf{W}_{1*}, \mathbf{W}_{3*}\} \subset \mathbf{W}_{\overline{2\to 3}}$, the received symbols $[\mathbf{Y}_{2*}]_1^{t-1}$, and the past channel realizations $\vec{\mathbf{z}} = [\mathbf{z}]_1^{t-1}$ can uniquely decide $[X_3]_1^t$, and thus also the received symbols $[\mathbf{Y}_{3*}]_1^{t-1}$ (since $[\mathbf{z}]_1^{t-1}$ is known). As a result, removing $[\mathbf{Y}_{3*}]_1^{t-1}$ in the first term of (F.34) will not change the conditional entropy; (F.36) follows from the definition of mutual information; and (F.37) follows from the definition (F.8).

Jointly (F.27), (F.28), and (F.37) imply (F.19).

The third case, $\vec{\mathbf{z}} \in \mathcal{Z}_3$, is symmetric to the case of $\vec{\mathbf{z}} \in \mathcal{Z}_2$. The proof of Claim D.0.3 is thus complete. ■

Proof of Claim D.0.4. We provide the proofs for the (in)equalities (D.17) to (D.19) in Claim D.0.4. We first show the proof for (D.17).

Proof of (D.17): Note that

$$
\begin{aligned}
I(\mathbf{W}_{*1}\,;\,\mathbf{W}_{1*},[\mathbf{Y}_{*1},\mathbf{Z}]_1^n) &= I(\mathbf{W}_{*1}\,;\,\mathbf{W}_{1*}) + I(\mathbf{W}_{*1}\,;\,[\mathbf{Y}_{*1},\mathbf{Z}]_1^n \,|\, \mathbf{W}_{1*}) && \text{(F.38)}\\
&= I(\mathbf{W}_{*1}\,;\,[\mathbf{Y}_{*1},\mathbf{Z}]_1^n \,|\, \mathbf{W}_{1*}) && \text{(F.39)}\\
&= I(\mathbf{W}_{*1}\,;\,[\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{1*})\\
&\quad + I(\mathbf{W}_{*1}\,;\,\mathbf{Z}(n) \,|\, [\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*})\\
&\quad + I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(n) \,|\, [\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*}, \mathbf{Z}(n)) && \text{(F.40)}\\
&= I(\mathbf{W}_{*1}\,;\,[\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{1*})\\
&\quad + I(\mathbf{W}_{*1}\,;\,\mathbf{Y}_{*1}(n) \,|\, [\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1}, \mathbf{W}_{1*}, \mathbf{Z}(n)), && \text{(F.41)}
\end{aligned}
$$

where (F.38) follows from the chain rule; (F.39) follows from the fact the messages $\mathbf{W}_{*1}$ and $\mathbf{W}_{1*}$ are independent with each other; (F.40) follows from the chain rule; and (F.41) can be obtained by showing that the second term of (F.40) is zero. The reason is because $\mathbf{Z}(n)$ is independent of $\mathbf{W}_{*1}$, $[\mathbf{Y}_{*1},\mathbf{Z}]_1^{n-1}$, and $\mathbf{W}_{1*}$. By iteratively applying the equalities (F.40) to (F.41) for $t = n-1$ back to $t = 1$, the result (D.17) follows. □

Secondly, we prove (D.18). The proof of (D.19) can be derived symmetrically by swapping the node indices 2 and 3.

Proof of (D.18): Note that

$$
\begin{aligned}
& I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{3*},[\mathbf{Y}_{*3},\mathbf{Z}]_1^n) \\
& \le I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{\{1,3\}*},\mathbf{W}_{2\to1},\mathbf{W}_{2\to31},[\mathbf{Y}_{2*},Y_{1\to3},\mathbf{Z}]_1^n) && \text{(F.42)} \\
& = I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{\{1,3\}*},\mathbf{W}_{2\to1},\mathbf{W}_{2\to31},[\mathbf{Y}_{2*},\mathbf{Z}]_1^n) && \text{(F.43)} \\
& = I(\mathbf{W}_{2\to3}\,;\,\mathbf{W}_{\overline{2\to3}}) + I(\mathbf{W}_{2\to3}\,;\,[\mathbf{Y}_{2*},\mathbf{Z}]_1^n \,|\, \mathbf{W}_{\overline{2\to3}}), && \text{(F.44)} \\
& = I(\mathbf{W}_{2\to3}\,;\,[\mathbf{Y}_{2*},\mathbf{Z}]_1^n \,|\, \mathbf{W}_{\overline{2\to3}}), && \text{(F.45)} \\
& = I(\mathbf{W}_{2\to3}\,;\,[\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{\overline{2\to3}}) \\
& \quad + I(\mathbf{W}_{2\to3}\,;\,\mathbf{Z}(n) \,|\, [\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1},\mathbf{W}_{\overline{2\to3}}) \\
& \quad + I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(n) \,|\, [\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1},\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(n)) && \text{(F.46)} \\
& = I(\mathbf{W}_{2\to3}\,;\,[\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1} \,|\, \mathbf{W}_{\overline{2\to3}}) \\
& \quad + I(\mathbf{W}_{2\to3}\,;\,\mathbf{Y}_{2*}(n) \,|\, [\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1},\mathbf{W}_{\overline{2\to3}},\mathbf{Z}(n)), && \text{(F.47)}
\end{aligned}
$$

where (F.42) follows from the fact that adding the observations $\mathbf{W}_{1*}$, $\mathbf{W}_{2\to1}$, $\mathbf{W}_{2\to31}$, and $[Y_{2\to1}]_1^n$ increases the mutual information; (F.43) follows from Lemma D.0.6 that $[X_1]_1^n$ is a function of $\mathbf{W}_{\{1,3\}*}$, $[\mathbf{Y}_{2*}]_1^{n-1}$, and $[\mathbf{Z}]_1^{n-1}$, which in turn implies that $[Y_{1\to3}]_1^n$ is a function of $\mathbf{W}_{\{1,3\}*}$, $[\mathbf{Y}_{2*}]_1^{n-1}$, and $[\mathbf{Z}]_1^n$ since $[Y_{1\to3}]_1^n$ is a function of $[X_1]_1^n$ and $[\mathbf{Z}]_1^n$. As a result, removing $[Y_{1\to3}]_1^n$ does not decrease the mutual information; (F.44) follows from the chain rule and the definition of $\mathbf{W}_{\overline{2\to3}}$ in (D.14); (F.45) follows from the fact the messages $\mathbf{W}_{2\to3}$ and $\mathbf{W}_{\overline{2\to3}}$ are independent of each other; (F.46) follows from the chain rule; and (F.47) follows from the second term of (F.46) being zero, since $\mathbf{Z}(n)$ is independent of $\mathbf{W}_{2\to3}$, $[\mathbf{Y}_{2*},\mathbf{Z}]_1^{n-1}$, and $\mathbf{W}_{\overline{2\to3}}$. By iteratively applying the equalities (F.46) to (F.47), the inequality (D.18) follows. □

The proof of Claim D.0.4 is thus complete. ■

G. LIST OF CODING TYPES FOR sFTs AND rFTs

We enumerate the 154 *Feasible Types* (FTs) defined in (4.7) that the source s can transmit in the following way:

sFTs $\triangleq$ {00000, 00010, 00020, 00030, 00070, 00110, 00130, 00170, 00220, 00230, 00270, 00330, 00370, 00570, 00770, 00A70, 00B70, 00F70, 00F71, 01010, 01030, 01070, 01110, 01130, 01170, 01230, 01270, 01330, 01370, 01570, 01770, 01A70, 01B70, 01F70, 01F71, 02020, 02030, 02070, 02130, 02170, 02220, 02230, 02270, 02330, 02370, 02570, 02770, 02A70, 02B70, 02F70, 02F71, 03030, 03070, 03130, 03170, 03230, 03270, 03330, 03370, 03570, 03770, 03A70, 03B70, 03F70, 03F71, 07070, 07170, 07270, 07370, 07570, 07770, 07A70, 07B70, 07F70, 07F71, 11110, 11130, 11170, 11330, 11370, 11570, 11770, 11B70, 11F70, 11F71, 13130, 13170, 13330, 13370, 13570, 13770, 13B70, 13F70, 13F71, 17170, 17370, 17570, 17770, 17B70, 17F70, 17F71, 22220, 22230, 22270, 22330, 22370, 22770, 22A70, 22B70, 22F70, 22F71, 23230, 23270, 23330, 23370, 23770, 23A70, 23B70, 23F70, 23F71, 27270, 27370, 27770, 27A70, 27B70, 27F70, 27F71, 33330, 33370, 33770, 33B70, 33F70, 33F71, 37370, 37770, 37B70, 37F70, 37F71, 57570, 57770, 57F70, 57F71, 77770, 77F70, 77F71, A7A70, A7B70, A7F70, A7F71, B7B70, B7F70, B7F71, F7F70, F7F71},

where each 5-digit index $\overline{\mathbf{b}}_1\overline{\mathbf{b}}_2\overline{\mathbf{b}}_3\overline{\mathbf{b}}_4\overline{\mathbf{b}}_5$ represent a 15-bitstring $\mathbf{b}$ of which $\overline{\mathbf{b}}_1$ is a hexadecimal of first four bits, $\overline{\mathbf{b}}_2$ is a octal of the next three bits, $\overline{\mathbf{b}}_3$ is a hexadecimal of the next four bits, $\overline{\mathbf{b}}_4$ is a octal of the next three bits, and $\overline{\mathbf{b}}_5$ is binary of the last bit. The subset of sFTs that the relay r can transmit, i.e., rFTs are listed separately in the following:

$$r\mathsf{FTs} \triangleq \{\texttt{00F71},\ \texttt{01F71},\ \texttt{02F71},\ \texttt{03F71},\ \texttt{07F71},\ \texttt{11F71},\ \texttt{13F71},\ \texttt{17F71},\ \texttt{22F71},\ \texttt{23F71},\ \texttt{27F71},\ \texttt{33F71},\ \texttt{37F71},\ \texttt{57F71},\ \texttt{77F71},\ \texttt{A7F71},\ \texttt{B7F71},\ \texttt{F7F71}\},$$

Recall that the b_{15} of a 15-bitstring $\mathbf{b}$ represents whether the coding subset belongs to $A_{15}(t)$ or not, and $A_{15}(t) \triangleq S_r(t-1)$ by definition (4.6). As a result, any coding type with $b_{15} = 1$ implies that it lies in the knowledge space of the relay r. The enumerated rFTs in the above is thus a collection of such coding subsets in sFTs with $\overline{\mathbf{b}}_5 = 1$.

H. LNC ENCODING OPERATIONS, PACKET MOVEMENT PROCESS, AND QUEUE INVARIANCE

H.1 For The Strong-Relaying Scenario of Proposition 4.2.1

In the following, we will describe all the LNC encoding operations and the corresponding packet movement process of Proposition 4.2.1 one by one, and then prove that the Queue Invariance explained in Section 4.2.1 always holds.

To simplify the analysis, we will ignore the null reception, i.e., none of $\{d_1, d_2, r\}$ receives a transmitted packet, because nothing will happen in the queueing network. Moreover, we exploit the following symmetry: For those variables whose superscript indicates the session information $k \in \{1, 2\}$ (either session-1 or session-2), here we describe session-1 ($k=1$) only. Those variables with $k=2$ in the superscript will be symmetrically explained by simultaneously swapping (a) session-1 and session-2 in the superscript; (b) X and Y; (c) i and j; and (d) d_1 and d_2, if applicable.

• s^1_{UC}: The source s transmits $X_i \in Q^1_\phi$. Depending on the reception status, the packet movement process following the inequalities in Proposition 4.2.1 is summarized as follows.

Departure	Reception Status	Insertion
$Q^1_\phi \xrightarrow{X_i}$	$\overline{d_1 d_2} r$	$\xrightarrow{X_i} Q^1_{\{r\}}$
	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{X_i} Q^1_{\{d_2\}}$
	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	$d_1 d_2 r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$

- **Departure**: One property for $X_i \in Q^1_\phi$ is that X_i must be unknown to any of $\{d_1, d_2, r\}$. As a result, whenever X_i is received by any of them, X_i must be removed from Q^1_ϕ for the Queue Invariance.

- **Insertion**: One can easily verify that the queue properties for $Q^1_{\{r\}}$, $Q^1_{\{d_2\}}$, Q^1_{dec}, and $Q^{[1]}_{\{rd_2\}}$ hold for the corresponding insertions.

- s^2_{UC}: s transmits $Y_j \in Q^2_\phi$. The movement process is symmetric to s^1_{UC}.
- s^1_{PM1}: s transmits a mixture $[X_i + Y_j]$ from $X_i \in Q^1_\phi$ and $Y_j \in Q^2_{\{r\}}$. The movement process is as follows.

$Q^1_\phi \xrightarrow{X_i}$	$\overline{d_1 d_2} r$	$\xrightarrow{X_i} Q^1_{\{r\}}$
$Q^1_\phi \xrightarrow{X_i}, Q^2_{\{r\}} \xrightarrow{Y_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{[X_i+Y_j]} Q^{m\|2}_{\{d_2\}\|\{r\}}$
	$d_1 \overline{d_2 r}$	$\xrightarrow{[X_i+Y_j]:Y_j} Q_{\mathsf{mix}}$
	$\overline{d_1} d_2 r$	$\xrightarrow{[X_i+Y_j]:X_i} Q_{\mathsf{mix}}$
	$d_1 \overline{d_2} r$ $d_1 d_2 \overline{r}$	$\xrightarrow{[X_i+Y_j]:Y_j} Q_{\mathsf{mix}}$
	$d_1 d_2 r$	$\xrightarrow{[X_i+Y_j]\text{: either } X_i \text{ or } Y_j} Q_{\mathsf{mix}}$

- **Departure**: The property for $X_i \in Q^1_\phi$ is that X_i must be unknown to any of $\{d_1, d_2, r\}$, even not flagged in $\mathsf{RL}_{\{d_1,d_2,r\}}$. As a result, whenever the mixture $[X_i + Y_j]$ is received by any of $\{d_1, d_2, r\}$, X_i must be removed from Q^1_ϕ. Similarly, the property for $Y_j \in Q^2_{\{r\}}$ is that Y_j must be unknown to any of $\{d_1, d_2\}$, even not flagged in $\mathsf{RL}_{\{d_1,d_2\}}$. Therefore, whenever the mixture is received by any of $\{d_1, d_2\}$, Y_j must be removed from $Q^2_{\{r\}}$.

- **Insertion**: When only r receives the mixture, r can use the known Y_j and the received $[X_i + Y_j]$ to extract the pure X_i. As a result, we can insert X_i to $Q^1_{\{r\}}$ as it is not flagged in $\mathsf{RL}_{\{d_1,d_2\}}$. The case when only d_2 receives the mixture satisfies the properties of $Q^{m|2}_{\{d_2\}|\{r\}}$ as r knows the pure Y_j only while d_2 knows the mixture $[X_i + Y_j]$ only. As a result, we can insert $[X_i + Y_j]$ to $Q^{m|2}_{\{d_2\}|\{r\}}$. The remaining reception cases fall into at least one of two conditions of Q_{mix}. For example when only d_1 receives the mixture, now $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_1\}}$ while Y_j is still known by r only. This corresponds to the first condition of Q_{mix}. One can easily verify that other cases satisfy either one of or both properties of Q_{mix}. Following the packet format for Q_{mix}, we insert $[X_i + Y_j] : W$ into Q_{mix} where W denotes the packet in r that can benefit both destinations when transmitted. From the previous example when only d_1 receives the mixture, we insert $[X_i + Y_j] : Y_j$ into Q_{mix} as sending the known Y_j from r simultaneously enables d_2 to receive the desired Y_j and d_1 to decode the desired X_i by subtracting Y_j from the received $[X_i + Y_j]$.

• s^2_{PM1}: s transmits a mixture $[X_i + Y_j]$ from $X_i \in Q^1_{\{r\}}$ and $Y_j \in Q^2_{\phi}$. The movement process is symmetric to s^1_{PM1}.

• s^1_{PM2}: s transmits a mixture $[X_i + Y_j]$ from $X_i \in Q^1_{\{r\}}$ and $Y_j \in Q^2_{\{d_1\}}$. The movement process is as follows.

$Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1 d_2} r$	$\xrightarrow[\text{Case 1}]{Y_j} Q^{[2]}_{\{rd_1\}}$
$Q^1_{\{r\}} \xrightarrow{X_i}$, $Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{[X_i+Y_j]:X_i} Q_{\mathsf{mix}}$
$Q^1_{\{r\}} \xrightarrow{X_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
$Q^1_{\{r\}} \xrightarrow{X_i}$, $Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1} d_2 r$	$\xrightarrow{[X_i+Y_j]:X_i} Q_{\mathsf{mix}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 1}]{Y_j} Q^{[2]}_{\{rd_1\}}$
	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 2}]{X_i(\equiv Y_j)} Q^{[2]}_{\{rd_1\}}$
	$d_1 d_2 r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 1}]{Y_j} Q^{[2]}_{\{rd_1\}}$

- **Departure**: The property for $X_i \in Q^1_{\{r\}}$ is that X_i must be unknown to any of $\{d_1, d_2\}$, even not flagged in $\mathsf{RL}_{\{d_1,d_2\}}$. As a result, whenever the mixture $[X_i + Y_j]$ is received by any of $\{d_1, d_2\}$, X_i must be removed from $Q^1_{\{r\}}$. Similarly, the property for $Y_j \in Q^2_{\{d_1\}}$ is that Y_j must be unknown to any of $\{d_2, r\}$, even not flagged in $\mathsf{RL}_{\{d_2,r\}}$. Therefore, whenever the mixture is received by any of $\{d_2, r\}$, Y_j must be removed from $Q^2_{\{d_1\}}$.

- **Insertion**: Whenever d_1 receives the mixture, d_1 can use the known Y_j and the received $[X_i + Y_j]$ to extract the pure/desired X_i. As a result, we can insert X_i into Q^1_{dec} whenever d_1 receives. The cases when d_2 receives but d_1 does not fall into the second condition of Q_{mix} as $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_2\}}$ and X_i is known by r only. Namely, r can benefit both destinations simultaneously by sending the known X_i. For those two reception status $\overline{d_1} d_2 \overline{r}$ and $\overline{d_1} d_2 r$, we can thus insert this mixture into Q_{mix} as $[X_i + Y_j] : X_i$. Whenever r receives the mixture, r can use the known X_i and the received $[X_i + Y_j]$ to extract the pure Y_j. Now Y_j is known by both r and d_1 but still unknown to d_2 even if d_2 receives this mixture $[X_i + Y_j]$ as well. As a result, Y_j can be moved to $Q^{[2]}_{\{rd_1\}}$ as the Case 1 insertion.

But for the reception status of $\overline{d_1}d_2r$, note from the previous discussion that we can insert the mixture into Q_{mix} since d_2 receives the mixture but d_1 does not. In this case, we chose to use more efficient Q_{mix} that can handle both sessions simultaneously. Finally when the reception status is $d_1d_2\overline{r}$, we have that X_i is known by both r and d_1 while the mixture $[X_i + Y_j]$ is received by d_2. Namely, X_i is still unknown to d_2 but when it is delivered, d_2 can use X_i and the received $[X_i+Y_j]$ to extract a desired session-2 packet Y_j. Moreover, X_i is already in Q^1_{dec} and thus can be used as an information-equivalent packet for Y_j. This scenario is exactly the same as the Case 2 of $Q^{[2]}_{\{rd_1\}}$ and thus we can move X_i into $Q^{[2]}_{\{rd_1\}}$ as the Case 2 insertion.

- s^2_{PM2}: s transmits a mixture $[X_i + Y_j]$ from $X_i \in Q^1_{\{d_2\}}$ and $Y_j \in Q^2_{\{r\}}$. The movement process is symmetric to s^1_{PM2}.
- s^1_{RC}: s transmits X_i of the mixture $[X_i + Y_j]$ in $Q^{m|2}_{\{d_2\}|\{r\}}$. The movement process is as follows.

$Q^{m\|2}_{\{d_2\}\|\{r\}} \xrightarrow{[X_i+Y_j]}$	$\overline{d_1d_2}r$	$\xrightarrow{[X_i+Y_j]:X_i} Q_{\mathsf{mix}}$
	$\overline{d_1}d_2\overline{r}$	$\xrightarrow{X_i} Q^1_{\{d_2\}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$
	$d_1\overline{d_2r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{X_i} Q^{(2)\|2}_{\{d_1\}\|\{r\}}$
	$\overline{d_1}d_2r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$
	$d_1\overline{d_2}r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 2}]{X_i(\equiv Y_j)} Q^{[2]}_{\{rd_1\}}$
	$d_1d_2\overline{r}$, d_1d_2r	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$

  - **Departure**: One condition for $[X_i + Y_j] \in Q^{m|2}_{\{d_2\}|\{r\}}$ is that X_i is unknown to any of $\{d_1, d_2, r\}$. As a result, whenever X_i is received by any of $\{d_1, d_2, r\}$, the mixture $[X_i + Y_j]$ must be removed from $Q^{m|2}_{\{d_2\}|\{r\}}$.
  - **Insertion**: From the conditions of $Q^{m|2}_{\{d_2\}|\{r\}}$, we know that X_i is unknown to d_1 and Y_j is known only by r. As a result, whenever d_1 receives X_i, d_1 receives the new session-1 packet and thus we can insert X_i into Q^1_{dec}. Whenever d_2 receives X_i, d_2 can use the known $[X_i + Y_j]$ and the received X_i to subtract the pure Y_j. We can thus insert Y_j into Q^2_{dec}. The case when only r receives X_i falls into the first condition of Q_{mix} as $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_2\}}$ and X_i is known by r only. In

this case, r can benefit both destinations simultaneously by sending the received X_i. For this reception status of $\overline{d_1 d_2} r$, we thus insert the mixture into Q_{mix} as $[X_i + Y_j] : X_i$. The remaining reception status to consider are $\overline{d_1} d_2 \overline{r}$, $d_1 \overline{d_2 r}$, $\overline{d_1} d_2 r$, and $d_1 \overline{d_2} r$. The first when only d_2 receives X_i falls into the property of $Q^1_{\{d_2\}}$ as X_i is known only by d_2 and not flagged in $\mathsf{RL}_{\{d_1,r\}}$. Thus we can insert X_i into $Q^1_{\{d_2\}}$. Obviously, d_2 can decode Y_j from the previous discussion. For the second when only d_1 receives X_i, we first have $X_i \in Q^1_{\mathsf{dec}}$ while X_i is unknown to any of $\{d_2, r\}$. Moreover, Y_j is known by r only and $[X_i + Y_j]$ is in $\mathsf{RL}_{\{d_2\}}$. This scenario falls exactly into $Q^2_{\{d_1\}}$ and thus we can insert X_i into $Q^2_{\{d_1\}}$. The third case when both d_2 and r receive X_i falls exactly into Case 1 of $Q^{[1]}_{\{rd_2\}}$ as X_i is now known by both d_2 and r but still unknown to d_1. And obviously, d_2 can decode Y_j from the previous discussion. For the fourth case when both d_1 and r receive X_i, we now have that r contains $\{X_i, Y_j\}$; d_1 contains X_i; and d_2 contains $[X_i + Y_j]$. That is, X_i is already in Q^1_{dec} and known by r as well but still unknown to d_2. Moreover, d_2 can decode the desired session-2 packet Y_j when it receives X_i further. As a result, X_i can be used as an information-equivalent packet for Y_j and can be moved into $Q^{[2]}_{\{rd_1\}}$ as the Case 2 insertion.

- s^2_{RC}: s transmits Y_j of $[X_i + Y_j] \in Q^{m|1}_{\{d_1\}|\{r\}}$. The movement process is symmetric to s^1_{RC}.
- s^1_{DX}: s transmits $X_i \in Q^1_{\{d_2\}}$. The movement process is as follows.

$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$\overline{d_1 d_2} r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
do nothing	$\overline{d_1} d_2 \overline{r}$	do nothing
$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	$d_1 d_2 \overline{r}$	
	$d_1 d_2 r$	

- **Departure**: One condition for $X_i \in Q^1_{\{d_2\}}$ is that X_i must be unknown to any of $\{d_1, r\}$. As a result, X_i must be removed from $Q^1_{\{d_2\}}$ whenever it is received by any of $\{d_1, r\}$.

- **Insertion**: Whenever d_1 receives X_i, it receives a new session-1 packet and thus we can insert X_i into Q^1_{dec}. If X_i is received by r but not by d_1, then X_i will be known by both d_2 and r (since d_2 already knows X_i) but still unknown to d_1. This falls exactly into the first-case scenario of $Q^{[1]}_{\{rd_2\}}$ and thus we can move X_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 1 insertion.

- s^2_{DX}: s transmits $Y_j \in Q^2_{\{d_1\}}$. The movement process is symmetric to s^1_{DX}.
- $s^{(1)}_{\mathsf{DX}}$: s transmits $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$. The movement process is as follows.

$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$\overline{d_1 d_2} r$	$\xrightarrow[\text{Case 2}]{Y_i} Q^{[1]}_{\{rd_2\}}$
do nothing	$\overline{d_1} d_2 \overline{r}$	do nothing
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i (\equiv Y_i)} Q^1_{\mathsf{dec}}$
	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 2}]{Y_i} Q^{[1]}_{\{rd_2\}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i (\equiv Y_i)} Q^1_{\mathsf{dec}}$
	$d_1 d_2 \overline{r}$	
	$d_1 d_2 r$	

- **Departure**: One property for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ is that Y_i must be unknown to any of $\{d_1, r\}$. As a result, whenever Y_i is received by any of $\{d_1, r\}$, Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$.

- **Insertion**: From the property of $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$, we know that $Y_i \in Q^2_{\mathsf{dec}}$; there exists a session-1 packet X_i still unknown to d_1 where $X_i \equiv Y_i$; and $[X_i + Y_i]$ is in $\mathsf{RL}_{\{d_1\}}$. As a result, whenever d_1 receives Y_i, d_1 can use the received Y_i and the known $[X_i + Y_i]$ to extract X_i and thus we can insert X_i into Q^1_{dec}. If Y_i is received by r but not by d_1, then Y_i will be known by both d_2 and r but unknown to d_1, where $[X_i + Y_i]$ is in $\mathsf{RL}_{\{d_1\}}$. Thus when d_1 receives Y_i, d_1 can further decode the desired X_i. Moreover, Y_i is already in Q^2_{dec}. As a result, we can move Y_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 2 insertion.

- $s^{(2)}_{\mathsf{DX}}$: s transmits $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$. The movement process is symmetric to $s^{(1)}_{\mathsf{DX}}$.

- $s_{\mathsf{CX};1}$: s transmits $[X_i + Y_j]$ from $X_i \in Q^1_{\{d_2\}}$ and $Y_j \in Q^2_{\{d_1\}}$. The movement process is as follows.

<table>
<tr><td>$Q^1_{\{d_2\}} \xrightarrow{X_i}$,
$Q^2_{\{d_1\}} \xrightarrow{Y_j}$</td><td>$\overline{d_1 d_2} r$</td><td>$\xrightarrow{[X_i+Y_j]} Q^{m_{\mathsf{CX}}}_{\{r\}}$</td></tr>
<tr><td>$Q^2_{\{d_1\}} \xrightarrow{Y_j}$</td><td>$\overline{d_1} d_2 \overline{r}$</td><td>$\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$</td></tr>
<tr><td>$Q^1_{\{d_2\}} \xrightarrow{X_i}$</td><td>$d_1 \overline{d_2 r}$</td><td>$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$</td></tr>
<tr><td rowspan="4">$Q^1_{\{d_2\}} \xrightarrow{X_i}$,
$Q^2_{\{d_1\}} \xrightarrow{Y_j}$</td><td>$\overline{d_1} d_2 r$</td><td>$\xrightarrow[\text{Case 3}]{[X_i+Y_j]} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$</td></tr>
<tr><td>$d_1 \overline{d_2} r$</td><td>$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 3}]{[X_i+Y_j]} Q^{[2]}_{\{rd_1\}}$</td></tr>
<tr><td>$d_1 d_2 \overline{r}$</td><td rowspan="2">$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$</td></tr>
<tr><td>$d_1 d_2 r$</td></tr>
</table>

- **Departure**: One condition for $X_i \in Q^1_{\{d_2\}}$ is that X_i must be unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{d_1,r\}}$. As a result, whenever the mixture is received by any of $\{d_1, r\}$, X_i must be removed from $Q^1_{\{d_2\}}$. Symmetrically for $Y_j \in Q^2_{\{d_1\}}$, whenever the mixture is received by any of $\{d_2, r\}$, Y_j must be removed from $Q^2_{\{d_1\}}$.

- **Insertion**: Whenever d_1 receives the mixture $[X_i + Y_j]$, d_1 can use the known $Y_j \in Q^2_{\{d_1\}}$ and the received $[X_i + Y_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known $X_i \in Q^1_{\{d_2\}}$ and the received $[X_i + Y_j]$ to extract the desired Y_j and thus we can insert Y_j into Q^2_{dec}. The remaining reception status are $\overline{d_1 d_2} r$, $\overline{d_1} d_2 r$, and $d_2 \overline{d_2} r$. The first when only r receives the mixture exactly falls into the first-case scenario of $Q^{m_{\mathsf{CX}}}_{\{r\}}$ as $[X_i + Y_j]$ is in $\mathsf{RL}_{\{r\}}$; $X_i \in Q^1_{\{d_2\}}$ is known by d_2 only; and $Y_j \in Q^2_{\{d_1\}}$ is known by d_1 only. As a result, r can then send this mixture $[X_i + Y_j]$ to benefit both destinations. The second case when both d_2 and r receive the mixture, jointly with the assumption $Y_j \in Q^2_{\{d_1\}}$, falls exactly into the third-case scenario of $Q^{[1]}_{\{rd_2\}}$ where W_i is a pure session-1 packet. As a result, we can move $[X_i + Y_j]$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 3 insertion. (And obviously, d_2 can decode Y_j from the previous discussion.) The third case when both d_1 and r receive the mixture follows symmetrically to the second case of $\overline{d_1} d_2 r$ and thus we can insert $[X_i + Y_j]$ into $Q^{[2]}_{\{rd_1\}}$ as the Case 3 insertion.

- $s_{\mathsf{CX};2}$:s transmits $[X_i{+}X_j]$ from $X_i \in Q^1_{\{d_2\}}$ and $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$. The movement process is as follows.

$Q^1_{\{d_2\}} \xrightarrow{X_i}$, $Q^{(2)|2}_{\{d_1\}|\{r\}} \xrightarrow{X_j}$	$\overline{d_1 d_2} r$	$\xrightarrow{[X_i+X_j]} Q^{m_{\mathsf{CX}}}_{\{r\}}$
$Q^{(2)|2}_{\{d_1\}|\{r\}} \xrightarrow{X_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$, $Q^{(2)|2}_{\{d_1\}|\{r\}} \xrightarrow{X_j}$	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 3}]{[X_i+X_j]} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 3}]{[X_i+X_j]} Q^{[2]}_{\{rd_1\}}$
	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
	$d_1 d_2 r$	

  - **Departure**: One condition for $X_i \in Q^1_{\{d_2\}}$ is that X_i must be unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{d_1,r\}}$. As a result, whenever the mixture $[X_i + X_j]$ is received by any of $\{d_1, r\}$, X_i must be removed from $Q^1_{\{d_2\}}$. From the property for $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$, we know that X_j is unknown to any of $\{d_2, r\}$, even not flagged in $\mathsf{RL}_{\{r\}}$. As a result, whenever r receives the mixture $[X_i + X_j]$, X_j must be removed from $Q^{(2)|2}_{\{d_1\}|\{r\}}$. Moreover, whenever d_2 receives this mixture, d_2 can use the known $X_i \in Q^1_{\{d_2\}}$ and the received $[X_i + X_j]$ to decode X_j and thus X_j must be removed from $Q^{(2)|2}_{\{d_1\}|\{r\}}$.

  - **Insertion**: From the properties of $X_i \in Q^1_{\{d_2\}}$ and $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$, we know that r contains Y_j (still unknown to d_2 and $Y_j \equiv X_j$); d_1 contains X_j; and d_2 contains $\{X_i, [Y_j + X_j]\}$ already. Therefore, whenever d_1 receives the mixture $[X_i + X_j]$, d_1 can use the known X_j and the received $[X_i{+}X_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known $\{X_i, [Y_j + X_j]\}$ and the received $[X_i + X_j]$ to extract the desired Y_j, and thus we can insert Y_j into Q^2_{dec}. The remaining reception status are $\overline{d_1 d_2} r$, $\overline{d_1} d_2 r$, and $d_2 \overline{d_2} r$. One can see that the case when only r receives the mixture exactly falls into the Case 2 scenario of $Q^{m_{\mathsf{CX}}}_{\{r\}}$. For the second case when both d_2 and r receive the mixture, now r contains $\{Y_j, [X_i + X_j]\}$; d_1 contained X_j before; and d_2 contains $\{X_i, [Y_j + X_j], [X_i + X_j]\}$. This falls exactly into the third-case scenario of $Q^{[1]}_{\{rd_2\}}$ where W_i is a pure session-1 packet X_i. As a result, we can move $[X_i{+}X_j]$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 3 insertion. (And obviously, d_2 can decode the desired Y_j from the previous discussion.) For the third case when both d_1 and r receive the mixture, now r contains $\{Y_j, [X_i + X_j]\}$; d_1 contains $\{X_j, [X_i + X_j]\}$; and d_2 contained $\{X_i, [Y_j + X_j]\}$ before, where we now have $X_i \in Q^1_{\mathsf{dec}}$ from the previous discussion. This falls exactly into the third-case scenario of $Q^{[2]}_{\{rd_1\}}$ where W_j is a pure session-1 packet $X_j \in Q^1_{\mathsf{dec}}$. Note that delivering $[X_i + X_j]$ will enable d_2 to further decode the desired Y_j. Thus we can move $[X_i + X_j]$ into $Q^{[2]}_{\{rd_1\}}$ as the Case 3 insertion.

- $s_{\mathsf{CX};3}$: s transmits $[Y_i + Y_j]$ from $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $Y_j \in Q^2_{\{d_1\}}$. The movement process is as follows.

$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$, $Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1 d_2} r$	$\xrightarrow{[Y_i+Y_j]} Q^{m_{\mathsf{CX}}}_{\{r\}}$
$Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$, $Q^2_{\{d_1\}} \xrightarrow{Y_j}$	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 3}]{[Y_i+Y_j]} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 3}]{[Y_i+Y_j]} Q^{[2]}_{\{rd_1\}}$
	$d_1 d_2 \overline{r}$, $d_1 d_2 r$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$

  - **Departure**: From the property for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$, we know that Y_i is unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{r\}}$. As a result, whenever r receives the mixture $[Y_i + Y_j]$, Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. Moreover, whenever d_1 receives this mixture, d_1 can use the known $Y_j \in Q^2_{\{d_1\}}$ and the received $[Y_i + Y_j]$ to decode Y_i and thus Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. One condition for $Y_j \in Q^2_{\{d_1\}}$ is that Y_j must be unknown to any of $\{d_2, r\}$, even not flagged in $\mathsf{RL}_{\{d_2,r\}}$. As a result, whenever the mixture $[Y_i + Y_j]$ is received by any of $\{d_2, r\}$, Y_j must be removed from $Q^2_{\{d_1\}}$.

  - **Insertion**: From the properties of $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $Y_j \in Q^2_{\{d_1\}}$, we know that r contains X_i (still unknown to d_1 and $X_i \equiv Y_i$); d_1 contains $\{Y_j, [X_i + Y_i]\}$; and d_2 contains Y_i already. Therefore, whenever d_1 receives the mixture $[Y_i + Y_j]$, d_1 can use the known $\{Y_j, [X_i + Y_i]\}$ and the received $[Y_i + Y_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known Y_i and the received $[Y_i + Y_j]$ to extract the desired Y_j, and thus we can insert Y_j into Q^2_{dec}. The remaining reception status are $\overline{d_1 d_2} r$, $\overline{d_1} d_2 r$, and $d_2 \overline{d_2} r$. One can see that the first case when only r receives the mixture exactly falls into the Case 3 scenario of $Q^{m_{\mathsf{CX}}}_{\{r\}}$. For the second case when both d_2 and r receive the mixture, now r contains $\{X_i, [Y_i + Y_j]\}$; d_1 contained $\{Y_j, [X_i + Y_i]\}$ before; and d_2 contains $\{Y_i, [Y_i + Y_j]\}$, where we now have $Y_j \in Q^2_{\mathsf{dec}}$ from the previous discussion. This falls exactly into the third-case scenario of $Q^{[1]}_{\{rd_2\}}$ where W_i is a pure session-2 packet Y_i. Note that delivering $[Y_i + Y_j]$ will enable d_1 to further decode the desired X_i. Thus we can move $[Y_i + Y_j]$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 3 insertion. For the third case when both d_1 and r receive the mixture, now r contains $\{X_i, [Y_i + Y_j]\}$; d_1 contains $\{Y_j, [X_i + Y_i], [Y_i + Y_j]\}$; and d_2 contained Y_i before. This falls exactly into the third-case scenario of $Q^{[2]}_{\{rd_1\}}$ where W_j is a pure session-2 packet Y_j. As a result, we can move $[Y_i + Y_j]$ into $Q^{[2]}_{\{rd_1\}}$ as the Case 3 insertion. (And obviously, d_1 can decode the desired X_i from the previous discussion.)

- $s_{\mathsf{CX};4}$: s transmits $[Y_i + X_j]$ from $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$. The movement process is as follows.

$Q^{(1)\vert 1}_{\{d_2\}\vert\{r\}} \xrightarrow{Y_i}$, $Q^{(2)\vert 2}_{\{d_1\}\vert\{r\}} \xrightarrow{X_j}$	$\overline{d_1 d_2} r$	$\xrightarrow{[Y_i+X_j]} Q^{m_{\mathsf{CX}}}_{\{r\}}$
$Q^{(2)\vert 2}_{\{d_1\}\vert\{r\}} \xrightarrow{X_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
$Q^{(1)\vert 1}_{\{d_2\}\vert\{r\}} \xrightarrow{Y_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$
$Q^{(1)\vert 1}_{\{d_2\}\vert\{r\}} \xrightarrow{Y_i}$, $Q^{(2)\vert 2}_{\{d_1\}\vert\{r\}} \xrightarrow{X_j}$	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 3}]{[Y_i+X_j]} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
	$d_1 \overline{d_2} r$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 3}]{[Y_i+X_j]} Q^{[2]}_{\{rd_1\}}$
	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j(\equiv X_j)} Q^2_{\mathsf{dec}}$
	$d_1 d_2 r$	

- **Departure**: From the property for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$, we know that Y_i is unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{r\}}$. As a result, whenever r receives the mixture $[Y_i + X_j]$, Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. Moreover, $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$ is known by d_1. As a result, whenever d_1 receives the mixture, d_1 can use the known X_j and the received $[Y_i + X_j]$ to decode Y_i and thus Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. Symmetrically for $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$, whenever the mixture is received by any of $\{d_2, r\}$, X_j must be removed from $Q^{(2)|2}_{\{d_1\}|\{r\}}$.

- **Insertion**: From the properties of $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$, we know that r contains $\{X_i, Y_j\}$ where X_i (resp. Y_j) is still unknown to d_1 (resp. d_2) and $X_i \equiv Y_i$ (resp. $Y_j \equiv X_j$); d_1 contains $\{[X_i + Y_i], X_j\}$; and d_2 contains $\{Y_i, [Y_j + X_j]\}$ already. Therefore, whenever d_1 receives the mixture $[Y_i + X_j]$, d_1 can use the known $\{[X_i + Y_i], X_j\}$ and the received $[Y_i + X_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known $\{Y_i, [Y_j + X_j]\}$ and the received $[Y_i + X_j]$ to extract the desired Y_j, and thus we can insert Y_j into Q^2_{dec}. The remaining reception status are $\overline{d_1 d_2} r$, $\overline{d_1} d_2 r$, and $d_2 \overline{d_2} r$. One can see that the first case when only r receives the mixture exactly falls into the Case 4 scenario of $Q^{m\mathsf{CX}}_{\{r\}}$. For the second case when both d_2 and r receive the mixture, now r contains $\{X_i, Y_j, [Y_i + X_j]\}$; d_1 contained $\{[X_i + Y_i], X_j\}$ before; and d_2 contains $\{Y_i, [Y_j + X_j], [Y_i + X_j]\}$ where we now have $X_j \in Q^1_{\mathsf{dec}}$ from the previous discussion. This falls exactly into the third-case scenario of $Q^{[1]}_{\{rd_2\}}$ where W_i is a pure session-2 packet Y_i. Note that delivering $[Y_i + X_j]$ will enable d_1 to further decode the desired X_i. Thus we can move $[Y_i + X_j]$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 3 insertion. For the third case when both d_1 and r receive the mixture, now r contains $\{X_i, Y_j, [Y_i + X_j]\}$; d_1 contains $\{[X_i + Y_i], X_j, [Y_i + X_j]\}$; and d_2 contained $\{Y_i, [Y_j + X_j]\}$ before, where we now have $Y_i \in Q^2_{\mathsf{dec}}$ from the previous discussion. This falls exactly into the third-case scenario of $Q^{[2]}_{\{rd_1\}}$ where W_j is a pure session-2 packet X_j. Note that delivering $[Y_i + X_j]$ will enable d_2 to further decode the desired Y_j. Thus we can move $[Y_i + X_j]$ into $Q^{[2]}_{\{rd_1\}}$ as the Case 3 insertion.

- $s_{\mathsf{CX};5}$:s transmits $[X_i+\overline{W}_j]$ from $X_i \in Q^1_{\{d_2\}}$ and $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$. The movement process is as follows.

$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$\overline{d_1 d_2} r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
$Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$, $Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$	$d_1 \overline{d_2} r$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
$Q^1_{\{d_2\}} \xrightarrow{X_i}$, $Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
	$d_1 d_2 r$	

  - **Departure**: The property for $X_i \in Q^1_{\{d_2\}}$ is that X_i must be unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{d_1,r\}}$. As a result, whenever the mixture $[X_i+\overline{W}_j]$ is received by any of $\{d_1, r\}$, X_i must be removed from $Q^1_{\{d_2\}}$. Similarly, one condition for $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$ is that $\overline{W}_j$ must be unknown to d_2. However when d_2 receives the mixture, d_2 can use the known $X_i \in Q^1_{\{d_2\}}$ and the received $[X_i+\overline{W}_j]$ to decode $\overline{W}_j$. Thus $\overline{W}_j$ must be removed from $Q^{[2]}_{\{rd_1\}}$ whenever d_2 receives.
  - **Insertion**: From the properties of $X_i \in Q^1_{\{d_2\}}$ and $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$, we know that r contains $\overline{W}_j$; d_1 contains $\overline{W}_j$; and d_2 contains X_i already. Therefore, whenever d_1 receives this mixture, d_1 can use the known $\overline{W}_j$ and the received $[X_i+\overline{W}_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known X_i and the received $[X_i + \overline{W}_j]$ to extract $\overline{W}_j$. We now need to consider case by case when $\overline{W}_j$ was inserted into $Q^{[2]}_{\{rd_1\}}$. If it was the Case 1 insertion, then $\overline{W}_j$ is a pure session-2 packet Y_j and thus we can simply insert Y_j into Q^2_{dec}. If it was the Case 2 insertion, then $\overline{W}_j$ is a pure session-2 packet $X_j \in Q^1_{\mathsf{dec}}$ and there exists a session-2 packet Y_j still unknown to d_2 where $Y_j \equiv X_j$. Moreover, d_2 has received $[Y_j + X_j]$. As a result, d_2 can further decode Y_j and thus we can insert Y_j into Q^2_{dec}. If it was the Case 3 insertion, then $\overline{W}_j$ is a mixed form of $[W_i + W_j]$ where W_i is already known by d_2 but W_j is not. As a result, d_2 can decode W_j upon receiving $\overline{W}_j = [W_i + W_j]$. Note that W_j in the Case 3 insertion $\overline{W}_j = [W_i + W_j] \in Q^{[2]}_{\{rd_1\}}$

comes from either $Q^2_{\{d_1\}}$ or $Q^{(2)|2}_{\{d_1\}|\{r\}}$. If W_j was coming from $Q^2_{\{d_1\}}$, then W_j is a session-2 packet Y_j and we can simply insert Y_j into Q^2_{dec}. If W_j was coming from $Q^{(2)|2}_{\{d_1\}|\{r\}}$, then W_j is a session-1 packet X_j and there also exists a session-2 packet Y_j still unknown to d_2 where $Y_j \equiv X_j$. Moreover, d_2 has received $[Y_j + X_j]$. As a result, d_2 can further use the known $[Y_j + X_j]$ and the extracted X_j to decode Y_j and thus we can insert Y_j into Q^2_{dec}. In a nutshell, whenever d_2 receives the mixture $[X_i + \overline{W}_j]$, a session-2 packet Y_j that was unknown to d_2 can be newly decoded. The remaining reception status are $\overline{d_1 d_2} r$ and $\overline{d_1} d_2 r$. For both cases when r receives the mixture but d_1 does not, r can use the known $\overline{W}_j$ and the received $[X_i + \overline{W}_j]$ to extract X_i. Since X_i is now known by both r and d_2 but unknown to d_1, we can thus move X_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 1 insertion.

- $s_{\mathsf{CX};6}$: s transmits $[\overline{W}_i + Y_j]$ from $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$ and $Y_j \in Q^2_{\{d_1\}}$. The movement process is symmetric to $s_{\mathsf{CX};5}$.
- $s_{\mathsf{CX};7}$: s transmits $[Y_i + \overline{W}_j]$ from $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$. The movement process is as follows.

$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$\overline{d_1 d_2} r$	$\xrightarrow[\text{Case 2}]{Y_i} Q^{[1]}_{\{rd_2\}}$
$Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$\overline{d_1} d_2 \overline{r}$	$\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$d_1 \overline{d_2 r}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$, $Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$\overline{d_1} d_2 r$	$\xrightarrow[\text{Case 2}]{Y_i} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$d_1 \overline{d_2} r$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$
$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$, $Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$d_1 d_2 \overline{r}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$,
	$d_1 d_2 r$	$\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$

- **Departure**: From the property for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$, we know that Y_i is unknown to any of $\{d_1, r\}$, even not flagged in $\mathsf{RL}_{\{r\}}$. As a result, whenever r receives the mixture $[Y_i + \overline{W}_j]$, Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. Moreover, $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$ is known by d_1. As a result, whenever d_1 receives the mixture, d_1 can use the known $\overline{W}_j$ and the received $[Y_i + \overline{W}_j]$ to decode Y_i and thus Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$. Similarly, one condition for $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$ is that $\overline{W}_j$ must be unknown to d_2. However when d_2 receives the mixture, d_2 can use the known $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and the received $[Y_i + \overline{W}_j]$ to decode $\overline{W}_j$. Thus $\overline{W}_j$ must be removed from $Q^{[2]}_{\{rd_1\}}$ whenever d_2 receives.

- **Insertion**: From the properties of $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$ and $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$, we know that r contains $\{X_i, \overline{W}_j\}$; d_1 contains $\{[X_i + Y_i], \overline{W}_j\}$; and d_2 contains Y_i already. Therefore, whenever d_1 receives this mixture, d_1 can use the known $\{[X_i + Y_i], \overline{W}_j\}$ and the received $[Y_i + \overline{W}_j]$ to extract the desired X_i and thus we can insert X_i into Q^1_{dec}. Similarly, whenever d_2 receives this mixture, d_2 can use the known Y_i and the received $[Y_i + \overline{W}_j]$ to extract $\overline{W}_j$. We now need to consider case by case when $\overline{W}_j$ was inserted into $Q^{[2]}_{\{rd_1\}}$. If it was the Case 1 insertion, then $\overline{W}_j$ is a pure session-2 packet Y_j and thus we can simply insert Y_j into Q^2_{dec}. If it was the Case 2 insertion, then $\overline{W}_j$ is a pure session-1 packet $X_j \in Q^1_{\mathsf{dec}}$ and there exists a session-2 packet Y_j still unknown to d_2 where $Y_j \equiv X_j$. Moreover, d_2 has received $[Y_j + X_j]$. As a result, d_2 can further decode Y_j and thus we can insert Y_j into Q^2_{dec}. If it was the Case 3 insertion, then $\overline{W}_j$ is a mixed form of $[W_i + W_j]$ where W_i is already known by d_2 but W_j is not. As a result, d_2 can decode W_j upon receiving $\overline{W}_j = [W_i + W_j]$. Note that W_j in the Case 3 insertion $\overline{W}_j = [W_i + W_j] \in Q^{[2]}_{\{rd_1\}}$ comes from either $Q^2_{\{d_1\}}$ or $Q^{(2)|2}_{\{d_1\}|\{r\}}$. If W_j was coming from $Q^2_{\{d_1\}}$, then W_j is a session-2 packet Y_j and we can simply insert Y_j into Q^2_{dec}. If W_j was coming from $Q^{(2)|2}_{\{d_1\}|\{r\}}$, then W_j is a session-1 packet X_j and there also exists a session-2 packet Y_j still unknown to d_2 where $Y_j \equiv X_j$. Moreover, d_2 has received $[Y_j + X_j]$. As a result, d_2 can further use the known $[Y_j + X_j]$ and the extracted X_j to decode Y_j and thus we can insert Y_j into Q^2_{dec}. In a nutshell, whenever d_2 receives the mixture $[Y_i + \overline{W}_j]$, a session-2 packet Y_j that was unknown to d_2 can be newly decoded. The remaining reception status are $\overline{d_1 d_2} r$ and $\overline{d_1} d_2 r$. For both cases when r receives the mixture but d_1 does not, r can use the known $\overline{W}_j$ and the received $[Y_i + \overline{W}_j]$ to extract Y_i. Since Y_i is now known by both r and d_2 but $[X_i + Y_i]$ is in $\mathsf{RL}_{\{d_1\}}$, we can thus move Y_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 2 insertion.

• $s_{\mathsf{CX};8}$: s transmits $[\overline{W}_i + X_j]$ from $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$ and $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$. The movement process is symmetric to $s_{\mathsf{CX};7}$.

• r^1_{UC}: r transmits X_i from $X_i \in Q^1_{\{r\}}$. The movement process is as follows.

$Q^1_{\{r\}} \xrightarrow{X_i}$	$\overline{d_1}d_2$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
	$d_1\overline{d_2}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$
	d_1d_2	

- **Departure**: One condition for $X_i \in Q^1_{\{r\}}$ is that X_i must be unknown to any of $\{d_1, d_2\}$. As a result, whenever X_i is received by any of $\{d_1, d_2\}$, X_i must be removed from $Q^1_{\{r\}}$.

- **Insertion**: From the above discussion, we know that X_i is unknown to d_1. As a result, whenever X_i is received by d_1, we can insert X_i to Q^1_{dec}. If X_i is received by d_2 but not by d_1, then X_i is now known by both d_2 and r but still unknown to d_1. This exactly falls into the first-case scenario of $Q^{[1]}_{\{rd_2\}}$ and thus we can move X_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 1 insertion.

• r^2_{UC}: r transmits Y_j from $Y_j \in Q^2_{\{r\}}$. The movement process is symmetric to r^1_{UC}.

• $r^{(1)}_{\mathsf{DT}}$: r transmits X_i that is known by r only and information equivalent from $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$. The movement process is as follows.

$Q^{(1)\|1}_{\{d_2\}\|\{r\}} \xrightarrow{Y_i}$	$\overline{d_1}d_2$	$\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$
	$d_1\overline{d_2}$	$\xrightarrow{X_i(\equiv Y_i)} Q^1_{\mathsf{dec}}$
	d_1d_2	

- **Departure**: From the property for $Y_i \in Q^{(1)|1}_{\{d_2\}|\{r\}}$, we know that there exists an information-equivalent session-1 packet X_i that is known by r but unknown to any of $\{d_1, d_2\}$. As a result, whenever X_i is received by any of $\{d_1, d_2\}$, Y_i must be removed from $Q^{(1)|1}_{\{d_2\}|\{r\}}$.

- **Insertion**: From the above discussion, we know that X_i is unknown to d_1 and thus we can insert X_i to Q^1_{dec} whenever X_i is received by d_1. If X_i is received by d_2 but not by d_1, then X_i is now known by both d_2 and r but still unknown to d_1. This exactly falls into the first-case scenario of $Q^{[1]}_{\{rd_2\}}$ and thus we can move X_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 1 insertion.

• $r^{(2)}_{\mathsf{DT}}$: r transmits Y_j that is known by r only and information equivalent from $X_j \in Q^{(2)|2}_{\{d_1\}|\{r\}}$. The movement process is symmetric to $r^{(1)}_{\mathsf{DT}}$.

• r_{RC}: r transmits W known by r for the packet of the form $[X_i + Y_j] : W \in Q_{\mathsf{mix}}$. The movement process is as follows.

$Q_{\mathsf{mix}} \xrightarrow{[X_i+Y_j]:W}$	$\overline{d_1}d_2$	either $\xrightarrow[\text{Case 1}]{X_i} Q^{[1]}_{\{rd_2\}}$ or $\xrightarrow[\text{Case 2}]{Y_j} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$
	$d_1\overline{d_2}$	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, either $\xrightarrow[\text{Case 1}]{Y_j} Q^{[2]}_{\{rd_1\}}$ or $\xrightarrow[\text{Case 2}]{X_i} Q^{[2]}_{\{rd_1\}}$
	d_1d_2	$\xrightarrow{X_i} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j} Q^2_{\mathsf{dec}}$

- **Departure**: From the conditions of $[X_i + Y_j] : W \in Q_{\mathsf{mix}}$, we know that Q_{mix} is designed to benefit both destinations simultaneously when r transmits W. That is, whenever d_1 (resp. d_2) receives W, d_1 (resp. d_2) can decode the desired X_i (resp. Y_j), regardless whether the packet W is of a session-1 or of a session-2. However from the conditions of Q_{mix}, we know that X_i is unknown to d_1 and Y_j is unknown to d_2. Therefore, whenever W is received by any of $\{d_1, d_2\}$, $[X_i + Y_j] : W$ must be removed from $Q^{[1]}_{\{rd_2\}}$.

- **Insertion**: From the above discussions, we know that d_1 (resp. d_2) can decode the desired X_i (resp. Y_j) when W is received by d_1 (resp. d_2). As a result, we can insert X_i into Q^1_{dec} (resp. Y_j into Q^2_{dec}) when d_1 (resp. d_2) receives W. We now consider two reception status $\overline{d_1}d_2$ and $d_1\overline{d_2}$. From the conditions of Q_{mix}, note that W is always known by r and can be either X_i or Y_j. Moreover, X_i (resp. Y_j) is unknown to d_1 (resp. d_2). For the first reception case $\overline{d_1}d_2$, if X_i was chosen as W to benefit both destinations, then X_i is now known by both d_2 and r but still unknown to d_1. This exactly falls into the first-case scenario of $Q^{[1]}_{\{rd_2\}}$ and thus we move X_i into $Q^{[1]}_{\{rd_2\}}$ as the Case 1 insertion. On the other hand, if Y_j was chosen as W to benefit both destinations, then we know that Y_j is now known by both d_2 and r, and that $[X_i + Y_j]$ is already in $\mathsf{RL}_{\{d_1\}}$. This exactly falls into the second-case scenario of $Q^{[1]}_{\{rd_2\}}$ and thus we can move $Y_j \in Q^2_{\mathsf{dec}}$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 2 insertion. The second reception case $d_1\overline{d_2}$ will follow the the previous arguments symmetrically.

- r_{XT}: r transmits $[W_i + W_j] \in Q^{m\mathsf{CX}}_{\{r\}}$. The movement process is as follows.

$Q^{m\mathsf{CX}}_{\{r\}} \xrightarrow{[W_i+W_j]}$	$\overline{d_1}d_2$	$\xrightarrow[\text{Case 3}]{[W_i+W_j]} Q^{[1]}_{\{rd_2\}}$, $\xrightarrow{Y_j(\equiv W_j)} Q^2_{\mathsf{dec}}$
	$d_1\overline{d_2}$	$\xrightarrow{X_i(\equiv W_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow[\text{Case 3}]{[W_i+W_j]} Q^{[2]}_{\{rd_1\}}$
	d_1d_2	$\xrightarrow{X_i(\equiv W_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j(\equiv W_j)} Q^2_{\mathsf{dec}}$

  - **Departure**: From the property for $[W_i+W_j] \in Q^{m\mathsf{CX}}_{\{r\}}$, we know that W_i is known only by d_2 and that W_j is only known by d_1. As a result, whenever d_1 receives this mixture, d_1 can use the known W_j and the received $[W_i + W_j]$ to extract W_i and thus the mixture must be removed from $Q^{m\mathsf{CX}}_{\{r\}}$. Similarly, whenever d_2 receives this mixture, d_2 can use the known W_i and the received $[W_i + W_j]$ to extract W_j and thus the mixture must be removed from $Q^{m\mathsf{CX}}_{\{r\}}$.

  - **Insertion**: From the above discussions, we have observed that whenever d_1 (resp. d_2) receives the mixture, d_1 (resp. d_2) can extract W_i (resp. W_j). From the four cases study of $Q^{m\mathsf{CX}}_{\{r\}}$, we know that d_1 (resp. d_2) can decode a desired session-1 packet X_i (resp. session-2 packet Y_j) whenever d_1 (resp. d_2) receives the mixture, and thus we can insert X_i (resp. Y_j) into Q^1_{dec} (resp. Q^2_{dec}). We now consider the reception status $\overline{d_1}d_2$ and $d_1\overline{d_2}$. If d_2 receives the mixture but d_1 does not, then d_1 contained W_j and d_2 now contains $[W_i + W_j]$. Moreover, $[W_i + W_j]$ was transmitted from r. This falls exactly into the third-case scenario of $Q^{[1]}_{\{rd_2\}}$. As a result, we can move $[W_i + W_j]$ into $Q^{[1]}_{\{rd_2\}}$ as the Case 3 insertion. The case when the reception status is $d_1\overline{d_2}$ can be symmetrically followed such that we can move $[W_i + W_j]$ into $Q^{[2]}_{\{rd_1\}}$ as the Case 3 insertion.

- $r^{[1]}_{\mathsf{DT}}$: r transmits $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$. The movement process is as follows.

do nothing	$\overline{d_1}d_2$	do nothing
$Q^{[1]}_{\{rd_2\}} \xrightarrow{\overline{W}_i}$	$d_1\overline{d_2}$, d_1d_2	$\xrightarrow{X_i(\equiv \overline{W}_i)} Q^1_{\mathsf{dec}}$

  - **Departure**: One condition for $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$ is that $\overline{W}_i$ is known by d_2 unknown to d_1. As a result, whenever d_1 receives, $\overline{W}_i$ must be removed from $Q^{[1]}_{\{rd_2\}}$. Since $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$ is already known by d_2, nothing happens if it is received by d_2.

- **Insertion**: From the previous observation, we only need to consider the reception status when d_1 receives $\overline{W}_i$. For those $d_1\overline{d_2}$ and d_1d_2, we need to consider case by case when $\overline{W}_i$ was inserted into $Q^{[1]}_{\{rd_2\}}$. If it was the Case 1 insertion, then $\overline{W}_i$ is a pure session-1 packet X_i and thus we can simply insert X_i into Q^1_{dec}. If it was the Case 2 insertion, then $\overline{W}_i$ is a pure session-2 packet $Y_i \in Q^2_{\mathsf{dec}}$ and there exists a session-1 packet X_i still unknown to d_1 where $X_i \equiv Y_i$. Moreover, d_1 has received $[X_i + Y_i]$. As a result, d_1 can further decode X_i and thus we can insert X_i into Q^1_{dec}. If it was the Case 3 insertion, then $\overline{W}_i$ is a mixed form of $[W_i + W_j]$ where W_j is already known by d_1 but W_i is not. As a result, d_1 can decode W_i upon receiving $\overline{W}_i = [W_i + W_j]$. Note that W_i in the Case 3 insertion $\overline{W}_i = [W_i + W_j] \in Q^{[1]}_{\{rd_2\}}$ comes from either $Q^1_{\{d_2\}}$ or $Q^{(1)|1}_{\{d_2\}|\{r\}}$. If W_i was coming from $Q^1_{\{d_2\}}$, then W_i is a session-1 packet X_i and we can simply insert X_i into Q^1_{dec}. If W_i was coming from $Q^{(1)|1}_{\{d_2\}|\{r\}}$, then W_i is a session-2 packet Y_i and there also exists a session-1 packet X_i still unknown to d_1 where $X_i \equiv Y_i$. Moreover, d_1 has received $[X_i + Y_i]$. As a result, d_1 can further use the known $[X_i + Y_i]$ and the extracted Y_i to decode X_i and thus we can insert X_i into Q^1_{dec}. In a nutshell, whenever d_1 receives $\overline{W}_i$, a session-1 packet X_i that was unknown to d_1 can be newly decoded.

- $r^{[2]}_{\mathsf{DT}}$: r transmits $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$. The movement process is symmetric to $r^{[1]}_{\mathsf{DT}}$.
- r_{CX}: r transmits $[\overline{W}_i + \overline{W}_j]$ from $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$ and $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$. The movement process is as follows.

$Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	$\overline{d_1}d_2$	$\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$
$Q^{[1]}_{\{rd_2\}} \xrightarrow{\overline{W}_i}$	$d_1\overline{d_2}$	$\xrightarrow{X_i(\equiv \overline{W}_i)} Q^1_{\mathsf{dec}}$
$Q^{[1]}_{\{rd_2\}} \xrightarrow{\overline{W}_i}$, $Q^{[2]}_{\{rd_1\}} \xrightarrow{\overline{W}_j}$	d_1d_2	$\xrightarrow{X_i(\equiv \overline{W}_i)} Q^1_{\mathsf{dec}}$, $\xrightarrow{Y_j(\equiv \overline{W}_j)} Q^2_{\mathsf{dec}}$

- **Departure**: From the property for $\overline{W}_i \in Q^{[1]}_{\{rd_2\}}$, we know that $\overline{W}_i$ is known by d_2 but unknown to d_1. Symmetrically, $\overline{W}_j \in Q^{[2]}_{\{rd_1\}}$ is known by d_1 but unknown to d_2. As result, whenever d_1 (resp. d_2) receives the mixture, d_1 (resp. d_2) can use the known $\overline{W}_j$ (resp. $\overline{W}_i$) and the received $[\overline{W}_i + \overline{W}_j]$ to extract $\overline{W}_i$ (resp.

$\overline{W}_j$). Therefore, we must remove $\overline{W}_i$ from $Q^{[1]}_{\{rd_2\}}$ whenever d_1 the mixture and remove $\overline{W}_j$ from $Q^{[2]}_{\{rd_1\}}$ whenever d_2 receives.

- **Insertion**: From the above discussions, we have observed that whenever d_1 (resp. d_2) receives the mixture, d_1 (resp. d_2) can extract $\overline{W}_i$ (resp. $\overline{W}_j$). We first focus on the case when d_1 receives the mixture. For those $d_1\overline{d_2}$ and d_1d_2, we can use the same arguments for $\overline{W}_i$ as described in the Insertion process of $r^{[1]}_{\mathsf{DT}}$. Following these case studies, one can see that a session-1 packet X_i that was unknown to d_1 can be newly decoded whenever d_1 receives $\overline{W}_i$. The reception status when d_2 receives the mixture can be followed symmetrically such that d_2 can always decode a new session-2 packet Y_j that was unknown before.

I. DETAILED DESCRIPTION OF ACHIEVABILITY SCHEMES IN FIG. 4.3

In the following, we describe (R_1, R_2) rate regions of each suboptimal achievability scheme used for the numerical evaluation in Section 4.3.

• **Intra-Flow Network Coding only:** The rate regions can be described by Proposition 4.2.1, if the variables $\{s^k_{\mathsf{PM1}}, s^k_{\mathsf{PM2}}, s^k_{\mathsf{RC}} : \text{for all } k \in \{1,2\}\}$, $\{s_{\mathsf{CX};l}\,(l\!=\!1,\cdots,8)\}$, $\{r_{\mathsf{RC}}, r_{\mathsf{XT}}, r_{\mathsf{CX}}\}$ are all hardwired to 0. Namely, we completely shut down all the variables dealing with cross-packet-mixtures. After such hardwirings, Proposition 4.2.1 is further reduced to the following form:

$$1 \geq \sum_{k\in\{1,2\}} \left(s^k_{\mathsf{UC}} + s^k_{\mathsf{DX}} + r^k_{\mathsf{UC}} + r^{[k]}_{\mathsf{DT}} \right),$$

and consider any $i, j \in (1, 2)$ satisfying $i \neq j$. For each (i, j) pair (out of the two choices $(1, 2)$ and $(2, 1)$),

$$\begin{aligned}
R_i &\geq s^i_{\mathsf{UC}} \cdot p_s(d_i, d_j, r), \\
s^i_{\mathsf{UC}} \cdot p_{s\to\overline{d_i d_j} r} &\geq r^i_{\mathsf{UC}} \cdot p_r(d_i, d_j), \\
s^i_{\mathsf{UC}} \cdot p_{s\to\overline{d_i} d_j \overline{r}} &\geq s^i_{\mathsf{DX}} \cdot p_s(d_i, r), \\
s^i_{\mathsf{UC}} \cdot p_{s\to\overline{d_i} d_j r} + s^i_{\mathsf{DX}} \cdot p_s(\overline{d_i} r) + r^i_{\mathsf{UC}} \cdot p_{r\to\overline{d_i} d_j} &\geq r^{[i]}_{\mathsf{DT}} \cdot p_r(d_i), \\
\left(s^i_{\mathsf{UC}} + s^i_{\mathsf{DX}} \right) \cdot p_s(d_i) + \left(r^i_{\mathsf{UC}} + r^{[i]}_{\mathsf{DT}} \right) \cdot p_r(d_i) &\geq R_i.
\end{aligned}$$

• **Always Relaying with NC:** This scheme requires that all the packets go through r, and then r performs 2-user broadcast channel NC. The corresponding rate regions can be described as follows:

$$\frac{R_1}{p_r(d_1)} + \frac{R_2}{p_r(d_1, d_2)} \leq 1 - \frac{R_1 + R_2}{p_s(r)},$$
$$\frac{R_1}{p_r(d_1, d_2)} + \frac{R_2}{p_r(d_2)} \leq 1 - \frac{R_1 + R_2}{p_s(r)}.$$

• **Always Relaying with routing:** This scheme requires that all the packets go through r as well, but r performs uncoded routing for the final delivery. The corresponding rate regions can be described as follows:

$$\frac{R_1}{p_r(d_1)} + \frac{R_2}{p_r(d_2)} \leq 1 - \frac{R_1 + R_2}{p_s(r)}.$$

• **[47] without Relaying:** This scheme completely ignores the relay r in the middle, and s just performs 2-user broadcast channel LNC of [47]. The corresponding rate regions can be described as follows:

$$\frac{R_1}{p_s(d_1)} + \frac{R_2}{p_s(d_1, d_2)} \leq 1,$$
$$\frac{R_1}{p_s(d_1, d_2)} + \frac{R_2}{p_s(d_2)} \leq 1.$$

• **Routing without Relaying:** This scheme completely ignores the relay r in the middle, and s just performs uncoded routing. The corresponding rate regions can be described as follows:

$$\frac{R_1}{p_s(d_1)} + \frac{R_2}{p_s(d_2)} \leq 1.$$

J. PROOFS OF PROPOSITIONS AND COROLLARIES FOR CHAPTER 5

J.1 Proofs of Propositions 5.4.1 and 5.4.2

We prove Proposition 5.4.1 as follows.

Proof of $\Rightarrow$ direction of Proposition 5.4.1: We prove this direction by contradiction. Suppose that $\mathbf{h}(\underline{\mathbf{x}})$ is linearly dependent. Then, there exists a set of coefficients $\{\alpha_k\}_{k=1}^N$ such that $\sum_{k=1}^N \alpha_k h_k(\underline{\mathbf{x}}) = 0$ and at least one of them is non-zero. Since $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N$ is row-invariant, we can perform elementary column operations on $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N$ using $\{\alpha_k\}_{k=1}^N$ to create an all-zero column. Thus, $\mathsf{det}([\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N)$ is a zero polynomial. ■

Proof of $\Leftarrow$ direction of Proposition 5.4.1: This direction is also proven by contradiction. Suppose that $\mathsf{det}([\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N)$ is a zero polynomial. We will prove that $\mathbf{h}(\underline{\mathbf{x}})$ is linearly dependent by induction on the value of N. For $N=1$, $\mathsf{det}([\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N)$ $= 0$ implies that $h_1(\underline{\mathbf{x}})$ is a zero polynomial, which by definition is linearly dependent.

Suppose that the statement holds for any $N < n_0$. When $N = n_0$, consider the (1,1)-th cofactor of $[\mathbf{h}(\underline{\mathbf{x}}^{(k)})]_{k=1}^N$, which is the determinant of the submatrix of the intersection of the 2nd to N-th rows and the 2nd to N-th columns. Consider the following two cases. Case 1: the $(1,1)$-th cofactor is a zero polynomial. Then by the induction assumption $\{h_2(\underline{\mathbf{x}}), ..., h_N(\underline{\mathbf{x}})\}$ is linearly dependent. By definition, so is $\mathbf{h}(\underline{\mathbf{x}})$. Case 2: the $(1,1)$-th cofactor is a non-zero polynomial. Since we assume a sufficiently large q, there exists an assignment $\hat{\mathbf{x}}_2 \in \mathbb{F}_q^{|\mathbf{x}|}$ to $\hat{\mathbf{x}}_N \in \mathbb{F}_q^{|\mathbf{x}|}$ such that the value of the (1,1)-th cofactor is non-zero when evaluated by $\hat{\mathbf{x}}_2$ to $\hat{\mathbf{x}}_N$. But note that by the Laplace expansion, we also have $\sum_{k=1}^N h_k(\underline{\mathbf{x}}^{(1)})\, C_{1k} = 0$ where C_{1k} is the $(1,k)$-th cofactor. By evaluating C_{1k} with $\{\hat{\mathbf{x}}_i\}_{i=2}^N$, we can conclude that $\mathbf{h}(\underline{\mathbf{x}})$ is linearly dependent since at least one of C_{1k} (specifically C_{11}) is non-zero. ■

We prove Proposition 5.4.2 as follows.

Proof of $\Leftarrow$ direction of Proposition 5.4.2: This can be proved by simply choosing $G' = G$. ■

Proof of $\Rightarrow$ direction of Proposition 5.4.2: Since we have $f(\{m_{e_i;e_i'}(\underline{\mathbf{x}}) : \forall i \in I\}) \equiv g(\{m_{e_i;e_i'}(\underline{\mathbf{x}}) : \forall i \in I\})$, we can assume $f(\{m_{e_i;e_i'}(\underline{\mathbf{x}}) : \forall i \in I\}) = \alpha g(\{m_{e_i;e_i'}(\underline{\mathbf{x}}) : \forall i \in I\})$ for some non-zero $\alpha \in \mathbb{F}_q$. Consider any subgraph G' containing all edges in $\{e_i, e_i' : \forall i \in I\}$ and the channel gain $m_{e_i;e_i'}(\underline{\mathbf{x}}')$ on G'. Then, $m_{e_i;e_i'}(\underline{\mathbf{x}}')$ can be derived from $m_{e_i;e_i'}(\underline{\mathbf{x}})$ by substituting those $\underline{\mathbf{x}}$ variables that are not in G' by zero. As a result, we immediately have $f(\{m_{e_i;e_i'}(\underline{\mathbf{x}}') : \forall i \in I\}) = \alpha g(\{m_{e_i;e_i'}(\underline{\mathbf{x}}') : \forall i \in I\})$ for the same α. The proof of this direction is thus complete. ■

J.2 Proofs of Corollaries 5.4.1 and 5.4.2

We prove Corollary 5.4.1 as follows.

Proof of $\Rightarrow$ direction of Corollary 5.4.1: We assume $(i_1, i_2) = (1, 2)$ and $(j_1, j_2) = (1, 3)$ without loss of generality. Since $\mathsf{EC}(\{s_1, s_2\}; \{d_1, d_3\}) = 1$, there exists an edge e^* that separates $\{d_1, d_3\}$ from $\{s_1, s_2\}$. Therefore, we must have $m_{11} = m_{e_{s_1};e^*} m_{e^*;e_{d_1}}$, $m_{13} = m_{e_{s_1};e^*} m_{e^*;e_{d_3}}$, $m_{21} = m_{e_{s_2};e^*}\ m_{e^*;e_{d_1}}$, and $m_{23} = m_{e_{s_2};e^*} m_{e^*;e_{d_3}}$. As a result, $m_{11} m_{23} \equiv m_{21} m_{13}$. ■

Proof of $\Leftarrow$ direction of Corollary 5.4.1: We prove this direction by contradiction. Suppose that $\mathsf{EC}(\{s_{i_1}, s_{i_2}\}; \{d_{j_1}, d_{j_2}\}) \geq 2$. In a $G_{3\text{ANA}}$ network, each source (resp. destination) has only one outgoing (resp. incoming) edge. Therefore, the assumption $\mathsf{EC}(\{s_{i_1}, s_{i_2}\}; \{d_{j_1}, d_{j_2}\}) \geq 2$ implies that at least one of the following two cases must be true: Case 1: There exists a pair of edge-disjoint paths $P_{s_{i_1} d_{j_1}}$ and $P_{s_{i_2} d_{j_2}}$; Case 2: There exists a pair of edge-disjoint paths $P_{s_{i_1} d_{j_2}}$ and $P_{s_{i_2} d_{j_1}}$. For Case 1, we consider the network variables that are along the two edge-disjoint paths, i.e., consider the collection $\underline{\mathbf{x}}'$ of network variables $x_{ee'} \in \underline{\mathbf{x}}$ such that either both e and e' are used by $P_{s_{i_1}d_{j_1}}$ or both e and e' are used by $P_{s_{i_2}d_{j_2}}$. We keep those variables in $\underline{\mathbf{x}}'$ intact and set the other network variables to be zero. As a result, we will have $m_{i_1j_1}(\underline{\mathbf{x}}')m_{i_2j_2}(\underline{\mathbf{x}}') = \prod_{\forall x_{ee'} \in \underline{\mathbf{x}}'} x_{ee'}$ and $m_{i_2j_1}(\underline{\mathbf{x}}')m_{i_1j_2}(\underline{\mathbf{x}}') = 0$ where the latter is due the edge-disjointness between two paths $P_{s_{i_1}d_{j_1}}$ and $P_{s_{i_2}d_{j_2}}$. This implies that before hardwiring the variables outside $\underline{\mathbf{x}}'$, we must have $m_{i_1j_1}(\underline{\mathbf{x}})m_{i_2j_2}(\underline{\mathbf{x}}) \not\equiv m_{i_2j_1}(\underline{\mathbf{x}})m_{i_1j_2}(\underline{\mathbf{x}})$. The proof of Case 1 is complete. Case 2 can be proven by swapping the labels of j_1 and j_2. ■

We prove Corollary 5.4.2 as follows.

Proof of Corollary 5.4.2: When $(i_1, j_1) = (i_2, j_2)$, obviously we have $m_{i_1j_1} = m_{i_2j_2}$ and $\mathsf{GCD}(m_{i_1j_1}, m_{i_2j_2}) \equiv m_{i_2j_2}$. Suppose that for some $(i_1, j_1) \neq (i_2, j_2)$, we have $\mathsf{GCD}(m_{i_1j_1}, m_{i_2j_2}) \equiv m_{i_2j_2}$. Without loss of generality, we assume $i_1 \neq i_2$. Since the channel gains are defined for two distinct sources, we must have $m_{i_1j_1} \not\equiv m_{i_2j_2}$. As a result, $\mathsf{GCD}(m_{i_1j_1}, m_{i_2j_2}) \equiv m_{i_2j_2}$ implies that $m_{i_1j_1}$ must be reducible. By Proposition 5.4.3, $m_{i_1j_1}$ must be expressed as $m_{i_1j_1} = m_{e_{s_{i_1}};e_1}\Big(\prod_{i=1}^{N-1} m_{e_i;e_{i+1}}\Big)m_{e_N;e_{d_{j_1}}}$ where each term corresponds to a pair of consecutive 1-edge cuts separating s_{i_1} and d_{j_1}. For $m_{i_1j_1}$ to contain $m_{i_2j_2}$ as a factor, the source edge $e_{s_{i_2}}$ must be one of the 1-edge cuts separating s_{i_1} and d_{j_1}. This contradicts the assumption that in a 3-unicast ANA network $|\mathsf{In}(s_i)| = 0$ for all i. The proof is thus complete. ■

J.3 Proof of Proposition 5.4.3

Proposition 5.4.3 will be proven through the concept of the line graph, which is defined as follows: The line graph of a DAG $G = (V, E)$ is represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with the vertex set $\mathcal{V} = E$ and edge set $\mathcal{E} = \{(e', e'') \in E^2 : \mathsf{head}(e') = \mathsf{tail}(e'')\}$ (the set representing the adjacency relationships between the edges of E). Provided that G is directed acyclic, its line graph $\mathcal{G}$ is also directed acyclic. The graph-theoretic notations for G defined in Section 5.1 are applied in the same way as in $\mathcal{G}$.

Note that the line graph translates the edges into vertices. Thus, a *vertex cut* in the line graph is the counterpart of the edge cut in a normal graph. Specifically, a

k-vertex cut separating vertex sets U and W is a collection of k vertices other than the vertices in U and W such that any path from any $u \in U$ to any $w \in W$ must use at least one of those k vertices. Moreover, the minimum value (number of vertices) of all the possible vertex cuts between vertex sets U and W is termed $\mathsf{VC}(U;W)$. For any nodes u and v in V, one can easily see that $\mathsf{EC}(u;v)$ in G is equal to $\mathsf{VC}(\tilde{u};\tilde{v})$ in $\mathcal{G}$ where $\tilde{u}$ and $\tilde{v}$ are the vertices in $\mathcal{G}$ corresponding to any incoming edge of u and any outgoing edge of v, respectively.

Once we focus on the line graph $\mathcal{G}$, the network variables $\underline{\mathbf{x}}$, originally defined over the (e', e'') pairs of the normal graph, are now defined on the edges of the line graph. We can thus define the channel gain from a vertex u to a vertex v on $\mathcal{G}$ as

$$\mathring{m}_{u;v} = \sum_{\forall\, P_{uv} \in \mathbf{P}_{uv}} \prod_{\forall\, e \in P_{uv}} x_e, \tag{J.1}$$

where $\mathbf{P}_{uv}$ denotes the collection of all distinct paths from u to v. For notational simplicity, we sometimes simply use "an edge e" to refer to the corresponding network variable x_e. Each x_e (or e) thus takes values in $\mathbb{F}_q$. When $u = v$, simply set $\mathring{m}_{u;v} = 1$.

The line-graph-based version of Proposition 5.4.3 is described as follows:

Corollary J.3.1. *Given the line graph $\mathcal{G}$ of a DAG G, $\mathring{m}$ defined above, and two distinct vertices s and d, the following is true:*

- *If $\mathsf{VC}(s;d) = 0$, then $\mathring{m}_{s;d} = 0$*
- *If $\mathsf{VC}(s;d) = 1$, then $\mathring{m}_{s;d}$ is reducible and can be expressed as*

$$\mathring{m}_{s;d} = \mathring{m}_{s;u_1} \left(\prod_{i=1}^{N-1} \mathring{m}_{u_i;u_{i+1}} \right) \mathring{m}_{u_N;d},$$

where $\{u_i\}_{i=1}^{N}$ are all the distinct 1-vertex cuts between s and d in the topological order (from the most upstream to the most downstream). Moreover, the polynomial factors $\mathring{m}_{s;u_1}$, $\{\mathring{m}_{u_i;u_{i+1}}\}_{i=1}^{N-1}$, and $\mathring{m}_{u_N;d}$ are all irreducible, and no two of them are equivalent.

- *If* $\mathsf{VC}(s;d)\geq 2$ *(including* ∞*), then* $\mathring{m}_{s;d}$ *is irreducible.*

Proof of Corollary J.3.1: We use the induction on the number of edges $|\mathcal{E}|$ of $\mathcal{G}=(\mathcal{V},\mathcal{E})$. When $|\mathcal{E}|=0$, then $\mathsf{VC}(s;d)=0$ since there are no edges in $\mathcal{G}$. Thus $\mathring{m}_{s;d}=0$ naturally.

Suppose that the above three claims are true for $|\mathcal{E}|=k-1$. We would like to prove that those claims also hold for the line graph $\mathcal{G}$ with $|\mathcal{E}|=k$.

Case 1: $\mathsf{VC}(s;d)=0$ on $\mathcal{G}$. In this case, s and d are already disconnected. Therefore, $\mathring{m}_{s;d}=0$.

Case 2: $\mathsf{VC}(s;d)=1$ on $\mathcal{G}$. Consider all distinct 1-vertex cuts $u_1,\cdots,u_N$ between s and d in the topological order. If we define $u_0\triangleq s$ and $u_{N+1}\triangleq d$, then we can express $\mathring{m}_{s;d}$ as $\mathring{m}_{s;d}=\prod_{i=0}^{N}\mathring{m}_{u_i;u_{i+1}}$. Since we considered all distinct 1-vertex cuts between s and d, we must have $\mathsf{VC}(u_i;u_{i+1})\geq 2$ for $i=0,\cdots,N$. By induction, $\{\mathring{m}_{u_i;u_{i+1}}\}_{i=0}^{N}$ are all irreducible. Also, since each sub-channel gain $\mathring{m}_{u_i;u_{i+1}}$ covers a disjoint portion of $\mathcal{G}$, no two of them are equivalent.

Case 3: $\mathsf{VC}(s;d)\geq 2$ on $\mathcal{G}$. Without loss of generality, we can also assume that s can reach any vertex $u\in\mathcal{V}$ and d can be reached from any vertex $u\in\mathcal{V}$. Consider two subcases: Case 3.1: all edges in $\mathcal{E}$ have their tails being s and their heads being d. In this case, $\mathring{m}_{s;d}=\sum_{e\in\mathcal{E}}x_e$. Obviously $\mathring{m}_{s;d}$ is irreducible. Case 3.2: at least one edge in $\mathcal{E}$ is not directly connecting s and d. In this case, there must exist an edge e' such that $s\prec\mathsf{tail}(e')$ and $\mathsf{head}(e')=d$. Arbitrarily pick one such edge e' and fix it. We denote the tail vertex of the chosen e' by w. By the definition of (J.1), we have

$$\mathring{m}_{s;d}=\mathring{m}_{s;w}x_{e'}+\mathring{m}'_{s;d}, \tag{J.2}$$

where $\mathring{m}_{s;w}$ is the channel gain from s to w, and $\mathring{m}'_{s;d}$ is the channel gain from s to d on the subgraph $\mathcal{G}'=\mathcal{G}\backslash\{e'\}$ that removes e' from $\mathcal{G}$. Note that there always exists a path from s to d not using w on $\mathcal{G}'$ otherwise w will be a cut separating s and d on $\mathcal{G}$, contradicting the assumption that $\mathsf{VC}(s;d)\geq 2$.

We now argue by contradiction that $\mathring{m}_{s;d}$ must be irreducible. Suppose not, then $\mathring{m}_{s;d}$ can be written as a product of two polynomials A and B with the degrees of A and B being larger than or equal to 1. We can always write $A = x_{e'}A_1 + A_2$ by singling out the portion of A that has $x_{e'}$ as a factor. Similarly we can write $B = x_{e'}B_1 + B_2$. We then have

$$\mathring{m}_{s;d} = (x'_e A_1 + A_2)(x'_e B_1 + B_2). \tag{J.3}$$

We first notice that by (J.2) there is no quadratic term of $x_{e'}$ in $\mathring{m}_{s;d}$. Therefore, one of A_1 and B_1 must be a zero polynomial. Assume $B_1 = 0$. Comparing (J.2) and (J.3) shows that $\mathring{m}_{s;w} = A_1 B_2$ and $\mathring{m}'_{s;d} = A_2 B_2$. Since the degree of B is larger than or equal to 1 and $B_1 = 0$, the degree of B_2 must be larger than equal to 1. As a result, we have $\mathsf{GCD}(\mathring{m}_{s;w}, \mathring{m}'_{s;d}) \not\equiv 1$ (having at least a non-zero polynomial B_2 as its common factor).

The facts that $\mathsf{GCD}(\mathring{m}_{s;w}, \mathring{m}'_{s;d}) \not\equiv 1$ and $w \prec d$ imply that one of the following three cases must be true: (i) Both $\mathring{m}_{s;w}$ and $\mathring{m}'_{s;d}$ are reducible; (ii) $\mathring{m}_{s;w}$ is reducible but $\mathring{m}'_{s;d}$ is not; and (iii) $\mathring{m}'_{s;d}$ is reducible but $\mathring{m}_{s;w}$ is not. For Case (i), by applying Proposition 5.4.3 to the subgraph $\mathcal{G}' = \mathcal{G} \backslash \{e'\}$, we know that $\mathsf{VC}(s;w) = \mathsf{VC}(s;d) = 1$ and both polynomials $\mathring{m}_{s;w}$ and $\mathring{m}'_{s;d}$ can be factorized according to their 1-vertex cuts, respectively. Since $\mathring{m}_{s;w}$ and $\mathring{m}'_{s;d}$ have a common factor, there exists a vertex u that is both a 1-vertex cut separating s and w and a 1-vertex cut separating s and d when focusing on $\mathcal{G}'$. As a result, such u is a 1-vertex cut separating s and d in the original graph $\mathcal{G}$. This contradicts the assumption $\mathsf{VC}(s;d) \geq 2$ in $\mathcal{G}$. For Case (ii), by applying Proposition 5.4.3 to $\mathcal{G}'$, we know that $\mathsf{VC}(s;w) = 1$ and $\mathring{m}_{s;w}$ can be factorized according to their 1-vertex cuts. Since $\mathring{m}_{s;w}$ and the irreducible $\mathring{m}'_{s;d}$ have a common factor, $\mathring{m}_{s;w}$ must contain $\mathring{m}'_{s;d}$ as a factor, which implies that d is a 1-vertex cut separating s and w in $\mathcal{G}'$. This contradicts the construction of $\mathcal{G}'$ where $w \prec d$. For Case (iii), by applying Proposition 5.4.3 to $\mathcal{G}'$, we know that $\mathsf{VC}(s;d) = 1$ and $\mathring{m}'_{s;d}$ can be factorized according to their 1-vertex cuts. Since $\mathring{m}'_{s;d}$ and the irreducible

$\mathring{m}_{s;w}$ have a common factor, $\mathring{m}'_{s;d}$ must contain $\mathring{m}_{s;w}$ as a factor, which implies that w is a 1-vertex cut separating s and d in $\mathcal{G}'$. As a result, w is a 1-vertex cut separating s and d in the original graph $\mathcal{G}$. This contradicts the assumption $\mathsf{VC}(s;d) \geq 2$ in $\mathcal{G}$. ■

K. PROOFS OF LEMMAS FOR CHAPTER 6

We prove Lemma 6.1.1 as follows.

Proof of Lemma 6.1.1: Consider indices $i \neq j$. By the definition, all paths from s_i to d_j must use all edges in $\overline{S}_i$ and all edges in $\overline{D}_j$. Thus, for any $e' \in \overline{S}_i$ and any $e'' \in \overline{D}_j$, one of the following statements must be true: $e' \prec e''$, $e' \succ e''$, or $e' = e''$. ■

We prove Lemma 6.1.2 as follows.

Proof of Lemma 6.1.2: Consider three indices i, j, and k taking distinct values in $\{1, 2, 3\}$. Consider an arbitrary edge $e \in \overline{D}_i \cap \overline{D}_j$. By definition, all paths from s_k to d_i, and all paths from s_k to d_j must use e. Therefore, $e \in \overline{S}_k$. ■

We prove Lemma 6.1.3 as follows.

Proof of Lemma 6.1.3: Without loss of generality, let $i=1$ and $j=2$. Choose the most downstream edge in $\overline{S}_1 \backslash \overline{D}_2$ and denote it as e'_*. Since e'_* belongs to $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(s_1; d_3)$ but not to $\mathsf{1cut}(s_3; d_2)$, there must exist as_3-to-$d_2$path P_{32} not using e'_*. In addition, for any $e'' \in \overline{D}_2$, we have either $e'' \prec e'_*$, $e'' \succ e'_*$, or $e'' = e'_*$ by Lemma 6.1.1. Suppose there exists an edge $e'' \in \overline{D}_2$ such that $e'' \prec e'_*$. Then by definition, anys_3-to-d_2 path must use e''. Also note that since $e'' \in \overline{D}_2$, there exists a path $P_{s_1\mathsf{tail}(e'')}$ from s_1 to $\mathsf{tail}(e'')$. Consider the concatenateds_1-to-$d_2$path $P_{s_1\mathsf{tail}(e'')}e''P_{32}$. We first note that since $e'' \prec e'_*$, the path segment $P_{s_1\mathsf{tail}(e'')}e''$ does not use e'_*. By our construction, P_{32} also does not use e'_*. Jointly, the above observations contradict the fact that $e'_* \in \overline{S}_1$ is a 1-edge cut separating s_1 and d_2. By contradiction, we must have $e'_* \preceq e''$. Note that since by our construction e'_* must not be in $\overline{D}_2$ while e'' is in $\overline{D}_2$, we must have $e'_* \neq e''$ and thus $e'_* \prec e''$. Since e'_* was chosen as the most downstream edge of $\overline{S}_1 \backslash \overline{D}_2$, we have $e' \prec e''$ for all $e' \in \overline{S}_1 \backslash \overline{D}_2$ and $e'' \in \overline{D}_2$. The proof is thus complete. ■

We prove Lemma 6.1.4 as follows.

Proof of $\Rightarrow$ direction of Lemma 6.1.4: We note that $(\overline{S}_i \cap \overline{D}_j) \supset (\overline{S}_i \cap \overline{D}_j \cap \overline{D}_k) = (\overline{D}_j \cap \overline{D}_k)$ where the equality follows from Lemma 6.1.2. As a result, when $\overline{D}_j \cap \overline{D}_k \neq \emptyset$, we also have $\overline{S}_i \cap \overline{D}_j \neq \emptyset$. ■

Proof of $\Leftarrow$ direction of Lemma 6.1.4: Consider three indices i, j, and k taking distinct values in $\{1,2,3\}$. Suppose that $\overline{S}_i \cap \overline{D}_j \neq \emptyset$ and $\overline{S}_i \cap \overline{D}_k \neq \emptyset$. Then, for any $e' \in \overline{S}_i \cap \overline{D}_j$ and any $e'' \in \overline{S}_i \cap \overline{D}_k$, we must have either $e' \prec e''$, $e' \succ e''$, or $e' = e''$ by Lemma 6.1.1. Suppose that $\overline{D}_j \cap \overline{D}_k = \emptyset$. Then we must have $e' \neq e''$, which leaves only two possibilities: either $e' \prec e''$ or $e' \succ e''$. However, $e' \prec e''$ contradicts Lemma 6.1.3 because $e' \in (\overline{S}_i \cap \overline{D}_j) \subset \overline{D}_j$ and $e'' \in (\overline{S}_i \cap \overline{D}_k) \subset (\overline{S}_i \backslash \overline{D}_j)$, the latter of which is due to the assumption of $\overline{D}_j \cap \overline{D}_k = \emptyset$. By swapping the roles of j and k, one can also show that it is impossible to have $e' \succ e''$. By contradiction, we must have $\overline{D}_j \cap \overline{D}_k \neq \emptyset$. The proof is thus complete. ■

We prove Lemma 6.1.5 as follows.

Proof of Lemma 6.1.5: Without loss of generality, consider $i = 1$ and $j = 2$. Note that by Lemma 6.1.1 any $e' \in \overline{S}_1 \cap \overline{S}_2$ and any $e'' \in \overline{D}_1 \cap \overline{D}_2$ must satisfy either $e' \prec e''$, $e' \succ e''$, or $e' = e''$. For the following, we prove this lemma by contradiction.

Suppose that there exists an edge $e''_* \in \overline{D}_1 \cap \overline{D}_2$ such that for all $e' \in \overline{S}_1 \cap \overline{S}_2$ we have $e''_* \prec e'$. For the following, we first prove that any path from s_i to d_j where $i, j \in \{1,2,3\}$ and $i \neq j$ must pass through e''_*. To that end, we first notice that by the definition of $\overline{D}_1$ and $\overline{D}_2$ and by the assumption $e''_* \in \overline{D}_1 \cap \overline{D}_2$, any path from $\{s_2, s_3\}$ to d_1, and any path from $\{s_1, s_3\}$ to d_2 must use e''_*. Thus, we only need to prove that any path from $\{s_1, s_2\}$ to d_3 must use e''_* as well.

Suppose there exists as_1-to-$d_3$path P_{13} that does not use e''_*. By the definition of $\overline{S}_1$, P_{13} must use all edges of $\overline{S}_1 \cap \overline{S}_2$, all of which are in the downstream of e''_* by the assumption. Also d_2 is reachable from any $e' \in \overline{S}_1 \cap \overline{S}_2$. Choose arbitrarily one edge $e'_* \in \overline{S}_1 \cap \overline{S}_2$ and a path $P_{\mathsf{head}(e'_*)d_2}$ from $\mathsf{head}(e'_*)$ to d_2. Then, we can create an path

$P_{13}\, e'_*\, P_{\mathsf{head}(e'_*)d_2}$ from s_1 to d_2 without using e''_*. The reason is that P_{13} does not use e''_* by our construction and $e'_* P_{\mathsf{head}(e'_*)d_2}$ does not use e''_* since $e''_* \prec e'_*$. Such ans_1-to-d_2 path not using e''_* thus contradicts the assumption of $e''_* \in (\overline{D}_1 \cap \overline{D}_2) \subset \mathsf{1cut}(s_1; d_2)$. Symmetrically, any$s_2$-to-$d_3$path must use e''_*.

In summary, we have shown that $e''_* \in \cap_{i=1}^{3} \left(\overline{S}_i \cap \overline{D}_i\right)$. However, this contradicts the assumption that e''_* is in the upstream of all $e' \in \overline{S}_1 \cap \overline{S}_2$, because we can simply choose $e' = e''_* \in \cap_{i=1}^{3} \left(\overline{S}_i \cap \overline{D}_i\right) \subset (\overline{S}_1 \cap \overline{S}_2)$ and e''_* cannot be an upstream edge of itself $e' = e''_*$. The proof is thus complete. ■

We prove Lemma 6.1.6 as follows.

Proof of Lemma 6.1.6: Without loss of generality, let $i = 1$, $j_1 = 1$, $j_2 = 2$, and $j_3 = 3$. Suppose that $\overline{S}_{1;\{1,2\}} \neq \emptyset$ and $\overline{S}_{1;\{1,3\}} \neq \emptyset$. For the following, we prove this lemma by contradiction.

Suppose that $\overline{S}_{1;\{1,2\}} \cap \overline{S}_{1;\{1,3\}} = \emptyset$. For any $e' \in \overline{S}_{1;\{1,2\}}$ and any $e'' \in \overline{S}_{1;\{1,3\}}$, since both e' and e'' are 1-edge cuts separating s_1 and d_1, it must be either $e' \prec e''$ or $e' \succ e''$, or $e' = e''$. The last case is not possible since we assume $\overline{S}_{1;\{1,2\}} \cap \overline{S}_{1;\{1,3\}} = \emptyset$. Consider the most downstream edges $e'_* \in \overline{S}_{1;\{1,2\}}$ and $e''_* \in \overline{S}_{1;\{1,3\}}$, respectively. We first consider the case $e'_* \prec e''_*$. If all paths from s_1 to d_3 use e'_*, which, by definition, use e''_*, then e'_* will belong to $\mathsf{1cut}(s_1; d_3)$, which contradicts the assumption that $\overline{S}_{1;\{1,2\}} \cap \overline{S}_{1;\{1,3\}} = \emptyset$. Thus, there exists a$s_1$-to-$d_3$path P_{13} using e''_* but not e'_*. Then, s_1 can follow P_{13} and reach d_1 via e''_* without using e'_*. Such as_1-to-$d_1$path contradicts the definition $e'_* \in \overline{S}_{1;\{1,2\}} \subset \mathsf{1cut}(s_1; d_1)$. Therefore, it is impossible to have $e'_* \prec e''_*$. By symmetric arguments, it is also impossible to have $e'_* \succ e''_*$. By definition, any edge in $\overline{S}_{1;\{1,2\}} \cap \overline{S}_{1;\{1,3\}}$ is a 1-edge cut separating s_1 and $\{d_2, d_3\}$, which implies that $\overline{S}_{1;\{2,3\}} \neq \emptyset$ and $\overline{S}_1 \neq \emptyset$. ■

We prove Lemma 6.1.7 as follows.

Proof of $\Rightarrow$ direction of Lemma 6.1.7: Suppose $\overline{S}_{i;\{j_1,j_2\}} \neq \emptyset$. By definition, there exists an edge $e \in \mathsf{1cut}(s_i; d_{j_1}) \cap \mathsf{1cut}(s_i; d_{j_2})$ in the downstream of the s_i-source edge

e_{s_i}. Then, the channel gains m_{ij_1} and m_{ij_2} have a common factor $m_{e_{s_i};e}$ and we thus have $\mathsf{GCD}(\,m_{ij_1},\,m_{ij_2})\not\equiv 1$. ■

Proof of ⇐ direction of Lemma 6.1.7: We prove this direction by contradiction. Suppose $\mathsf{GCD}(\,m_{ij_1}, m_{ij_2})\not\equiv 1$. By Corollary 5.4.2, we know that $\mathsf{GCD}(\,m_{ij_1},\,m_{ij_2})$ must not be m_{ij_1} nor m_{ij_2}. Thus, both must be reducible and by Proposition 5.4.3 can be expressed as the product of irreducibles, for which each factor corresponds to the consecutive 1-edge cuts in $\mathsf{1cut}(s_i; d_{j_1})$ and $\mathsf{1cut}(s_i; d_{j_2})$, respectively. Since they have at least one common irreducible factor, there exists an edge $e \in \mathsf{1cut}(s_i; d_{j_1}) \cap \mathsf{1cut}(s_i; d_{j_2})$ in the downstream of the s_i-source edge e_{s_i}. Thus, $e \in \overline{S}_{i;\{j_1,j_2\}}$. The case for $\mathsf{GCD}(\,m_{j_1 i},\,m_{j_2 i})\equiv 1$ can be proven symmetrically. The proof is thus complete. ■

L. THE REFERENCE TABLE FOR THE PROOF OF PROPOSITION 6.3.1

Table L.1: The reference table for the proof of Proposition 6.3.1

The Logic Statements for the Proof of Proposition 6.3.1

C0 to **C6**	defined in p. 224.	**G7** to **G15**	defined in p. 234.
D1 to **D6**	defined in p. 225.	**G16** to **G26**	defined in p. 246.
E0 to **E2**	defined in p. 223.	**G27** to **G31**	defined in p. 265.
G0	defined in p. 213.	**G32** to **G36**	defined in p. 270.
G1, **G2**	defined in p. 100.	**G37** to **G43**	defined in p. 274.
G3, **G4**	defined in p. 213.	**H1**, **H2**, **K1**, **K2**	defined in p. 100.
G5, **G6**	defined in p. 215.	**LNR**	defined in p. 100.

The Logic Relationships for the Proof of Proposition 6.3.1

N1 to **N9**	defined in p. 215, to help proving Corollary M.2.1, the general structured proof for the necessity of Proposition 6.3.1.
R1 to **R10**	defined in p. 235, to help proving **S11**.
R11 to **R25**	defined in p. 247, to help proving **S13**.
R26 to **R33**	defined in p. 266, to help proving **S14**.
R34 to **R40**	defined in p. 270, to help proving **R28**.
R41 to **R47**	defined in p. 275, to help proving **R29**.
S1 to **S14**	defined in p. 225, to help proving Corollary N.2.1, the general structured proof for the sufficiency of Proposition 6.3.1.

For the ease of exposition, we provide the Table L.1, the reference table. The reference table helps finding where to look for the individual logic statements and relationships for the entire proof of Proposition 6.3.1.

M. GENERAL STRUCTURED PROOF FOR THE NECESSITY OF PROPOSITION 6.3.1

In this appendix, we provide Corollary M.2.1, which will be used to prove the graph-theoretic necessary direction of 3-unicast ANA network for arbitrary n values. Since we already provided the proof for "$\mathbf{LNR} \wedge \mathbf{G1} \Leftarrow \mathbf{H1}$" in Proposition 6.3.1, here we focus on proving "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Leftarrow \mathbf{H2}$, $\mathbf{K1}$, $\mathbf{K2}$". After introducing Corollary M.2.1, the main proof of "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Leftarrow \mathbf{H2}$, $\mathbf{K1}$, $\mathbf{K2}$" will be provided in Appendix M.3.

Before proceeding, we need the following additional logic statements to describe the general proof structure.

M.1 The first set of logic statements

Consider the following logic statements.

- **G0:** $m_{11}m_{23}m_{32} = R + L$.
- **G3:** $\overline{S}_2 \cap \overline{D}_3 = \emptyset$.
- **G4:** $\overline{S}_3 \cap \overline{D}_2 = \emptyset$.

Several implications can be made when **G3** is true. We term those implications *the properties of* **G3**. Several properties of **G3** are listed as follows, for which their proofs are provided in Appendix M.4.

Consider the case in which **G3** *is true.* Use e_2^* to denote the most downstream edge in $\mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_2; d_3)$. Since the source edge e_{s_2} belongs to both $\mathsf{1cut}(s_2; d_1)$ and $\mathsf{1cut}(s_2; d_3)$, such e_2^* always exists. Similarly, use e_3^* to denote the most upstream edge in $\mathsf{1cut}(s_1; d_3) \cap \mathsf{1cut}(s_2; d_3)$. The properties of **G3** can now be described as follows.

$\diamond$ **Property 1 of G3:** $e_2^* \prec e_3^*$ and the channel gains m_{13}, m_{21}, and m_{23} can be expressed as $m_{13} = m_{e_{s_1};e_3^*} m_{e_3^*;e_{d_3}}$, $m_{21} = m_{e_{s_2};e_2^*} m_{e_2^*;e_{d_1}}$, and $m_{23} = m_{e_{s_2};e_2^*} m_{e_2^*;e_3^*} m_{e_3^*;e_{d_3}}$.
$\diamond$ **Property 2 of G3:** $\mathsf{GCD}(m_{e_{s_1};e_3^*},\ m_{e_{s_2};e_2^*} m_{e_2^*;e_3^*}) \equiv 1$, $\mathsf{GCD}(m_{e_2^*;e_3^*} m_{e_3^*;e_{d_3}},\ m_{e_2^*;e_{d_1}}) \equiv 1$, $\mathsf{GCD}(m_{13},\ m_{e_2^*;e_3^*}) \equiv 1$, and $\mathsf{GCD}(m_{21},\ m_{e_2^*;e_3^*}) \equiv 1$.

On the other hand, when **G3** is false, or equivalently when $\neg$ **G3** is true where "$\neg$" is the NOT operator, we can also derive several implications, termed *the properties of* $\neg$ **G3**.

Consider the case in which **G3** *is false.* Use e_u^{23} (resp. e_v^{23}) to denote the most upstream (resp. the most downstream) edge in $\overline{S}_2 \cap \overline{D}_3$. By definition, it must be $e_u^{23} \preceq e_v^{23}$. We now describe the following properties of $\neg$ **G3**.
$\diamond$ **Property 1 of $\neg$ G3:** The channel gains m_{13}, m_{21}, and m_{23} can be expressed as $m_{13} = m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}}$, $m_{21} = m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_1}}$, and $m_{23} = m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}}$.
$\diamond$ **Property 2 of $\neg$ G3:** $\mathsf{GCD}(m_{e_{s_1};e_u^{23}},\ m_{e_{s_2};e_u^{23}}) \equiv 1$ and $\mathsf{GCD}(m_{e_v^{23};e_{d_1}},\ m_{e_v^{23};e_{d_3}}) \equiv 1$.
$\diamond$ **Property 3 of $\neg$ G3:** $\{e_u^{23}, e_v^{23}\} \subset \mathsf{1cut}(s_1; \mathsf{head}(e_v^{23}))$ and $\{e_u^{23}, e_v^{23}\} \subset \mathsf{1cut}(\mathsf{tail}(e_u^{23}); d_1)$. This further implies that for any s_1-to-d_1 path P, if there exists a vertex $w \in P$ satisfying $\mathsf{tail}(e_u^{23}) \preceq w \preceq \mathsf{head}(e_v^{23})$, then we must have $\{e_u^{23}, e_v^{23}\} \subset P$.

Symmetrically, we define the following properties of **G4** and $\neg$ **G4**.

Consider the case in which **G4** *is true.* Use e_3^* to denote the most downstream edge in $\mathsf{1cut}(s_3; d_1) \cap \mathsf{1cut}(s_3; d_2)$, and use e_2^* to denote the most upstream edge in $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(s_3; d_2)$. We now describe the following properties of **G4**.
$\diamond$ **Property 1 of G4:** $e_3^* \prec e_2^*$ and the channel gains m_{12}, m_{31}, and m_{32} can be expressed as $m_{12} = m_{e_{s_1};e_2^*} m_{e_2^*;e_{d_2}}$, $m_{31} = m_{e_{s_3};e_3^*} m_{e_3^*;e_{d_1}}$, and $m_{32} = m_{e_{s_3};e_3^*} m_{e_3^*;e_2^*} m_{e_2^*;e_{d_2}}$.
$\diamond$ **Property 2 of G4:** $\mathsf{GCD}(m_{e_{s_1};e_2^*},\ m_{e_{s_3};e_3^*} m_{e_3^*;e_2^*}) \equiv 1$, $\mathsf{GCD}(m_{e_3^*;e_2^*} m_{e_2^*;e_{d_2}},\ m_{e_3^*;e_{d_1}}) \equiv 1$, $\mathsf{GCD}(m_{12},\ m_{e_3^*;e_2^*}) \equiv 1$, and $\mathsf{GCD}(m_{31},\ m_{e_3^*;e_2^*}) \equiv 1$.

Consider the case in which **G4** *is false.* Use e_u^{32} (resp. e_v^{32}) to denote the most upstream (resp. the most downstream) edge in $\overline{S}_3 \cap \overline{D}_2$. By definition, it must be $e_u^{32} \preceq e_v^{32}$. We now describe the following properties of $\neg$ **G4**.

⋄ **Property 1 of ¬ G4:** The channel gains m_{12}, m_{31}, and m_{32} can be expressed as $m_{12} = m_{e_{s_1};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_2}}$, $m_{31} = m_{e_{s_3};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_1}}$, and $m_{32} = m_{e_{s_3};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{23};e_{d_2}}$.

⋄ **Property 2 of ¬ G4:** $\mathsf{GCD}(m_{e_{s_1};e_u^{32}}, m_{e_{s_3};e_u^{32}}) \equiv 1$ and $\mathsf{GCD}(m_{e_v^{32};e_{d_1}}, m_{e_v^{32};e_{d_2}}) \equiv 1$

⋄ **Property 3 of ¬ G4:** $\{e_u^{32}, e_v^{32}\} \subset \mathsf{1cut}(s_1; \mathsf{head}(e_v^{32}))$ and $\{e_u^{32}, e_v^{32}\} \subset \mathsf{1cut}(\mathsf{tail}(e_u^{32}); d_1)$. This further implies that for any s_1-to-d_1 path P, if there exists a vertex $w \in P$ satisfying $\mathsf{tail}(e_u^{32}) \preceq w \preceq \mathsf{head}(e_v^{32})$, then we must have $\{e_u^{32}, e_v^{32}\} \subset P$.

The following logic statements are well-defined if and only if $(\neg\,$**G3**$) \wedge (\neg\,$**G4**$)$ is true. Recall the definition of e_u^{23}, e_v^{23}, e_u^{32}, and e_v^{32} when $(\neg\,$**G3**$) \wedge (\neg\,$**G4**$)$ is true.

- **G5:** Either $e_u^{23} \prec e_u^{32}$ or $e_u^{23} \succ e_u^{32}$.
- **G6:** Any vertex w' where $\mathsf{tail}(e_u^{23}) \preceq w' \preceq \mathsf{head}(e_v^{23})$ and any vertex w'' where $\mathsf{tail}(e_u^{32}) \preceq w'' \preceq \mathsf{head}(e_v^{32})$ are not reachable from each other. (That is, neither $w' \preceq w''$ nor $w'' \preceq w'$.)

It is worth noting that a *statement* being well-defined does not mean that it is true. Any well-defined logic statement can be either true or false. For comparison, a *property* of **G3** is both well-defined and true whenever **G3** is true.

M.2 General Necessity Proof Structure

The following "logic relationships" are proved in Appendix M.5, which will be useful for the proof of the following Corollary M.2.1.

- **N1: H2** $\Rightarrow$ **LNR** $\wedge$ **G1**.
- **N2: K1** $\Rightarrow$ **LNR** $\wedge$ **G1**.
- **N3: K2** $\Rightarrow$ **LNR** $\wedge$ **G1**.
- **N4:** $(\neg\,$**G2**$) \wedge$ **G3** $\wedge$ **G4** $\Rightarrow$ false.
- **N5: G1** $\wedge (\neg\,$**G2**$) \wedge (\neg\,$**G3**$) \wedge$ **G4** $\Rightarrow$ false.
- **N6: G1** $\wedge (\neg\,$**G2**$) \wedge$ **G3** $\wedge (\neg\,$**G4**$) \Rightarrow$ false.
- **N7: LNR** $\wedge (\neg\,$**G3**$) \wedge (\neg\,$**G4**$) \wedge (\neg\,$**G5**$) \Rightarrow$ **G6**.
- **N8: G1** $\wedge (\neg\,$**G2**$) \wedge (\neg\,$**G3**$) \wedge (\neg\,$**G4**$) \wedge$ **G5** $\Rightarrow$ false.

• **N9:** $(\neg\,\mathbf{G2})\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\neg\,\mathbf{G5})\wedge\mathbf{G6}\Rightarrow\mathbf{G0}$.

Corollary M.2.1. *Let $\mathbf{h}(\underline{\mathbf{x}})$ be a set of (arbitrarily chosen) polynomials based on the 9 channel gains m_{ij} of the 3-unicast ANA network, and define $\mathbf{X}$ to be the logic statement that $\mathbf{h}(\underline{\mathbf{x}})$ is linearly independent. If we can prove that $\mathbf{X}\Rightarrow\mathbf{LNR}\wedge\mathbf{G1}$ and $\mathbf{X}\wedge\mathbf{G0}\Rightarrow$ false, then the logic relationship $\mathbf{X}\Rightarrow\mathbf{LNR}\wedge\mathbf{G1}\wedge\mathbf{G2}$ must hold.*

Proof of Corollary M.2.1: Suppose $\mathbf{X}\Rightarrow\mathbf{LNR}\wedge\mathbf{G1}$ and $\mathbf{X}\wedge\mathbf{G0}\Rightarrow$ false. We first see that **N7** and **N9** jointly imply

$$\mathbf{LNR}\wedge(\neg\,\mathbf{G2})\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\neg\,\mathbf{G5})\Rightarrow\ \mathbf{G0}.$$

Combined with **N8**, we thus have

$$\mathbf{LNR}\wedge\mathbf{G1}\wedge(\neg\,\mathbf{G2})\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\Rightarrow\ \mathbf{G0}.$$

This, jointly with **N4**, **N5**, and **N6**, further imply

$$\mathbf{LNR}\wedge\mathbf{G1}\wedge(\neg\,\mathbf{G2})\Rightarrow\ \mathbf{G0}.$$

Together with the assumption that $\mathbf{X}\wedge\mathbf{G0}\Rightarrow$ false, we have $\mathbf{X}\wedge\mathbf{LNR}\wedge\mathbf{G1}\wedge(\neg\,\mathbf{G2})\Rightarrow$ false. Combining with the assumption that $\mathbf{X}\Rightarrow\mathbf{LNR}\wedge\mathbf{G1}$ then yields

$$\mathbf{X}\wedge(\neg\,\mathbf{G2})\Rightarrow\ \text{false},$$

which equivalently implies that $\mathbf{X}\Rightarrow\mathbf{G2}$. The proof is thus complete. ■

M.3 Proof of "$\mathbf{LNR}\wedge\mathbf{G1}\wedge\mathbf{G2}\Leftarrow\mathbf{K1}\vee\mathbf{H2}\vee\mathbf{K2}$"

Thanks to Corollary M.2.1 and the logic relationships **N1**, **N2**, and **N3**, we only need to show that (i) $\mathbf{K1}\wedge\mathbf{G0}\Rightarrow$ false; (ii) $\mathbf{H2}\wedge\mathbf{G0}\Rightarrow$ false; and (iii) $\mathbf{K2}\wedge\mathbf{G0}\Rightarrow$ false.

We prove "$\mathbf{K1}\wedge\mathbf{G0}\Rightarrow$ false" as follows.

Proof. We prove an equivalent form: **G0** $\Rightarrow$ ($\neg$**K1**). Suppose **G0** is true. Consider $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ which contains 3 polynomials (see (6.10) when $n=1$):

$$\mathbf{k}_1^{(1)}(\underline{\mathbf{x}}) = \{\, m_{11}m_{23}m_{31}L,\; m_{21}m_{13}m_{31}L,\; m_{21}m_{13}m_{31}R \,\}. \tag{M.1}$$

Since $L = m_{13}m_{32}m_{21}$, the first polynomial in $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ is equivalent to $m_{11}m_{23}m_{32}$ $m_{21}m_{13}m_{31}$. Then $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ becomes linearly dependent by substituting $R + L$ for $m_{11}m_{23}m_{32}$ (from **G0** being true). The proof is thus complete. ■

We prove "**H2** $\wedge$ **G0** $\Rightarrow$ false" as follow.

Proof. We prove an equivalent form: **G0** $\Rightarrow$ ($\neg$**H2**). Suppose **G0** is true. Consider $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ in (5.27). Substituting $R + L$ for $m_{11}m_{23}m_{32}$ (from **G0** being true) and $L = m_{21}m_{13}m_{32}$ to the expression of $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$, then we have

$$\mathbf{h}_1^{(n)}(\underline{\mathbf{x}}) = \{\, (R+L)R^n,\; (R+L)R^{n-1}L,\; \cdots,\; (R+L)L^n,\; R^nL,\; R^{n-1}L^2,\; \cdots,\; RL^n \,\}.$$

One can see that $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ becomes linearly dependent when $n \geq 2$. The proof is thus complete. ■

We prove "**K2** $\wedge$ **G0** $\Rightarrow$ false" as follow.

Proof. Similarly following the proof of "**K1** $\wedge$ **G0** $\Rightarrow$ false", we further have

$$\begin{aligned}\mathbf{k}_1^{(n)}(\underline{\mathbf{x}}) = m_{21}m_{13}m_{31}\,\{\, &(R+L)L^{n-1},\; (R+L)L^{n-2}R,\\ &\cdots,(R+L)R^{n-1},\; L^n,\; L^{n-1}R,\; \cdots,LR^{n-1},\; R^n \,\},\end{aligned}$$

which becomes linearly dependent when $n \geq 2$. The proof is thus complete. ■

M.4 Proofs of the properties of G3, G4, $\neg$ G3, and $\neg$ G4

We prove Properties 1 and 2 of **G3** as follows.

Proof. Suppose **G3** is true, that is, $\overline{S}_2 \cap \overline{D}_3 = \emptyset$. Consider e_2^*, the most downstream edge of $\mathsf{1cut}(s_2;d_1) \cap \mathsf{1cut}(s_2;d_3)$ and e_3^*, the most upstream edge of $\mathsf{1cut}(s_1;d_3) \cap \mathsf{1cut}(s_2;d_3)$. If either $e_2^* = e_{s_2}$ or $e_3^* = e_{d_3}$ (or both), we must have $e_2^* \prec e_3^*$ otherwise it contradicts definitions (ii) and (iii) of the 3-unicast ANA network. Consider the case in which both $e_2^* \neq e_{s_2}$ and $e_3^* \neq e_{d_3}$. Recall the definitions of $\overline{S}_2 \triangleq \mathsf{1cut}(s_2;d_1) \cap \mathsf{1cut}(s_2;d_3) \backslash \{e_{s_2}\}$ and $\overline{D}_3 \triangleq \mathsf{1cut}(s_1;d_3) \cap \mathsf{1cut}(s_2;d_3) \backslash \{e_{d_3}\}$. We thus have $e_2^* \in \overline{S}_2$ and $e_3^* \in \overline{D}_3$. By the assumption $\overline{S}_2 \cap \overline{D}_3 = \emptyset$ and Lemma 6.1.3, we must have $e_2^* \prec e_3^*$ as well.

From the construction of e_2^* and e_3^*, the channel gains m_{13}, m_{21}, and m_{23} can be expressed as $m_{13} = m_{e_{s_1};e_3^*} m_{e_3^*;e_{d_3}}$, $m_{21} = m_{e_{s_2};e_2^*} m_{e_2^*;e_{d_1}}$, and $m_{23} = m_{e_{s_2};e_2^*} m_{e_2^*;e_3^*} m_{e_3^*;e_{d_3}}$. Moreover, we have both $\mathsf{GCD}(\, m_{e_{s_1};e_3^*},\ m_{e_{s_2};e_2^*} m_{e_2^*;e_3^*}) \equiv 1$ and $\mathsf{GCD}(\, m_{e_2^*;e_3^*} m_{e_3^*;e_{d_3}},\ m_{e_2^*;e_{d_1}})$ $\equiv 1$ otherwise it violates that e_2^* (resp. e_3^*) is the most downstream (resp. upstream) edge of $\overline{S}_2$ (resp. $\overline{D}_3$). The same argument also leads to $\mathsf{GCD}(\, m_{13},\ m_{e_2^*;e_3^*}) \equiv 1$ and $\mathsf{GCD}(\, m_{21},\ m_{e_2^*;e_3^*}) \equiv 1$. ■

We prove Properties 1, 2, and 3 of $\neg$ **G3** as follows.

Proof. Suppose $\neg$ **G3** is true, i.e., $\overline{S}_2 \cap \overline{D}_3 \neq \emptyset$. Choose the most upstream e_u^{23} and most downstream e_v^{23} edges in $\overline{S}_2 \cap \overline{D}_3$. Then, the channel gains m_{13}, m_{21}, and m_{23} can be expressed as $m_{13} = m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}}$, $m_{21} = m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}}\ m_{e_v^{23};e_{d_1}}$, and $m_{23} = m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}}$. Moreover, we must have both $\mathsf{GCD}(m_{e_{s_1};e_u^{23}}, m_{e_{s_2};e_u^{23}}) \equiv 1$ and $\mathsf{GCD}(m_{e_v^{23};e_{d_3}}, m_{e_v^{23};e_{d_1}}) \equiv 1$ otherwise it violates Lemma 6.1.3 and/or e_u^{23} (resp. e_v^{23}) being the most upstream (resp. downstream) edge among $\overline{S}_2 \cap \overline{D}_3$. For example, if $\mathsf{GCD}(\, m_{e_{s_1};e_u^{23}},\ m_{e_{s_2};e_u^{23}}) \not\equiv 1$, then by Lemma 6.1.7 and the assumption $e_u^{23} \in \overline{S}_2 \cap \overline{D}_3 \subset \overline{D}_3$, there must exist an edge $e \in \overline{D}_3$ such that $e \prec e_u^{23}$. If such edge e is also in $\overline{S}_2$, then this e violates the construction that e_u^{23} is the most upstream edge of $\overline{S}_2 \cap \overline{D}_3$. If such edge e is not in $\overline{S}_2$, then it contradicts the conclusion in Lemma 6.1.3.

We now prove Property 3 of $\neg$ **G3**. Suppose that at least one of $\{e_u^{23}, e_v^{23}\}$ is not an 1-edge cut separating s_1 and $\mathsf{head}(e_v^{23})$. Say $e_u^{23} \notin \mathsf{1cut}(s_1; \mathsf{head}(e_v^{23}))$, then s_1 can reach $\mathsf{head}(e_v^{23})$ without using e_u^{23}. Since $\mathsf{head}(e_v^{23})$ reaches d_3, we can create ans_1-to-$d_3$path not using e_u^{23}. This contradicts the construction that $e_u^{23} \in \overline{S}_2 \cap \overline{D}_3 \subset \overline{D}_3$. Similarly, we can also prove that $e_v^{23} \notin \mathsf{1cut}(s_1; \mathsf{head}(e_v^{23}))$ leads to a contradiction. Therefore, we have proven $\{e_u^{23}, e_v^{23}\} \subset \mathsf{1cut}(s_1; \mathsf{head}(e_v^{23}))$. Symmetrically applying the above arguments, we can also prove that $\{e_u^{23}, e_v^{23}\} \subset \mathsf{1cut}(\mathsf{tail}(e_u^{23}); d_1)$.

Now consider an s_1-to-d_1 path P such that there exists one vertex $w \in P$ satisfying $\mathsf{tail}(e_u^{23}) \preceq w \preceq \mathsf{head}(e_v^{23})$. If the path of interest P does not use e_u^{23} and $w = \mathsf{tail}(e_u^{23})$, then $\mathsf{tail}(e_u^{23})$ can follow P to d_1 without using e_u^{23}, which contradicts $e_u^{23} \in \mathsf{1cut}(\mathsf{tail}(e_u^{23}); d_1)$. If P does not use e_u^{23} and $\mathsf{tail}(e_u^{23}) \prec w \preceq \mathsf{head}(e_v^{23})$, then s_1 can follow P to w and reach $\mathsf{head}(e_v^{23})$ without using e_u^{23}, which contradicts $e_u^{23} \in \mathsf{1cut}(s_1; \mathsf{head}(e_v^{23}))$. By the similar arguments, we can also prove the case when P does not use e_v^{23} leads to a contradiction. Therefore, we must have $\{e_u^{23}, e_v^{23}\} \subset P$. The proof is complete. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the above proofs can also be used to prove Properties 1 and 2 of **G4** and Properties 1, 2, and 3, of $\neg$ **G4**.

M.5 Proofs of N1 to N9

We prove **N1** as follows.

Proof. Instead of proving directly, we prove **H2** $\Rightarrow$ **H1** and use the existing result of "**LNR** $\wedge$ **G1** $\Leftarrow$ **H1**" established in the proof of Proposition 6.3.1. **H2** $\Rightarrow$ **H1** is straightforward since $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$ is a subset of the polynomials $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ (multiplied by a common factor) and whenever $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent, so is $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$. The proof is thus complete. ■

We prove **N2** as follows.

Proof. We prove an equivalent relationship: $(\neg$ **LNR**$) \vee (\neg$ **G1**$) \Rightarrow (\neg$ **K1**$)$. Consider $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ as in (M.1). Suppose $G_{3\text{ANA}}$ satisfies $(\neg$ **LNR**$) \vee (\neg$ **G1**$)$, which means $G_{3\text{ANA}}$

satisfies either $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$ or $m_{11}m_{23} \equiv m_{21}m_{13}$ or $m_{11}m_{32} \equiv m_{31}\ m_{12}$. If $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$, then we notice that $m_{21}m_{13}m_{31}L \equiv m_{21}m_{13}m_{31}R$ and $\mathbf{k}_1^{(1)}(\underline{\mathbf{x}})$ is thus linearly dependent. If $m_{11}\ m_{23} \equiv m_{21}m_{13}$, then we notice $m_{11}m_{23}m_{31}L \equiv m_{21}m_{13}m_{31}\ L$. Similarly if $m_{11}m_{32} \equiv m_{31}m_{12}$, then we have $m_{11}m_{23}m_{31}\ L \equiv m_{21}m_{13}m_{31}R$. The proof is thus complete. ■

Following similar arguments used in proving **N2**, i.e., **K2** ⇒ **K1**, one can easily prove **N3**.

We prove **N4** as follows.

Proof. $(\neg\,\mathbf{G2}) \wedge \mathbf{G3} \wedge \mathbf{G4}$ implies that s_1 cannot reach d_1 on $G_{3\text{ANA}}$. This violates the definition (iv) of the 3-unicast ANA network. ■

We prove **N5** as follow.

Proof. We prove an equivalent relationship: $(\neg\,\mathbf{G2}) \wedge (\neg\,\mathbf{G3}) \wedge \mathbf{G4} \Rightarrow (\neg\,\mathbf{G1})$. Suppose $(\neg\,\mathbf{G2}) \wedge (\neg\,\mathbf{G3}) \wedge \mathbf{G4}$ is true. Then the most upstream edge of $\overline{S}_2 \cap \overline{D}_3$ is an 1-edge cut separating s_1 and d_1. Therefore we have $\mathsf{EC}(\{s_1, s_2\}; \{d_1, d_3\}) = 1$ and thus by Corollary 5.4.1, $m_{11}m_{23} \equiv m_{21}m_{13}$. This further implies that **G1** is false. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the above **N5** proof can also be used to prove **N6**.

We prove **N7** as follows.

Proof. Suppose $\mathbf{LNR} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge (\neg\,\mathbf{G5})$ is true. From **LNR** being true, any $\overline{S}_2 \cap \overline{D}_3$ edge and any $\overline{S}_3 \cap \overline{D}_2$ edge must be distinct, otherwise (if there exists an edge $e \in \overline{S}_2 \cap \overline{S}_3 \cap \overline{D}_2 \cap \overline{D}_3$) it contradicts the assumption **LNR** by Proposition 6.2.1. From **G5** being false, we have either $e_u^{23} = e_u^{32}$ or both e_u^{23} and e_u^{32} are not reachable from each other. But $e_u^{23} = e_u^{32}$ cannot be true by the assumption **LNR**.

Now we prove **G6**, i.e., any vertex w' where $\mathsf{tail}(e_u^{23}) \preceq w' \preceq \mathsf{head}(e_v^{23})$ and any vertex w'' where $\mathsf{tail}(e_u^{32}) \preceq w'' \preceq \mathsf{head}(e_v^{32})$ are not reachable from each other. Suppose not and assume that some vertex w' satisfying $\mathsf{tail}(e_u^{23}) \preceq w' \preceq \mathsf{head}(e_v^{23})$ and some vertex w'' satisfying $\mathsf{tail}(e_u^{32}) \preceq w'' \preceq \mathsf{head}(e_v^{32})$ are reachable from each other. Since s_1 can reach $\mathsf{tail}(e_u^{23})$ or $\mathsf{tail}(e_u^{32})$ and d_1 can be reached from $\mathsf{head}(e_v^{23})$ or $\mathsf{head}(e_v^{32})$ by Property 1 of $\neg$ **G3** and $\neg$ **G4**, we definitely have an s_1-to-d_1 path P who uses both w' and w''. The reason is that if $w' \preceq w''$, then s_1 can first reach $\mathsf{tail}(e_u^{23})$, visit w', w'', and $\mathsf{head}(e_v^{32})$, and finally arrive at d_1. The case when $w'' \preceq w'$ can be proven by symmetry. By Property 3 of $\neg$ **G3**, such path must use $\{e_u^{23}, e_v^{23}\}$. Similarly by Property 3 of $\neg$ **G4**, such path must also use $\{e_u^{32}, e_v^{32}\}$. Together with the above discussion that any $\overline{S}_2 \cap \overline{D}_3$ edge and any $\overline{S}_3 \cap \overline{D}_2$ edge are distinct, this implies that all four edges $\{e_u^{23}, e_v^{23}, e_u^{32}, e_v^{32}\}$ are not only distinct but also used by a single path P. However, this contradicts the assumption **LNR** $\wedge$ ($\neg$ **G5**) that e_u^{23} and e_u^{32} are not reachable from each other. ■

We prove **N8** as follows.

Proof. Suppose **G1** $\wedge$ ($\neg$ **G2**) $\wedge$ ($\neg$ **G3**) $\wedge$ ($\neg$ **G4**) $\wedge$ **G5** is true. Consider e_u^{23} and e_u^{32}, the most upstream edges of $\overline{S}_2 \cap \overline{D}_3$ and $\overline{S}_3 \cap \overline{D}_2$, respectively. Say we have $e_u^{23} \prec e_u^{32}$. Then $\neg$ **G2** implies that removing e_u^{23} will disconnect s_1 and d_1. Therefore, $e_u^{23} \in \overline{S}_2 \cap \overline{D}_3$ also belongs to $\mathsf{1cut}(s_1; d_1)$. This further implies that we have $\mathsf{EC}(\{s_1, s_2\}; \{d_1, d_3\}) = 1$ and thus $G_{3\text{ANA}}$ satisfies $m_{11}m_{23} \equiv m_{13}m_{21}$. However, this contradicts the assumption that **G1** is true. Similar arguments can be applied to show that the case when $e_u^{32} \prec e_u^{23}$ also contradicts **G1**. The proof of **N8** is thus complete. ■

We prove **N9** as follows.

Proof. Suppose that ($\neg$ **G2**) $\wedge$ ($\neg$ **G3**) $\wedge$ ($\neg$ **G4**) $\wedge$ ($\neg$ **G5**) $\wedge$ **G6** is true. Consider e_u^{23} and e_u^{32}, the most upstream edges of $\overline{S}_2 \cap \overline{D}_3$ and $\overline{S}_3 \cap \overline{D}_2$, respectively. From ($\neg$ **G5**) $\wedge$ **G6** being true, one can see that e_u^{23} and e_u^{32} are not only distinct but also not reachable from each other. Thus by $\neg$ **G2** being true, $\{e_u^{23}, e_u^{32}\}$ constitutes an edge cut separating s_1 and d_1. Note from Property 1 of $\neg$ **G3** and $\neg$ **G4** that s_1 can reach d_1 through either e_u^{23} or e_u^{32}. Since e_u^{23} and e_u^{32} are not reachable from each other, both have to be removed to disconnect s_1 and d_1 (removing only one of them is not enough).

From **G6** being true, any vertex w' where $\mathsf{tail}(e_u^{23}) \preceq w' \preceq \mathsf{head}(e_v^{23})$ and any vertex w'' where $\mathsf{tail}(e_u^{32}) \preceq w'' \preceq \mathsf{head}(e_v^{32})$ are not reachable from each other. Thus e_u^{23} (resp. e_u^{32}) cannot reach e_v^{32} (resp. e_v^{23}). Moreover, e_v^{23} and e_v^{32} are not only distinct but also not reachable from each other. This implies that e_u^{23} (resp. e_u^{32}) can only reach e_v^{23} (resp. e_v^{32}) if $e_u^{23} \neq e_v^{23}$ (resp. $e_u^{32} \neq e_v^{32}$). Then the above discussions further that imply $\{e_v^{23}, e_v^{32}\}$ is also an edge cut separating s_1 and d_1.

Let $m'_{11} = m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_1}}$, which takes into account the overall path gain from s_1 to d_1 for all paths that use both e_u^{23} and e_v^{23}. Similarly denote $m''_{11} = m_{e_{s_1};e_u^{32}} m_{e_u^{32};e_v^{32}}\, m_{e_v^{32};e_{d_1}}$ to be the overall path gain from s_1 to d_1 for all paths that use both e_u^{32} and e_v^{32}. Then the discussions so far imply that the channel gain m_{11} consists of two polynomials: $m_{11} = m'_{11} + m''_{11}$. Then, it follows that

$$
\begin{aligned}
& m_{11}m_{23}m_{32} = (m'_{11} + m''_{11})\, m_{23}m_{32} \\
& = (m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_1}})\, m_{23}m_{32} \\
& \;+ (m_{e_{s_1};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_1}})\, m_{23}m_{32} \\
& = (m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_1}})(m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}})\, m_{32} \\
& \;+ (m_{e_{s_1};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_1}})\, m_{23}\, (m_{e_{s_3};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_2}}) \\
& = (m_{e_{s_1};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_3}})\, m_{32}\, (m_{e_{s_2};e_u^{23}} m_{e_u^{23};e_v^{23}} m_{e_v^{23};e_{d_1}}) \\
& \;+ (m_{e_{s_1};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_2}})\, m_{23}\, (m_{e_{s_3};e_u^{32}} m_{e_u^{32};e_v^{32}} m_{e_v^{32};e_{d_1}}) \\
& = m_{13}m_{32}m_{21} + m_{12}m_{23}m_{31} = L + R.
\end{aligned}
$$

where the third and fourth equalities follow from the Property 1 of both $\neg$ **G3** and $\neg$ **G4**. The proof is thus complete. ■

N. GENERAL STRUCTURED PROOF FOR THE SUFFICIENCY OF PROPOSITION 6.3.1

In this appendix, we provide Corollary N.2.1, which will be used to prove the graph-theoretic sufficient direction of 3-unicast ANA network for arbitrary $n>0$ values. We need the following additional logic statements to describe the general proof structure.

N.1 The second set of logic statements

Given a 3-unicast ANA network $G_{3\text{ANA}}$, recall the definitions $L = m_{13}m_{32}m_{21}$ and $R = m_{12}m_{23}m_{31}$ (we drop the input argument $\underline{\mathbf{x}}$ for simplicity). By the definition of $G_{3\text{ANA}}$, any channel gains are non-trivial, and thus R and L are non-zero polynomials. Let $\psi_\alpha^{(n)}(R, L)$ and $\psi_\beta^{(n)}(R, L)$ to be some polynomials with respect to $\underline{\mathbf{x}}$, represented by

$$\psi_\alpha^{(n)}(R,L)=\sum_{i=0}^{n}\alpha_i R^{n-i}L^i, \qquad \psi_\beta^{(n)}(R,L)=\sum_{j=0}^{n}\beta_j R^{n-j}L^j,$$

with some set of coefficients $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$, respectively. Basically, given a value of n and the values of $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$, $\psi_\alpha^{(n)}(R, L)$ (resp. $\psi_\beta^{(n)}(R, L)$) represents a linear combination of $\{R^n,\ R^{n-1}L,\ \cdots,\ RL^{n-1},\ L^n\}$, the set of Vandermonde polynomials

We need the following additional logic statements.

• **E0:** Let $I_{3\text{ANA}}$ be a finite index set defined by $I_{3\text{ANA}} = \{(i, j) :\ i, j \in \{1, 2, 3\}$ and $i \neq j\}$. Consider two non-zero polynomial functions $f : \mathbb{F}_q^{|I_{3\text{ANA}}|} \mapsto \mathbb{F}_q$ and $g : \mathbb{F}_q^{|I_{3\text{ANA}}|} \mapsto$

$\mathbb{F}_q$. Then given a $G_{3\text{ANA}}$ of interest, there exists some coefficient values $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$ such that

$$m_{11}\, f(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\})\, \psi_\alpha^{(n)}(R,L) \;=\; g(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\})\, \psi_\beta^{(n)}(R,L),$$

with (i) At least one of coefficients $\{\alpha_i\}_{i=0}^n$ is non-zero; and (ii) At least one of coefficients $\{\beta_j\}_{j=0}^n$ is non-zero.

Among $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$, define i_{st} (resp. j_{st}) as the smallest i (resp. j) such that $\alpha_i \neq 0$ (resp. $\beta_j \neq 0$). Similarly, define i_{end} (resp. j_{end}) as the largest i (resp. j) such that $\alpha_i \neq 0$ (resp. $\beta_j \neq 0$).[1] Then, we can rewrite the above equation as follows:

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i\, m_{11}\, f(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\})\, R^{n-i}L^i = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j\, g(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\})\, R^{n-j}L^j. \tag{N.1}$$

• **E1:** Continue from the definition of **E0**. The considered $G_{3\text{ANA}}$ satisfies (N.1) with (i) $f(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\}) = m_{23}$; and (ii) $g(\{m_{ij} : \forall\,(i,j)\in I_{3\text{ANA}}\}) = m_{13}m_{21}$. Then, (N.1) reduces to

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{n-i} L^i = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{n-j} L^j. \tag{N.2}$$

• **E2:** Continue from the definition of **E0**. The chosen coefficients $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$ which satisfy (N.1) in the given $G_{3\text{ANA}}$ also satisfy (i) $\alpha_k \neq \beta_k$ for some $k\in\{0,...,n\}$; and (ii) either $\alpha_0\neq 0$ or $\beta_n\neq 0$ or $\alpha_k\neq\beta_{k-1}$ for some $k\in\{1,...,n\}$.

One can see that whether the above logic statements are true or false depends on the polynomials m_{ij} and on the $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$ values being considered.

The following logic statements are well-defined if and only if **E0** is true. Whether the following logic statements are true depends on the values of i_{st}, i_{end}, j_{st}, and j_{end}.

• **C0:** $i_{\text{st}} > j_{\text{st}}$ and $i_{\text{end}} = j_{\text{end}}$.

• **C1:** $i_{\text{st}} < j_{\text{st}}$.

[1]From definition, $0 \leq i_{\text{st}} \leq i_{\text{end}} \leq n$ and $0 \leq j_{\text{st}} \leq j_{\text{end}} \leq n$.

- **C2:** $i_{\mathrm{st}} > j_{\mathrm{st}}$.
- **C3:** $i_{\mathrm{st}} = j_{\mathrm{st}}$.
- **C4:** $i_{\mathrm{end}} < j_{\mathrm{end}}$.
- **C5:** $i_{\mathrm{end}} > j_{\mathrm{end}}$.
- **C6:** $i_{\mathrm{end}} = j_{\mathrm{end}}$.

We also define the following statements for the further organization.

- **D1:** $\mathsf{GCD}(m_{12}^{l_1} m_{23}^{l_1} m_{31}^{l_1},\, m_{32}) = m_{32}$ for some integer $l_1 > 0$.
- **D2:** $\mathsf{GCD}(m_{13}^{l_2} m_{32}^{l_2} m_{21}^{l_2},\, m_{23}) = m_{23}$ for some integer $l_2 > 0$.
- **D3:** $\mathsf{GCD}(\, m_{11} m_{13}^{l_3} m_{32}^{l_3} m_{21}^{l_3},\, m_{12} m_{31}) = m_{12} m_{31}$ for some integer $l_3 > 0$.
- **D4:** $\mathsf{GCD}(\, m_{11} m_{12}^{l_4} m_{23}^{l_4} m_{31}^{l_4},\, m_{13} m_{21}) = m_{13} m_{21}$ for some integer $l_4 > 0$.
- **D5:** $\mathsf{GCD}(\, m_{11} m_{12}^{l_5} m_{23}^{l_5} m_{31}^{l_5},\, m_{32}) = m_{32}$ for some integer $l_5 > 0$.
- **D6:** $\mathsf{GCD}(\, m_{11} m_{13}^{l_6} m_{32}^{l_6} m_{21}^{l_6},\, m_{23}) = m_{23}$ for some integer $l_6 > 0$.

N.2 General Sufficiency Proof Structure

We prove the following "logic relationships," which will be used for the proof of Corollary N.2.1.

- **S1**: **D1** $\Rightarrow$ **D5**.
- **S2**: **D2** $\Rightarrow$ **D6**.
- **S3**: **E0** $\wedge$ **E1** $\wedge$ **C1** $\Rightarrow$ **D4** $\wedge$ **D5**.
- **S4**: **E0** $\wedge$ **E1** $\wedge$ **C2** $\Rightarrow$ **D1**.
- **S5**: **G1** $\wedge$ **E0** $\wedge$ **E1** $\wedge$ **C3** $\Rightarrow$ **D4**.
- **S6**: **E0** $\wedge$ **E1** $\wedge$ **C4** $\Rightarrow$ **D2** $\wedge$ **D3**.
- **S7**: **E0** $\wedge$ **E1** $\wedge$ **C5** $\Rightarrow$ **D3**.
- **S8**: **G1** $\wedge$ **E0** $\wedge$ **E1** $\wedge$ **C6** $\Rightarrow$ **D2**.
- **S9**: **E0** $\wedge$ **E1** $\wedge$ **C0** $\Rightarrow$ **E2**.
- **S10**: **G1** $\wedge$ **E0** $\wedge$ **E1** $\wedge$ $(\neg$ **C0**$)$ $\Rightarrow$ $($**D1** $\wedge$ **D3**$)$ $\vee$ $($**D2** $\wedge$ **D4**$)$ $\vee$ $($**D3** $\wedge$ **D4**$)$.
- **S11**: **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D1** $\wedge$ **D3** $\Rightarrow$ false.
- **S12**: **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D2** $\wedge$ **D4** $\Rightarrow$ false.

- **S13**: $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{E2} \wedge \mathbf{D1} \wedge \mathbf{D2} \Rightarrow$ false.
- **S14**: $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \Rightarrow$ false.

The proofs of **S1** to **S10** are relegated to Appendix N.5. The proofs of **S11** to **S14** are relegated to Appendices N.6, N.7, N.8, and N.9, respectively. Note that the above **S1** to **S14** relationships greatly simplify the analysis of finding the graph-theoretic conditions for the feasibility of the 3-unicast ANA network. This observation is summarized in Corollary N.2.1.

Corollary N.2.1. *Let* $\mathbf{h}(\underline{\mathbf{x}})$ *be a set of (arbitrarily chosen) polynomials based on the* 9 *channel gains* m_{ij} *of the 3-unicast ANA network, and define* $\mathbf{X}$ *to be the logic statement that* $\mathbf{h}(\underline{\mathbf{x}})$ *is linearly independent. Let* $\mathbf{G}$ *to be an arbitrary logic statement in the 3-unicast ANA network. If we can prove that*

(A) $\mathbf{G} \wedge (\neg\, \mathbf{X}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\, \mathbf{C0})$,

then the logic relationship $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G} \wedge (\neg\, \mathbf{X}) \Rightarrow$ *false must also hold.*

Also, if we can prove that

(B) $\mathbf{G} \wedge (\neg\, \mathbf{X}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0}$,

then the logic relationship $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge \mathbf{G} \wedge (\neg\, \mathbf{X}) \Rightarrow$ *false must also hold.*

Proof of Corollary N.2.1: First, notice that **S11**, **S12**, and **S14** jointly imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \{(\mathbf{D1} \wedge \mathbf{D3}) \vee (\mathbf{D2} \wedge \mathbf{D4}) \vee (\mathbf{D3} \wedge \mathbf{D4})\} \Rightarrow \text{false}. \tag{N.3}$$

Then, (N.3), jointly with **S10** further imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\, \mathbf{C0}) \Rightarrow \text{false}. \tag{N.4}$$

Note that by definition **C0** is equivalent to $\mathbf{C2} \wedge \mathbf{C6}$. Then **S4** and **S8** jointly imply

$$\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0} \Rightarrow \mathbf{D1} \wedge \mathbf{D2}. \tag{N.5}$$

Then, (N.5), **S9**, and **S13** jointly imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0} \Rightarrow \text{ false.} \tag{N.6}$$

Now we prove the result using (N.4) and (N.6). Suppose we can also prove (A) $\mathbf{G} \wedge (\neg\,\mathbf{X}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\,\mathbf{C0})$. Then, one can see that this, jointly with (N.4), implies $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G} \wedge (\neg\,\mathbf{X}) \Rightarrow$ false. Similarly, (B) $\mathbf{G} \wedge (\neg\,\mathbf{X}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0}$ and (N.6) jointly imply $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge \mathbf{G} \wedge (\neg\,\mathbf{X}) \Rightarrow$ false. The proof is thus complete. ■

N.3 The insight on proving the sufficiency

To prove the sufficiency directions, we need to show that a set of polynomials is linearly independent given any 3-unicast ANA network, for example, "$\mathbf{LNR} \wedge \mathbf{G} \Rightarrow \mathbf{X}$". To that end, we prove the equivalent relationship "$\mathbf{LNR} \wedge \mathbf{G} \wedge (\neg\,\mathbf{X}) \Rightarrow$ false." Focusing on the linear dependence condition $\neg\,\mathbf{X}$, although there are many possible cases, allows us to use the subgraph property (Proposition 2) to simplify the proof. Further, we use the logic statements **S3** to **S10** to convert all the cases of the linear dependence condition into the greatest common divisor statements **D1** to **D6**, for which the channel gain property (Proposition 3) further helps us to find the corresponding graph-theoretic implication.

N.4 Proofs of "$\mathbf{LNR} \wedge \mathbf{G1} \Rightarrow \mathbf{H1}$" and "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{K1} \vee \mathbf{H2} \vee \mathbf{K2}$"

As discussed in Appendix N, we use Corollary N.2.1 to prove the sufficiency directions. We first show that (i) $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{H2}$; and (ii) $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{K2}$. Then the remaining sufficiency directions "$\mathbf{LNR} \wedge \mathbf{G1} \Rightarrow \mathbf{H1}$" and "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{K1}$" are derived using simple facts of "$\mathbf{H2} \Rightarrow \mathbf{H1}$" and "$\mathbf{K2} \Rightarrow \mathbf{K1}$", respectively. Note that $\mathbf{H2} \Rightarrow \mathbf{H1}$ is straightforward since $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$ is a subset of the polynomials

$\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ (multiplied by a common factor) and whenever $\mathbf{h}_1^{(n)}(\underline{\mathbf{x}})$ is linearly independent, so is $\mathbf{h}_1^{(1)}(\underline{\mathbf{x}})$. Similarly, we have $\mathbf{K2} \Rightarrow \mathbf{K1}$.

We prove "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{H2}$" as follows.

Proof. By the definition of linear dependence, $\neg\,\mathbf{H2}$ implies that there exist two sets of coefficients $\{\alpha_i\}_{i=0}^{n}$ and $\{\beta_j\}_{j=0}^{n-1}$ such that

$$\sum_{i=0}^{n} \alpha_i m_{11} m_{23} R^{n-i} L^i = \sum_{j=0}^{n-1} \beta_j m_{13} m_{21} R^{n-j} L^j. \tag{N.7}$$

We will now argue that at least one of $\{\alpha_i\}_{i=0}^{n}$ and at least one of $\{\beta_j\}_{j=0}^{n-1}$ are non-zero if $L \not\equiv R$. The reason is as follows. For example, suppose that all $\{\beta_j\}_{j=0}^{n-1}$ are zero. By definition (iv) of the 3-unicast ANA network, any channel gain is non-trivial. Thus $m_{11}m_{23}$ is a non-trivial polynomial. Then, (N.7) becomes $\sum_{i=0}^{n} \alpha_i R^{n-i} L^i = 0$, which implies that the set of $(n+1)$ polynomials, $\tilde{\mathbf{h}}(\underline{\mathbf{x}}) = \{R^n,\, R^{n-1}L,\, ...,\, RL^{n-1},\, L^n\}$, is linearly dependent. By Proposition 5.4.1, the determinant of the Vandermonde matrix $[\tilde{\mathbf{h}}(\underline{\mathbf{x}}^{(k)})]_{k=1}^{n+1}$ is thus zero, which implies $L(\underline{\mathbf{x}}) \equiv R(\underline{\mathbf{x}})$. This contradicts the assumption $\mathbf{LNR}$. The fact that not all $\{\alpha_i\}_{i=0}^{n}$ are zero can be proven similarly.

As a result, there exist two sets of coefficients $\{\alpha_i\}_{i=0}^{n}$ and $\{\beta_j\}_{j=0}^{n-1}$ with at least one of each group being non-zero such that the following logic relationship holds:

$$\mathbf{LNR} \wedge (\neg\,\mathbf{H2}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1}. \tag{N.8}$$

Then, note that (N.8) implies

$$\mathbf{LNR} \wedge (\neg\,\mathbf{C0}) \wedge (\neg\,\mathbf{H2}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\,\mathbf{C0}),$$
$$\text{and} \quad \mathbf{LNR} \wedge \mathbf{C0} \wedge (\neg\,\mathbf{H2}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0}.$$

Applying Corollary N.2.1(A) (substituting $\mathbf{G}$ by $\mathbf{LNR} \wedge (\neg\,\mathbf{C0})$ and $\mathbf{X}$ by $\mathbf{H2}$, respectively), the former implies $\mathbf{LNR} \wedge \mathbf{G1} \wedge (\neg\,\mathbf{C0}) \wedge (\neg\,\mathbf{H2}) \Rightarrow$ false. By Corollary N.2.1(B), the latter implies $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge \mathbf{C0} \wedge (\neg\, \mathbf{H2}) \Rightarrow$ false. These jointly imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \wedge (\neg\, \mathbf{H2}) \Rightarrow \text{ false},$$

which is equivalent to $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{H2}$. The proof is thus complete. ■

We prove "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{K2}$" as follows.

Proof. We will only show the logic relationship "$\mathbf{LNR} \wedge (\neg\, \mathbf{K2}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1}$" so that the rest can be proved by Corollary N.2.1 as in the proof of "$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{G2} \Rightarrow \mathbf{H2}$". Suppose $\neg\, \mathbf{K2}$ is true. Then, there exists two sets of coefficients $\{\alpha_i\}_{i=1}^{n}$ and $\{\beta_j\}_{j=0}^{n}$ such that

$$\sum_{i=1}^{n} \alpha_i m_{11} m_{23} R^{n-i} L^i = \sum_{j=0}^{n} \beta_j m_{13} m_{21} R^{n-j} L^j. \tag{N.9}$$

One can easily see that, similarly to the above proof, the assumption **LNR** results in the not-being-all-zero condition on both $\{\alpha_i\}_{i=1}^{n}$ and $\{\beta_j\}_{j=0}^{n}$, which in turn implies that "$\mathbf{LNR} \wedge (\neg\, \mathbf{K2}) \Rightarrow \mathbf{E0} \wedge \mathbf{E1}$". The proof is thus complete. ■

N.5 Proofs of S1 to S10

We prove **S1** as follows.

Proof. Suppose **D1** is true, that is, $G_{3\text{ANA}}$ satisfies $\mathsf{GCD}(\, m_{12}^{l_1} m_{23}^{l_1} m_{31}^{l_1},\, m_{32}) = m_{32}$ for some integer $l_1 > 0$. Then $G_{3\text{ANA}}$ also satisfies $\mathsf{GCD}(\, m_{11} m_{12}^{l_1} m_{23}^{l_1} m_{31}^{l_1},\, m_{32}) = m_{32}$ obviously. Thus we have **D5**. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the proof for **S1** can be applied symmetrically to the proof for **S2**.

We prove **S3** as follows.

Proof. Suppose $\mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C1}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, $G_{3\text{ANA}}$ of interest satisfies (N.2). By the definition of **C1**, we have $i_{\text{st}} < j_{\text{st}}$.

By (N.2), we can divide $L^{i_{\text{st}}}$ on both sides. Then we have

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{n-i} L^{i-i_{\text{st}}} = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{n-j} L^{j-i_{\text{st}}}.$$

Since $i_{\text{st}} < j_{\text{st}}$, each term with non-zero β_j in the right-hand side (RHS) has L as a common factor. Similarly, each term with non-zero α_i on the left-hand side (LHS) has L as a common factor except for the first term (since $\alpha_{i_{\text{st}}} \neq 0$). Therefore the first term $\alpha_{i_{\text{st}}} m_{11} m_{23} R^{n-i_{\text{st}}}$ must contain $L = m_{13} m_{32} m_{21}$ as a factor, which implies $\mathsf{GCD}(\, m_{11} m_{12}^{n-i_{\text{st}}} m_{23}^{n-i_{\text{st}}+1} m_{31}^{n-i_{\text{st}}},\, m_{13} m_{32} m_{21}) = m_{13} m_{32}\ m_{21}$. Since $i_{\text{st}} < j_{\text{st}} \leq n$, we have $n - i_{\text{st}} \geq 1$. Hence, we have $\mathsf{GCD}(\, m_{11} m_{12}^{k} m_{23}^{k+1} m_{31}^{k},\, m_{13} m_{32} m_{21}) = m_{13} m_{32} m_{21}$ for some integer $k \geq 1$. This observation implies the following two statements. Firstly, $\mathsf{GCD}(\, m_{11} m_{12}^{l_4} m_{23}^{l_4} m_{31}^{l_4},\, m_{13} m_{21}) = m_{13} m_{21}$ when $l_4 = k+1 \geq 2$ and thus we have proven **D4**. Secondly, $\mathsf{GCD}(\, m_{11} m_{12}^{l_5} m_{23}^{l_5} m_{31}^{l_5},\, m_{32}) = m_{32}$ when $l_5 = k + 1 \geq 2$ and thus we have proven **D5**. The proof is thus complete. ■

We prove **S4** as follows.

Proof. Suppose $\mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C2}$ is true. Then $G_{3\text{ANA}}$ of interest satisfies (N.2) and we have $i_{\text{st}} > j_{\text{st}}$.

We now divide $L^{j_{\text{st}}}$ on both sides of (N.2), which leads to

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{n-i} L^{i-j_{\text{st}}} = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{n-j} L^{j-j_{\text{st}}}.$$

Each term with non-zero α_i on the LHS has L as a common factor. Similarly, each term with non-zero β_j on the RHS has L as a common factor except for the first term (since $\beta_{j_{\text{st}}} \neq 0$). As a result, the first term $\beta_{j_{\text{st}}} m_{13} m_{21} R^{n-j_{\text{st}}}$ must contain $L = m_{13} m_{32} m_{21}$ as a factor. This implies that $\mathsf{GCD}(\, R^{n-j_{\text{st}}},\, m_{32}) = m_{32}$. Since $j_{\text{st}} < i_{\text{st}} \leq n$, we have $n - j_{\text{st}} \geq 1$ and thus $\mathsf{GCD}(\, R^{k},\, m_{32}) = m_{32}$ for some positive integer k, which is equivalent to **D1**. The proof is thus complete. ■

We prove **S5** as follows.

Proof. Suppose $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C3}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, $G_{3\text{ANA}}$ of interest satisfies (N.2). Since $i_{\text{st}} = j_{\text{st}}$, we can divide $L^{i_{\text{st}}} = L^{j_{\text{st}}}$ on both sides of (N.2), which leads to

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{n-i} L^{i-i_{\text{st}}} = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{n-j} L^{j-j_{\text{st}}}.$$

Note that if $i_{\text{st}} = j_{\text{st}} = n$ meaning that $i_{\text{st}} = j_{\text{st}} = i_{\text{end}} = j_{\text{end}} = n$, then (N.2) reduces to $m_{11}m_{23} \equiv m_{13}m_{21}$ (since $\alpha_{i_{\text{st}}} \neq 0$ and $\beta_{j_{\text{st}}} \neq 0$). This contradicts the assumption **G1**.

Thus for the following, we only consider the case when $i_{\text{st}} = j_{\text{st}} \leq n-1$. Note that each term with non-zero β_j on the RHS has a common factor $m_{13}m_{21}$. Similarly, each term with non-zero α_i on the LHS has a common factor $L = m_{13}m_{32}m_{21}$ except for the first term ($i = i_{\text{st}}$). As a result, the first term $\alpha_{i_{\text{st}}} m_{11} m_{23} R^{n-i_{\text{st}}}$ must contain $m_{13}m_{21}$ as a factor. Since $i_{\text{st}} \leq n-1$, we have $\mathsf{GCD}(\, m_{11} m_{12}^{k} m_{23}^{k+1} m_{31}^{k},\, m_{13}m_{21}) = m_{13}m_{21}$ for some integer $k \geq 1$. Therefore, we have **D4**. ∎

We prove **S6** as follows.

Proof. Suppose $\mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C4}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, $G_{3\text{ANA}}$ of interest satisfies (N.2). Since $i_{\text{end}} < j_{\text{end}}$, we can divide $R^{n-j_{\text{end}}}$ on both sides of (N.2). Then, we have

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{j_{\text{end}}-i} L^{i} = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{j_{\text{end}}-j} L^{j}.$$

Each term with non-zero α_i on the LHS has R as a common factor. Similarly, each term with non-zero β_j on the RHS has R as a common factor except for the last term (since $\beta_{j_{\text{end}}} \neq 0$). Thus, the last term $\beta_{j_{\text{end}}} m_{13} m_{21} L^{j_{\text{end}}}$ must be divisible by $R = m_{12}m_{23}m_{31}$, which implies that $\mathsf{GCD}(\, m_{13}^{k+1} m_{32}^{k} m_{21}^{k+1},\, m_{12}m_{23}m_{31}) = m_{12}m_{23}m_{31}$ for some integer $k = j_{\text{end}} \geq i_{\text{end}} + 1 \geq 1$. This observation has two implications. Firstly, $\mathsf{GCD}(\, m_{11} m_{13}^{l_3} m_{32}^{l_3} m_{21}^{l_3},\, m_{12}m_{31}) = m_{12}m_{31}$ for some positive integer $l_3 = k+1$ and thus we have proven **D3**. Secondly, we also have $\mathsf{GCD}(\, m_{13}^{l_2} m_{32}^{l_2} m_{21}^{l_2},\, m_{23}) = m_{23}$ for some positive integer $l_2 = k+1$ and thus we have proven **D2**. The proof is thus complete. ■

We prove **S7** as follows.

Proof. Suppose $\mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C5}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, $G_{3\text{ANA}}$ of interest satisfies (N.2). Since $i_{\text{end}} > j_{\text{end}}$, we can divide $R^{n-i_{\text{end}}}$ on both sides of (N.2). Then we have

$$\sum_{i=i_{\text{st}}}^{i_{\text{end}}} \alpha_i m_{11} m_{23} R^{i_{\text{end}}-i} L^i = \sum_{j=j_{\text{st}}}^{j_{\text{end}}} \beta_j m_{13} m_{21} R^{i_{\text{end}}-j} L^j.$$

Each term on the RHS has R as a common factor. Similarly, each term on the LHS has R as a common factor except for the last term (since $\alpha_{i_{\text{end}}} \neq 0$). Thus, the last term $\alpha_{i_{\text{end}}} m_{11} m_{23} L^{i_{\text{end}}}$ must be divisible by $R = m_{12} m_{23} m_{31}$, which implies that $\mathsf{GCD}(\, m_{11} L^k,\, m_{12} m_{31}) = m_{12} m_{31}$ for some integer $k = i_{\text{end}} \geq j_{\text{end}} + 1 \geq 1$. This further implies **D3**. ■

We prove **S8** as follows.

Proof. Suppose $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C6}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, $G_{3\text{ANA}}$ of interest satisfies (N.2). Since **G5**, $i_{\text{end}} = j_{\text{end}}$, is true, define $t = i_{\text{end}} = j_{\text{end}}$ and $m = \min\{i_{\text{st}}, j_{\text{st}}\}$. Then by dividing R^{n-t} and L^m from both sides of (N.2), we have

$$\sum_{i=i_{\text{st}}}^{t} \alpha_i m_{11} m_{23} R^{t-i} L^{i-m} = \sum_{j=j_{\text{st}}}^{t} \beta_j m_{13} m_{21} R^{t-j} L^{j-m}. \tag{N.10}$$

Each term with non-zero α_i on the LHS has a common factor m_{23}. We first consider the case of $m < t$. Then each term with non-zero β_j on the RHS has a common factor $R = m_{12} m_{23} m_{31}$ except the last term $\beta_t m_{13} m_{21} L^{t-m}$. As a result, $\beta_t m_{13} m_{21} L^{t-m}$ must be divisible by m_{23}, which implies that $\mathsf{GCD}(\, m_{13}^{k+1} m_{32}^{k} m_{21}^{k+1},\, m_{23}) = m_{23}$ for some $k = t - m \geq 1$. This implies **D2**.

On the other hand, we argue that we cannot have $m = t$. If so, then $i_{\text{st}} = j_{\text{st}} = i_{\text{end}} = j_{\text{end}}$ and (N.2) reduces to $m_{11}m_{23} \equiv m_{13}m_{21}$. However, this contradicts the assumption **G1**. The proof is thus complete. ■

We prove **S9** as follows.

Proof. Suppose $\mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{C0}$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, G_{3ANA} of interest satisfies (N.2) with not-being-all-zero coefficients $\{\alpha_i\}_{i=0}^{n}$ and $\{\beta_j\}_{j=0}^{n}$. Our goal is to prove that, when $i_{\text{st}} > j_{\text{st}}$ and $i_{\text{end}} = j_{\text{end}}$, we have **E2**: (i) $\alpha_k \neq \beta_k$ for some $k \in \{0, ..., n\}$; and (ii) either $\alpha_0 \neq 0$ or $\beta_n \neq 0$ or $\alpha_k \neq \beta_{k-1}$ for some $k \in \{1, ..., n\}$.

Note that (i) is obvious since $i_{\text{st}} > j_{\text{st}}$. Note by definition that i_{st} (resp. j_{st}) is the smallest i (resp. j) among $\alpha_i \neq 0$ (resp. $\beta_j \neq 0$). Then, $i_{\text{st}} > j_{\text{st}}$ implies that $\alpha_{j_{\text{st}}} = 0$ while $\beta_{j_{\text{st}}} \neq 0$. Thus simply choosing $k = j_{\text{st}}$ proves (i).

We now prove (ii). Suppose (ii) is false such that $\alpha_0 = 0$; $\beta_n = 0$; and $\alpha_k = \beta_{k-1}$ for all $k \in \{1, ..., n\}$. Since $\beta_n = 0$, by definition, j_{end} must be less than or equal to $n - 1$. Since we assumed $i_{\text{end}} = j_{\text{end}}$, this in turn implies that $\alpha_n = 0$. Then β_{n-1} must be zero because $\beta_{n-1} = \alpha_n$. Again this implies $j_{\text{end}} \leq n - 2$. Applying iteratively, we have all zero coefficients $\{\alpha_i\}_{i=0}^{n}$ and $\{\beta_j\}_{j=0}^{n}$. However, this contradicts the assumption **E0** since we assumed that at least one of each coefficient group is non-zero. The proof of **S9** is thus complete. ■

We prove **S10** as follows.

Proof. Suppose $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\, \mathbf{C0})$ is true. By $\mathbf{E0} \wedge \mathbf{E1}$ being true, G_{3ANA} of interest satisfies (N.2) with some values of i_{st}, j_{st}, i_{end}, and j_{end}. Investigating their relationships, there are total 9 possible cases that G_{3ANA} can satisfy (N.2): (i) $i_{\text{st}} < j_{\text{st}}$ and $i_{\text{end}} < j_{\text{end}}$; (ii) $i_{\text{st}} < j_{\text{st}}$ and $i_{\text{end}} > j_{\text{end}}$; (iii) $i_{\text{st}} < j_{\text{st}}$ and $i_{\text{end}} = j_{\text{end}}$; (iv) $i_{\text{st}} > j_{\text{st}}$ and $i_{\text{end}} < j_{\text{end}}$; (v) $i_{\text{st}} > j_{\text{st}}$ and $i_{\text{end}} > j_{\text{end}}$; (vi) $i_{\text{st}} > j_{\text{st}}$ and $i_{\text{end}} = j_{\text{end}}$; (vii) $i_{\text{st}} = j_{\text{st}}$ and $i_{\text{end}} < j_{\text{end}}$; (viii) $i_{\text{st}} = j_{\text{st}}$ and $i_{\text{end}} > j_{\text{end}}$; and (ix) $i_{\text{st}} = j_{\text{st}}$ and $i_{\text{end}} = j_{\text{end}}$.

Note that **C0** is equivalent to (vi). Since we assumed that **C0** is false, G_{3ANA} can satisfy (N.2) with all the possible cases except (vi). We also note that (i) is equivalent to $\mathbf{C1} \wedge \mathbf{C4}$, (ii) is equivalent to $\mathbf{C1} \wedge \mathbf{C5}$, etc. By applying **S3** and **S6**, we have

• $\mathbf{E0} \wedge \mathbf{E1} \wedge \text{(i)} \Rightarrow (\mathbf{D4} \wedge \mathbf{D5}) \wedge (\mathbf{D2} \wedge \mathbf{D3})$.

By similarly applying **S3** to **S8**, we have the following relationships:

• $\mathbf{E0} \wedge \mathbf{E1} \wedge \text{(ii)} \Rightarrow (\mathbf{D4} \wedge \mathbf{D5}) \wedge \mathbf{D3}$.

• $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \text{(iii)} \Rightarrow (\mathbf{D4} \wedge \mathbf{D5}) \wedge \mathbf{D2}$.

• $\mathbf{E0} \wedge \mathbf{E1} \wedge \text{(iv)} \Rightarrow \mathbf{D1} \wedge (\mathbf{D2} \wedge \mathbf{D3})$.

• $\mathbf{E0} \wedge \mathbf{E1} \wedge \text{(v)} \Rightarrow \mathbf{D1} \wedge \mathbf{D3}$.

• $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \text{(vii)} \Rightarrow \mathbf{D4} \wedge (\mathbf{D2} \wedge \mathbf{D3})$.

• $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \text{(viii)} \Rightarrow \mathbf{D4} \wedge \mathbf{D3}$.

• $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \text{(ix)} \Rightarrow \mathbf{D4} \wedge \mathbf{D2}$.

Then, the above relationships jointly imply $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge (\neg\, \mathbf{C0}) \Rightarrow (\mathbf{D1} \wedge \mathbf{D3}) \vee (\mathbf{D2} \wedge \mathbf{D4}) \vee (\mathbf{D3} \wedge \mathbf{D4})$. The proof of **S10** is thus complete. ■

N.6 Proof of S11

N.6.1 The third set of logic statements

To prove **S11**, we need the third set of logic statements.

• **G7:** There exists an edge $\tilde{e}$ such that both the following conditions are satisfied: (i) $\tilde{e}$ can reach d_1 but cannot reach any of d_2 and d_3; and (ii) $\tilde{e}$ can be reached from s_1 but not from any of s_2 nor s_3.

• **G8:** $\overline{S}_3 \neq \emptyset$ and $\overline{D}_2 \neq \emptyset$.

The following logic statements are well-defined if and only if $\mathbf{G4} \wedge \mathbf{G8}$ is true. Recall the definition of e_3^* and e_2^* when **G4** is true.

• **G9:** $\{e_3^*, e_2^*\} \subset \mathsf{1cut}(s_2; d_3)$.

• **G10:** $e_3^* \in \mathsf{1cut}(s_2; d_1)$.

• **G11:** $e_3^* \in \mathsf{1cut}(s_1; d_1)$.

• **G12:** $e_2^* \in \mathsf{1cut}(s_1; d_3)$.

• **G13:** $e_2^* \in \mathsf{1cut}(s_1; d_1)$.

The following logic statements are well-defined if and only if $\neg\, \mathbf{G4}$ is true. Recall the definition of e_u^{32} and e_v^{32} when $\neg\, \mathbf{G4}$ is true.

• **G14:** $e_u^{32} \not\in \mathsf{1cut}(s_1; d_1)$.

• **G15:** Let $\tilde{e}_u$ denote the most downstream edge among $\mathsf{1cut}(s_1; d_1) \cap \mathsf{1cut}(s_1; \mathsf{tail}(e_u^{32}))$. Also let $\tilde{e}_v$ denote the most upstream edge among $\mathsf{1cut}(s_1; d_1) \cap \mathsf{1cut}(\mathsf{head}(e_v^{32}); d_1)$. Then we have (a) $\mathsf{head}(\tilde{e}_u) \prec \mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32}) \prec \mathsf{tail}(\tilde{e}_v)$; there exists a s_1-to-d_1 path P_{11}^* through $\tilde{e}_u$ and $\tilde{e}_v$ satisfying the following two conditions: (b) P_{11}^* is vertex-disjoint from any s_3-to-d_2 path; and (c) there exists an edge $\tilde{e} \in P_{11}^*$ where $\tilde{e}_u \prec \tilde{e} \prec \tilde{e}_v$ that is not reachable from any of $\{e_u^{32}, e_v^{32}\}$.

N.6.2 The skeleton of proving S11

We prove the following relationships, which jointly prove **S11**. The proofs for the following statements are relegated to Appendix N.6.3.

• **R1:** $\mathbf{D1} \Rightarrow \mathbf{G8}$.

• **R2:** $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{D1} \Rightarrow \mathbf{G9}$.

• **R3:** $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge \mathbf{D3} \Rightarrow (\mathbf{G10} \vee \mathbf{G11}) \wedge (\mathbf{G12} \vee \mathbf{G13})$.

• **R4:** $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge (\neg\,\mathbf{G10}) \wedge \mathbf{G11} \wedge \mathbf{E0} \Rightarrow$ false.

• **R5:** $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge (\neg\,\mathbf{G12}) \wedge \mathbf{G13} \wedge \mathbf{E0} \Rightarrow$ false.

• **R6:** $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge \mathbf{G10} \wedge \mathbf{G12} \Rightarrow (\neg\,\mathbf{LNR})$.

• **R7:** $\mathbf{G1} \wedge (\neg\,\mathbf{G4}) \Rightarrow \mathbf{G14}$.

• **R8:** $(\neg\,\mathbf{G4}) \wedge \mathbf{G14} \Rightarrow \mathbf{G15}$.

• **R9:** $(\neg\,\mathbf{G4}) \wedge \mathbf{G14} \wedge \mathbf{D3} \Rightarrow \mathbf{G7}$.

• **R10:** $\mathbf{G7} \wedge \mathbf{E0} \Rightarrow$ false.

One can easily verify that jointly **R4** to **R6** imply

$$\mathbf{LNR} \wedge \mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge \mathbf{E0} \wedge (\mathbf{G10} \vee \mathbf{G11}) \wedge (\mathbf{G12} \vee \mathbf{G13}) \Rightarrow \text{false}. \tag{N.11}$$

Together with **R3**, (N.11) reduces to

$$\mathbf{LNR} \wedge \mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge \mathbf{E0} \wedge \mathbf{D3} \Rightarrow \text{false}. \tag{N.12}$$

Jointly with **R1** and **R2**, (N.12) further reduces to

$$\mathbf{LNR} \wedge \mathbf{G4} \wedge \mathbf{E0} \wedge \mathbf{D1} \wedge \mathbf{D3} \Rightarrow \text{ false.} \tag{N.13}$$

In addition, **R7**, **R9**, and **R10** jointly imply

$$\mathbf{G1} \wedge (\neg\, \mathbf{G4}) \wedge \mathbf{E0} \wedge \mathbf{D3} \Rightarrow \text{ false.} \tag{N.14}$$

One can easily verify that jointly (N.13) and (N.14) imply **S11**. The skeleton of the proof of **S11** is complete.

N.6.3 Proofs of R1 to R10

We prove **R1** as follows.

Proof. Suppose **D1** is true. By Corollary 5.4.2, any channel gain cannot have the other channel gain as a factor. Therefore, m_{32} must be reducible. Furthermore we must have $\mathsf{GCD}(\, m_{12},\, m_{32}) \not\equiv 1$ since m_{12} is the only channel gain in the LHS of **D1** that reaches d_2. (See the proof of Lemma 6.2.1 for detailed discussion). Similarly, we must have $\mathsf{GCD}(\, m_{31},\, m_{32}) \not\equiv 1$. Lemma 6.1.7 then implies $\overline{S}_3 \neq \emptyset$ and $\overline{D}_2 \neq \emptyset$. ■

We prove **R2** as follows.

Proof. Suppose $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{D1}$ is true. From $\mathbf{G4} \wedge \mathbf{G8}$ being true, by definition, e_3^* (resp. e_2^*) is the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_2$) and $e_3^* \prec e_2^*$. For the following, we will prove that $\{e_3^*, e_2^*\} \subset \mathsf{1cut}(s_2; d_3)$.

We now consider $m_{e_3^*;e_2^*}$, a part of m_{32}. From **D1** and Property 2 of **G4**, we have

$$\mathsf{GCD}(\, m_{23}^{l_1},\ m_{e_3^*;e_2^*}) = m_{e_3^*;e_2^*}, \tag{N.15}$$

for some positive integer l_1. This implies that $m_{e_3^*;e_2^*}$ is a factor of m_{23}. By Proposition 5.4.3, we have $\{e_3^*, e_2^*\} \subset \mathsf{1cut}(s_2; d_3)$. The proof is thus complete. ■

We prove **R3** as follows.

Proof. Suppose $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge \mathbf{D3}$ is true. Therefore, the e_3^* (resp. e_2^*) defined in the properties of **G4** must also be the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_2$). Moreover, since $\{e_3^*, e_2^*\} \subset \mathsf{1cut}(s_2; d_3)$, we can express m_{23} as $m_{23} = m_{e_{s_2}; e_3^*} m_{e_3^*; e_2^*} m_{e_2^*; e_{d_3}}$. For the following, we will prove that $e_3^* \in \mathsf{1cut}(s_1; d_1) \cup \mathsf{1cut}(s_2; d_1)$.

We use the following observation: For any edge $e' \in \mathsf{1cut}(s_3; d_2)$ that is in the upstream of e_2^*, there must exist a path from s_1 to $\mathsf{tail}(e_2^*)$ that does not use such e'. Otherwise, $e' \in \mathsf{1cut}(s_3; d_2)$ is also a 1-edge cut separating s_1 and d_2, which contradicts that e_2^* is the most upstream edge of $\overline{D}_2$.

We now consider $m_{e_3^*; e_{d_1}}$, a factor of m_{31}. From **D3** and Property 2 of **G4**, we have $\mathsf{GCD}(\, m_{11} m_{13}^{l_3} m_{21}^{l_3},\, m_{e_3^*; e_{d_1}}) = m_{e_3^*; e_{d_1}}$. By Proposition 5.4.3, we must have $e_3^* \in \mathsf{1cut}(s_1; d_1) \cup \mathsf{1cut}(s_1; d_3) \cup \mathsf{1cut}(s_2; d_1)$. We also note that by the observation in the beginning of this proof, there exists a path from s_1 to $\mathsf{tail}(e_2^*)$ not using e_3^*. Furthermore, $e_2^* \in \mathsf{1cut}(s_2; d_3)$ implies that e_2^* can reach d_3. These jointly shows that there exists a path from s_1 through e_2^* to d_3 without using e_3^*, which means $e_3^* \notin \mathsf{1cut}(s_1; d_3)$. Therefore, e_3^* belongs to $\mathsf{1cut}(s_1; d_1) \cup \mathsf{1cut}(s_2; d_1)$. The proof of $e_2^* \in \mathsf{1cut}(s_1; d_1) \cup \mathsf{1cut}(s_1; d_3)$ can be derived similarly. The proof **R3** is thus complete. ■

We prove **R4** as follows.

Proof. Assume $\mathbf{G4} \wedge \mathbf{G8} \wedge \mathbf{G9} \wedge (\neg\, \mathbf{G10}) \wedge \mathbf{G11} \wedge \mathbf{E0}$ is true. Recall that e_3^* is the most downstream edge in $\overline{S}_3$ and e_2^* is the most upstream edge in $\overline{D}_2$. For the following we construct 8 path segments that interconnects s_1 to s_3, d_1 to d_3, and two edges e_3^* and e_2^*.

- P_1: a path from s_1 to $\mathsf{tail}(e_2^*)$ without using e_3^*. This is always possible due to Properties 1 and 2 of **G4**.
- P_2: a path from s_2 to $\mathsf{tail}(e_3^*)$. This is always possible due to **G8** and **G9** being true.

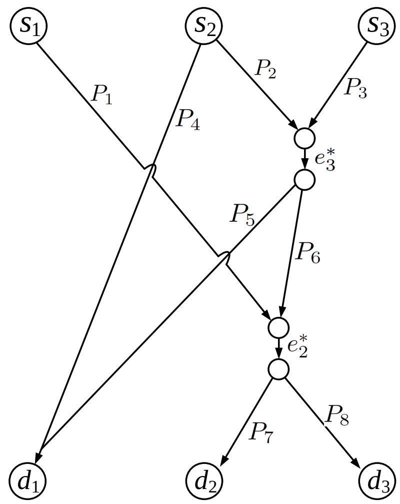


Fig. N.1. The subgraph G' of the 3-unicast ANA network $G_{3\text{ANA}}$ induced by the union of the 8 paths plus two edges e_3^* and e_2^* in the proof of **R4**.

- P_3: a path from s_3 to $\mathsf{tail}(e_3^*)$. This is always possible due to **G4** and **G8** being true.
- P_4: a path from s_2 to d_1 without using e_3^*. This is always possible due to **G10** being false.
- P_5: a path from $\mathsf{head}(e_3^*)$ to d_1 without using e_2^*. This is always possible due to Properties 1 and 2 of **G4**.
- P_6: a path from $\mathsf{head}(e_3^*)$ to $\mathsf{tail}(e_2^*)$. This is always possible due to Property 1 of **G4**.
- P_7: a path from $\mathsf{head}(e_2^*)$ to d_2. This is always possible due to **G4** and **G8** being true.
- P_8: a path from $\mathsf{head}(e_2^*)$ to d_3. This is always possible due to **G8** and **G9** being true.

Fig. N.1 illustrates the relative topology of these 8 paths. We now consider the subgraph G' induced by the 8 paths and two edges e_3^* and e_2^*. One can easily check that s_i can reach d_j for any $i \neq j$. In particular, s_1 can reach d_2 through $P_1 e_2^* P_7$; s_1 can reach d_3 through $P_1 e_2^* P_8$; s_2 can reach d_1 through either P_4 or $P_2 e_3^* P_5$; s_2 can reach d_3 through $P_2e_3^*P_6e_2^*P_8$; s_3 can reach d_1 through $P_3e_3^*P_5$; and s_3 can reach d_2 through $P_3e_3^*P_6e_2^*P_7$.

We first show the following topological relationships: P_1 is vertex-disjoint with P_2, P_3, and P_4, respectively, in the induced subgraph G'. From **G9**, $\{P_1, P_2\}$ must be vertex-disjoint paths otherwise s_2 can reach d_3 without using $e_3^* \in \mathsf{1cut}(s_2; d_3)$. Similarly from the fact that $e_3^* \in \overline{S}_3$, $\{P_1, P_3\}$ must be vertex-disjoint paths. Also notice that by **G11**, e_3^* is a 1-edge cut separating s_1 and d_1 in the original graph. Therefore anys_1-to-$d_1$path in the subgraph must use e_3^* as well. But by definition, both P_1 and P_4 do not use e_3^* and s_1 can reach d_1 if they share a vertex. This thus implies that $\{P_1, P_4\}$ are vertex-disjoint paths.

The above topological relationships further imply that s_1 cannot reach d_1 in the induced subgraph G'. The reason is as follows. We first note that P_1 is the only path segment that s_1 can use to reach other destinations, and anys_1-to-$d_1$path, if exists, must use path segment P_1 in the very beginning. Since P_1 ends at $\mathsf{tail}(e_2^*)$, using path segment P_1 alone is not possible to reach d_1. Therefore, if as_1-to-$d_1$path exists, then at some point, it must use one of the other 7 path segments P_2 to P_8. On the other hand, we also note that $e_3^* \in \mathsf{1cut}(s_1; d_1)$ and the path segments P_5 to P_8 are in the downstream of e_3^*. Therefore, for anys_1-to-$d_1$path, if it uses any of the vertices of P_5 to P_8, it must first go through $\mathsf{tail}(e_3^*)$, the end point of path segments P_2 and P_3. As a result, we only need to consider the scenario in which one of $\{P_2, P_3, P_4\}$ is used by thes_1-to-$d_1$path when this path switches from P_1 to a new path segment. But we have already showed that P_1 and $\{P_2, P_3, P_4\}$ are vertex-disjoint with each other. As a result, nos_1-to-$d_1$path can exist. Thus s_1 cannot reach d_1 on the induced graph G'.

By **E0** being true and Proposition 5.4.2, any subgraph who contains the source and destination edges (hence G') must satisfy **E0**. Note that we already showed there is nos_1-to-$d_1$path on G'. Recalling (N.1), its LHS becomes zero. Thus, we have $g(\{m_{ij} : \forall\,(i,j) \in I_{3\text{ANA}}\})\, \psi_\beta^{(n)}(R, L) = 0$ with at least one non-zero coefficient β_j. But note also that any channel gain m_{ij} where $i \neq j$ is non-trivial on G'. Thus R, L, and $g(\{m_{ij} : \forall\,(i,j) \in I_{3\text{ANA}}\})$ are all non-zero polynomials. Therefore, G' must satisfy $\psi_\beta^{(n)}(R, L) = 0$ with at least one non-zero coefficient β_j and this further implies that the set of polynomials $\{R^n, R^{n-1}L, \cdots, RL^{n-1}, L^n\}$ is linearly dependent on G'. Since this is the Vandermonde form, it is equivalent to that $L \equiv R$ holds on G'.

For the following, we further show that in the induced graph G', the following three statements are true: (a) $\overline{S}_2 \cap \overline{S}_3 = \emptyset$; (b) $\overline{S}_1 \cap \overline{S}_2 = \emptyset$; and (c) $\overline{S}_1 \cap \overline{S}_3 = \emptyset$, which implies by Proposition 6.2.1 that G' must have $L \not\equiv R$. We thus have a contradiction.

(a) $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G': Suppose there is an edge $e \in \overline{S}_2 \cap \overline{S}_3$ on G'. Since $e \in \overline{S}_2$, such e must belong to P_4 and anys_2-to-$d_3$path. Since both $e \in P_4$ and $e_3^* \notin P_4$ belong to $\mathsf{1cut}(s_2; d_3)$, we have either $e \prec e_3^*$ or $e \succ e_3^*$. We first note that e must not be in the downstream of e_3^*. Otherwise, s_2 can use P_4 to reach e without using e_3^* and finally to d_3 (since $e \in \overline{S}_2$), which contradicts the assumption of **G9** that $e_3^* \in \mathsf{1cut}(s_2; d_3)$. As a result, $e \prec e_3^*$ and any path from s_2 to $\mathsf{tail}(e_3^*)$ must use e. This in turn implies that P_2 uses e. We now argue that P_3 must also use e. The reason is that thes_3-to-$d_1$path $P_3 e_3^* P_5$ must use e since $e \in \overline{S}_3$ and $e \prec e_3^*$. Then these jointly contradict that $e_3^* \in \overline{S}_3$ since s_3 can follow P_3, switch to P_4 through e, and reach d_1 without using e_3^*.

(b) $\overline{S}_1 \cap \overline{S}_2 = \emptyset$ on G': Suppose there is an edge $e \in \overline{S}_1 \cap \overline{S}_2$. Since $e \in \overline{S}_2$, by the same arguments as used in proving (a), we know that $e \prec e_3^*$ and e must be used by both P_2 and P_4. We then note that e must also be used by thes_1-to-$d_3$path $P_1 e_2^* P_8$ since $e \in \overline{S}_1$. This in turn implies that P_1 must use e since $e \prec e_3^* \prec e_2^*$. However, these jointly contradict the fact P_1 and $\{P_2, P_3, P_4\}$ being vertex-disjoint, which were proved previously. The proof of (b) is complete.

(c) $\overline{S}_1 \cap \overline{S}_3 = \emptyset$ on G': Suppose there is an edge $e \in \overline{S}_1 \cap \overline{S}_3$. We then note that e must be used by thes_1-to-$d_3$path $P_1 e_2^* P_8$ since $e \in \overline{S}_1$. Then e must be either e_3^* or used by P_3 since e_3^* is the most downstream edge of $\overline{S}_3$. Therefore, P_1 must use e (since $e_3^* \prec e_2^*$). In addition, since by our construction P_1 does not use e_3^*, it is P_3 who uses e. However, P_1 and P_3 are vertex-disjoint with each other, which contradicts what we just derived $e \in P_1 \cap P_3$. The proof of (c) is complete. ■

We prove **R5** as follows.

Proof. We notice that **R5** is a symmetric version of **R4** by simultaneously reversing the roles of sources and destinations and relabeling flow 2 by flow 3, i.e., we swap the roles of the following three pairs: (s_1, d_1), (s_2, d_3), and (s_3, d_2). We can then reuse the proof of **R4**. ■

We prove **R6** as follows.

Proof. Assume $\mathbf{G4} \wedge \mathbf{G8}$ is true and recall that e_3^* is the most downstream edge in $\overline{S}_3$ and e_2^* is the most upstream edge in $\overline{D}_2$. From $\mathbf{G9} \wedge \mathbf{G10} \wedge \mathbf{G12}$ being true, we further have $e_3^* \in \mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_2; d_3)$ and $e_2^* \in \mathsf{1cut}(s_1; d_3) \cap \mathsf{1cut}(s_2; d_3)$. This implies that e_3^* (resp. e_2^*) belongs to $\overline{S}_2 \cap \overline{S}_3$ (resp. $\overline{D}_2 \cap \overline{D}_3$). We thus have $\neg\,\mathbf{LNR}$ by Proposition 6.2.1. ■

We prove **R7** as follows.

Proof. We prove an equivalent relationship: $(\neg\,\mathbf{G4}) \wedge (\neg\,\mathbf{G14}) \Rightarrow (\neg\,\mathbf{G1})$. From **G4** being false, we have $e_u^{32} \in \overline{S}_3 \cap \overline{D}_2 \subset \mathsf{1cut}(s_3; d_2) \cap \mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(s_3; d_1)$. From **G14** being false, we have $e_u^{32} \in \mathsf{1cut}(s_1; d_1)$. As a result, e_u^{32} is a 1-edge cut separating $\{s_1, s_3\}$ and $\{d_1, d_2\}$. This implies $m_{11}m_{32} \equiv m_{12}m_{31}$ and thus $\neg\,\mathbf{G1}$. The proof of **R7** is thus complete. ■

We prove **R8** as follows.

Proof. Suppose that $(\neg\,\mathbf{G4}) \wedge \mathbf{G14}$ is true. From Property 3 of $\neg\,\mathbf{G4}$, any s_1-to-d_1 path who uses a vertex w where $\mathsf{tail}(e_u^{32}) \preceq w \preceq \mathsf{head}(e_v^{32})$ must use both e_u^{32} and e_v^{32}. Since we have $e_u^{32} \notin \mathsf{1cut}(s_1; d_1)$ from **G14**, there must exist a s_1-to-d_1 path not using e_u^{32}. Then, these jointly imply that there exists a s_1-to-d_1 path which does not use any vertex in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$. Fix arbitrarily one such path as P_{11}^*.

If the chosen P_{11}^* shares a vertex with any path segment from s_3 to $\mathsf{tail}(e_u^{32})$, then s_3 can reach d_1 without using e_u^{32}, contradicting $e_u^{32} \in \overline{S}_3 \cap \overline{D}_2 \subset \mathsf{1cut}(s_3; d_1)$. By the similar argument, P_{11}^* should not share a vertex with any path segment from

$\mathsf{head}(e_v^{32})$ to d_2. Then jointly with the above discussion, we can conclude that P_{11}^* is vertex-disjoint with anys_3-to-$d_2$path. We thus have proven (b) of **G15**.

Now consider $\tilde{e}_u$ (we have at least the s_1-source edge e_{s_1}) and $\tilde{e}_v$ (we have at least the d_1-destination edge e_{d_1}) defined in **G15**. By definition, $\tilde{e}_u \prec e_u^{32}$ and $e_v^{32} \prec \tilde{e}_v$, and the chosen P_{11}^* must use both $\tilde{e}_u$ and $\tilde{e}_v$. Thus if $\mathsf{head}(\tilde{e}_u) = \mathsf{tail}(e_u^{32})$, then this contradicts the above discussion since $\mathsf{tail}(e_u^{32}) \in P_{11}^*$. Therefore, it must be $\mathsf{head}(\tilde{e}_u) \prec \mathsf{tail}(e_u^{32})$. Similarly, it must also be $\mathsf{head}(e_v^{32}) \prec \mathsf{tail}(\tilde{e}_v)$. Thus we have proven (a) of **G15**.

We now prove (c) of **G15**. We prove this by contradiction. Fix arbitrarily one edge $e \in P_{11}^*$ where $\tilde{e}_u \prec e \prec \tilde{e}_v$ and assume that this edge e is reachable from either e_u^{32} or e_v^{32} or both. We first prove that whenever e_u^{32} reaches e, then e must be in the downstream of e_v^{32}. The reason is as follows. If e_u^{32} reaches e, then $e \in P_{11}^*$ should not reach e_v^{32} because it will be located in-between e_u^{32} and e_v^{32}, and this contradicts the above discussion. The case when e are e_v^{32} are not reachable from each other is also not possible because s_1 can first reach e through e_u^{32} and follow P_{11}^* to d_1 without using e_v^{32}, which contradicts the Property 3 of $\neg$**G4**. Thus, if $e_u^{32} \prec e$, then it must be $e_v^{32} \prec e$. By the similar argument, we can show that if $e \prec e_v^{32}$, it must be $e \prec e_u^{32}$. Therefore, only two cases are possible when e is reachable from either e_u^{32} or e_v^{32} or both: either $e \prec e_u^{32}$ or $e_v^{32} \prec e$. Extending this result to every edges of P_{11}^* from $\tilde{e}_u$ to $\tilde{e}_v$, we can group them into two: edges in the upstream of e_u^{32}; and edges in the downstream of e_v^{32}. Since $\tilde{e}_u \prec e_u^{32} \prec e_v^{32} \prec \tilde{e}_v$, this further implies that the chosen P_{11}^* must be disconnected. This, however, contradicts the construction P_{11}^*. Therefore, there must exist an edge $\tilde{e} \in P_{11}^*$ where $\tilde{e}_u \prec e \prec \tilde{e}_v$ that is not reachable from any of $\{e_u^{32}, e_v^{32}\}$. We thus have proven (c) of **G15**. The proof of **R8** is complete. ■

We prove **R9** as follows.

Proof. Suppose $(\neg$**G4**$) \wedge$ **G14** $\wedge$ **D3** is true. From **R8**, **G15** must also be true, and we will use thes_1-to-$d_1$path P_{11}^*, the two edges $\tilde{e}_u$ and $\tilde{e}_v$, and the edge $\tilde{e} \in P_{11}^*$ defined in **G15**. For the following, we will prove that the specified $\tilde{e}$ satisfies **G7**. Since

$\tilde{e} \in P_{11}^*$, we only need to prove that $\tilde{e}$ cannot be reached by any of $\{s_2, s_3\}$ and cannot reach any of $\{d_2, d_3\}$.

We first claim that $\tilde{e}$ cannot be reached from s_3. Suppose not. Then we can consider a new s_3-to-d_1 path: s_3 can reach $\tilde{e}$ and follow P_{11}^* to d_1. Since $\tilde{e}$ is not reachable from any of $\{e_u^{32}, e_v^{32}\}$ by (c) of **G15**, this new s_3-to-d_1 path must not use any of $\{e_u^{32}, e_v^{32}\}$. However, this contradicts the construction $\{e_u^{32}, e_v^{32}\} \subset \overline{S}_3 \cap \overline{D}_2 \subset \mathsf{1cut}(s_3; d_1)$. We thus have proven the first claim that $\tilde{e}$ cannot be reached from s_3. Symmetrically, we can also prove that $\tilde{e}$ cannot reach d_2.

What remains to be proven is that $\tilde{e}$ cannot be reached from s_2 and cannot reach d_3. Since **D3** is true, there exists a positive integer l_3 satisfying $\mathsf{GCD}(m_{11} m_{13}^{l_3} m_{32}^{l_3} m_{21}^{l_3},\, m_{12} m_{31}) = m_{12} m_{31}$. Consider $m_{e_{s_1}; e_u^{32}}$, a part of m_{12}, and $m_{e_v^{32}; e_{d_1}}$, a part of m_{31}. By Property 1 of $\neg$ **G4**, we have

$$\mathsf{GCD}(m_{11} m_{13}^{l_3} m_{21}^{l_3},\ m_{e_{s_1}; e_u^{32}} m_{e_v^{32}; d_{d_1}}) = m_{e_{s_1}; e_u^{32}} m_{e_v^{32}; d_{d_1}}.$$

Recall the definition of $\tilde{e}_u$ (resp. $\tilde{e}_v$) being the most downstream (resp. upstream) edge among $\mathsf{1cut}(s_1; \mathsf{tail}(e_v^{32})) \cap \mathsf{1cut}(s_1; d_1)$ (resp. $\mathsf{1cut}(\mathsf{head}(e_v^{32}); d_1) \cap \mathsf{1cut}(s_1; d_1)$). Then we can further factorize $m_{e_{s_1}; e_u^{32}} = m_{e_{s_1}; \tilde{e}_u} m_{\tilde{e}_u; e_u^{32}}$ and $m_{e_v^{32}; e_{d_1}} = m_{e_v^{32}; \tilde{e}_v} m_{\tilde{e}_v; e_{d_1}}$, respectively. Since both $\tilde{e}_u$ and $\tilde{e}_v$ separate s_1 and d_1, we can express m_{11} as $m_{11} = m_{e_{s_1}; \tilde{e}_u} m_{\tilde{e}_u; \tilde{e}_v} m_{\tilde{e}_v; e_{d_1}}$. Then one can see that the middle part of m_{11}, i.e., $m_{\tilde{e}_u; \tilde{e}_v}$, must be co-prime to both $m_{\tilde{e}_u; e_u^{32}}$ and $m_{e_v^{32}; \tilde{e}_v}$, otherwise it violates the construction of $\tilde{e}_u$ (resp. $\tilde{e}_v$) being the most downstream (resp. upstream) edge among $\mathsf{1cut}(s_1; \mathsf{tail}(e_v^{32})) \cap \mathsf{1cut}(s_1; d_1)$ (resp. $\mathsf{1cut}(\mathsf{head}(e_v^{32}); d_1) \cap \mathsf{1cut}(s_1; d_1)$). The above equation thus reduces to

$$\mathsf{GCD}(m_{13}^{l_3} m_{21}^{l_3},\ m_{\tilde{e}_u; e_u^{32}} m_{e_v^{32}; \tilde{e}_v}) = m_{\tilde{e}_u; e_u^{32}} m_{e_v^{32}; \tilde{e}_v}. \tag{N.16}$$

Using (N.16) and the previous constructions, we first prove that $\tilde{e}$ cannot reach d_3. Since $\mathsf{head}(\tilde{e}_u) \prec \mathsf{tail}(e_u^{32})$ by (a) of **G15**, we must have $0 < \mathsf{EC}(\mathsf{head}(\tilde{e}_u); \mathsf{tail}(e_u^{32})) < \infty$. By Proposition 5.4.3, $m_{\tilde{e}_u; e_u^{32}}$ is either irreducible or the product of irreducibles corresponding to the consecutive edges among $\tilde{e}_u$, $\mathsf{1cut}(\mathsf{head}(\tilde{e}_u);\mathsf{tail}(e_u^{32}))$, and e_u^{32}. Consider the following edge set $E_u = \{\tilde{e}_u\} \cup \mathsf{1cut}(\mathsf{head}(\tilde{e}_u);\mathsf{tail}(e_u^{32})) \cup \{e_u^{32}\}$, the collection of $\mathsf{1cut}(\mathsf{head}(\tilde{e}_u);\mathsf{tail}(e_u^{32}))$ and two edges $\tilde{e}_u$ and e_u^{32}. Note that in the proof of **R8**, P_{11}^* was chosen arbitrarily such that $\tilde{e}_u \in P_{11}^*$ and $e_u^{32} \notin P_{11}^*$ but there was no consideration for the 1-edge cuts from $\mathsf{head}(\tilde{e}_u)$ to $\mathsf{tail}(e_u^{32})$ if non-empty. In other words, when s_1 follow the chosen P_{11}^*, it is obvious that it first meets $\tilde{e}_u$ but it is not sure when it starts to deviate not to use e_u^{32} if we have non-empty $\mathsf{1cut}(\mathsf{head}(\tilde{e}_u);\mathsf{tail}(e_u^{32}))$. Let e_1^u denote the most downstream edge of $E_u \cap P_{11}^*$ (we have at least $\tilde{e}_u$) and let e_2^u denote the most upstream edge of $E_u \backslash P_{11}^*$ (we have at least e_u^{32}). From the constructions of P_{11}^* and E_u, the defined edges $e_1^u \in P_{11}^*$ and $e_2^u \notin P_{11}^*$ are edges of E_u such that $\tilde{e}_u \preceq e_1^u \prec e_2^u \preceq e_u^{32}$; $m_{e_1^u;e_2^u}$ is irreducible; and $m_{\tilde{e}_u;e_u^{32}}$ contain $m_{e_1^u;e_2^u}$ as a factor. By doing this way, we can clearly specify the location (in-between $e_1^u \in P_{11}^*$ and $e_2^u \notin P_{11}^*$) when P_{11}^* starts to deviate not to use e_u^{32}.

For the following, we first argue that $\mathsf{GCD}(m_{13}, m_{e_1^u;e_2^u}) \not\equiv 1$. Suppose not then we have $\mathsf{GCD}(m_{21}, m_{e_1^u;e_2^u}) = m_{e_1^u;e_2^u}$ from (N.16). By Proposition 5.4.3, we have $\{e_1^u, e_2^u\} \subset \mathsf{1cut}(s_2;d_1)$. However from the above construction, $e_1^u \in \mathsf{1cut}(s_2;d_1)$ implies that s_2 can first reach $e_1^u \in P_{11}^*$ and then follow P_{11}^* to d_1 without using e_2^u since $e_1^u \prec e_2^u$ and $e_2^u \notin P_{11}^*$. This contradicts $e_2^u \in \mathsf{1cut}(s_2;d_1)$ that we just established. This thus proves that $\mathsf{GCD}(m_{13}, m_{e_1^u;e_2^u}) \not\equiv 1$. Since $m_{e_1^u;e_2^u}$ is irreducible, again by Proposition 5.4.3, we further have $\{e_1^u, e_2^u\} \subset \mathsf{1cut}(s_1;d_3)$.

We now argue that $\tilde{e}$ cannot reach d_3. Suppose not and assume that there exists a path segment Q from $\tilde{e}$ to d_3. Since $\tilde{e} \in P_{11}^*$ is not reachable from any of $\{e_u^{32}, e_v^{32}\}$ by (c) of **G15**, it is obvious that $\tilde{e}$ must be in the downstream of $e_1^u \in P_{11}^*$ since $e_1^u \prec e_u^{32}$ from the above construction. Then when s_1 follow P_{11}^* to $\tilde{e}$ (through e_1^u) and switch to Q to reach d_3, it will not use e_2^u unless $\tilde{e} \prec e_2^u$ and $e_2^u \in Q$, but $\tilde{e}$ cannot be in the upstream of e_2^u since $e_2^u \preceq e_u^{32}$ from the above construction. Therefore, thiss_1-to-$d_3$path $P_{11}^*\tilde{e}Q$ will not use e_2^u and thus contradicts $e_2^u \in \mathsf{1cut}(s_1;d_3)$ that we just established. As a result, $\tilde{e}$ cannot reach d_3.

The proof that $\tilde{e}$ cannot be reached from s_2 can be derived symmetrically. In particular, we can apply the above proof arguments ($\tilde{e}$ cannot reach d_3) by symmetrically using the following: the edge set $E_v = \{e_v^{32}\} \cup \mathsf{1cut}(\mathsf{head}(e_v^{32}); \mathsf{tail}(\tilde{e}_v)) \cup \{\tilde{e}_v\}$ and denote e_1^v (resp. e_2^v) be the most downstream (resp. upstream) edge of $E_v \backslash P_{11}^*$ (resp. $E_v \cap P_{11}^*$) such that $\{e_1^v, e_2^v\} \subset \mathsf{1cut}(s_2; d_1)$ from (N.16).

Therefore we have proven that $\tilde{e}$ cannot be reached from s_2 and cannot reach d_3. The proof of **R9** is thus complete. ■

We prove **R10** as follows.

Proof. We prove an equivalent relationship: **G7** $\Rightarrow$ ($\neg$ **E0**). Suppose **G7** is true and consider the edge $\tilde{e}$ defined in **G7**. Consider ans_1-to-$d_1$path P_{11} that uses $\tilde{e}$ and an edge $e \in P_{11}$ that is immediate downstream of $\tilde{e}$ along this path, i.e., $\mathsf{head}(\tilde{e}) = \mathsf{tail}(e)$. Such edge e always exists since $\tilde{e}$ cannot be the d_1-destination edge e_{d_1}. (Recall that $\tilde{e}$ cannot be reached by s_2.) We now observe that since **G7** is true, such e cannot reach any of $\{d_2, d_3\}$ (otherwise $\tilde{e}$ can reach one of $\{d_2, d_3\}$). Now consider a local kernel $x_{\tilde{e}e}$ from $\tilde{e}$ to e. Then, one can see that by the facts that $\tilde{e}$ cannot be reached by any of $\{s_2, s_3\}$ and e cannot reach any of $\{d_2, d_3\}$, any channel gain m_{ij} where $i \neq j$ cannot depend on $x_{\tilde{e}e}$. On the other hand, the channel gain polynomial m_{11} has degree 1 in $x_{\tilde{e}e}$ since both $\tilde{e}$ and e are used by a path P_{11}.

Since any channel gain m_{ij} where $i \neq j$ is non-trivial on a given G_{3ANA}, the above discussion implies that $f(\{m_{ij} : \forall\,(i,j) \in I_{\text{3ANA}}\})$, $g(\{m_{ij} : \forall\,(i,j) \in I_{\text{3ANA}}\})$, R, and L become all non-zero polynomials, any of which does not depend on $x_{\tilde{e}e}$. Thus recalling (N.1), its RHS does not depend on $x_{\tilde{e}e}$. However, the LHS of (N.1) has a common factor m_{11} and thus has degree 1 in $x_{\tilde{e}e}$. This implies that G_{3ANA} does not satisfy (N.1) if we have at least one non-zero coefficient α_i and β_j, respectively. This thus implies $\neg$ **E0**. ■

N.7 Proof of S12

If we swap the roles of sources and destinations, then the proof of **S11** in Appendix N.6 can be directly applied to show **S12**. More specifically, note that **D1** (resp. **D3**) are converted back and forth from **D2** (resp. **D4**) by such $(s, d\,)$-swapping. Also, one can easily verify that **LNR**, **G1**, and **E0** remain the same after the index swapping. Thus we can see that **S11** becomes **S12** after reverting flow indices. The proofs of **S11** in Appendix N.6 can thus be used to prove **S12**.

N.8 Proof of S13

N.8.1 The fourth set of logic statements

To prove **S13**, we need the fourth set of logic statements.

- **G16:** There exists a subgraph $G' \subset G_{3\text{ANA}}$ such that in G' both the following conditions are true: (i) s_i can reach d_j for all $i \neq j$; and (ii) s_1 can reach d_1.
- **G17:** Continue from the definition of **G16**. The considered subgraph G' also contains an edge $\tilde{e}$ such that both the following conditions are satisfied: (i) $\tilde{e}$ can reach d_1 but cannot reach any of $\{d_2, d_3\}$; (ii) $\tilde{e}$ can be reached from s_1 but not from any of $\{s_2, s_3\}$.
- **G18:** Continue from the definition of **G16**. There exists a subgraph $G'' \subset G'$ such that (i) s_i can reach d_j for all $i \neq j$; and (ii) s_1 can reach d_1. Moreover, the considered subgraph G'' also satisfies (iii) $m_{11}m_{23} = m_{13}m_{21}$; and (iv) $L \not\equiv R$.
- **G19:** Continue from the definition of **G16**. There exists a subgraph $G'' \subset G'$ such that (i) s_i can reach d_j for all $i \neq j$; and (ii) s_1 can reach d_1. Moreover, the considered subgraph G'' also satisfies (iii) $m_{11}m_{32} = m_{12}m_{31}$; and (iv) $L \not\equiv R$.
- **G20:** $\overline{S}_2 \neq \emptyset$ and $\overline{D}_3 \neq \emptyset$.

The following logic statements are well-defined if and only if **G3** $\wedge$ **G20** is true. Recall the definition of e_2^* and e_3^* when **G3** is true.

- **G21:** $\{e_2^*, e_3^*\} \subset \mathsf{1cut}(s_3; d_2)$.

The following logic statements are well-defined if and only if $(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})$ is true. Recall the definition of e_u^{23}, e_v^{23}, e_u^{32}, and e_v^{32} when $(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})$ is true.

• **G22:** There exists a path P_{11}^* from s_1 to d_1 who does not use any vertex in-between $\mathsf{tail}(e_u^{23})$ and $\mathsf{head}(e_v^{23})$, and any vertex in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$.

• **G23:** $e_u^{23} \prec e_u^{32}$.

• **G24:** $e_u^{32} \prec e_u^{23}$.

• **G25:** $e_u^{32} \prec e_v^{23}$.

• **G26:** $e_u^{23} \prec e_v^{32}$.

N.8.2 The skeleton of proving S13

We prove the following relationships, which jointly prove **S13**. The proofs for the following statements are relegated to Appendix N.8.3.

• **R11:** $\mathbf{D1} \Rightarrow \mathbf{G8}$ (identical to **R1**).

• **R12:** $\mathbf{G4}\wedge\mathbf{G8}\wedge\mathbf{D1} \Rightarrow \mathbf{G9}$ (identical to **R2**).

• **R13:** $\mathbf{LNR}\wedge\mathbf{G4}\wedge\mathbf{G8}\wedge\mathbf{G9}\wedge\mathbf{D2} \Rightarrow$ false.

• **R14:** $\mathbf{D2} \Rightarrow \mathbf{G20}$.

• **R15:** $\mathbf{G3}\wedge\mathbf{G20}\wedge\mathbf{D2} \Rightarrow \mathbf{G21}$.

• **R16:** $\mathbf{LNR}\wedge\mathbf{G3}\wedge\mathbf{G20}\wedge\mathbf{G21}\wedge\mathbf{D1} \Rightarrow$ false.

• **R17:** $\mathbf{LNR}\wedge\mathbf{G2}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\neg\,\mathbf{G5}) \Rightarrow \mathbf{G7}$.

• **R18:** $\mathbf{G16}\wedge\mathbf{G17}\wedge\mathbf{E0} \Rightarrow$ false.

• **R19:** $\mathbf{G16}\wedge(\mathbf{G18}\vee\mathbf{G19})\wedge\mathbf{E0}\wedge\mathbf{E1}\wedge\mathbf{E2} \Rightarrow$ false.

• **R20:** $\mathbf{G1}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\neg\,\mathbf{G22})\wedge\mathbf{G23} \Rightarrow \mathbf{G16}\wedge\mathbf{G18}$.

• **R21:** $\mathbf{LNR}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G22}\wedge\mathbf{G23}\wedge\mathbf{G25} \Rightarrow \mathbf{G16}\wedge\mathbf{G17}$.

• **R22:** $\mathbf{LNR}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G22}\wedge\mathbf{G23}\wedge(\neg\,\mathbf{G25}) \Rightarrow \mathbf{G16}\wedge(\mathbf{G17}\vee\mathbf{G18})$.

• **R23:** $\mathbf{G1}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\neg\,\mathbf{G22})\wedge\mathbf{G24} \Rightarrow \mathbf{G16}\wedge\mathbf{G19}$.

• **R24:** $\mathbf{LNR}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G22}\wedge\mathbf{G24}\wedge\mathbf{G26} \Rightarrow \mathbf{G16}\wedge\mathbf{G17}$.

• **R25:** $\mathbf{LNR}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G22}\wedge\mathbf{G24}\wedge(\neg\,\mathbf{G26}) \Rightarrow \mathbf{G16}\wedge(\mathbf{G17}\vee\mathbf{G19})$.

One can easily verify that jointly **R11** to **R13** imply

$$\mathbf{LNR} \wedge \mathbf{G4} \wedge \mathbf{D1} \wedge \mathbf{D2} \Rightarrow \text{ false.} \tag{N.17}$$

Similarly, **R14** to **R16** jointly imply

$$\mathbf{LNR} \wedge \mathbf{G3} \wedge \mathbf{D1} \wedge \mathbf{D2} \Rightarrow \text{ false.} \tag{N.18}$$

Thus, (N.17) and (N.18) together imply

$$\mathbf{LNR} \wedge (\mathbf{G3} \vee \mathbf{G4}) \wedge \mathbf{D1} \wedge \mathbf{D2} \Rightarrow \text{ false.} \tag{N.19}$$

Now recall **R10**, i.e., $\mathbf{G7} \wedge \mathbf{E0} \Rightarrow$ false. Then, jointly **R10** and **R17** imply

$$\mathbf{LNR} \wedge \mathbf{G2} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge (\neg\,\mathbf{G5}) \wedge \mathbf{E0} \Rightarrow \text{false.} \tag{N.20}$$

One can easily verify that jointly **R18** and **R19** imply

$$\mathbf{G16} \wedge (\mathbf{G17} \vee \mathbf{G18} \vee \mathbf{G19}) \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{E2} \Rightarrow \text{false.} \tag{N.21}$$

One can see that jointly (N.21), **R20**, **R21**, and **R22** imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge \mathbf{G23} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{E2} \Rightarrow \text{false.} \tag{N.22}$$

By similar arguments as used in deriving (N.22), jointly (N.21), **R23**, **R24**, and **R25** imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge \mathbf{G24} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{E2} \Rightarrow \text{false.} \tag{N.23}$$

Since by definition $(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G5}\Rightarrow(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge(\mathbf{G23}\vee\mathbf{G24})$, jointly (N.22) and (N.23) imply

$$\mathbf{LNR}\wedge\mathbf{G1}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G5}\wedge\mathbf{E0}\wedge\mathbf{E1}\wedge\mathbf{E2}\Rightarrow\text{ false.}\tag{N.24}$$

By similar arguments as used in deriving (N.22), (N.24) and (N.20) further imply

$$\mathbf{LNR}\wedge\mathbf{G1}\wedge\mathbf{G2}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{E0}\wedge\mathbf{E1}\wedge\mathbf{E2}\Rightarrow\text{ false.}\tag{N.25}$$

Finally, one can easily verify that jointly (N.19) and (N.25) imply that we have $\mathbf{LNR}\wedge\mathbf{G1}\wedge\mathbf{G2}\wedge\mathbf{E0}\wedge\mathbf{E1}\wedge\mathbf{E2}\wedge\mathbf{D1}\wedge\mathbf{D2}\Rightarrow$ false, which proves **S13**. The skeleton of the proof of **S13** is complete.

N.8.3 Proofs of R11 to R25

Since **R11** and **R12** is identical to **R1** and **R2**, respectively, see Appendix N.6.3 for their proofs.

We prove **R13** as follows.

Proof. We prove an equivalent relationship: $\mathbf{G4}\wedge\mathbf{G8}\wedge\mathbf{G9}\wedge\mathbf{D2}\Rightarrow\neg\,\mathbf{LNR}$. Suppose $\mathbf{G4}\wedge\mathbf{G8}\wedge\mathbf{G9}$ is true. The e_3^* (resp. e_2^*) defined in the properties of **G4** must be the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_2$), both of which belongs to $\mathsf{1cut}(s_2;d_3)$.

For the following, we will prove that there exists an edge in-between $\{e_{s_2},e_{s_3}\}$ and e_3^* who belongs to $\overline{S}_2\cap\overline{S}_3$. We will also prove that there exists an edge in-between e_2^* and $\{e_{d_2},e_{d_3}\}$ who belongs to $\overline{D}_2\cap\overline{D}_3$. By Proposition 6.2.1 we thus have **LNR** being false.

Define a node $u=\mathsf{tail}(e_3^*)$. Since $e_3^*\in\mathsf{1cut}(s_2;d_3)$, u is reachable from s_2. Since $e_3^*\in\overline{S}_3$, u is also reachable form s_3. Consider the set of edges $\{\mathsf{1cut}(s_2;u)\cap\mathsf{1cut}(s_3;u)\}\cup\{e_3^*\}$ and choose e'' as the most upstream one (we have at least e_3^*). Let e' denote the most downstream edge of $\mathsf{1cut}(s_2;\mathsf{tail}(e''))$ (we have at least the s_2-source edge e_{s_2}).

Since we choose e' to be the most downstream one, by Proposition 5.4.3 the channel gain $m_{e';e''}$ must be irreducible.

Moreover, since $e_3^* \in \mathsf{1cut}(s_2;d_3)$, both e' and e'' must be in $\mathsf{1cut}(s_2;d_3)$. The reason is that by $e_3^* \in \mathsf{1cut}(s_2;d_3)$ any path from s_2 to d_3 must use e_3^*, which in turn implies that any path from s_2 to d_3 must use e'' since e'' separates s_2 and $\mathsf{tail}(e_3^*)$. Therefore $e'' \in \mathsf{1cut}(s_2;d_3)$. Similarly, any$s_2$-to-$d_3$path must use e'', which means any s_2-to-$d_3$path must use e' as well since $e' \in \mathsf{1cut}(s_2;\mathsf{tail}(e''))$. As a result, the channel gain m_{23} contains $m_{e';e''}$ as a factor.

Since **D2** is true, it implies that $m_{e';e''}$ must be a factor of one of the following three channel gains m_{13}, m_{32}, and m_{21}. We first argue that $m_{e';e''}$ is not a factor of m_{32}. The reason is that if $m_{e';e''}$ is a factor of m_{32}, then $e' \in \mathsf{1cut}(s_3;d_2)$, which means that $e' \in \mathsf{1cut}(s_3;\mathsf{tail}(e_3^*))$. Since e' is also in $\mathsf{1cut}(s_2;\mathsf{tail}(e_3^*))$, this contradicts the construction that e'' is the most upstream edge of $\mathsf{1cut}(s_2;\mathsf{tail}(e_3^*)) \cap \mathsf{1cut}(s_3;\mathsf{tail}(e_3^*))$.

Now we argue that $\mathsf{GCD}(m_{13}, m_{e';e''}) \equiv 1$. Suppose not. Then since $m_{e';e''}$ is irreducible, Proposition 5.4.3 implies that $\{e', e''\}$ are 1-edge cuts separating s_1 and d_3. Also from Property 1 of **G4**, there always exists a path segment from s_1 to e_2^* without using e_3^*. Since $e_2^* \in \mathsf{1cut}(s_2;d_3)$, e_2^* can reach d_3 and we thus have as_1-to-d_3 path without using e_3^*. However by the assumption that $e' \in \mathsf{1cut}(s_1;d_3)$, this chosen path must use e'. As a result, s_2 can first reach e' and then reach d_3 through the chosen path without using e_3^*, which contradicts the assumption **G9**, i.e., $e_3^* \in \mathsf{1cut}(s_2;d_3)$.

From the above discussion $m_{e';e''}$ must be a factor of m_{21}, which by Proposition 5.4.3 implies that $\{e', e''\}$ also belong to $\mathsf{1cut}(s_2;d_1)$. Since by our construction e'' satisfies $e'' \in \overline{S}_3 \cap \mathsf{1cut}(s_2;d_3)$, we have thus proved that $e'' \in \overline{S}_2 \cap \overline{S}_3$. The proof for the existence of an edge satisfying $\overline{D}_2 \cap \overline{D}_3$ can be followed symmetrically. The proof of **R12** is thus complete. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the proofs of **R11** to **R13** can also be used to prove **R14** to **R16**, respectively. More specifically, **D1** and **D2** are converted back and forth from each other when swapping the flow indices. The same thing happens between **G3** and **G4**; between **G20** and **G8**; and between **G21** and **G9**. Moreover, **LNR** remains the same after the index swapping. The above proofs can thus be used to prove **R14** to **R16**.

We prove **R17** as follows.

Proof. Suppose $\mathbf{LNR} \wedge \mathbf{G2} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge (\neg\,\mathbf{G5})$ is true. Recall the definitions of e_u^{23}, e_u^{32}, e_v^{23}, and e_v^{32} from Properties of both $\neg\,\mathbf{G3}$ and $\neg\,\mathbf{G4}$. Since $\mathbf{LNR} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4})$ is true, we have **G6** if we recall **N7**. Together with $\neg\,\mathbf{G5}$, e_u^{23} and e_u^{32} are distinct and not reachable from each other. Thus from **G2** being true, there must exist a s_1-to-$d_1$path who does not use any of $\{e_u^{23}, e_u^{32}\}$. Combined with Property 3 of $\neg\,\mathbf{G3}$ and $\neg\,\mathbf{G4}$, this further implies that suchs_1-to-$d_1$path does not use any of $\{e_v^{23}, e_v^{32}\}$. Fix one suchs_1-to-$d_1$path as P_{11}^*.

We will now show that there exists an edge in P_{11}^* satisfying **G7**. To that end, we will show that if an edge $e \in P_{11}^*$ can be reached from s_2, then it must be in the downstream of e_v^{23}. We first argue that e_v^{23} and e are reachable from each other. The reason is that we now have as_2-to-$d_1$path by first going from s_2 to $e \in P_{11}^*$ and then use P_{11}^* to d_1. Since $e_v^{23} \in \mathsf{1cut}(s_2; d_1)$ by definition, such path must use e_v^{23}. As a result, we either have $e_v^{23} \prec e$ or $e \prec e_v^{23}$. ($e = e_v^{23}$ is not possible since $e_v^{23} \notin P_{11}^*$.) We then prove that $e \prec e_v^{23}$ is not possible. The reason is that P_{11}^* does not use e_u^{23} and thus s_1 must not reach e_v^{23} through P_{11}^* due to Property 3 of $\neg\,\mathbf{G3}$. As a result, we must have $e_v^{23} \prec e$. By symmetric arguments, any $e \in P_{11}^*$ that can be reached from reach s_3 must be in the downstream of e_v^{32} and any $e \in P_{11}^*$ that can reach d_3 (resp. d_2) must be in the upstream of e_u^{23} (resp. e_u^{32}).

For the following, we prove that there exists an edge $\tilde{e} \in P_{11}^*$ that cannot reach any of $\{d_2, d_3\}$, and that cannot be reached from any of $\{s_2, s_3\}$. Since $\tilde{e} \in P_{11}^*$, this will imply **G7**. Let e' denote the most downstream edge of P_{11}^* that can reach at least one of $\{d_2, d_3\}$ (we have at least the s_1-source edge e_{s_1}). Among all the edges in P_{11}^* that are downstream of e', let e'' denote the most upstream one that can be reached by at least one of $\{s_2, s_3\}$ (we have at least the d_1-destination edge e_{d_1}). In the next paragraph, we argue that e'' is not the immediate downstream edge of e', i.e., $\mathsf{head}(e') \prec \mathsf{tail}(e'')$. This conclusion directly implies that we have at least one edge $\tilde{e}$ that satisfies **G7** (which is in-between e' and e'').

Without loss of generality, assume that $\mathsf{head}(e') = \mathsf{tail}(e'')$ and e' can reach d_2. Then, by our previous arguments, e' is an upstream edge of e_u^{32}. Consider two cases: Case 1: Suppose e'' is reachable from s_3, then by our previous arguments, e'' is a downstream edge of e_v^{32}. However, this implies that we can go from $\mathsf{head}(e')$ through e_u^{32} to e_v^{32} and then back to $\mathsf{tail}(e'') = \mathsf{head}(e')$, which contradicts the assumption that G is acyclic. Consider the Case 2: e'' is reachable from s_2. Then by our previous arguments, e'' is a downstream edge of e_v^{23}. Then we can go from e_u^{23} to e_v^{23}, then to $\mathsf{tail}(e'') = \mathsf{head}(e')$ and then to e_u^{32}. This contradicts the assumption of $\neg$**G5**. The proof of **R17** is thus complete. ■

We prove **R18** as follows.

Proof. Suppose **G16** $\wedge$ **G17** $\wedge$ **E0** is true. From **E0** being true, G_{3ANA} satisfies (N.1) with at least two non-zero coefficients α_i and β_j. From **G16** being true, the considered subgraph G' has the non-trivial channel gain polynomials m_{ij} for all $i \neq j$ and m_{11}. By Proposition 5.4.2, G' also satisfies (N.1) with the same set of non-zero coefficients α_i and β_j.

From **G17** being true, consider the defined edge $\tilde{e} \in G'$ that cannot reach any of $\{d_2, d_3\}$ (but reaches d_1) and cannot be reached by any of $\{s_2, s_3\}$ (but reached from s_1). This chosen $\tilde{e}$ must not be the s_1-source edge e_{s_1} otherwise ($\tilde{e} = e_{s_1}$) $\tilde{e}$ will reach d_2 or d_3 and thus contradict the assumption **G17**.

Choose an edge $e \in G'$ such that $e_{s_1} \preceq e$ and $\mathsf{head}(e) = \mathsf{tail}(\tilde{e})$. This is always possible because s_1 can reach $\tilde{e}$ and $e_{s_1} \prec \tilde{e}$ on G'. Then, this chosen edge e should not be reached from s_2 or s_3 otherwise s_2 or s_3 can reach $\tilde{e}$ and this contradicts the assumption **G17**. Now consider a local kernel $x_{e\tilde{e}}$ from e to $\tilde{e}$. Then, one can quickly see that the channel gains m_{21}, m_{23}, m_{31}, and m_{32} must not have $x_{e\tilde{e}}$ as a variable since e is not reachable from s_2 nor s_3. Also m_{12} and m_{13} must not have $x_{e\tilde{e}}$ as a variable since $\tilde{e}$ doe not reach any of $\{d_2, d_3\}$.

This further implies that the RHS of (N.1) does not depend on $x_{e\tilde{e}}$. However, the LHS of (N.1) has a common factor m_{11} and thus has degree 1 in $x_{e\tilde{e}}$. This contradicts the above discussion that G' also satisfies (N.1). ■

We prove **R19** as follows.

Proof. Equivalently, we prove the following two relationships: **G16** ∧ **G18** ∧ **E0** ∧ **E1** ∧ **E2** ⇒ false; and **G16** ∧ **G19** ∧ **E0** ∧ **E1** ∧ **E2** ⇒ false.

We first prove the former. Suppose that **G16** ∧ **G18** ∧ **E0** ∧ **E1** ∧ **E2** is true. From **E0** ∧ **E1** ∧ **E2** being true, there exists some coefficient values $\{\alpha_i\}_{i=0}^n$ and $\{\beta_j\}_{j=0}^n$ such that G_{3ANA} of interest satisfies

$$m_{11}m_{23}\,\psi_\alpha^{(n)}(R,L) = m_{13}m_{21}\,\psi_\beta^{(n)}(R,L), \tag{N.26}$$

with (i) At least one of α_i is non-zero; (ii) At least one of β_j is non-zero; (iii) $\alpha_k \neq \beta_k$ for some $k \in \{0,...,n\}$; and (iv) either $\alpha_0 \neq 0$ or $\beta_n \neq 0$ or $\alpha_k \neq \beta_{k-1}$ for some $k \in \{1,...,n\}$.

From the assumption that **G16** is true, consider a subgraph G' which has the non-trivial channel gain polynomials m_{ij} for all $i \neq j$ and m_{11}. Thus by Proposition 5.4.2, G' also satisfies (N.26) with the same coefficient values.

Now from **G18** being true, we will prove the first relationship, i.e., **G16** ∧ **G18** ∧ **E0** ∧ **E1** ∧ **E2** ⇒ false. Since **G18** is true, there exists a subgraph $G'' \subset G'$ which also has the non-trivial channel gains m_{ij} for all $i \neq j$ and m_{11}. Thus again by Proposition 5.4.2, G'' satisfies (N.26) with the same coefficients. Since G'' also satisfies $m_{11}m_{23} = m_{13}m_{21}$, by (N.26), we know that G'' satisfies

$$\psi_\alpha^{(n)}(R,L) = \psi_\beta^{(n)}(R,L). \tag{N.27}$$

Note that by (iii), the coefficient values were chosen such that $\alpha_k \neq \beta_k$ for some $k \in \{0,...,n\}$. Then (N.27) further implies that G'' satisfies $\sum_{k=0}^n \gamma_k R^{n-k} L^k = 0$ with at least one non-zero γ_k. Equivalently, this means that the set of polynomials $\{R^n, R^{n-1}L, \cdots, RL^{n-1}, L^n\}$ is linearly dependent. Since this is the Vandermonde form, it is equivalent to that $L \equiv R$ holds on G''. However, this contradicts the assumption **G18** that G'' satisfies $L \not\equiv R$.

To prove the second relationship, i.e., $\mathbf{G16} \wedge \mathbf{G19} \wedge \mathbf{E0} \wedge \mathbf{E1} \wedge \mathbf{E2} \Rightarrow$ false, we assume **G19** is true. Since **G19** is true, there exists a subgraph $G'' \subset G'$ which also has the non-trivial channel gains m_{ij} for all $i \neq j$ and m_{11}. Thus again by Proposition 5.4.2, G'' satisfies (N.26) with the same coefficients. Moreover, G'' satisfies $m_{11}m_{32} = m_{12}m_{31}$, which together with (N.26) imply that G'' also satisfies

$$R\,\psi_{\alpha}^{(n)}(R, L) = L\,\psi_{\beta}^{(n)}(R, L), \tag{N.28}$$

where we first multiply m_{32} on both sides of (N.26).

Expanding (N.28), we have

$$\begin{aligned} R\,\psi_{\alpha}^{(n)}(R, L) - L\,\psi_{\beta}^{(n)}(R, L) &= \alpha_0 R^{n+1} + \sum_{k=1}^{n} (\alpha_k - \beta_{k-1}) R^{n+1-k} L^k + \beta_n L^{n+1} \\ &= \sum_{k=0}^{n+1} \gamma_k R^{n+1-k} L^k = 0 \end{aligned} \tag{N.29}$$

By (iv), the coefficient values were chosen such that either $\alpha_0 \neq 0$ or $\beta_n \neq 0$ or $\alpha_k \neq \beta_{k-1}$ for some $k \in \{1, ..., n\}$. Then (N.29) further implies that G'' satisfies $\sum_{k=0}^{n+1} \gamma_k R^{n+1-k} L^k = 0$ with some non-zero γ_k. Equivalently, this means that the set of polynomials $\{R^{n+1}, R^n L, \cdots, RL^n, L^{n+1}\}$ is linearly dependent, and thus G'' satisfies $L \equiv R$. This contradicts the assumption **G19** that $L \not\equiv R$ holds on G''. The proof of **R19** is thus complete. ∎

We prove **R20** as follows.

Proof. Suppose $\mathbf{G1} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge (\neg\,\mathbf{G22}) \wedge \mathbf{G23}$ is true. Recall the definitions of e_u^{23}, e_u^{32}, e_v^{23}, and e_v^{32} when $(\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4})$ is true. From Property 1 of both $\neg\,\mathbf{G3}$ and $\neg\,\mathbf{G4}$, we know that s_1 can reach e_u^{23} (resp. e_u^{32}) and then use e_v^{23} (resp. e_v^{32}) to arrive at d_1. Note that $\neg\,\mathbf{G22}$ being true implies that everys_1-to-$d_1$path must use a vertex w in-between $\mathsf{tail}(e_u^{23})$ and $\mathsf{head}(e_v^{23})$ or in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$

or both. Combined with Property 3 of both $\neg$ **G3** and $\neg$ **G4**, this further implies that every s_1-to-d_1 path must use $\{e_u^{23}, e_v^{23}\}$ or $\{e_u^{32}, e_v^{32}\}$ or both.

From **G23** being true, we have $e_u^{23} \prec e_u^{32}$. For the following we prove that (i) $\mathsf{head}(e_v^{23}) \prec \mathsf{tail}(e_u^{32})$; and (ii) there exists a path segment from $\mathsf{head}(e_v^{23})$ to d_1 which is vertex-disjoint with any vertex in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$. First we note that e_u^{23} is not an 1-edge cut separating s_1 and $\mathsf{tail}(e_u^{32})$. The reason is that if $e_u^{23} \in \mathsf{1cut}(s_1; \mathsf{tail}(e_u^{32}))$, then e_u^{23} must be an 1-edge cut separating s_1 and d_1 since any s_1-to-d_1 path must use $\{e_u^{23}, e_v^{23}\}$ or $\{e_u^{32}, e_v^{32}\}$ or both. However, since $e_u^{23} \in \overline{S}_2 \cap \overline{D}_3$, this implies $e_u^{23} \in \mathsf{1cut}(\{s_1, s_2\}; \{d_1, d_3\})$. This contradicts the assumption **G1**. We now consider all the possible cases: either $e_v^{23} \prec e_u^{32}$ or $e_u^{32} \preceq e_v^{23}$ or not reachable from each other. We first show that the last case is not possible. The reason is that suppose e_v^{23} and e_u^{32} are not reachable from each other, then s_1 can first reach e_u^{23}, then reach e_u^{32} to d_1 without using e_v^{23}. This contradicts Property 3 of $\neg$ **G3**. Similarly, the second case is not possible because when $e_u^{32} \preceq e_v^{23}$, we can find a path from s_1 to e_u^{32} to e_v^{23} to d_1 not using e_u^{23} since $e_u^{23} \notin \mathsf{1cut}(s_1; \mathsf{tail}(e_u^{32}))$. This also contradicts Property 3 of $\neg$ **G3**. We thus have shown $e_v^{23} \prec e_u^{32}$. Now we still need to show that e_v^{23} and e_u^{32} are not immediate neighbors: $\mathsf{head}(e_v^{23}) \prec \mathsf{tail}(e_u^{32})$. Suppose not, i.e., $\mathsf{head}(e_v^{23}) = \mathsf{tail}(e_u^{32})$. Then by Property 3 of $\neg$ **G3**, we know that any path from $\mathsf{head}(e_v^{23}) = \mathsf{tail}(e_u^{32})$ to d_1 must use both e_u^{32} and e_v^{32}. By the conclusion in the first paragraph of this proof, we know that this implies $\{e_u^{32}, e_v^{32}\} \subset \mathsf{1cut}(s_1; d_1)$. However, this further implies that $\{e_u^{32}, e_v^{32}\} \subset \mathsf{1cut}(\{s_1, s_3\}; \{d_1, d_2\})$, which contradicts **G1**. The proof of (i) is complete.

We now prove (ii). Suppose that every path from $\mathsf{head}(e_v^{23})$ to d_1 has at least one vertex w that satisfies $\mathsf{tail}(e_u^{32}) \preceq w \preceq \mathsf{head}(e_v^{32})$. Then by Property 3 of $\neg$ **G3**, every s_1-to-d_1 path that uses e_v^{23} must use both e_u^{32} and e_v^{32}. By the findings in the first paragraph of this proof, this also implies that any s_1-to-d_1 path must use both e_u^{32} and e_v^{32}. However, this further implies that $\{e_u^{32}, e_v^{32}\} \subset \mathsf{1cut}(\{s_1, s_3\}; \{d_1, d_2\})$. This contradicts **G1**. We have thus proven (ii).

Using the assumptions and the above discussions, we construct the following 11 path segments.

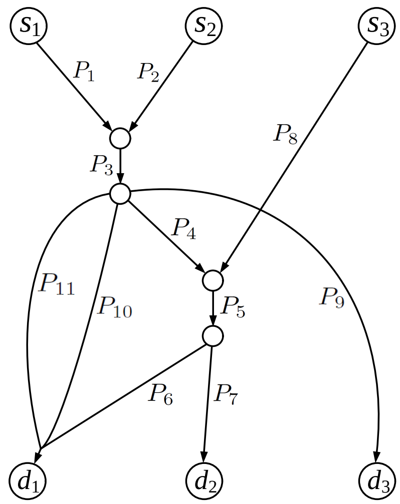


Fig. N.2. The subgraph G' of the 3-unicast ANA network $G_{3\text{ANA}}$ induced by the union of the 11 paths in the proof of **R20**.

- P_1: a path from s_1 to $\mathsf{tail}(e_u^{23})$. This is always possible due to **G3** being false.
- P_2: a path from s_2 to $\mathsf{tail}(e_u^{23})$, which is edge-disjoint with P_1. This is always possible due to **G3** being false.
- P_3: a path starting from e_u^{23} and ending at e_v^{23}. This is always possible due to **G3** being false.
- P_4: a path from $\mathsf{head}(e_v^{23})$ to $\mathsf{tail}(e_u^{23})$. This is always possible since we showed (i) in the above discussion.
- P_5: a path starting from e_u^{32} and ending at e_v^{32}. This is always possible due to **G4** being false.
- P_6: a path from $\mathsf{head}(e_v^{32})$ to d_1. This is always possible due to **G4** being false.
- P_7: a path from $\mathsf{head}(e_v^{32})$ to d_2, which is edge-disjoint with P_6. This is always possible due to **G4** being false and Property 2 of $\neg$**G4**.
- P_8: a path from s_3 to $\mathsf{tail}(e_u^{32})$. This is always possible due to **G4** being false.
- P_9: a path from $\mathsf{head}(e_v^{23})$ to d_3. This is always possible due to **G3** being false.
- P_{10}: a path from $\mathsf{head}(e_v^{23})$ to d_1, which is vertex-disjoint with P_5. This is always possible since we showed (ii) in the above discussion.

• P_{11}: a path from $\mathsf{head}(e_v^{23})$ to d_1, which is edge-disjoint with P_9. This is always possible due to **G3** being false.

Fig. N.2 illustrates the relative topology of these 11 paths. We now consider the subgraph G' induced by the above 11 path segments. First, one can see that s_i can reach d_j for all $i \neq j$. In particular, s_1 can reach d_2 through $P_1P_3P_4P_5P_7$; s_1 can reach d_3 through $P_1P_3P_9$; s_2 can reach d_1 through either $P_2P_3P_4P_5P_6$ or $P_2P_3P_{10}$ or $P_2P_3P_{11}$; s_2 can reach d_3 through $P_2P_3P_9$; s_3 can reach d_1 through $P_8P_5P_6$; and s_3 can reach d_2 through $P_8P_5P_7$. Moreover, s_1 can reach d_1 through either $P_1P_3P_4P_5P_6$ or $P_1P_3P_{10}$ or $P_1P_3P_{11}$. Thus we showed **G16**.

For the following, we will prove that $m_{11}m_{23} = m_{13}m_{21}$ and $L \not\equiv R$ hold in the above G'. Note that $\{P_1, P_2, P_3, P_{10}\}$ must be vertex-disjoint with P_8, otherwise s_3 can reach d_1 without using P_5 and this contradicts $\{e_u^{32}, e_v^{32}\} \subset \overline{S}_3 \cap \overline{D}_2 \subset \mathsf{1cut}(s_3; d_1)$. Since P_8 is vertex-disjoint from $\{P_1, P_2\}$, one can easily see that removing P_3 separates $\{s_1, s_2\}$ and $\{d_1, d_3\}$. Thus G' satisfies $m_{11}m_{23} = m_{13}m_{21}$.

To show that $L \not\equiv R$ holds on G', we make the following arguments. First, we show that G' satisfies $\overline{S}_2 \cap \overline{S}_3 = \emptyset$. Note that any $\overline{S}_2$ edge can exist only as one of four cases: (i) P_2; (ii) P_3; (iii) an edge that P_4, P_9, P_{10}, and P_{11} share; and (iv) an edge that P_6, P_9, P_{10}, and P_{11} share. Note also that any $\overline{S}_3$ edge can exist only as one of three cases: (i) P_8; (ii) P_5; and (iii) an edge that P_6 and P_7 shares. But since P_6 and P_7 were chosen to be edge-disjoint with each other from the above construction, any $\overline{S}_3$ edge can exist on either P_8 or P_5. However, P_5 was chosen to be vertex-disjoint with P_{10} from the above construction and we also showed that P_8 is vertex-disjoint with $\{P_2, P_3, P_{10}\}$. Thus, $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G'.

Second, we show that G' satisfies $\overline{D}_1 \cap \overline{D}_2 = \emptyset$. Note that any $\overline{D}_1$ edge can exist on an edge that all P_6, P_{10}, and P_{11} share since P_6 cannot share an edge with any of its upstream paths (in particular P_2, P_3, P_4, and P_5); P_5 cannot share an edge with P_{10} due to vertex-disjointness; and P_8 cannot share edge with $\{P_2, P_3, P_{10}\}$ otherwise there will be ans_3-to-$d_1$path not using P_5. Note also that any $\overline{D}_2$ edge can exist on (i) an edge that both P_4 and P_8 share; (ii) P_5; and (iii) P_7. However, P_7 was chosen to be edge-disjoint with P_6, and P_5 was chosen to be vertex-disjoint with P_{10}. Moreover, we already showed that P_8 is vertex-disjoint with P_{10}. Thus, $\overline{D}_1 \cap \overline{D}_2 = \emptyset$ on G'.

Third, we show that G' satisfies $\overline{D}_1 \cap \overline{D}_3 = \emptyset$. Note that any $\overline{D}_1$ edge can exist on an edge that P_6, P_{10} and P_{11} share. Note also that any $\overline{D}_3$ edge can exist on (i) P_3; and (ii) P_9. However, all P_6, P_{10} and P_{11} are the downstream paths of P_3. Moreover, P_9 was chosen to be edge-disjoint with P_{11} by our construction. Thus, $\overline{D}_1 \cap \overline{D}_3 = \emptyset$ on G'.

Hence, the above discussions, together with Proposition 6.2.1, implies that the considered G' satisfies $L \not\equiv R$. Thus we have proven **G18** being true. The proof is thus complete. ∎

We prove **R21** as follows.

Proof. Suppose $\mathbf{LNR} \wedge (\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4}) \wedge \mathbf{G22} \wedge \mathbf{G23} \wedge \mathbf{G25}$ is true. Recall the definitions of e_u^{23}, e_u^{32}, e_v^{23}, and e_v^{32} when $(\neg\,\mathbf{G3}) \wedge (\neg\,\mathbf{G4})$ is true. From Property 1 of both $\neg\,\mathbf{G3}$ and $\neg\,\mathbf{G4}$, s_1 reaches e_u^{23} and e_u^{32}, respectively. From **G22** being true, there exists as_1-to-$d_1$path P_{11}^* who does not use any vertex in-between $\mathsf{tail}(e_u^{23})$ and $\mathsf{head}(e_v^{23})$, and any vertex in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$.

Note that $\mathbf{G23} \wedge \mathbf{G25}$ implies $e_u^{23} \prec e_u^{32} \prec e_v^{23}$. For the following, we prove that $e_v^{32} \prec e_v^{23}$. Note that by our construction $e_u^{32} \preceq e_v^{32}$. As a result, we have $e_u^{23} \prec e_u^{32} \preceq e_v^{32} \prec e_v^{23}$. To that end, we consider all the possible cases between e_v^{32} and e_v^{23}: $e_v^{32} \prec e_v^{23}$; or $e_v^{23} \prec e_v^{32}$; or $e_v^{32} = e_v^{23}$; or they are not reachable from each other. We first show that the third case is not possible. The reason is that if $e_v^{32} = e_v^{23}$, then we have $\overline{S}_2 \cap \overline{S}_3 \cap \overline{D}_2 \cap \overline{D}_3 \neq \emptyset$, which contradicts the assumption **LNR**. The last case in which e_v^{32} and e_v^{23} are not reachable from each other is also not possible. The reason is that by our construction, there is always ans_1-to-$d_1$path through e_u^{23}, e_u^{32}, and e_v^{32} without using e_v^{23}. Note that by Property 3 of $\neg\,\mathbf{G3}$, suchs_1-to-$d_1$path must use e_v^{23}, which is a contradiction. We also claim that the second case, $e_v^{23} \prec e_v^{32}$, is not possible. The reason is that if $e_v^{23} \prec e_v^{32}$, then together with the assumption $\mathbf{G23} \wedge \mathbf{G25}$ we have $e_u^{23} \prec e_u^{32} \prec e_v^{23} \prec e_v^{32}$. We also note that e_u^{32} must be an 1-edge cut separating s_1

and $\mathsf{tail}(e_v^{23})$, otherwise s_1 can reach $\mathsf{tail}(e_v^{23})$ without using e_u^{32} and then use e_v^{23} and e_v^{32} to arrive at d_2. This contradicts the construction $e_u^{32} \in \overline{S}_3 \cap \overline{D}_2 \subset \mathsf{1cut}(s_1; d_2)$. Since $e_v^{23} \in \overline{S}_2 \cap \overline{D}_3$ is also an 1-edge cut separating s_1 and d_3, this in turn implies that $e_u^{32} \in \mathsf{1cut}(s_1; d_3)$. Symmetrically following this argument, we can also prove that $e_v^{23} \in \mathsf{1cut}(s_3; d_1)$. Since $e_u^{32} \in \overline{S}_3 \cap \overline{D}_2$ and $e_v^{23} \in \overline{S}_2 \cap \overline{D}_3$, these further imply that $e_u^{32} \in \overline{S}_1 \cap \overline{S}_3 \cap \overline{D}_2$ and $e_v^{23} \in \overline{S}_2 \cap \overline{D}_1 \cap \overline{D}_3$, which contradicts the assumption **LNR** by Proposition 6.2.1. We have thus established $e_v^{32} \prec e_v^{23}$ and together with the assumption $\mathbf{G23} \wedge \mathbf{G25}$, we have $e_u^{23} \prec e_u^{32} \preceq e_v^{32} \prec e_v^{23}$.

Using the assumptions and the above discussions, we construct the following 7 path segments.

- P_1: a path from s_1 to $\mathsf{tail}(e_u^{23})$. This is always possible due to **G3** being false.
- P_2: a path from s_2 to $\mathsf{tail}(e_u^{23})$ which is edge-disjoint with P_1. This is always possible due to **G3** being false and Property 2 of $\neg$ **G3**.
- P_3: a path starting from e_u^{23}, using e_u^{32} and e_v^{32}, and ending at e_v^{23}. This is always possible from the above discussion.
- P_4: a path from $\mathsf{head}(e_v^{23})$ to d_1. This is always possible due to **G3** being false.
- P_5: a path from $\mathsf{head}(e_v^{23})$ to d_3 which is edge-disjoint with P_4. This is always possible due to **G3** being false and Property 2 of $\neg$ **G3**.
- P_6: a path from s_3 to $\mathsf{tail}(e_u^{32})$. This is always possible due to **G4** being false.
- P_7: a path from $\mathsf{head}(e_v^{32})$ to d_2. This is always possible due to **G4** being false.

We now consider the subgraph G' induced by the above 7 path segments and P_{11}^*. First, one can easily check that s_i can reach d_j for all $i \neq j$. In particular, s_1 can reach d_2 through $P_1 P_3 e_v^{32} P_7$; s_1 can reach d_3 through $P_1 P_3 P_5$; s_2 can reach d_1 through $P_2 P_3 P_4$; s_2 can reach d_3 through $P_2 P_3 P_5$; s_3 can reach d_1 through $P_6 e_u^{32} P_3 P_4$; and s_3 can reach d_2 through $P_6 e_u^{32} P_3 e_v^{32} P_7$. Moreover, s_1 can reach d_1 through either P_{11}^* or $P_1 P_3 P_4$. As a result, **G16** must hold.

We now prove **G17**. To that end, we will show that there exists an edge $\tilde{e} \in P_{11}^*$ that cannot reach any of $\{d_2, d_3\}$, and cannot be reached from any of $\{s_2, s_3\}$. Note from **G22** being true that P_{11}^* was chosen to be vertex-disjoint with P_3. Note that

P_{11}^* must also be vertex-disjoint with P_2 (resp. P_6) otherwise s_2 (resp. s_3) can reach d_1 without using P_3 (resp. $e_u^{32}P_3e_v^{32}$). Similarly, P_{11}^* must also be vertex-disjoint with P_5 (resp. P_7) otherwise s_1 can reach d_3 (resp. d_2) without using P_3 (resp. $e_u^{32}P_3e_v^{32}$). Hence, among 7 path segments constructed above, the only path segments that can share a vertex with P_{11}^* are P_1 and P_4. Without loss of generality, we also assume that P_1 is chosen such that it overlaps with P_{11}^* in the beginning but then "branches out". That is, let u^* denote the most downstream vertex among those who are used by both P_1 and P_{11}^* and we can then replace P_1 by $s_1P_{11}^*u^*P_1\mathsf{tail}(e_u^{23})$. Note that the new construction still satisfies the requirement that P_1 and P_2 are edge-disjoint since P_{11}^* is vertex-disjoint with P_2. Similarly, we also assume that P_4 is chosen such that it does not overlap with P_{11}^* in the beginning but then "merges" with P_{11}^* whenever P_4 shares a vertex v^* with P_{11}^* for the first time. The new construction of P_4, i.e., $\mathsf{head}(e_v^{23})P_4v^*P_{11}^*d_1$ is still edge-disjoint from P_5. Then in the considered subgraph G', in order for an edge $e\in P_{11}^*$ to reach d_2 or d_3, we must have $\mathsf{head}(e)\preceq u^*$. Similarly, in order for an edge $e\in P_{11}^*$ to be reached from s_2 or s_3, this edge e must satisfy $v^*\preceq\mathsf{tail}(e)$. If there does not exist such an edge $\tilde{e}\in P_{11}^*$ satisfying **G17**, then it means that $u^*=v^*$. This, however, contradicts the assumption that G is acyclic because now we can walk from u^* through $P_1P_3P_4$ back to $v^*=u^*$. Therefore, we thus have **G17**. The proof of **R21** is thus complete. ■

We prove **R22** as follows.

Proof. Suppose $\mathbf{LNR}\wedge(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})\wedge\mathbf{G22}\wedge\mathbf{G23}\wedge(\neg\,\mathbf{G25})$ is true. Recall the definitions of e_u^{23}, e_u^{32}, e_v^{23}, and e_v^{32} when $(\neg\,\mathbf{G3})\wedge(\neg\,\mathbf{G4})$ is true. From Property 1 of both $\neg\,\mathbf{G3}$ and $\neg\,\mathbf{G4}$, s_1 reaches e_u^{23} and e_u^{32}, respectively. From **G22** being true, there exists as_1-to-$d_1$path P_{11}^* who does not use any vertex in-between $\mathsf{tail}(e_u^{23})$ and $\mathsf{head}(e_v^{23})$, and any vertex in-between $\mathsf{tail}(e_u^{32})$ and $\mathsf{head}(e_v^{32})$.

Note that **G23** implies $e_u^{23}\prec e_u^{32}$. For the following, we prove that $\mathsf{head}(e_v^{23})\prec\mathsf{tail}(e_u^{32})$. To that end, we consider all the possible cases by $\neg\,\mathbf{G25}$ being true: either $e_v^{23}\prec e_u^{32}$ or $e_v^{23}=e_u^{32}$ or not reachable from each other. We first show that the second case is not possible. The reason is that if $e_v^{23}=e_u^{32}$, then we have $\overline{S}_2\cap\overline{S}_3\cap\overline{D}_2\cap\overline{D}_3\neq\emptyset$, which contradicts the assumption **LNR**. The third case in which e_v^{23} and e_u^{32} are not reachable from each other is also not possible. The reason is that by our construction, there is always an s_1-to-d_1 path through e_u^{23}, e_u^{32}, and e_v^{32} without using e_v^{23}. Note that by Property 3 of $\neg$ **G3**, such s_1-to-d_1 path must use e_v^{23}, which is a contradiction. We have thus established $e_v^{23}\prec e_u^{32}$. We still need to show that e_v^{23} and e_u^{32} are not immediate neighbors since we are proving $\mathsf{head}(e_v^{23})\prec\mathsf{tail}(e_u^{32})$. We prove this by contradiction. Suppose not, i.e., $w=\mathsf{head}(e_v^{23})=\mathsf{tail}(e_u^{32})$. Since $e_u^{32}\in\overline{S}_3\cap\overline{D}_2\subset\mathsf{1cut}(s_1;d_2)$, any s_1-to-d_2 path must use its tail w. By Property 3 of $\neg$ **G3** we have $e_v^{23}\in\mathsf{1cut}(s_1;w)$. This in turn implies that e_v^{23} is also an 1-edge cut separating s_1 and d_2. By symmetry, we can also prove $e_u^{32}\in\mathsf{1cut}(s_2;d_1)$. Jointly the above argument implies that $e_v^{23}\in\overline{S}_1\cap\overline{S}_2\cap\overline{D}_3$ and $e_u^{32}\in\overline{S}_3\cap\overline{D}_1\cap\overline{D}_2$, which contradicts the assumption **LNR** by Proposition 6.2.1.

Based on the above discussions, we construct the following 9 path segments.

- P_1: a path from s_1 to $\mathsf{tail}(e_u^{23})$. This is always possible due to **G3** being false.
- P_2: a path from s_2 to $\mathsf{tail}(e_u^{23})$ which is edge-disjoint with P_1. This is always possible due to **G3** being false and Property 2 of $\neg$ **G3**.
- P_3: a path starting from e_u^{23} and ending at e_v^{23}. This is always possible due to **G3** being false.
- P_4: a path from $\mathsf{head}(e_v^{23})$ to $\mathsf{tail}(e_u^{32})$. This is always possible from the above discussion.
- P_5: a path starting from e_u^{32} and ending at e_v^{32}. This is always possible due to **G4** being false.
- P_6: a path from $\mathsf{head}(e_v^{32})$ to d_1. This is always possible due to **G4** being false.
- P_7: a path from $\mathsf{head}(e_v^{32})$ to d_2 which is edge-disjoint with P_6. This is always possible due to **G4** being false and Property 2 of $\neg$ **G4**.
- P_8: a path from s_3 to $\mathsf{tail}(e_u^{32})$. This is always possible due to **G4** being false.
- P_9: a path from $\mathsf{head}(e_v^{23})$ to d_3. This is always possible due to **G3** being false.

From **G22** being true, P_{11}^* was chosen to be vertex-disjoint with $\{P_3, P_5\}$. Note that P_{11}^* must also be vertex-disjoint with P_2 (resp. P_8) otherwise s_2 (resp. s_3) can reach d_1 without using P_3 (resp. P_5). Similarly, P_{11}^* must also be vertex-disjoint with P_7 (resp. P_9) otherwise s_1 can reach d_2 (resp. d_3) without using P_5 (resp. P_3). Hence, among 9 path segments constructed above, the only path segments that can share a vertex with P_{11}^* are P_1, P_4, and P_6.

We now consider the subgraph G' induced by the above 9 path segments and P_{11}^*. First, one can easily check that s_i can reach d_j for all $i \neq j$. In particular, s_1 can reach d_2 through $P_1P_3P_4P_5P_7$; s_1 can reach d_3 through $P_1P_3P_9$; s_2 can reach d_1 through $P_2P_3P_4P_5P_6$; s_2 can reach d_3 through $P_2P_3P_9$; s_3 can reach d_1 through $P_8P_5P_6$; and s_3 can reach d_2 through $P_8P_5P_7$. Moreover, s_1 can reach d_1 through either P_{11}^* or $P_1P_3P_4P_5P_6$. Thus we showed **G16**.

Case 1: P_{11}^* is also vertex-disjoint with P_4. In this case, we will prove that **G17** is satisfied. Namely, we claim that there exists an edge $\tilde{e} \in P_{11}^*$ that cannot reach any of $\{d_2, d_3\}$, and cannot be reached from any of $\{s_2, s_3\}$. Note that only path segments that P_{11}^* can share a vertex with are P_1 and P_6. Without loss of generality, we assume that P_1 is chosen such that it overlaps with P_{11}^* in the beginning but then "branches out". That is, let u^* denote the most downstream vertex among those who are used by both P_1 and P_{11}^* and we can then replace P_1 by $s_1P_{11}^*u^*P_1\mathsf{tail}(e_u^{23})$. Note that the new construction still satisfies the requirement that P_1 and P_2 are edge-disjoint since P_{11}^* is vertex-disjoint with P_2. Similarly, we also assume that P_6 is chosen such that it does not overlap with P_{11}^* in the beginning but then "merges" with P_{11}^* whenever P_6 shares a vertex v^* with P_{11}^* for the first time. The new construction of P_6, i.e, $\mathsf{head}(e_v^{32})P_6v^*P_{11}^*d_1$, is still edge-disjoint from P_7. Then in the considered subgraph G', in order for an edge $e \in P_{11}^*$ to reach d_2 or d_3, we must have $\mathsf{head}(e) \preceq u^*$. Similarly, in order for an edge $e \in P_{11}^*$ to be reached from s_2 or s_3, this edge e must satisfy $v^* \preceq \mathsf{tail}(e)$. If there does not exist such an edge $\tilde{e} \in P_{11}^*$ satisfying **G17**, then it means that $u^* = v^*$. This, however, contradicts the assumption that G is acyclic because now we can walk from u^* through $P_1P_3P_4P_5P_6$ back to $v^* = u^*$. Therefore, we thus have **G17** for **Case 1**.

Case 2: P_{11}^* shares a vertex with P_4. In this case, we will prove that **G18** is true. Since P_{11}^* is vertex-disjoint with $\{P_3, P_5\}$, P_{11}^* must share a vertex w with P_4 where $\mathsf{head}(e_v^{23}) \prec w \prec \mathsf{tail}(e_u^{32})$. Choose the most downstream vertex among those who are used by both P_{11}^* and P_4 and denote it as w'. Then, denote the path segment $\mathsf{head}(e_v^{23})P_4w'P_{11}^*d_1$ by P_{10}. Note that we do not introduce new paths but only introduce a new notation as shorthand for a combination of some existing path segments. We observe that there may be some edge overlap between P_4 and P_9 since both starts from $\mathsf{head}(e_v^{23})$. Let $\tilde{w}$ denote the most downstream vertex that is used by both P_4 and P_9. We then replace P_9 by $\tilde{w}P_9d_3$, i.e., we truncate P_9 so that P_9 is now edge-disjoint from P_4.

Since the path segment $w'P_{10}d_1$ originally comes from P_{11}^*, $w'P_{10}d_1$ is also vertex-disjoint with $\{P_2, P_3, P_5, P_7, P_8, P_9\}$. In addition, P_8 must be vertex-disjoint with $\{P_1, P_2, P_3, P_{10}\}$, otherwise s_3 can reach d_1 without using P_5.

Now we consider the another subgraph $G'' \subset G'$ induced by the path segments P_1 to P_8, the redefined P_9, and newly constructed P_{10}, i.e., when compared to G', we replace P_{11}^* by P_{10}. One can easily verify that s_i can reach d_j for all $i \neq j$, and s_1 can reach d_1 on this new subgraph G''. Using the above topological relationships between these constructed path segments, we will further show that the induced G'' satisfies $m_{11}m_{23} = m_{13}m_{21}$ and $L \not\equiv R$.

Since P_8 is vertex-disjoint from $\{P_1, P_2\}$, one can see that removing P_3 separates $\{s_1, s_2\}$ and $\{d_1, d_3\}$. Thus, the considered G'' also satisfies $m_{11}m_{23} = m_{13}m_{21}$.

To prove $L \not\equiv R$, we first show that G'' satisfies $\overline{S}_2 \cap \overline{S}_3 = \emptyset$. Note that any $\overline{S}_2$ edge can exist only as one of three cases: (i) P_2; (ii) P_3; (iii) an edge that P_4 and P_{10} share, whose head is in the upstream of or equal to $\tilde{w}$, i.e., $\{e \in P_4 \cap P_{10} : \mathsf{head}(e) \preceq \tilde{w}\}$ (may or may not be empty); and (iv) an edge that P_6, P_9, and P_{10} share. Note also that any $\overline{S}_3$ edge can exist only as on of three cases: (i) P_8; (ii) P_5; and (iii) an edge that P_6 and P_7 share. But since P_6 and P_7 were chosen to be edge-disjoint from the above construction, any $\overline{S}_3$ edge can exist on either P_8 or P_5. We then notice that P_8 is vertex-disjoint with $\{P_2, P_3, P_{10}\}$. Also, P_5 was chosen to be vertex-disjoint with P_{10}

and both P_2 and P_3 are in the upstream of P_5. The above arguments show that no edge can be simultaneously in $\overline{S}_2$ and $\overline{S}_3$. We thus have $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G''.

Second, we show that G'' satisfies $\overline{D}_1 \cap \overline{D}_2 = \emptyset$. Note that any $\overline{D}_1$ edge can exist only an edge that both P_6 and P_{10} share since any of $\{P_5, P_8\}$ does not share an edge with any of $\{P_2, P_3, P_{10}\}$. Note also that any $\overline{D}_2$ edge can exist only as one of three cases: (i) an edge that both P_4 and P_8 share; (ii) P_5; and (iii) P_7. However, P_7 was chosen to be edge-disjoint with P_6, and we have shown that P_5 is vertex-disjoint with P_{10}. Moreover, we already showed that P_8 is vertex-disjoint with P_{10}. Thus, $\overline{D}_1 \cap \overline{D}_2 = \emptyset$ on G''.

Third, we show that G'' satisfies $\overline{D}_1 \cap \overline{D}_3 = \emptyset$. Note that any $\overline{D}_1$ edge can exist only on an edge that both P_{10} and P_6 share. Note also that any $\overline{D}_3$ edge can exist only as one of three cases: (i) a P_3 edge; (ii) a P_4 edge whose head is in the upstream of or equal to $\tilde{w}$, i.e., $\{e \in P_4 : \mathsf{head}(e) \preceq \tilde{w}\}$ (may or may not be empty); and (iii) P_9. However, P_6 is in the downstream of P_3 and P_4. Moreover, P_9 is edge-disjoint with P_{11}^* and thus edge-disjoint with $w' P_{10} d_1$. As a result, no edge can be simultaneously in $\overline{D}_1$ and $\overline{D}_3$. Thus $\overline{D}_1 \cap \overline{D}_3 = \emptyset$ on G''.

Hence, the above discussions, together with Proposition 6.2.1, implies that the considered G'' satisfies $L \not\equiv R$. We thus have proven **G18** being true for **Case 2**. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the proofs of **R20** to **R22** can also be used to prove **R23** to **R25**, respectively. More specifically, **G3** and **G4** are converted back and forth from each other when swapping the flow indices. The same thing happens between **G23** and **G24**; between **G25** and **G26**; and between **G18** and **G19**. Moreover, **LNR**, **G1**, **G16**, **G17**, and **G22** remain the same after the index swapping. Thus the above proofs of **R20** to **R22** can thus be used to prove **R23** to **R25**.

N.9 Proof of S14

N.9.1 The fifth set of logic statements

To prove **S14**, we need the fifth set of logic statements.

- **G27:** $\overline{S}_2 \cap \overline{D}_1 = \emptyset$.
- **G28:** $\overline{S}_3 \cap \overline{D}_1 = \emptyset$.
- **G29:** $\overline{D}_2 \cap \overline{S}_1 = \emptyset$.
- **G30:** $\overline{D}_3 \cap \overline{S}_1 = \emptyset$.
- **G31:** $\overline{S}_i \neq \emptyset$ and $\overline{D}_i \neq \emptyset$ for all $i \in \{1, 2, 3\}$.

Several implications can be made when **G27** is true. We term those implications *the properties of* **G27**. Several properties of **G27** are listed as follows, for which their proofs are provided in Appendix N.9.3.

Consider the case in which **G27** *is true.* Use e_2^* to denote the most downstream edge in $\mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_2; d_3)$. Since the source edge e_{s_2} belongs to both $\mathsf{1cut}(s_2; d_1)$ and $\mathsf{1cut}(s_2; d_3)$, such e_2^* always exists. Similarly, use e_1^* to denote the most upstream edge in $\mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_3; d_1)$. The properties of **G27** can now be described as follows.

◇ **Property 1 of G27:** $e_2^* \prec e_1^*$ and the channel gains m_{21}, m_{23}, and m_{31} can be expressed as $m_{21} = m_{e_{s_2}; e_2^*} m_{e_2^*; e_1^*}\, m_{e_1^*; e_{d_1}}$, $m_{23} = m_{e_{s_2}; e_2^*} m_{e_2^*; e_{d_3}}$, and $m_{31} = m_{e_{s_3}; e_1^*} m_{e_1^*; e_{d_1}}$.

◇ **Property 2 of G27:** $\mathsf{GCD}(\, m_{e_{s_3}; e_1^*},\ m_{e_{s_2}; e_2^*} m_{e_2^*; e_1^*}) \equiv 1$, $\mathsf{GCD}(\, m_{e_2^*; e_1^*} m_{e_1^*; e_{d_1}},\ m_{e_2^*; e_{d_3}}) \equiv 1$, $\mathsf{GCD}(\, m_{31},\ m_{e_2^*; e_1^*}) \equiv 1$, and $\mathsf{GCD}(\, m_{23},\ m_{e_2^*; e_1^*}) \equiv 1$.

On the other hand, when **G27** is false, we can also derive several implications, which are termed *the properties of* $\neg$ **G27**.

Consider the case in which **G27** *is false.* Use e_u^{21} (resp. e_v^{21}) to denote the most upstream (resp. the most downstream) edge in $\overline{S}_2 \cap \overline{D}_1$. By definition, it must be $e_u^{21} \preceq e_v^{21}$. We now describe the following properties of $\neg$ **G27**.

◇ **Property 1 of ¬ G27:** The channel gains m_{21}, m_{23}, and m_{31} can be expressed as $m_{21} = m_{e_{s_2}; e_u^{21}} m_{e_u^{21}; e_v^{21}} m_{e_v^{21}; e_{d_1}}$, $m_{23} = m_{e_{s_2}; e_u^{21}} m_{e_u^{21}; e_v^{21}} m_{e_v^{21}; e_{d_3}}$, and $m_{31} = m_{e_{s_3}; e_u^{21}} m_{e_u^{21}; e_v^{21}}$ $m_{e_v^{21}; e_{d_1}}$.

$\diamond$ **Property 2 of $\neg$ G27:** $\mathsf{GCD}(\, m_{e_{s_2};e_u^{21}},\; m_{e_{s_3};e_u^{21}})\equiv 1$ and $\mathsf{GCD}(\, m_{e_v^{21};e_{d_1}},\; m_{e_v^{21};e_{d_3}})\equiv 1$.

Symmetrically, we define the following properties of **G28** and $\neg$ **G28**.

Consider the case in which **G28** *is true.* Use e_3^* to denote the most downstream edge in $\mathsf{1cut}(s_3; d_1) \cap \mathsf{1cut}(s_3; d_2)$, and use e_1^* to denote the most upstream edge in $\mathsf{1cut}(s_2; d_1) \cap \mathsf{1cut}(s_3; d_1)$. We now describe the following properties of **G28**.

$\diamond$ **Property 1 of G28:** $e_3^* \prec e_1^*$ and the channel gains m_{31}, m_{32}, and m_{21} can be expressed as $m_{31} = m_{e_{s_3};e_3^*} m_{e_3^*;e_1^*}\, m_{e_1^*;e_{d_1}}$, $m_{32} = m_{e_{s_3};e_3^*} m_{e_3^*;e_{d_2}}$, and $m_{21} = m_{e_{s_2};e_1^*} m_{e_1^*;e_{d_1}}$.

$\diamond$ **Property 2 of G28:** $\mathsf{GCD}(\, m_{e_{s_2};e_1^*},\; m_{e_{s_3};e_3^*} m_{e_3^*;e_1^*})\equiv 1$, $\mathsf{GCD}(\, m_{e_3^*;e_1^*} m_{e_1^*;e_{d_1}},\; m_{e_3^*;e_{d_2}})\equiv 1$, $\mathsf{GCD}(\, m_{21},\; m_{e_3^*;e_1^*})\equiv 1$, and $\mathsf{GCD}(\, m_{32},\; m_{e_3^*;e_1^*})\equiv 1$.

Consider the case in which **G28** *is false.* Use e_u^{31} (resp. e_v^{31}) to denote the most upstream (resp. the most downstream) edge in $\overline{S}_3 \cap \overline{D}_1$. By definition, it must be $e_u^{31} \preceq e_v^{31}$. We now describe the following properties of $\neg$ **G28**.

$\diamond$ **Property 1 of $\neg$ G28:** The channel gains m_{31}, m_{32}, and m_{21} can be expressed as $m_{31} = m_{e_{s_3};e_u^{31}} m_{e_u^{31};e_v^{31}} m_{e_v^{31};e_{d_1}}$, $m_{32} = m_{e_{s_3};e_u^{31}} m_{e_u^{31};e_v^{31}} m_{e_v^{23};e_{d_2}}$, and $m_{21} = m_{e_{s_2};e_u^{31}} m_{e_u^{31};e_v^{31}}\, m_{e_v^{31};e_{d_1}}$.

$\diamond$ **Property 2 of $\neg$ G28:** $\mathsf{GCD}(\, m_{e_{s_2};e_u^{31}},\; m_{e_{s_3};e_u^{31}})\equiv 1$ and $\mathsf{GCD}(\, m_{e_v^{31};e_{d_1}},\; m_{e_v^{31};e_{d_2}})\equiv 1$.

N.9.2 The skeleton of proving S14

We prove the following relationships, which jointly prove **S14**.

- **R26:** **D3** $\wedge$ **D4** $\Rightarrow$ **G31**.
- **R27:** **LNR** $\wedge$ ($\neg$ **G27**) $\wedge$ ($\neg$ **G28**) $\wedge$ ($\neg$ **G29**) $\wedge$ ($\neg$ **G30**) $\Rightarrow$ false.
- **R28:** **D3** $\wedge$ **D4** $\wedge$ **G27** $\wedge$ **G28** $\Rightarrow$ false.
- **R29:** **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D3** $\wedge$ **D4** $\wedge$ ($\neg$ **G27**) $\wedge$ **G28** $\Rightarrow$ false.
- **R30:** **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D3** $\wedge$ **D4** $\wedge$ **G27** $\wedge$ ($\neg$ **G28**) $\Rightarrow$ false.
- **R31:** **D3** $\wedge$ **D4** $\wedge$ **G29** $\wedge$ **G30** $\Rightarrow$ false.
- **R32:** **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D3** $\wedge$ **D4** $\wedge$ ($\neg$ **G29**) $\wedge$ **G30** $\Rightarrow$ false.
- **R33:** **LNR** $\wedge$ **G1** $\wedge$ **E0** $\wedge$ **D3** $\wedge$ **D4** $\wedge$ **G29** $\wedge$ ($\neg$ **G30**) $\Rightarrow$ false.

One can see that **R28** and **R31** imply, respectively,

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \Rightarrow \text{ false}, \tag{N.30}$$

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G29} \wedge \mathbf{G30} \Rightarrow \text{ false}. \tag{N.31}$$

Also **R27** implies

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge (\neg\,\mathbf{G28}) \wedge (\neg\,\mathbf{G29}) \wedge (\neg\,\mathbf{G30}) \Rightarrow \text{ false}. \tag{N.32}$$

R29, **R30**, **R32**, **R33**, (N.30), (N.31), and (N.32) jointly imply

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \Rightarrow \text{ false},$$

which proves **S14**. The proofs of **R26** and **R27** are relegated to Appendix N.9.4. The proofs of **R28**, **R29**, and **R30** are provided in Appendices N.9.5, N.9.6, and N.9.7, respectively.

The logic relationships **R31** to **R33** are the symmetric versions of **R28** to **R30**. Specifically, if we swap the roles of sources and destinations, then the resulting graph is still a 3-unicast ANA network; **D3** is now converted to **D4**; **D4** is converted to **D3**; **G27** is converted to **G29**; and **G28** is converted to **G30**. Therefore, the proof of **R28** can serve as a proof of **R31**. Further, after swapping the roles of sources and destinations, the **LNR** condition (see (6.3)) remains the same; **G1** remains the same (see (6.4)); and **E0** remains the same. Therefore, the proof of **R29** (resp. **R30**) can serve as a proof of **R32** (resp. **R33**).

N.9.3 Proofs of the properties of G27, G28, ¬ G27, and ¬ G28

We prove Properties 1 and 2 of **G27** as follows.

Proof. By swapping the roles of s_1 and s_3, and the roles of d_1 and d_3, the proof of the properties of **G3** in Appendix M.4 can be used to prove the properties of **G27**. ■

We prove Properties 1 and 2 of $\neg$**G27** as follows.

Proof. By swapping the roles of s_1 and s_3, and the roles of d_1 and d_3, the proof of Properties 1 and 2 of $\neg$**G3** in Appendix M.4 can be used to prove the properties of $\neg$**G27**. ■

By swapping the roles of s_2 and s_3, and the roles of d_2 and d_3, the above proofs can also be used to prove Properties 1 and 2 of **G28** and Properties 1 and 2 of $\neg$**G28**.

N.9.4 Proofs of R26 and R27

We prove **R26** as follows.

Proof. Suppose **D3** $\wedge$ **D4** is true. By Corollary 5.4.2, we know that any channel gain cannot have any other channel gain as a factor. Since **D3** $\wedge$ **D4** is true, any one of the four channel gains m_{12}, m_{31}, m_{13}, and m_{21} must be reducible.

Since **D4** is true, we must also have for some positive integer l_4 such that

$$\mathsf{GCD}(\, m_{11}m_{12}^{l_4}m_{23}^{l_4}m_{31}^{l_4},\ m_{21}) = m_{21}. \tag{N.33}$$

We first note that m_{23} is the only channel gain starting from s_2 out of the four channel gains $\{m_{11}, m_{12}, m_{23}, m_{31}\}$. Therefore, we must have $\mathsf{GCD}(\, m_{23}, m_{21}) \not\equiv 1$ since "we need to cover the factor of m_{21} that emits from s_2." Lemma 6.1.7 then implies that $\overline{S}_2 \neq \emptyset$.

Further, **D4** implies $\mathsf{GCD}(\, m_{11}m_{12}^{l_4}m_{23}^{l_4}m_{31}^{l_4},\ m_{13}) = m_{13}$ for some positive integer l_4, which, by similar arguments, implies $\mathsf{GCD}(\, m_{23}, m_{13}) \not\equiv 1$. Lemma 6.1.7 then implies that $\overline{D}_3 \neq \emptyset$. By similar arguments but focusing on **D3** instead, we can also prove that $\overline{S}_3 \neq \emptyset$ and $\overline{D}_2 \neq \emptyset$.

We also notice that out of the four channel gains $\{m_{11}, m_{12}, m_{23}, m_{31}\}$, both m_{11} and m_{12} are the only channel gains starting from s_1. By **D4**, we thus have for some positive integer l_4 such that

$$\mathsf{GCD}(\, m_{11}m_{12}^{l_4},\ m_{13}) \not\equiv 1. \tag{N.34}$$

Similarly, by **D3** and **D4**, we have for some positive integers l_2 and l_4 such that

$$\mathsf{GCD}(\, m_{11}m_{31}^{l_4},\ m_{21}) \not\equiv 1, \tag{N.35}$$

$$\mathsf{GCD}(\, m_{11}m_{13}^{l_2},\ m_{12}) \not\equiv 1, \tag{N.36}$$

$$\mathsf{GCD}(\, m_{11}m_{21}^{l_2},\ m_{31}) \not\equiv 1. \tag{N.37}$$

For the following, we will prove $\overline{S}_1 \neq \emptyset$. Consider the following subcases: Subcase 1: If $\mathsf{GCD}(\, m_{12},\, m_{13}) \not\equiv 1$, then by Lemma 6.1.7, $\overline{S}_1 \neq \emptyset$. Subcase 2: If $\mathsf{GCD}(\, m_{12},\, m_{13}) \equiv 1$, then (N.34) and (N.36) jointly imply both $\mathsf{GCD}(\, m_{11},\, m_{13}) \not\equiv 1$ and $\mathsf{GCD}(\, m_{11},\, m_{12}) \not\equiv 1$. Then by first applying Lemma 6.1.7 and then applying Lemma 6.1.6, we have $\overline{S}_1 \neq \emptyset$. The proof of $\overline{D}_1 \neq \emptyset$ can be derived similarly by focusing on (N.35) and (N.37). The proof of **R26** is complete. ■

We prove **R27** as follows.

Proof. We prove an equivalent relationship: $(\neg\,$**G27**$) \wedge (\neg\,$**G28**$) \wedge (\neg\,$**G29**$) \wedge (\neg\,$**G30**$)$ $\Rightarrow \neg\,$**LNR**. Suppose $(\neg\,$**G27**$) \wedge (\neg\,$**G28**$) \wedge (\neg\,$**G29**$) \wedge (\neg\,$**G30**$)$ is true. By Lemma 6.1.4, we know that $(\neg\,$**G27**$) \wedge (\neg\,$**G28**$)$ is equivalent to $\overline{S}_2 \cap \overline{S}_3 \neq \emptyset$. Similarly, $(\neg\,$**G29**$) \wedge$ $(\neg\,$**G30**$)$ is equivalent to $\overline{D}_2 \cap \overline{D}_3 \neq \emptyset$. By Proposition 6.2.1, we have $L \equiv R$. The proof is thus complete. ■

N.9.5 Proof of R28

The additional set of logic statements

To prove **R28**, we need an additional set of logic statements. The following logic statements are well-defined if and only if **G27** $\wedge$ **G28** is true. Recall the definition of e_2^*, e_3^*, and e_1^* in Appendix N.9 when **G27** $\wedge$ **G28** is true.

- **G32:** $e_2^* \neq e_3^*$ and $\mathsf{GCD}(\, m_{e_{s_2};e_2^*} m_{e_2^*;e_1^*},\, m_{e_{s_3};e_3^*} m_{e_3^*;e_1^*}) \equiv 1$.
- **G33:** $\mathsf{GCD}(\, m_{11},\, m_{e_2^*;e_1^*}) \equiv 1$.
- **G34:** $\mathsf{GCD}(\, m_{11},\, m_{e_3^*;e_1^*}) \equiv 1$.

The following logic statements are well-defined if and only if **G27** $\wedge$ **G28** $\wedge$ **G31** is true.

- **G35:** $\{e_2^*, e_1^*\} \subset \mathsf{1cut}(s_1; d_2)$.
- **G36:** $\{e_3^*, e_1^*\} \subset \mathsf{1cut}(s_1; d_3)$.

The skeleton of proving R28

We prove the following logic relationships, which jointly proves **R28**.

- **R34: G27** $\wedge$ **G28** $\Rightarrow$ **G32**.
- **R35: D4** $\wedge$ **G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ **G33** $\Rightarrow$ **G35**.
- **R36: D3** $\wedge$ **G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ **G34** $\Rightarrow$ **G36**.
- **R37: G27** $\wedge$ **G28** $\wedge$ $(\neg$ **G33**$)$ $\wedge$ $(\neg$ **G34**$)$ $\Rightarrow$ false.
- **R38: G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ $(\neg$ **G33**$)$ $\wedge$ **G36** $\Rightarrow$ false.
- **R39: G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ $(\neg$ **G34**$)$ $\wedge$ **G35** $\Rightarrow$ false.
- **R40: G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ **G35** $\wedge$ **G36** $\Rightarrow$ false.

Specifically, **R35** and **R39** jointly imply that

$$\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G33} \wedge (\neg\, \mathbf{G34}) \Rightarrow \text{false}.$$

Moreover, **R36** and **R38** jointly imply that

$$\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\, \mathbf{G33}) \wedge \mathbf{G34} \Rightarrow \text{ false.}$$

Furthermore, **R35**, **R36**, and **R40** jointly imply that

$$\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G33} \wedge \mathbf{G34} \Rightarrow \text{ false.}$$

Finally, **R37** implies that

$$\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\, \mathbf{G33}) \wedge (\neg\, \mathbf{G34}) \Rightarrow \text{ false.}$$

The above four relationships jointly imply $\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \Rightarrow$ false. By **R26** in Appendix N.9, i.e., $\mathbf{D3} \wedge \mathbf{D4} \Rightarrow \mathbf{G31}$, we thus have $\mathbf{D3} \wedge \mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28}$ $\Rightarrow$ false. The proof of **R28** is thus complete. The detailed proofs of **R34** to **R40** are provided in the next subsection.

The proofs of R34 to R40

We prove **R34** as follows.

Proof. Suppose $\mathbf{G27} \wedge \mathbf{G28}$ is true. Since e_1^* is the most upstream 1-edge cut separating d_1 from $\{s_2, s_3\}$, there must exist two edge-disjoint paths connecting $\{s_2, s_3\}$ and $\mathsf{tail}(e_1^*)$. By Property 1 of **G27** and **G28**, one path must use e_2^* and the other must use e_3^*. Due to the edge-disjointness, $e_2^* \neq e_3^*$. Since we have two edge-disjoint paths from s_2 (resp. s_3) to $\mathsf{tail}(e_1^*)$, we also have $\mathsf{GCD}(\, m_{e_{s_2};e_2^*} m_{e_2^*;e_1^*},\, m_{e_{s_3};e_3^*} m_{e_3^*;e_1^*}) \equiv 1$. ■

We prove **R35** as follows.

Proof. Suppose $\mathbf{D4} \wedge \mathbf{G27} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G33}$ is true. By the Properties of **G27** and **G28** and by **G31**, e_2^* (resp. e_3^*) is the most downstream edge of $\overline{S}_2$ (resp. $\overline{S}_3$). And both e_2^* and e_3^* are in the upstream of e_1^* where e_1^* is the most upstream edge of $\overline{D}_1$. Consider $m_{e_2^*;e_1^*}$, a factor of m_{21}. From Property 2 of **G27**, we have $\mathsf{GCD}(m_{23}, m_{e_2^*;e_1^*}) \equiv 1$. In addition, since **G27** $\wedge$ **G28** $\Rightarrow$ **G32** as established in **R34**, we have $\mathsf{GCD}(m_{31}, m_{e_2^*;e_1^*}) \equiv 1$. Together with the assumption that **D4** is true, we have for some positive integer l_4 such that

$$\mathsf{GCD}(m_{11}m_{12}^{l_4},\ m_{e_2^*;e_1^*}) = m_{e_2^*;e_1^*}. \tag{N.38}$$

Since we assume that **G33** is true, (N.38) further implies $\mathsf{GCD}(m_{12}^{l_4}, m_{e_2^*;e_1^*}) = m_{e_2^*;e_1^*}$. By Proposition 5.4.3, we must have **G35**: $\{e_2^*, e_1^*\} \subset \mathsf{1cut}(s_1; d_2)$. The proof is thus complete. ■

R36 is a symmetric version of **R35** and can be proved by relabeling (s_2, d_2) as (s_3, d_3), and relabeling (s_3, d_3) as (s_2, d_2) in the proof of **R35**.

We prove **R37** as follows.

Proof. Suppose **G27** $\wedge$ **G28** $\wedge (\neg$ **G33**$) \wedge (\neg$ **G34**$)$ is true. Since **G27** $\wedge$ **G28** is true, we have two edge-disjoint paths $P_{s_2\mathsf{tail}(e_1^*)}$ through e_2^* and $P_{s_3\mathsf{tail}(e_1^*)}$ through e_3^* if we recall **R34**. Consider $m_{e_2^*;e_1^*}$, a factor of m_{21}, and $m_{e_3^*;e_1^*}$, a factor of m_{31}. Since $\neg$ **G33** is true, there is an irreducible factor of $m_{e_2^*;e_1^*}$ that is also a factor of m_{11}. Since that factor is also a factor of m_{21}, by Proposition 5.4.3 and Property 1 of **G27**, there must exist at least one edge e' satisfying (i) $e_2^* \preceq e' \prec e_1^*$; (ii) $e' \in \overline{D}_{1;\{1,2\}}$; and (iii) $e' \in P_{s_2\mathsf{tail}(e_1^*)}$. Similarly, $\neg$ **G34** implies that there exists at least one edge e'' satisfying (i) $e_3^* \preceq e'' \prec e_1^*$; (ii) $e'' \in \overline{D}_{1;\{1,3\}}$; and (iii) $e'' \in P_{s_3\mathsf{tail}(e_1^*)}$. Then the above observation implies that $e' \in P_{s_2\mathsf{tail}(e_1^*)} \cap \mathsf{1cut}(s_1; d_1)$ and $e'' \in P_{s_3\mathsf{tail}(e_1^*)} \cap \mathsf{1cut}(s_1; d_1)$. Since $P_{s_2\mathsf{tail}(e_1^*)}$ and $P_{s_3\mathsf{tail}(e_1^*)}$ are edge-disjoint paths, it must be $e' \neq e''$. But both e' and e'' are 1-edge cuts separating s_1 and d_1. Thus e' and e'' must be reachable from each other: either $e' \prec e''$ or $e'' \prec e'$. However, both cases are impossible because one in the upstream can always follow the corresponding $P_{s_2\mathsf{tail}(e_1^*)}$ or $P_{s_3\mathsf{tail}(e_1^*)}$ path to e_1^* without using the one in the downstream. For example, if $e' \prec e''$, then s_1 can first reach e' and follow $P_{s_2\mathsf{tail}(e_1^*)}$ to arrive at $\mathsf{tail}(e_1^*)$ without using e''. Since $e_1^* \in \overline{D}_1$

reaches d_1, this contradicts $e'' \in \mathsf{1cut}(s_1; d_1)$. Since neither case can be true, the proof is thus complete. ■

We prove **R38** as follows.

Proof. Suppose **G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ ($\neg$ **G33**) $\wedge$ **G36** is true. By the Properties of **G27** and **G28** and by **G31**, e_2^* (resp. e_3^*) is the most downstream edge of $\overline{S}_2$ (resp. $\overline{S}_3$). And both e_2^* and e_3^* are in the upstream of e_1^* where e_1^* is the most upstream edge of $\overline{D}_1$. Since e_1^* is the most upstream $\overline{D}_1$ edge, there exist three edge-disjoint paths $P_{s_2 \mathsf{tail}(e_1^*)}$, $P_{s_3 \mathsf{tail}(e_1^*)}$, and $P_{\mathsf{head}(e_1^*) d_1}$. Fix any arbitrary construction of these paths. Obviously, $P_{s_2 \mathsf{tail}(e_1^*)}$ uses e_2^* and $P_{s_3 \mathsf{tail}(e_1^*)}$ uses e_3^*.

Since $\neg$ **G33** is true, there is an irreducible factor of $m_{e_2^*; e_1^*}$ that is also a factor of m_{11}. Since that factor is also a factor of m_{21}, by Proposition 5.4.3, there must exist an edge e satisfying (i) $e_2^* \preceq e \prec e_1^*$; (ii) $e \in \mathsf{1cut}(s_1; d_1) \cap \mathsf{1cut}(s_2; d_1)$. By (i), (ii), and the construction $e_1^* \in \overline{D}_1 \subset \mathsf{1cut}(s_2; d_1)$, the pre-defined path $P_{s_2 \mathsf{tail}(e_1^*)}$ must use such e.

Since **G36** is true, e_3^* is reachable from s_1 and e_1^* reaches to d_3. Choose arbitrarily one path $P_{s_1 \mathsf{tail}(e_3^*)}$ from s_1 to $\mathsf{tail}(e_3^*)$ and one path $P_{\mathsf{head}(e_1^*) d_3}$ from $\mathsf{head}(e_1^*)$ to d_3. We argue that $P_{s_1 \mathsf{tail}(e_3^*)}$ must be vertex-disjoint with $P_{s_2 \mathsf{tail}(e_1^*)}$. Suppose not and let v denote a vertex shared by $P_{s_1 \mathsf{tail}(e_3^*)}$ and $P_{s_2 \mathsf{tail}(e_1^*)}$. Then there is a s_1-to-d_3 path $P_{s_1 \mathsf{tail}(e_3^*)} v P_{s_2 \mathsf{tail}(e_1^*)} e_1^* P_{\mathsf{head}(e_1^*) d_3}$ without using e_3^*. This contradicts the assumption **G36** since **G36** implies $e_3^* \in \mathsf{1cut}(s_1; d_3)$. However, if $P_{s_1 \mathsf{tail}(e_3^*)}$ is vertex-disjoint with $P_{s_2 \mathsf{tail}(e_1^*)}$, then there is an s_1-to-d_1 path $P_{s_1 \mathsf{tail}(e_3^*)}\ e_3^* P_{s_3 \mathsf{tail}(e_1^*)} e_1^* P_{\mathsf{head}(e_1^*) d_1}$ not using the edge e defined in the previous paragraph since $e \in P_{s_2 \mathsf{tail}(e_1^*)}$ and $P_{s_2 \mathsf{tail}(e_1^*)}$ is edge-disjoint with $P_{s_3 \mathsf{tail}(e_1^*)}$. This also contradicts (ii). Since neither case can be true, the proof of **R38** is thus complete. ■

R39 is a symmetric version of **R38** and can be proved by swapping the roles of s_2 and s_3, and the roles of d_2 and d_3 in the proof of **R38**.

We prove **R40** as follows.

Proof. Suppose **G27** $\wedge$ **G28** $\wedge$ **G31** $\wedge$ **G35** $\wedge$ **G36** is true. By the Properties of **G27** and **G28** and by **G31**, e_2^* (resp. e_3^*) is the most downstream edge of $\overline{S}_2$ (resp. $\overline{S}_3$). Also $e_2^* \prec e_1^*$ and $e_3^* \prec e_1^*$ where e_1^* is the most upstream $\overline{D}_1$ edge.

By **G36**, there exists a path from s_1 to e_3^*. Since $e_3^* \in \overline{S}_3$, there exists a path from e_3^* to d_2 without using e_1^*. As a result, there exists a path from s_1 to d_2 through e_3^* without using e_1^*. This contradicts the assumption **G35** since **G35** implies $e_1^* \in \mathsf{1cut}(s_1; d_2)$. The proof is thus complete. ■

N.9.6 Proof of R29

The additional set of logic statements

To prove **R29**, we need some additional sets of logic statements. The following logic statements are well-defined if and only if **G28** is true. Recall the definition of e_3^* and e_1^* when **G28** is true.

- **G37:** $e_3^* \in \mathsf{1cut}(s_1; d_1)$.
- **G38:** $e_3^* \in \mathsf{1cut}(s_1; d_3)$.
- **G39:** $e_1^* \in \mathsf{1cut}(s_1; d_1)$.
- **G40:** $e_1^* \in \mathsf{1cut}(s_1; d_3)$.
- **G41:** $e_3^* \in \mathsf{1cut}(s_1; d_2)$.

The following logic statements are well-defined if and only if $(\neg$ **G27**$)$ $\wedge$ **G28** is true. Recall the definition of e_u^{21}, e_v^{21}, e_3^*, and e_1^* when $(\neg$ **G27**$)$ $\wedge$ **G28** is true.

- **G42:** $e_1^* = e_u^{21}$.
- **G43:** Let e' be the most downstream edge of $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(s_1; \mathsf{tail}(e_3^*))$ and also let e'' be the most upstream edge of $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(\mathsf{head}(e_3^*); d_2)$. Then, e' and e'' simultaneously satisfy the following two conditions: (i) both e' and e'' belong to $\mathsf{1cut}(s_1; d_3)$; and (ii) $e'' \in \mathsf{1cut}(\mathsf{head}(e_v^{21}); \mathsf{tail}(e_{d_3}))$ and $e'' \prec e_{d_2}$.

The skeleton of proving R29

We prove the following relationships, which jointly proves **R29**.

- **R41:** $(\neg\,\mathbf{G27}) \wedge \mathbf{G28} \Rightarrow \mathbf{G42}$.
- **R42:** $\mathbf{D3} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \Rightarrow (\mathbf{G37} \vee \mathbf{G38}) \wedge (\mathbf{G39} \vee \mathbf{G40})$.
- **R43:** $\mathbf{G1} \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \Rightarrow \neg\,\mathbf{G41}$.
- **R44:** $\mathbf{D3} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \wedge (\neg\,\mathbf{G41}) \Rightarrow \mathbf{G43}$.
- **R45:** $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \Rightarrow$ false.
- **R46:** $(\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\,\mathbf{G37}) \wedge \mathbf{G38} \wedge \mathbf{G39} \Rightarrow$ false.
- **R47:** $\mathbf{LNR} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\,\mathbf{G37}) \wedge \mathbf{G38} \wedge \mathbf{G40} \Rightarrow$ false.

One can easily verify that jointly **R46** and **R47** imply

$$\mathbf{LNR} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\,\mathbf{G37}) \wedge \mathbf{G38} \wedge (\mathbf{G39} \vee \mathbf{G40}) \Rightarrow \text{false}.$$

From the above logic relationship and by **R42**, we have

$$\mathbf{LNR} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\,\mathbf{G37}) \wedge \mathbf{G38} \Rightarrow \text{false}.$$

From the above logic relationship and by **R45**, we have

$$\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\mathbf{G37} \vee \mathbf{G38}) \Rightarrow \text{false}.$$

By applying **R42** and **R26**, we have $\mathbf{LNR} \wedge \mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge \mathbf{D4} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \Rightarrow$ false, which proves **R29**. The detailed proofs for **R41** to **R47** are provided in the next subsection.

The proofs of R41 to R47

We prove **R41** as follows.

Proof. Suppose $(\neg\,\mathbf{G27}) \wedge \mathbf{G28}$ is true. By $\neg\,\mathbf{G27}$ being true and its Property 1, we have e_u^{21} (resp. e_v^{21}), the most upstream (resp. downstream) edge of $\overline{S}_2 \cap \overline{D}_1$. Since

$\neg$ **G27** implies that $\overline{D}_1 \neq \emptyset$, by Property 1 of **G28**, we also have e_1^*, the most upstream $\overline{D}_1$ edge.

Since $\overline{D}_1 \cap \overline{S}_2 \neq \emptyset$, we can partition the non-empty $\overline{D}_1$ by $\overline{D}_1 \backslash \overline{S}_2$ and $\overline{D}_1 \cap \overline{S}_2$. By the (s, d)-symmetric version of Lemma 6.1.3, if $\overline{D}_1 \backslash \overline{S}_2 \neq \emptyset$, then any $\overline{D}_1 \backslash \overline{S}_2$ edge must be in the downstream of $e_v^{21} \in \overline{D}_1 \cap \overline{S}_2 \subset \overline{S}_2$. Thus, e_u^{21}, the most upstream $\overline{D}_1 \cap \overline{S}_2$ edge, must also be the most upstream edge of $\overline{D}_1$. Therefore, $e_1^* = e_u^{21}$. The proof is thus complete. ■

We prove **R42** as follows.

Proof. Suppose $\mathbf{D3} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31}$ is true. Since $(\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31}$ is true, e_3^* (resp. e_1^*) is the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_1$) and $e_3^* \prec e_1^*$. By **R41**, **G42** is also true and thus e_1^* is also the most upstream edge of $\overline{S}_2 \cap \overline{D}_1$.

Consider $m_{e_3^*;e_1^*}$, a factor of m_{31}. From Property 2 of **G28**, $\mathsf{GCD}(\,m_{32},\, m_{e_3^*;e_1^*}) \equiv 1$. By **G42** being true and Property 2 of $\neg$ **G27**, we also have $\mathsf{GCD}(\,m_{e_{s_2};e_1^*},\, m_{e_{s_3};e_3^*} m_{e_3^*;e_1^*}) \equiv 1$, which implies that $\mathsf{GCD}(\,m_{21},\, m_{e_3^*;e_1^*}) \equiv 1$. Then since **D3** is true, we have for some positive integer l_2 such that

$$\mathsf{GCD}(\,m_{11} m_{13}^{l_2},\; m_{e_3^*;e_1^*}) = m_{e_3^*;e_1^*}.$$

Proposition 5.4.3 then implies that both e_3^* and e_1^* must be in $\mathsf{1cut}(s_1; d_1) \cup \mathsf{1cut}(s_1; d_3)$. This is equivalent to $(\mathbf{G37} \vee \mathbf{G38}) \wedge (\mathbf{G39} \vee \mathbf{G40})$ being true. The proof of **R42** is complete. ■

We prove **R43** as follows.

Proof. We prove an equivalent form: $\mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \wedge \mathbf{G41} \Rightarrow \neg\,\mathbf{G1}$. Suppose $\mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \wedge \mathbf{G41}$ is true. Since $\mathbf{G28} \wedge \mathbf{G31}$ is true, we have e_3^* being the most downstream edge of $\overline{S}_3$. Therefore $e_3^* \in \mathsf{1cut}(s_3; d_1) \cap \mathsf{1cut}(s_3; d_2)$. Since $\mathbf{G37} \wedge \mathbf{G41}$ is also true, e_3^* belongs to $\mathsf{1cut}(s_1; d_1) \cap \mathsf{1cut}(s_1; d_2)$ as well. As a result, $\mathsf{EC}(\{s_1, s_3\}; \{d_1, d_2\}) = 1$, which, by Corollary 5.4.2 implies $\neg$ **G1**. ■

We prove **R44** as follows.

Proof. Suppose that $\mathbf{D3} \wedge (\neg\,\mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37} \wedge (\neg\,\mathbf{G41})$ is true, which by **R41** implies that **G42** is true as well. Since $\mathbf{G28} \wedge \mathbf{G31}$ is true, e_3^* (resp. e_1^*) is the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_1$) and $e_3^* \prec e_1^*$. Recall the definition in **G43** that e' is the most downstream edge of $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(s_1; \mathsf{tail}(e_3^*))$ and e'' is the most upstream edge of $\mathsf{1cut}(s_1; d_2) \cap \mathsf{1cut}(\mathsf{head}(e_3^*); d_2)$. By the constructions of e' and e'', we must have $e_{s_1} \preceq e' \prec e_3^* \prec e'' \preceq e_{d_2}$. Then, we claim that the above construction together with $\neg\,\mathbf{G41}$ implies $\mathsf{EC}(\mathsf{head}(e'); \mathsf{tail}(e'')) \geq 2$. The reason is that if $\mathsf{EC}(\mathsf{head}(e'); \mathsf{tail}(e'')) = 1$, then we can find an 1-edge cut separating $\mathsf{head}(e')$ and $\mathsf{tail}(e'')$ and by $\neg\,\mathbf{G41}$ such edge cut must not be e_3^*. Hence, such edge cut is either an upstream or a downstream edge of e_3^*. However, either case is impossible, because the edge cut being in the upstream of e_3^* will contradict that e' is the most downstream one during its construction. Similarly, the edge cut being in downstream of e_3^* will contradict the construction of e''. The conclusion $\mathsf{EC}(\mathsf{head}(e'); \mathsf{tail}(e'')) \geq 2$ further implies $m_{e';e''}$ is irreducible.

Further, because e_3^* is the most downstream $\overline{S}_3$ edge and e'', by construction, satisfies $e'' \in \mathsf{1cut}(s_3; d_2)$, e'' must not belong to $\mathsf{1cut}(s_3; d_1)$, which in turn implies $e'' \notin \mathsf{1cut}(\mathsf{head}(e_3^*); d_1)$. Since **G37** is true, s_1 can reach e_3^*. Therefore, there exists an s_1-to-$d_1$path using e_3^* but not using e''. As a result, $e'' \notin \mathsf{1cut}(s_1; d_1)$. Together with $m_{e';e''}$ being irreducible, we thus have $\mathsf{GCD}(m_{11}, m_{e';e''}) \equiv 1$ by Proposition 5.4.3.

Now we argue that $\mathsf{GCD}(m_{21}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, we must have e' being an 1-edge cut separating s_2 and d_1. Since e_1^* is the most upstream $\overline{D}_1$ edge, by Property 2 of **G28**, there exists as_2-to-$d_1$path P_{21} not using e_3^*. By the construction of e', s_1 reaches e'. Choose arbitrarily a path $P_{s_1e'}$ from s_1 to e'. Then, the followings_1-to-$d_1$path $P_{s_1e'}e'P_{21}$ does not use e_3^*, which contradicts **G37**. As a result, we must have $\mathsf{GCD}(m_{21}, m_{e';e''}) \equiv 1$.

Now we argue that $\mathsf{GCD}(m_{32}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, both e' and e'' must belong to $\mathsf{1cut}(s_3; d_2)$ and there is no 1-edge cut of $\mathsf{1cut}(s_3; d_2)$ that is strictly being downstream to e' and being upstream to e''. This, however, contradicts the above construction that $e' \prec e_3^* \prec e''$ and $e_3^* \in \overline{S}_3 \subset \mathsf{1cut}(s_3; d_2)$. As a result, we must have $\mathsf{GCD}(\, m_{32},\, m_{e';e''}) \equiv 1$.

Together with the assumption that **D3** is true and the fact that $m_{e';e''}$ is a factor of m_{12}, we have for some positive integer l_2 such that

$$\mathsf{GCD}(\, m_{13}^{l_2},\ m_{e';e''}) = m_{e';e''}.$$

Proposition 5.4.3 then implies $\{e', e''\} \subset \mathsf{1cut}(s_1; d_3)$, which shows the first half of **G43**.

Therefore, anys_1-to-$d_3$path must use e''. Since $e_3^* \prec e''$ and s_1 can reach e_3^*, any path from $\mathsf{head}(e_3^*)$ to d_3 must use e''. Note that when we establish $\mathsf{GCD}(\, m_{11},\, m_{e';e''}) \equiv 1$ in the beginning of this proof, we also proved that $e'' \notin \mathsf{1cut}(s_1; d_1)$. Thus, there exists a path from $\mathsf{head}(e_3^*)$ to d_1 not using e''. Then such path must use e_v^{21} because e_v^{21} is also an 1-edge cut separating $\mathsf{head}(e_3^*)$ and d_1 by the facts that $e_v^{21} \in \overline{S}_2 \cap \overline{D}_1 \subset \mathsf{1cut}(s_3; d_1)$; $e_3^* \prec e_v^{21}$; s_3 reaches e_3^*. Moreover, since $e_v^{21} \in \overline{S}_2 \cap \overline{D}_1 \subset \mathsf{1cut}(s_2; d_3)$, $\mathsf{head}(e_v^{21})$ can reach d_3. In sum, we have shown that (i) any path from $\mathsf{head}(e_3^*)$ to d_3 must use e''; (ii) there exists a path from e_3^* to e_v^{21} not using e''; (iii) $\mathsf{head}(e_v^{21})$ can reach d_3. Jointly (i) to (iii) imply that any path from $\mathsf{head}(e_v^{21})$ to d_3 must use e''. As a result, we have $e'' \in \mathsf{1cut}(\mathsf{head}(e_v^{21}); d_3)$. Also e'' must not be the d_3-destination edge e_{d_3} since by construction $e'' \preceq e_{d_2}$, $e_{d_2} \neq e_{d_3}$, and $|\mathsf{Out}(d_3)| = 0$. This further implies that e'' must not be the d_2-destination edge e_{d_2} since $e'' \prec e_{d_3}$ and $|\mathsf{Out}(d_2)| = 0$. We have thus proven the second half of **G43**: $e'' \in \mathsf{1cut}(\mathsf{head}(e_v^{21}); \mathsf{tail}(e_{d_3}))$ and $e'' \prec e_{d_2}$. The proof of **R44** is complete. ■

We prove **R45** as follows.

Proof. Suppose $\mathbf{G1} \wedge \mathbf{E0} \wedge \mathbf{D3} \wedge (\neg\, \mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge \mathbf{G37}$ is true. By **R41**, **R43**, and **R44**, we know that **G42**, $\neg$ **G41**, and **G43** are true as well. For the following we construct 10 path segments that interconnects s_1 to s_3, d_1 to d_3, and three edges e'', e_3^*, and e_1^*.

• P_1: a path starting from e_{s_1} and ending at e'. This is always possible due to **G43** being true.

• P_2: a path from s_2 to $\mathsf{tail}(e_1^*)$ without using e_3^*. This is always possible due to the properties of **G28**.

• P_3: a path from s_3 to $\mathsf{tail}(e_3^*)$. This is always possible due to **G28** and **G31** being true. We also impose that P_3 is edge-disjoint with P_2. Again, this is always possible due to Property 2 of **G28**.

• P_4: a path from $\mathsf{head}(e')$ to $\mathsf{tail}(e'')$. This is always possible due to **G43** being true. We also impose the condition that $e_3^* \notin P_4$. Again this is always possible since $\neg$ **G41** being true, which implies that one can always find a path from s_1 to d_2 not using e_3^* but uses both e' and e'' (due to the construction of e' and e'' of **G43**).

• P_5: a path from $\mathsf{head}(e_3^*)$ to $\mathsf{tail}(e_1^*)$. We also impose the condition that P_5 is edge-disjoint with P_2. The construction of such P_5 is always possible due to the Properties of **G28**.

• P_6: a path from $\mathsf{head}(e_1^*)$ to d_1. This is always possible due to $(\neg$ **G27**$) \wedge$ **G28** being true. We also impose the condition that $e'' \notin P_6$. Again this is always possible. The reason is that e_3^* is the most downstream $\overline{S}_3$ edge and thus there are two edge-disjoint paths connecting $\mathsf{head}(e_3^*)$ and $\{d_1, d_2\}$. By our construction e'' must be in the latter path while we can choose P_6 to be part of the first path.

• P_7: a path from $\mathsf{head}(e_3^*)$ to $\mathsf{tail}(e'')$, which is edge-disjoint with $\{P_5, e_1^*, P_6\}$. This is always possible due to the property of e_3^* and the construction of **G43**.

• P_8: a path from $\mathsf{head}(e'')$ to d_2, which is edge-disjoint with $\{P_5, e_1^*, P_6\}$. This is always possible due to the property of e_3^* and the construction of **G43**.

• P_9: a path from $\mathsf{head}(e_1^*)$ to $\mathsf{tail}(e'')$. This is always possible due to **G43** being true (in particular the (ii) condition of **G43**).

• P_{10}: a path from $\mathsf{head}(e'')$ to d_3. This is always possible due to **G43** being true (in particular the (ii) condition of **G43**).

Fig. N.3 illustrates the relative topology of these 10 paths. We now consider the subgraph G' induced by the 10 paths plus the three edges e'', e_3^*, and e_1^*. One can

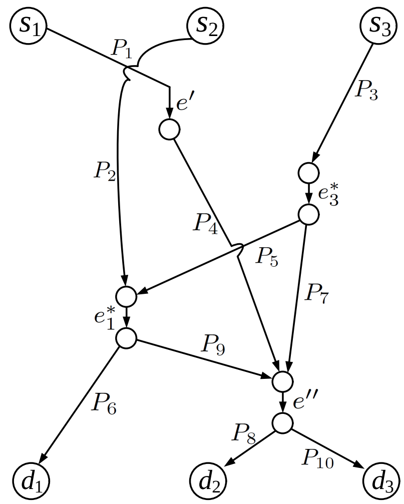


Fig. N.3. The subgraph G' of the 3-unicast ANA network $G_{3\text{ANA}}$ induced by 10 paths and three edges e'', e_3^*, and e_1^* in the proof of **R45**.

easily check that s_i can reach d_j for all $i \neq j$. In particular, s_1 can reach d_2 through $P_1P_4e''P_8$; s_1 can reach d_3 through $P_1P_4e''P_{10}$; s_2 can reach d_1 through $P_2e_1^*P_6$; s_2 can reach d_3 through $P_2e_1^*P_9e''P_{10}$; s_3 can reach d_1 through $P_3e_3^*P_5e_1^*P_6$; and s_3 can reach d_2 through either $P_3e_3^*P_5e_1^*P_9e''P_8$ or $P_3e_3^*P_7e''P_8$. Furthermore, topologically, the 6 paths P_5 to P_{10} are all in the downstream of e_3^*.

For the following we argue that s_1 cannot reach d_1 in the induced subgraph G'. To that end, we first notice that by **G37**, $e_3^* \in \mathsf{1cut}(s_1; d_1)$ in the original graph. Therefore anys_1-to-$d_1$path in the subgraph must use e_3^* as well. Since P_5 to P_{10} are in the downstream of e_3^*, we only need to consider P_1 to P_4.

By definition, P_3 reaches e_3^*. We now like to show that $e_3^* \notin P_2$, and $\{P_2, P_3\}$ are vertex-disjoint paths. The first statement is done by our construction. Suppose P_2 and P_3 share a common vertex v (v can possibly be $\mathsf{tail}(e_3^*)$), then there exists as_3-to-$d_1$path $P_3vP_2e_1^*P_6$ not using e_3^*. This contradicts **G28** (specifically $e_3^* \in \overline{S}_3 \subset \mathsf{1cut}(s_3; d_1)$). The above arguments show that the first time a path enters/touches part of P_3 (including $\mathsf{tail}(e_3^*)$) must be along either P_1 or P_4 (cannot be along P_2). As a result, when deciding whether there exists ans_1-to-$d_1$path using e_3^*, we only need to consider whether P_1 (and/or P_4) can share a vertex with P_3. To that end, we will prove that (i) $e_3^* \notin P_1$; (ii)

$\{P_1, P_3\}$ are vertex-disjoint paths; (iii) $e_3^* \notin P_4$; and (iv) $\{P_3, P_4\}$ are vertex-disjoint paths. Once (i) to (iv) are true, then there is nos_1-to-$d_1$path in the subgraph G'.

We now notice that (i) is true since $e' \prec e_3^*$; (iii) is true due to our construction; (ii) is true otherwise let v denote the shared vertex and there will exist as_3-to-$d_2$path $P_3 v P_1 P_4 e'' P_8$ without using e_3^*, which contradicts **G28** ($e_3^* \in \overline{S}_3 \subset \mathsf{1cut}(s_3; d_2)$); and by the same reason, (iv) is true otherwise let v denote the shared vertex and there will exist as_3-to-$d_2$path $P_3 v P_4 e'' P_8$ without using e_3^*. We have thus proven that there is nos_1-to-$d_1$path in G'.

Since **E0** is true, $G_{3\text{ANA}}$ must satisfy (N.1) with at least one non-zero coefficients α_i and β_j, respectively. Applying Proposition 5.4.2 implies that the subgraph G' must satisfy (N.1) with the same coefficient values. Note that there is no path from s_1 to d_1 on G' but any channel gain m_{ij} for all $i \neq j$ is non-trivial on G'. Recalling the expression of (N.1), its LHS becomes zero since it contains the zero polynomial m_{11} as a factor. We have $g(\{m_{ij} : \forall\, (i,j) \in I_{3\text{ANA}}\})\, \psi_\beta^{(n)}(R, L) = 0$ and thus $\psi_\beta^{(n)}(R, L) = 0$ with at least one non-zero coefficients β_j. This further implies that the set of polynomials $\{R^n, R^{n-1}L, \cdots, RL^{n-1}, L^n\}$ is linearly dependent on G'. Since this is the Vandermonde form, it is equivalent to that $L \equiv R$ holds on G'. However for the following, we will show that (a) $\overline{D}_1 \cap \overline{D}_2 = \emptyset$; (b) $\overline{S}_1 \cap \overline{S}_3 = \emptyset$; and (c) $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G', which implies by Proposition 6.2.1 that G' indeed satisfies $L \not\equiv R$. This is a contradiction and thus proves **R45**.

(a) $\overline{D}_1 \cap \overline{D}_2 = \emptyset$ on G': Note that any $\overline{D}_1$ edge can exist on (i) e_1^*; and (ii) P_6. Note also that any $\overline{D}_2$ edge can exist on (i) e''; and (ii) P_8. But from the above constructions, P_6 was chosen not to use e''. In addition, P_8 was chosen to be edge-disjoint with $\{e_1^*, P_6\}$. Moreover, $e_1^* \prec e''$. Thus, we must have $\overline{D}_1 \cap \overline{D}_2 = \emptyset$ on G'.

(b) $\overline{S}_1 \cap \overline{S}_3 = \emptyset$ on G': Note that any $\overline{S}_1$ edge can exist on (i) P_1; (ii) P_4; (iii) e''; and (iv) an edge that P_8 and P_{10} shares. Note also that any $\overline{S}_3$ can exist on (i) P_3; and (ii) e_3^*. But e_3^* is in the upstream of e'', P_8, and P_{10}. Also, e_3^* is in the downstream of e', ending edge of P_1. In addition, P_4 was chosen not to use e_3^*. Moreover, we already showed that $\{P_1, P_3\}$ are vertex-disjoint paths; and $\{P_3, P_4\}$ are vertex-disjoint paths. Thus, we must have $\overline{S}_1 \cap \overline{S}_3 = \emptyset$ on G'.

(c) $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G': Note that any $\overline{S}_2$ edge can exist on (i) P_2; (ii) e_1^*; (iii) an edge that P_6 and P_9 shares; and (iv) an edge that P_6 and P_{10} share. Note also that any $\overline{S}_3$ edge can exist on (i) P_3; and (ii) e_3^*. However, e_3^* is in the upstream of e_1^*, P_6, P_9, and P_{10}. In addition, P_2 was chosen not to use e_3^*. Moreover, we already showed that $\{P_2, P_3\}$ are vertex-disjoint paths. Thus, we must have $\overline{S}_2 \cap \overline{S}_3 = \emptyset$ on G'. ■

We prove **R46** as follows.

Proof. Suppose that $(\neg\, \mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\, \mathbf{G37}) \wedge \mathbf{G38} \wedge \mathbf{G39}$ is true. By **R41**, **G42** is true as well. Since $\mathbf{G28} \wedge \mathbf{G31}$ is true, e_3^* (resp. e_1^*) is the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_1$). From $(\neg\, \mathbf{G37}) \wedge \mathbf{G38} \wedge \mathbf{G39}$ being true, we also have $e_3^* \in \mathsf{1cut}(s_1; d_3) \backslash \mathsf{1cut}(s_1; d_1)$ and $e_1^* \in \mathsf{1cut}(s_1; d_1)$.

Since **G42** is true, we have $e_1^* = e_u^{21}$ is in $\overline{S}_2$. Any arbitrary s_2-to-d_3 path P_{23} thus must use e_1^*. Since $e_3^* \notin \mathsf{1cut}(s_1; d_1)$ and $e_1^* \in \mathsf{1cut}(s_1; d_1)$, there exists an s_1-to-d_1 path Q_{11} using e_1^* but not using e_3^*. Then, we can create a s_1-to-d_3 path $Q_{11} e_1^* P_{23}$ not using e_3^*, which contradicts $e_3^* \in \mathsf{1cut}(s_1; d_3)$. The proof of **R46** is complete. ■

We prove **R47** as follows.

Proof. Suppose that $\mathbf{LNR} \wedge \mathbf{D4} \wedge (\neg\, \mathbf{G27}) \wedge \mathbf{G28} \wedge \mathbf{G31} \wedge (\neg\, \mathbf{G37}) \wedge \mathbf{G38} \wedge \mathbf{G40}$ is true. Since $\mathbf{G28} \wedge \mathbf{G31}$ is true, e_3^* (resp. e_1^*) is the most downstream (resp. upstream) edge of $\overline{S}_3$ (resp. $\overline{D}_1$). Since $(\neg\, \mathbf{G27}) \wedge \mathbf{G28}$ implies **G42**, e_1^* also belongs to $\overline{S}_2$, which implies that $e_1^* \in \mathsf{1cut}(s_2; d_3)$. Since **G40** is true, we have $e_1^* \in \mathsf{1cut}(s_1; d_3)$. Jointly the above arguments imply $e_1^* \in \overline{D}_1 \cap \overline{D}_3$. Also, **G38** being true implies $e_3^* \in \overline{S}_3 \cap \mathsf{1cut}(s_1; d_3)$. Since **LNR** is true and $\overline{D}_1 \cap \overline{D}_3 \neq \emptyset$, by Proposition 6.2.1 we must have $\overline{S}_1 \cap \overline{S}_3 = \emptyset$, which implies that e_3^* cannot belong to $\mathsf{1cut}(s_1; d_2)$.

Let a node u be the tail of the edge e_3^*. Since $e_3^* \in \mathsf{1cut}(s_1; d_3)$, u is reachable from s_1. Since $e_3^* \in \overline{S}_3$, u is also reachable form s_3. Consider the collection of edges, $\mathsf{1cut}(s_1; u) \cap \mathsf{1cut}(s_3; u)$ (may be empty), all edges of which are in the upstream of

e_3^* if non-empty. Note that $(\mathsf{1cut}(s_1;u) \cap \mathsf{1cut}(s_3;u)) \cup \{e_3^*\}$ is always non-empty (since it contains at least e_3^*). Then, we use e'' to denote the most upstream edge of $(\mathsf{1cut}(s_1;u) \cap \mathsf{1cut}(s_3;u)) \cup \{e_3^*\}$. Let e' denote the most downstream edge among all edges in $\mathsf{1cut}(s_1;\mathsf{tail}(e''))$. Such choice is always possible since $\mathsf{1cut}(s_1;\mathsf{tail}(e''))$ contains at least one edge (the s_1-source edge e_{s_1}) and thus we have $e_{s_1} \preceq e' \prec e'' \preceq e_3^*$. Since we choose e' to be the most downstream one, by Proposition 5.4.3 the channel gain $m_{e';e''}$ must be irreducible. Moreover, since $e_3^* \in \mathsf{1cut}(s_1;d_3)$, any path from s_1 to d_3 must use e_3^*. Consequently since $e'' \in \mathsf{1cut}(s_1;u) \cup \{e_3^*\}$, any path from s_1 to d_3 must also use e''. Consequently since $e' \in \mathsf{1cut}(s_1;\mathsf{tail}(e''))$, any path from s_1 to d_3 must also use e'. As a result, $\{e', e''\} \subset \mathsf{1cut}(s_1;d_3)$. Therefore $m_{e';e''}$ is a factor of m_{13}.

Now we argue that $\mathsf{GCD}(m_{31}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, by Proposition 5.4.3 we must have $e' \in \mathsf{1cut}(s_3;d_1)$. Note that $e' = e_{s_1}$ cannot be a 1-edge cut separating s_3 and d_1 from the definitions (i) and (ii) of the 3-unicast ANA network. Thus, we only need to consider the case when $e_{s_1} \prec e'$ since $e_{s_1} \preceq e'$ from the construction of e'. Since $e_3^* \in \mathsf{1cut}(s_3;d_1)$ and $e' \prec e_3^*$ is an 1-edge cut separating s_3 and d_1, we must have $e' \in \mathsf{1cut}(s_3;u)$. Note that the most downstream $\mathsf{1cut}(s_1;\mathsf{tail}(e''))$ edge e' also belongs to $\mathsf{1cut}(s_1;u)$ from our construction. Therefore, jointly, this contradicts the construction that e'' is the most upstream edge of $(\mathsf{1cut}(s_1;u) \cap \mathsf{1cut}(s_3;u)) \cup \{e_3^*\}$ since $e' \prec e''$.

Now we argue that $\mathsf{GCD}(m_{23}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, we must have $e' \in \mathsf{1cut}(s_2;d_3)$ and thus $e_{s_1} \prec e'$. Choose arbitrarily a path from s_1 to e'. Since we have already established $e_3^* \prec e_1^*$ and e_1^* is the most upstream edge of $\overline{D}_1$, there exists a path $P_{s_2\mathsf{tail}(e_1^*)}$ from s_2 to $\mathsf{tail}(e_1^*)$ not using e_3^*. Since e_1^* is also in $\overline{D}_3$, $\mathsf{head}(e_1^*)$ can reach d_3. Note that the chosen path $P_{s_2\mathsf{tail}(e_1^*)}$ must use e' since $e' \in \mathsf{1cut}(s_2;d_3)$. As a result, s_1 can reach d_3 by going to e' first, and then following $P_{s_2\mathsf{tail}(e_1^*)}$ to e_1^*, and then going to d_3, without using e_3^*. This contradicts the assumption that $e_3^* \in \mathsf{1cut}(s_1;d_3)$.

Now we argue that $\mathsf{GCD}(m_{12}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, we must have $e'' \in \mathsf{1cut}(s_1; d_2)$. Since we have established $\neg$ **G41** (i.e., $e_3^* \notin \mathsf{1cut}(s_1; d_2)$), we only need to consider the case when $e'' \prec e_3^*$. Then by construction there exists as_1-to-$d_2$path P_{12} going through e'' but not e_3^*. However, since by construction e'' is reachable from s_3, there exists a path from s_3 to e'' first and then use P_{12} to arrive at d_2. Such as_3-to-$d_2$path does not use e_3^*, which contradicts the assumption that $e_3^* \in \overline{S}_3 \subset \mathsf{1cut}(s_3; d_2)$.

Now we argue that $\mathsf{GCD}(m_{11}, m_{e';e''}) \equiv 1$. Suppose not. Since $m_{e';e''}$ is irreducible, we must have $e'' \in \mathsf{1cut}(s_1; d_1)$. Since $\neg$ **G37** is true (i.e., $e_3^* \notin \mathsf{1cut}(s_1; d_1)$), we only need to consider the case when $e'' \prec e_3^*$. Then by construction there exists as_1-to-$d_1$path P_{11} going through e'' but not e_3^*. However, since by construction e'' is reachable from s_3, there exists a path from s_3 to e'' first and then use P_{11} to arrive at d_1. Such as_3-to-d_1 path does not use e_3^*, which contradicts the assumption that $e_3^* \in \overline{S}_3 \subset \mathsf{1cut}(s_3; d_1)$.

The four statements in the previous paragraphs shows that

$$\mathsf{GCD}(m_{11}m_{12}m_{23}m_{31},\ m_{e';e''}) \equiv 1.$$

This, however, contradicts the assumption that **D4** is true since we have shown that $m_{e';e''}$ is a factor of m_{13}. The proof of **R47** is thus complete. ■

N.9.7 Proof of R30

If we swap the roles of s_2 and s_3, and the roles of d_2 and d_3, then the proof of **R29** in Appendix N.9.6 can be directly applied to show **R30**. More specifically, note that both **D3** and **D4** are converted back and forth from each other when swapping the flow indices. Similarly, the index swapping also converts **G27** to **G28** and vice versa. Since **LNR**, **G1**, and **E0** remain the same after swapping the flow indices, we can see that **R29** becomes **R30** after swapping the flow indices. The proofs of **R29** in Appendix N.9.6 can thus be used to prove **R30**.

VITA

VITA

Jaemin Han received his B.S. and M.S. in Electrical and Electronic Engineering at 2007 and 2009, respectively, both in Yonsei University, South Korea. He is working towards the Ph.D. degree in Electrical and Computer Engineering from Purdue University, West Lafayette, Indiana, under the supervision of Professor Chih-Chun Wang since 2010. His research interest mainly covers algebraic, graph-theoretic, and information-theoretic characterizations of multi-session network coding and scheduling algorithms.