Characterizing positive-rate secure multicast network coding with eavesdropping nodes

Michael Langberg Michelle Effros

2024 IEEE International Symposium on Information Theory (ISIT) | 979-8-3503-8284-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/ISIT57864.2024.10619660

***Abstract*—Motivated by the study of multi-source multi-terminal key-dissemination, here called "key-cast," the work at hand presents a combinatorial characterization of when positive-rate secure multicast network coding in the presence of eavesdropping nodes is possible. In key-cast, introduced by the authors in [ITW2022], network nodes hold independent random bits, and one seeks a communication scheme that allows all terminal nodes to share a secret key K. We here address positive (albeit, arbitrarily small) rate key-cast under the security requirement that no single non-terminal network node can gain information about the shared key K; this scenario is useful in cryptographic settings. The work at hand studies key-dissemination protocols based on secure network coding and presents a combinatorial characterization of networks that support positive-rate multicast resilient to eavesdroppers that control individual network nodes. The secure-multicast capacity solution in the same setting is a known open problem.**

I. Introduction

The resource of shared secret randomness, i.e., a shared secret key, plays a fundamental role in the theory and practice of network communication systems; applications include cryptographic encryption, randomized coding technologies, distributed computing, statistical inference, distributed learning, distributed authentication, identification, local differential-privacy, and more (e.g., [1]–[24]). Motivated by the central role of shared randomness in such a wide range of distributed applications, the work at hand addresses the problem of disseminating common randomness over noiseless networks, i.e., in the context of Network Coding [25]–[29]; we call this the *key-cast* problem. In key-cast, first studied by the authors in [30], network nodes hold independent random bits, and one seeks a communication scheme that allows all terminal nodes to share a secret key K.

In this work, we focus on the cryptographically-motivated setting of key-cast in which one is only required to disseminate a *positive-rate* key, which, once shared among a collection of terminals, can be used to generate long sequences of common pseudo-random bits; the pseudo-random bits, in turn, can be used in applications like those mentioned above. Our interest lies in *secret* key dissemination under a natural secrecy condition in which the shared key K is independent of the information available at any non-terminal network node. As a result, in our setting, no network node, not even the nodes where random bits originate, other than the terminal nodes themselves that share the secret key K learns any information about K.

This work characterizes the combinatorial requirements that allow the design of a certain key-cast scheme based on the notion of *secure-multicast*. In secure-multicast one seeks to securely communicate source information to a collection of terminals in the presence of an eavesdropper with predefined eavesdropping capabilities. The model of secure multicast network coding includes source nodes, which have access to message information, and additional nodes that generate independent randomness used to enable secure communication. Most prior works on secure multicast, e.g., [31]–[37], consider a single source setting in which the source s generates both source messages and independent randomness, while no other network nodes can generate randomness. They further apply a *uniform* security assumption in which the eavesdropper can access any collection of at most z unit-capacity network links for a given security parameter z. A major result in this context includes a characterization of the secure multicast capacity and a demonstration that the capacity can be efficiently obtained using linear codes [31], [32], [34]–[36]. A more general model of secure-multicast, where several network nodes can generate messages and/or independent randomness and eavesdroppers have access to edge sets with varying capacities (e.g., the setting of eavesdropping on nodes) is studied in, e.g., [38]–[44]; in this general setting, the capacity is not fully characterized. In fact, determining its value is known, in certain cases, to be NP-hard [40] or as hard as determining the capacity of the k-unicast problem [42] (a well known open problem in the study of network codes, e.g., [39], [45], [46]).

Our study focuses on the design of positive-rate key-cast schemes that are resilient against non-uniform eavesdroppers that can access the information available at any single node. Our scheme builds on a corresponding positive-rate secure-multicast scheme in the setting in which any network node can generate randomness or messages and under the security requirement that no single internal network node can gain information about the transmitted message(s). Towards that end, in this work we ask and solve the following question (stated roughly below, and with greater rigor in Section II).

Question 1 (Positive-rate secure-multicast)**.** *Given a communication network G in which any network node can generate independent randomness, and given a set of terminal nodes D, is it possible to securely multicast a message m from a source s to nodes in D such that no non-terminal network node (except s) can gain information about m?*

M. Langberg is with the Department of Electrical Engineering at the University at Buffalo (State University of New York). Email: mikel@buffalo.edu

M. Effros is with the Department of Electrical Engineering at the California Institute of Technology. Email: effros@caltech.edu

This work is supported in part by NSF grant CCF-2245204.

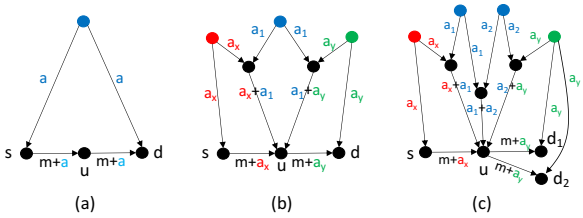

Fig. 1: A number of examples corresponding to Question 1, highlighting the major ideas used in our combinatorial characterization of networks that allow positive-rate secure multicast.

Figure 1 depicts a number of examples corresponding to Question 1. In the examples, any node can generate independent uniformly distributed random bits. In what follows, we review the examples in Figure 1, highlighting the major ideas used in our answer to Questions 1.

• **Secure multicast.** The networks depicted in Figure 1 allow the secure communication of message m from source s to terminal set D, where $D = \{d\}$ in Figure 1(a), Figure 1(b) and $D = \{d_1, d_2\}$ in Figure 1(c). In Figure 1(a), the vertex u is a **cut-vertex** that separates s and d. So, naively, one may conclude that u has the capabilities to gain information about any message transmitted between s and d. However, as noticed in prior works on secure network coding, e.g., [38]–[44], the information traversing the cut-vertex u can at times be *protected* using (a collection of) one-time pads. We refer to such vertices u as **protected cut-vertices**. Cut-vertices (and protected cut-vertices) play a major role in our analysis; see, Definitions III.2 and III.3. Indeed, in Figure 1(a) the blue node can generate a uniformly distributed bit a that is independent of m. As this node is connected to both s and d, a one time pad is established and u does not gain information about m, implying secure communication.

The padding protocols protecting the information traversing u may be more advanced than that of Figures1(a). Additional examples are given in Figures 1(b) and 1(c). The nodes colored in blue, red, and green, generate various uniformly distributed and independent bits $a_x, a_y, \{a_i\}$, and through certain connectivity requirements (related to the notion of *alternating paths*, see Definition III.4) allow the protection of source information m traversing u. We formally define these requirements and the corresponding **"padding" protocol** in Definitions III.3 and Protocol III.1, respectively. The combinatorial characterization of networks for which the answer to Question 1 is positive is given in Theorem III.1.

• **Secure key-cast.** The examples depicted in Figures 1(a)-1(c) also allow secure key-cast. Recall that for secure key-cast no network node (including the sources) gain any information on the shared key K. In the case of a single terminal d, this is trivial, since d can trivially generate its own key K. But even in the case of Figure 1(c) in which $D = \{d_1, d_2\}$ one can establish a shared key K by sending an additional uniformly distributed and independent bit m' from u to d_1 and d_2. This allows the terminals access to m and m', and accordingly to the key $K = m + m'$ which is independent of the information available at any non-terminal network node. We note that while it is not always the case that networks allowing secure multicast (when $|D| > 1$) also allow secure key-cast, not much is needed (with respect to the network topology) to convert a secure multicast scheme to a secure key-cast one. We elaborate on such extended schemes in Theorem IV.1.

The remainder of our presentation is structured as follows. In Section II, we present our detailed model and formalize Question 1. The combinatorial characterization of networks that allow positive-rate secure multicast (i.e., for which the answer to Question 1 is "yes") is given in Section III. Section IV designs positive-rate secure key-cast schemes using positive-rate secure-multicast. We conclude with a brief discussion on recent work on secure key-cast in Section V. Due to space limitations, some of the proofs are omitted and appear in the full version of this work [47].

II. Model

We follow the notation of [30], modified here to address the positive-rate setting. For any $\ell > 0$, $[\ell] \triangleq \{1, 2, \ldots, \lceil \ell \rceil\}$.

• **Key-cast Instance:** An instance $\mathcal{I} = (G, S, D, \mathcal{B})$ of the key-cast problem includes an acyclic directed network $G = (V, E)$, a collection of source nodes $S \subseteq V$, a collection of terminal nodes $D \subseteq V$, and a collection $\mathcal{B} = \{\beta_1, \ldots, \beta_{|\mathcal{B}|}\}$ of subsets of edges specifying the secrecy requirements. Each source node $s_i \in S$ holds an unlimited collection $M_i = \{b_{ij}\}_j$ of independent, uniformly distributed bits. Let $M = \cup_{s_i \in S} M_i$ denote all random bits available at the source nodes. Following a convention common in the study of acyclic network coding, we assume that the terminals $d \in D$ have no outgoing edges.

• **Key-Codes:** A network code $(\mathcal{F}, \mathcal{G}) = (\{f_e\}, \{g_i\})$, here called a *key-code*, is an assignment of an alphabet $\mathcal{X}_e$ and a (local) encoding function f_e for each edge $e \in E$ and a decoding function g_i for each terminal $d_i \in D$. For every edge $e = (u, v)$, the edge message $X_e \in \mathcal{X}_e$ from u to v equals the evaluation of encoding function f_e on inputs $X_{\mathtt{In}(u)}$; here, for a generic node u_0, In-edges(u_0) is the collection $(e : e = (v, u_0) \in E)$ of edges incoming to u_0, $X_{\mathtt{In}(u_0)} = ((X_e : e \in \text{In-edges}(u_0)), (\{b_{ij}\}_j : u_0 = s_i))$ captures all information available to node u_0 during the communication process, and, similarly, In-nodes(u_0) is the collection of nodes v such that $(v, u_0) \in E$. In order to ensure that $X_{\mathtt{In}(u)}$ is available to node u before it encodes, communication proceeds according to a predetermined topological order on E.

A key-code with target rate $R > 0$ is considered successful if, for every terminal $d_i \in D$, the evaluation of decoding functions g_i on the vector of random variables $X_{\mathtt{In}(d_i)}$ equals the reproduction of a uniform random variable K over alphabet $\mathcal{K} = [2^R]$ such that the following criteria are satisfied. First, key K meets secrecy constraints $\mathcal{B}$, which specifies that for every $\beta \in \mathcal{B}$, $I(K; (X_e : e \in \beta)) = 0$. Second, each terminal d_i decodes key K. Notice that the alphabets $\mathcal{X}_e$ chosen in code design may be set to be arbitrarily large. We thus refer to the setting at hand as "positive-rate" since the rate per time step resulting from choosing a large alphabet size $\mathcal{X}_e$ may be very small but still greater than zero.

Definition II.1 (Secure key-cast feasibility)**.** *Instance $\mathcal{I}$ is said to have positive key-cast rate $R_{\mathtt{key}} > 0$ if there exists a key-code $(\mathcal{F}, \mathcal{G})$ such that*

- **Key Rate:** *K is a uniform random variable over $[2^{R_{\mathtt{key}}}]$.*
- **Decoding:** *For all $d_i \in D$, $H(K|X_{\mathrm{In}(d_i)}) = 0$.*
- **Secrecy:** *$I(K; (X_e : e \in \beta)) = 0$ for any subset $\beta \in \mathcal{B}$.*

• **Secure-multicast:** In the secure-multicast setting, one distinguishes between source nodes S_m that hold message information and source nodes S_r that hold independent randomness used for masking. The two subsets may intersect. As before, we assume that every node s_i in $S_m \cup S_r$ holds an unlimited collection of independent bits $\{b_{ij}\}_j$.

Definition II.2 (Secure-multicast feasibility)**.** *Instance $\mathcal{I} = (G, (S_m, S_r), D, \mathcal{B})$ is said to have positive secure-multicast rate $R_{\mathtt{sec}} > 0$ if there exists a network code $(\mathcal{F}, \mathcal{G})$ such that*

- **Message Rate:** *K is a uniform random variable over $[2^{R_{\mathtt{sec}}}]$ such that $K = M' \subset \{b_{ij}\}_{i \in S_m, j}$, where M' is a subset of the source-bits generated by sources in S_m.*
- **Decoding:** *For all $d_i \in D$, $H(K|X_{\mathrm{In}(d_i)}) = 0$.*
- **Secrecy:** *$I(K; (X_e : e \in \beta)) = 0$ for any subset $\beta \in \mathcal{B}$.*

Notice that in both Definition II.1 and Definition II.2, the random variable K is shared between the terminals in D. In the key-cast setting (II.1), K denotes the secret key, which may be a (uniformly distributed) function of source bits; the source bits themselves are not necessarily decoded at terminals in D. In the secure multicast setting (II.2), K denotes the secret message generated at sources in S_m and decoded at each terminal in D. It is thus evident that the task of key-cast is more flexible than that of secure multicast: roughly speaking, instance $\mathcal{I}$ has positive key-cast rate $R_{\mathtt{key}} > 0$ according to Definition II.1 if $\mathcal{I}$ has positive secure-multicast rate $R_{\mathtt{sec}} > 0$ according to Definition II.2, but $R_{\mathtt{key}} > 0$ in Definition II.1 does not ensure $R_{\mathtt{sec}} > 0$ in Definition II.2 since $R_{\mathtt{key}} > 0$ does not ensure decodability of even a single bit from S_m.

The work at hand addresses instances in which each network node can generate uniformly distributed independent random bits, i.e., the setting that $S = V$ in Definition II.1 and $S_r = V$ in Definition II.2. Moreover, we consider eavesdroppers that have access to any individual network node (except terminal nodes). Namely, for $v \in V$, in Definition II.1 we consider $\mathcal{B} = \{\beta_v \mid v \in V \setminus D, \beta_v = \text{In-edges}(v)\}$.[1] Similarly, in Definition II.2 we require the message to be kept secret from any non-terminal node excluding message-generating sources.

In Section III, below, we seek to combinatorially characterize instances $\mathcal{I}$ with positive secure-multicast rate. We note that if there exists a secure-multicast scheme over $\mathcal{I} = (G, (S_m, S_r), D, \mathcal{B})$ communicating positive rate K with $|S_m| > 1$, then there exists a positive-rate *single message-source* secure-multicast scheme over $\mathcal{I} = (G, (\{s\}, S_r), D, \mathcal{B})$ for each $s \in S_m$ that generates message-bits in K. We can construct the latter code from the former by replacing all random message bits in K generated by nodes in $S_m \setminus \{s\}$ by constants. We thus, without loss of generality, consider instances in which $S_m = \{s\}$. We seek to answer the following question, which formalizes Question 1 from the Introduction.

Question 1 (Positive-rate secure multicast)**.** *For which instances $\mathcal{I} = (G, (\{s\}, V), D, \mathcal{B})$ with $\mathcal{B} = \{\beta_v \mid v \in V \setminus (D \cup \{s\}), \beta_v = \textit{In-edges}(v)\}$ can we achieve $R_{\mathtt{sec}} > 0$?*

III. Answering Question 1

In this section we consider secure multicast instances $\mathcal{I} = (G, (S_m = \{s\}, S_r = V), D, \mathcal{B})$ in which $\mathcal{B} = \{\beta_v \mid v \in V \setminus (D \cup \{s\}), \beta_v = \text{In-edges}(v)\}$. That is, in $\mathcal{I}$, we require the information multicast from s to D to be independent of the information available at any network node except the source s and terminals in D. We first consider the case where $|D| = 1$, i.e., $D = \{d\}$, and analyze secure communication from s to d. We then address general D. We start with a number of definitions followed by a subroutine to be used in our analysis. The definitions and subroutine are illustrated by Figure 2.

A. Preliminary notation and definitions ($D = \{d\}$)

Definition III.1. *A vertex $u \in V$ is called a* cut-vertex *for the source-terminal pair (s, d) if the removal of u separates s from d in G. Equivalently, all paths from s to d go through u.*

Let $D = \{d\}$. We define a partition of V that describes each node's connectivity to d and from s.

Definition III.2 (Partition (U_0, U_1, U_2))**.** *In the partition (U_0, U_1, U_2) of V, U_0 is the set of vertices $v \in V$ that are not reachable from s but from which d is reachable, U_1 is the set of vertices $v \in V$ that are reachable from s and from which d is reachable, and U_2 is the set of vertices $v \in V$ from which d is not reachable.*

We next define the notion of "U_0-protected cut-vertices." Roughly speaking, a cut-vertex $u \in U_1$ is U_0-protected if random variables generated at nodes in U_0 can be used to *mask* the message transmitted by s, thereby preventing u from learning anything about the message from s. Figure 2 depicts this definition.

Definition III.3 (U_0-protected vertices)**.** *Let (U_0, U_1, U_2) be the partition of Definition III.2. For any cut-vertex $u \in U_1$, consider the subsets W_u, X_u, and Y_u of U_0, where*

- *$v \in W_u$ if there exists a path $Q_w(v, u)$ from v to u in which all vertices except u are in U_0,*
- *$v \in X_u$ if there exist a path $Q_x(v, x)$ from v to some $x \in \textit{In-nodes}(u) \cap U_1$, and*
- *$v \in Y_u$ if there exists a path $Q_y(v, d)$ from v to d such that the first (in topological order) vertex y in $Q_y(v, d) \cap U_1$ has topological order greater than u;*

*sets W_u, X_u, and Y_u can intersect. Let $\{v_1, v_2, \ldots, v_\kappa\} = \textit{In-node}(u) \cap U_0$. For $i \in [\kappa]$, let $(W_{u,i}, X_{u,i}, Y_{u,i})$ be the vertices in (W_u, X_u, Y_u), respectively, that are connected by a path to v_i. Note that $X_{u,i}$ and $Y_{u,i}$ are included in $W_{u,i}$. Now, u is said to be U_0-***protected*** if $X_u \cap Y_u \neq \phi$, or if for*

[1] The security requirement expressed by $\mathcal{B}$ implies that $I(K; (X_e : e \in \text{In-edges}(v))) = 0$. Notice, by the definition of $X_{\mathrm{In}}(v)$, that this implies $I(K; X_{\mathrm{In}}(v)) = I(K; (X_e : e \in \text{In-edges}(v)), M_v) = 0$ as well for the independent bits M_v generated at v. That is, K is independent of all information available to v.

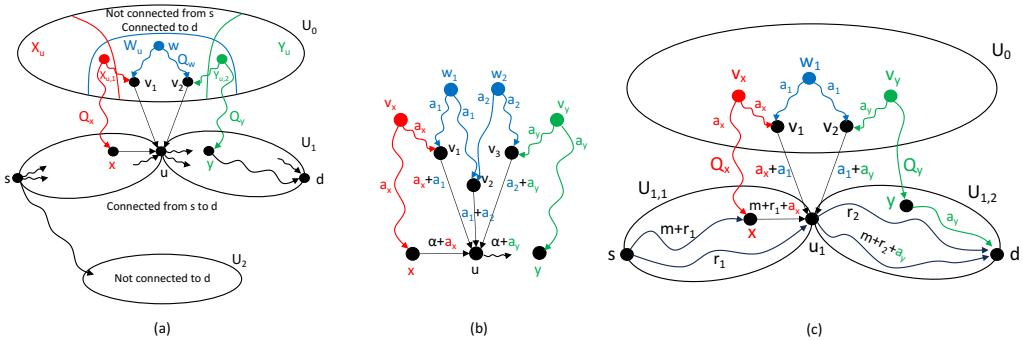

Fig. 2: Fig. 2(a) illustrates the sets (U_0, U_1, U_2) from Definition III.2 and the subsets W_u, X_u, Y_u for cut-vertex u with the refined subsets $X_{u,1}$ and $Y_{u,2}$ from Definition III.3. Vertex w is connected by an alternating path from s (see Definition III.4). Fig. 2(b) illustrates Protocol III.1 for u in the case where $\ell = 3$ and $\beta = \gamma = 0$. The red node v_x is in $X_{u,1} \cap W_{u,1}$, the blue node w_1 is in $W_{u,1} \cap W_{u,2}$, the blue node w_2 is in $W_{u,2} \cap W_{u,3}$, and the green node v_y is in $Y_{u,3} \cap W_{u,3}$. After receiving $\alpha + a_x$, node u does not learn anything about α and can output $\alpha + a_y$. Fig. 2(c) illustrates an example for the achievability of Theorem III.1 in which $c = 1$.

$\ell \geq 2$, there exists $i_1, i_2, \ldots, i_\ell \in [\kappa]$ such that $X_{u,i_1} \neq \phi$, $Y_{u,i_\ell} \neq \phi$, and $W_{u,i_j} \cap W_{u,i_{j+1}} \neq \phi$ for all $j \in [\ell - 1]$.

The above definitions are closely related to the notion of *alternating paths*. See Figure 2.

Definition III.4 (Alternating path)**.** *Given a directed graph G with vertex set V and edge set E, its undirected variant, denoted by $\bar{G}$, has vertex set V and undirected edge set $\bar{E} = \{(u, v) \mid (u, v) \in E\}$. Nodes v_1 and v_2 are connected by an alternating path in G, if v_1 and v_2 are connected by a path $\bar{P}_{v_1,v_2}$ in $\bar{G}$. The alternating path $P_{\mathtt{alt}}(v_1, v_2)$ in G connecting v_1 and v_2 consists of the set of edges $\{e \in E \mid \bar{e} \in \bar{P}\}$.*

Claim III.1. *If a cut-vertex u, with respect to (s, d), is U_0-protected then there exists an alternating path $P_{\mathtt{alt}}(s, d)$ connecting s and d that does not include u.*

We are now ready to state the main theorem for this section.

Theorem III.1. *Let $\mathcal{I} = (G, (S_m = \{s\}, S_r = V), D = \{d\}, \mathcal{B})$ with $\mathcal{B} = \{\beta_v \mid v \in V \setminus (D \cup \{s\}), \beta_v = \text{In-edges}(v)\}$. Then $\mathcal{I}$ has secure-multicast rate $R_{\mathtt{sec}} > 0$ according to Definition II.2 if and only if every cut-vertex u is U_0-protected.*

B. The "padding" protocol

Before addressing the proof of Theorem III.1 we preset our padding protocol (depicted in Figure 2), specifying how information can be transmitted from vertices in U_0 to a U_0-protected cut-vertex $u \in U_1$ (and additional vertices in U_1) to assist in masking the message information transmitted by the source node s. The secure-multicast scheme suggested shortly to answer Question 1 uses the padding protocol repeatedly.

Protocol III.1 (Padding protocol for U_0-protected u)**.** *Let u be a cut-vertex in U_1 that is U_0-protected according to Definition III.3. In this case, either $X_u \cap Y_u \neq \phi$ or, for some $\ell \geq 2$, there exists $\{i_1, i_2, \ldots, i_\ell\} \subset [\kappa]$ such that $X_{u,i_1} \neq \phi$, $Y_{u,i_\ell} \neq \phi$, and, for $j \in [\ell - 1]$, it holds that $W_{u,i_j} \cap W_{u,i_{j+1}} \neq \phi$. We describe the protocol for the latter, more general, case and defer the missing details to the full version [47] of this work.*

Fix $\ell \geq 2$ and $\{i_1, i_2, \ldots, i_\ell\} \subset [\kappa]$ such that $X_{u,i_1} \neq \phi$, $Y_{u,i_\ell} \neq \phi$, and, for $j \in [\ell - 1]$, $W_{u,i_j} \cap W_{u,i_{j+1}} \neq \phi$. Let $v_x \in X_{u,i_1} \subseteq X_u$. Then, there exists a path $Q_x(v_x, x)$ from v_x to some $x \in \text{In-nodes}(u) \cap U_1$. Let a_x be a uniformly distributed bit generated at v_x. For $j \in [\ell - 1]$, let a_j be a uniformly distributed bit generated at $w_j \in W_{u,i_j} \cap W_{u,i_{j+1}}$. Let $v_y \in Y_{u,i_\ell} \subseteq Y_u$. Then, there exists a path $Q_y(v_y, d)$ from v_y to d such that the first (by topological order) vertex y in $Q_y(v_y, d) \cap U_1$ has topological order greater than u. Let a_y be a uniformly distributed bit generated at v_y. Finally let $\alpha, \beta, \gamma \in \{0, 1\}$ be random variables such that the variables in the collection $\{\alpha, \beta, \gamma, a_x, a_1, \ldots, a_{\ell-1}, a_y\}$ are mutually independent. The protocol described below assumes that

- *node x has access to $\alpha + \beta + \gamma$ (additions are $\bmod 2$),*
- *node u has access to γ, and*
- *node u may or may-not have access to β,*

and guarantees that

- *node u is able to compute $\alpha + \beta + a_y$.*
- *node u does not gain any information about α.*

The protocol proceeds as follows.

- *Node v_x sends a_x to x through $Q_x(v_x, x)$ and to v_{i_1} through $Q_w(v_x, v_{i_1})$. Node x, with access to $\alpha + \beta + \gamma$ and a_x, sends $\alpha + \beta + \gamma + a_x$ to u.*
- *For $j \in [\ell - 1]$, node w_j sends a_j to v_{i_j} through $Q_w(w_j, v_{i_j})$ and to $v_{i_{j+1}}$ through $Q_w(w_j, v_{i_{j+1}})$.*
- *Node v_y sends a_y to y and then to d through $Q_y(v_y, d)$ and to v_{i_ℓ} through $Q_w(v_y, v_\ell)$.*
- *The incoming nodes of u in U_0 now compute the following functions to be forwarded to u. Node v_{i_1} computes and forwards $a_x + a_1$. For $j \in \{2, \ldots, \ell - 1\}$, node v_{i_j} computes and forwards $a_{j-1} + a_j$. Node v_{i_ℓ} computes and forwards $a_{\ell-1} + a_y$.*

The justification for the assumptions in the padding protocol, in addition to the precise information content of α, β, and

γ, are presented later when the padding protocol is used in our secure-multicast scheme given in Theorem III.1.

Under the given definitions, each of the nodes v_x,$w_1, \ldots, w_{\ell-1}$, and v_y is the source of a one-time pad (here denoted by $a_x, a_1, \ldots, a_{\ell-1}, a_y$) independent of all other random variables in the network. These one-time pads are sequentially added and then removed from α (representing the source message) to ensure that α remains protected. Protection at the bottleneck u is achieved by transmitting to the bottleneck not the protection bits themselves but sums of consecutive pairs of those bits. Namely,

Claim III.2. *After running Protocol III.1, it holds that (i) node u is able to compute $\alpha + \beta + a_y$, and (ii) node u does not gain any information about α.*

Proof: For (i), the proof follows from the fact that node u, knowing γ and using incoming information from $x, v_{i_1}, \ldots, v_{i_\ell}$ can compute $\gamma + (\alpha + \beta + \gamma + a_x) + (a_x + a_1) + (a_1 + a_2) + \cdots + (a_{\ell-2} + a_{\ell-1}) + (a_{\ell-1} + a_y) = \alpha + \beta + a_y$. For (ii), note that the incoming information to u is independent of α; namely, $I(\gamma, \alpha + \beta + \gamma + a_x, \beta, a_x + a_1, a_1 + a_2, \ldots, a_{\ell-2} + a_{\ell-1}, a_{\ell-1} + a_y; \alpha) = 0$. ∎

C. Proof of Theorem III.1

Proof: We start by proving achievability. The proof is depicted in Figure 2(c). If there are no cut vertices in V, then s and t are 2-vertex connected, meaning that there exist two vertex-disjoint paths, $P_1(s,d)$ and $P_2(s,d)$, in G between s and d [48]. Then $\mathbf{R}_{\mathtt{SR}} > 0$ since, given independent uniformly distributed bits m and r, the source can send $m + r$ on P_1 and r on P_2. The resulting scheme is secure.

Otherwise, let $u_1, \ldots, u_c$ be the collection of cut-vertices ordered topologically. We here present a proof sketch for the case $c = 1$, i.e., when there is a single cut-vertex u_1 separating s and d. As u_1 is U_0-protected, there exist a path $Q_x(v_x, x)$ for $v_x \in X_{u_1}$ and $x \in \text{In-nodes}(u_1) \cap U_1$, and a path $Q_y(v_y, d)$ through y for $v_y \in Y_{u_1}$ and $y \in U_1$ of topological order greater than that of u_1; here X_{u_1} and Y_{u_1} are the subsets of U_0 corresponding to u_1 in Definition III.3. By Protocol III.1 and Claim III.2, assuming that u_1 receives information $\alpha+\beta+\gamma+a_x$ and γ for a_x generated at v_x, vertex u_1 can compute $\alpha + \beta + a_y$ in a way that keeps all incoming information to u_1 collectively independent of α.

Let $U_{1,1}, U_{1,2}$ be a partition of U_1 implied by u_1 in which $U_{1,1}$ includes all vertices of U_1 of topological order at most that of u_1. From $c = 1$, it follows that either s and u_1 are connected by two vertex-disjoint paths $P_1(s, u_1)$ and $P_2(s, u_1)$ or that $(s, u_1) \in E$. Similarly for $U_{1,2}$, u_1 and d are either connected by two vertex-disjoint paths or $(u_1, d) \in E$. We here assume the former, more general, case for both pairs (s, u_1) and (u_1, d). We also assume that $P_1(s, u_1)$ passes through vertex x defined above. Full proof, for $c > 1$ without the above assumptions, appears in [47].

We are now ready to suggest a secure communication scheme in which the source s securely sends a uniform bit m to d. Source s sends $m + r_1$ on path $P_1(s, u_1)$ until vertex x, and send r_1 on path $P_2(s, u_1)$, where r_1 is an independent uniformly distributed bit. Let $\alpha = m$, let $\beta \equiv 0$ be constant, and let $\gamma = r_1$. It holds that x has access to $\alpha + \beta + \gamma = m + r_1$ and u_1 has access to $\gamma = r_1$. Applying Protocol III.1 on u_1 now guarantees by Claim III.2 that u_1 can compute $\alpha + \beta + a_y = m + a_y$ without gaining information about $\alpha = m$; in addition, a_y is forwarded on $Q_y(v_y, d)$ to d. Notice that all other nodes in $U_{1,1}$ do not gain information about $\alpha = m$ either. Vertex u_1 prepares to send $m + a_y$ in the next step of communication.

In $U_{1,2}$, u_1 sends $m + r_2 + a_y$ for a uniform and independent bit r_2 on one of the two vertex disjoint paths connecting u_1 and d, and sends r_2 on the other. It follows that each node in $U_{1,2} \setminus \{d\}$ gains no information about m (even under the assumption that it knows a_y). As d has access to a_y it can decode m. This concludes the achievability proof. We omit the converse proof due to space limitations. ∎

D. Multiple terminals

When D includes more than a single terminal node, we can perform the scheme described in Theorem III.1 for each terminal node d_i in parallel with independent randomness; this yields a legitimate code since our positive-rate model allows arbitrary edge alphabets. We conclude the following corollary.

Corollary III.1. *Let $\mathcal{I} = (G, (S_m = \{s\}, S_r = V), D, \mathcal{B})$ with $\mathcal{B} = \{\beta_v \mid v \in V \setminus (D \cup \{s\}), \beta_v = \text{In-edges}(v)\}$. Then $\mathcal{I}$ has secure-multicast rate $R_{\mathtt{sec}} > 0$ if and only if, for every $d \in D$, every cut-vertex u with respect to (s, d) is U_0-protected.*

IV. Secure key-cast through secure multicast

We here present a sufficient condition for secure key-cast in the setting in which we require the key K delivered to the terminals in D to be independent of the information available at *any* network node (except the terminals in D themselves). The ability to achieve security at all nodes, including source nodes that generate information, is a unique property of key-cast that distinguishes key-cast from secure multicast.

Theorem IV.1. *Let $\mathcal{I} = (G, V, D, \mathcal{B})$ in which $\mathcal{B} = \{\beta_v \mid v \in V \setminus D, \beta_v = \text{In-edges}(v)\}$. Then $\mathcal{I}$ has key-cast rate $R_{\mathtt{key}} > 0$ if (i) there exists a node s for which for every $d \in D$ every cut-vertex u separating s and d is U_0-protected, and, in addition, (ii) there exists a node s' such that for all $d \in D$ there is a path from s' to d that does not pass through s.*

V. Conclusions and open problems

In this work, we characterize positive-rate secure-multicast instances under the notion of node-security. We show that instances $\mathcal{I}$ with positive secure-multicast rate $R_{\mathtt{sec}} > 0$ imply (under additional connectivity conditions) positive key-cast rate $R_{\mathtt{key}} > 0$ under the security requirement that information available at *any* single network node (excluding the terminals in D, but including the source nodes) is independent of the shared key K. The opposite assertion, that $R_{\mathtt{key}} > 0$ implies $R_{\mathtt{sec}} > 0$, does not hold. A combinatorial characterization of instances $\mathcal{I}$ for which $R_{\mathtt{key}} > 0$ was left open in this work and has been recently resolved in the extended version [47] (the latter, finalized after conference submission).

REFERENCES

[1] Claude E. Shannon. Communication theory of secrecy systems. *The Bell system technical journal*, 28(4):656–715, 1949.

[2] Ueli M. Maurer. Secret key agreement by public discussion from common information. *IEEE Transactions on Information Theory*, 39(3):733–742, 1993.

[3] Rudolf Ahlswede and Imre Csiszár. Common randomness in information theory and cryptography. I. Secret sharing. *IEEE Transactions on Information Theory*, 39(4):1121–1132, 1993.

[4] Rudolf Ahlswede and Imre Csiszár. Common randomness in information theory and cryptography. II. CR capacity. *IEEE Transactions on Information Theory*, 44(1):225–240, 1998.

[5] Rudolf Ahlswede. Elimination of correlation in random codes for arbitrarily varying channels. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 44(2), 1978.

[6] Imre Csiszár and Prakash Narayan. The capacity of the arbitrarily varying channel revisited: Positivity, constraints. *IEEE Transactions on Information Theory*, 34(2):181–193, 1988.

[7] Imre Csiszár and Prakash Narayan. Capacity of the Gaussian arbitrarily varying channel. *IEEE Transactions on Information Theory*, 37(1):18–26, 1991.

[8] Amos Lapidoth and Prakash Narayan. Reliable communication under channel uncertainty. *IEEE Transactions on Information Theory*, 44(6):2148–2177, 1998.

[9] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge Univ. Press, 2006.

[10] Jakub Konečnỳ, Brendan H. McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[11] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. *Advances in Neural Information Processing Systems*, 30, 2017.

[12] Vaikkunth Mugunthan, Antigoni Polychroniadou, David Byrd, and Tucker Hybinette Balch. SMPAI: Secure multi-party computation for federated learning. In *NeurIPS Workshop on Robust AI in Financial Services*, 2019.

[13] Jayadev Acharya, Clément L. Canonne, and Himanshu Tyagi. Distributed signal detection under communication constraints. In *Conference on Learning Theory*, pages 41–63. PMLR, 2020.

[14] Nir Shlezinger, Mingzhe Chen, Yonina C. Eldar, Vincent H. Poor, and Shuguang Cui. Federated learning with quantization constraints. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8851–8855, 2020.

[15] Botond Szabó, Lasse Vuursteen, and Harry van Zanten. Optimal high-dimensional and nonparametric distributed testing under communication constraints. *arXiv preprint arXiv:2202.00968*, 2022.

[16] Mingzhe Chen, Nir Shlezinger, Vincent H. Poor, Yonina C. Eldar, and Shuguang Cui. Communication-efficient federated learning. *Proceedings of the National Academy of Sciences*, 118(17), 2021.

[17] Jayadev Acharya, Clément L. Canonne, Ziteng Sun, and Himanshu Tyagi. The role of interactivity in structured estimation. *arXiv preprint arXiv:2203.06870*, 2022.

[18] Daihyun Lim, Jae W. Lee, Blaise Gassend, Edward G. Suh, Marten Van Dijk, and Srinivas Devadas. Extracting secret keys from integrated circuits. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 13(10):1200–1205, 2005.

[19] Ying Su, Jeremy Holleman, and Brian P. Otis. A digital 1.6 pj/bit chip identification circuit using process variations. *IEEE Journal of Solid-State Circuits*, 43(1):69–77, 2008.

[20] G. Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *2007 44th ACM/IEEE Design Automation Conference*, pages 9–14, 2007.

[21] Haile Yu, Philip Heng Wai Leong, Heiko Hinkelmann, Leandro Moller, Manfred Glesner, and Peter Zipf. Towards a unique FPGA-based identification circuit using process variations. In *International Conference on Field Programmable Logic and Applications*, pages 397–402, 2009.

[22] Jayadev Acharya and Ziteng Sun. Communication complexity in locally private distribution estimation and heavy hitters. In *International Conference on Machine Learning*, pages 51–60. PMLR, 2019.

[23] David Byrd and Antigoni Polychroniadou. Differentially private secure multi-party computation for federated learning in financial applications. In *First ACM International Conference on AI in Finance*, pages 1–9, 2020.

[24] Rudolf Ahlswede, Alexander Ahlswede, Ingo Althöfer, Christian Deppe, and Ulrich Tamm. *Identification and Other Probabilistic Models*. Springer, 2021.

[25] Rudolf Ahlswede, Ning Cai, S-YR Li, and Raymond W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, 2000.

[26] S-YR Li, Raymond W. Yeung, and Ning Cai. Linear network coding. *IEEE Transactions on Information Theory*, 49(2):371–381, 2003.

[27] Ralf Koetter and Muriel Médard. An algebraic approach to network coding. *IEEE/ACM Transactions on Networking (TON)*, 11(5):782–795, 2003.

[28] Sidharth Jaggi, Peter Sanders, Philip A. Chou, Michelle Effros, Sebastian Egner, Kamal Jain, and Ludo M. G. M. Tolhuizen. Polynomial time algorithms for multicast network code construction. *IEEE Transactions on Information Theory*, 51(6):1973–1982, 2005.

[29] Tracey Ho, Muriel Médard, Ralf Koetter, David R. Karger, Michelle Effros, Jun Shi, and Ben Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52(10):4413–4430, 2006.

[30] Michael Langberg and Michelle Effros. Network coding multicast key-capacity. In *IEEE Information Theory Workshop (ITW)*, pages 422–427, 2022.

[31] Ning Cai and Raymond W. Yeung. Secure network coding. In *IEEE International Symposium on Information Theory*, page 323, 2002.

[32] Jon Feldman, Tal Malkin, C. Stein, and Rocco A. Servedio. On the capacity of secure network coding. In *42nd Annual Allerton Conference on Communication, Control, and Computing*, pages 63–68, 2004.

[33] Ning Cai and Raymond W. Yeung. A security condition for multi-source linear network coding. *IEEE International Symposium on Information Theory*, pages 561–565, 2007.

[34] Ning Cai and Raymond W. Yeung. On the optimality of a construction of secure network codes. *IEEE International Symposium on Information Theory*, pages 166–170, 2008.

[35] Salim El Rouayheb, Emina Soljanin, and Alex Sprintson. Secure network coding for wiretap networks of type II. *IEEE Transactions on Information Theory*, 58(3):1361–1371, 2012.

[36] Danilo Silva and Frank R. Kschischang. Universal secure network coding via rank-metric codes. *IEEE Transactions on Information Theory*, 57(2):1124–1135, 2011.

[37] Sidharth Jaggi and Michael Langberg. Secure network coding: Bounds and algorithms for secret and reliable communications. In *Chapter 7 of Network Coding: Fundamentals and applications (Muriel Médard and Alex Sprintson ed.)*, pages 183–215. Academic Press, 2012.

[38] Terence H. Chan and Alex Grant. Capacity bounds for secure network coding. *Australian Communications Theory Workshop*, pages 95–100, 2008.

[39] Terence H. Chan and Alex Grant. Network coding capacity regions via entropy functions. *IEEE Transactions on Information Theory*, 60(9):5347–5374, 2014.

[40] Tao Cui, Tracy Ho, and Joerg Kliewer. On secure network coding with nonuniform or restricted wiretap sets. *IEEE Transactions on Information Theory*, 59(1):166–176, 2012.

[41] Debaditya Chaudhuri. *Characterization of Rate Regions in Secure Network Coding over General Wiretap Networks*. PhD thesis, University at Buffalo, State University of New York, 2021.

[42] Wentao Huang, Tracey Ho, Michael Langberg, and Joerg Kliewer. Single-unicast secure network coding and network error correction are as hard as multiple-unicast network coding. *IEEE Transactions on Information Theory*, 64(6):4496–4512, 2018.

[43] Debaditya Chaudhuri and Michael Langberg. Trade-offs between rate and security in linear multicast network coding. In *IEEE International Symposium on Information Theory (ISIT)*, pages 846–850, 2018.

[44] Debaditya Chaudhuri, Michael Langberg, and Michelle Effros. Secure network coding in the setting in which a non-source node may generate random keys. In *IEEE International Symposium on Information Theory (ISIT)*, pages 2309–2313, 2019.

[45] Raymond W. Yeung and Ning Cai. *Network coding theory*. Now Publishers Inc, 2006.

[46] Michael Langberg and Muriel Médard. On the multiple unicast network coding conjecture. In *47th Annual Allerton Conference on Communication, Control, and Computing*, pages 222–227, 2009.

[47] Michael Langberg and Michelle Effros. Characterizing positive-rate key-cast (and multicast network-coding) with eavesdropping nodes. *Manuscript, availiable on* arXiv.org, 2024.

[48] Karl Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.