On Extremal Rates of Secure Storage Over Graphs

Zhou Li, *Graduate Student Member, IEEE*, and Hua Sun, *Member, IEEE*

***Abstract*— A secure storage code maps K source symbols, each of L_w bits, to N coded symbols, each of L_v bits, such that each coded symbol is stored in a node of a graph (one may view a node as a server). Each edge of the graph is either associated with D of the K source symbols such that from the pair of nodes connected by the edge, we can decode the D source symbols and learn no information about the remaining $K-D$ source symbols; or the edge is associated with no source symbols such that from the pair of nodes connected by the edge, nothing about the K source symbols is revealed. The ratio L_w/L_v is called the symbol rate of a secure storage code and the highest possible symbol rate is called the capacity. We characterize all graphs over which the capacity of a secure storage code is equal to 1, when $D = 1$. This result is generalized to $D > 1$, i.e., we characterize all graphs over which the capacity of a secure storage code is equal to $1/D$ under a mild condition that for any node, the source symbols associated with each of its connected edges do not include a common element. Further, we characterize all graphs over which the capacity of a secure storage code is equal to $2/D$.**

***Index Terms*— Capacity, extremal rate, secure storage codes.**

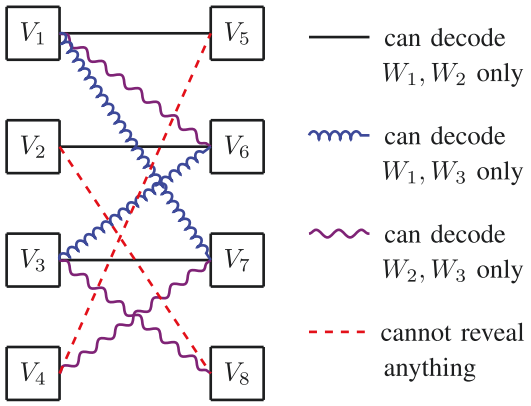


Fig. 1. An example graph of a secure storage problem with $K = 3$ source symbols and $N = 8$ coded symbols, whose capacity turns out to be $1/2$ (refer to Theorem 2. See Fig. 6 for a code construction). We may interpret this instance as storing 3 files W_1, W_2, W_3 over 8 servers $V_1, \cdots, V_8$ such that from some pair of servers, we may securely retrieve some files.

I. Introduction

MODERN datasets are usually massive and stored in a distributed manner. Providing flexible accessibility and security control over a variety of network topologies with limited storage budget is a challenging task. Motivated by such secure storage tasks, in this work we model a distributed storage system and its data access structure using a graph and aim to find storage efficient codes that satisfy the accessibility and security constraints specified by the graph.

A secure storage code is a mapping from K source symbols (e.g., files), $W_1, \cdots, W_K$, each of L_w bits, to N coded symbols, $V_1, \cdots, V_N$, each of L_v bits. Each coded symbol is stored in a node of a graph G (e.g., a server), so the node set of the graph is $\mathcal{V} = \{V_1, \cdots, V_N\}$. Note that we use V_n to denote both a coded symbol and a node as they have a one-to-one mapping. The data accessibility and security constraints are given through the edges of the graph. An edge $\{V_i, V_j\}$ of the graph G is associated either with D of the K source symbols or no source symbols. In the former case, the requirement is that from (V_i, V_j), we can decode the D source symbols and learn nothing about the remaining $K - D$ source symbols; in the latter case, (V_i, V_j) must be independent of the K source symbols such that no information is leaked. An example of a secure storage problem over a graph is given in Fig. 1. We can now see how a graph representation is used to capture the various data access patterns and security requirements of storing K files over N servers. The storage efficiency of a secure storage code is measured by its symbol rate, defined as L_w/L_v, i.e., out of the (L_v) bits of each coded symbol, how many bits (L_w) of each source symbol can be securely stored. Our objective is to characterize for a given graph G, the highest possible symbol rate, termed the capacity $C = \sup L_w/L_v$, of a secure storage code.

While this work is presented in a storage system context (i.e., how to securely store files over a graph based distributed storage system), the problem of secure storage has intimate relations to a few communication network contexts. First, when the graph G is bipartite (e.g., Fig. 1), the secure storage problem can be viewed a generalization of the conditional disclosure of secrets (CDS) problem [1], [2], [3], [4], [5]. To see this, we view the nodes on one side (e.g., V_1, V_2, V_3, V_4 in Fig. 1) as the transmit signal sent by Alice and view the nodes on the other side (e.g., V_5, V_6, V_7, V_8 in Fig. 1) as the transmit signal sent by Bob. If and only if the signal indices (node indices) satisfy some function (i.e., the type of the edge corresponds to some source symbols), Carol who receives both signals can recover the corresponding secrets. Compared to the classic CDS problem where there is only one secret (source symbol) to disclose, the secure storage problem generalizes to include multiple secrets [5]; further, an arbitrary subset of all secrets can be conditionally disclosed. As a result, our secure storage problem can be applied to conditional

Manuscript received 14 October 2022; revised 13 May 2023 and 5 July 2023; accepted 8 July 2023. Date of publication 26 July 2023; date of current version 8 August 2023. This work was supported in part by NSF under Grant CCF-2007108 and Grant CCF-2045656. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stefano Tomasin. *(Corresponding author: Zhou Li.)*

The authors are with the Department of Electrical Engineering, University of North Texas, Denton, TX 76205 USA (e-mail: zhouli@my.unt.edu; hua.sun@unt.edu).

Digital Object Identifier 10.1109/TIFS.2023.3299183

disclosure of multiple secrets, e.g., there are multiple business plans to share from Alice and Bob to Carol and if and only if certain condition is satisfied for Alice and Bob (modelled through the graph), correspoding business plans are revealed to Carol. As CDS is closely related to secret sharing [6], secure storage carries over such connection, especially to secret sharing where access structure is modelled through graphs [7]. Note that extensions of secret sharing to multiple users are present in the literature [8], [9], [10], but differences exist in the message setting (e.g., how many secrets are associated with each user) and performance metrics (e.g., symmetric rate versus sum rate) when compared to our work. Second, the secure storage problem can be interpreted as a secure network coding problem [11], [12], [13] over a class of combination networks. While previous work mainly considers multicast or special message structures (e.g., nested) over combination networks (often with no security) [14], [15], [16], [17], our focus in this work is on the interplay between data access and security pattern modelled by graphs other than the network topology graph. As a consequence, our results on the secure storage problem correspond to the solution of a class of secure network coding problem, which can be interpreted as a class of communication network problem with security constraints.

Main Result and Technique: Characterizing the capacity of secure storage codes appears to be a formidable task, mainly due to the fact that the constraint graph G can be arbitrary. Different classes of graphs (or hypergraphs) G can be used to model various well-known network information theory problems, such as index coding [18] and coded caching [19], as noticed in [20], which considers a similar graph storage problem with no security constraints (but the coded symbols may have different sizes). As a result, allowing arbitrary G will involve well-established hard capacity questions.

Instead of fixing a graph G first and then pursuing the capacity, the perspective we take in this work is to focus on *extremal* rate values and the associated *extremal* graphs whose secure storage capacity values are extremal. A natural starting point is the setting of $D = 1$ and $C = 1$, where it is easily seen that the capacity of secure storage code cannot exceed 1, i.e., the size of each coded symbol must be at least the size of each source symbol, as long as there exist security constraints. Our first main result is a full characterization of all such extremal graphs whose capacity is $C = 1$ (refer to Theorem 1), i.e., if a graph belongs to this class, we construct a secure storage code that achieves the highest possible symbol rate of 1 and otherwise if a graph does not belong to this class, the symbol rate of any secure storage code must be strictly smaller than 1. The key to this extremal rate characterization result is an alignment view of the space of the source symbols, the noise symbols (required to ensure information theoretic security), and the coded symbols. Such an alignment view is first introduced in [3], where all extremal graphs with $C = 1$ are found with $K = 1$ source symbol. This work generalizes this result to an arbitrary number of source symbols, i.e., from $K = 1$ to any K. While only noise alignment and signal (coded symbol) alignment are needed in [3] as there is only $K = 1$ source symbol, here we further need interference alignment to take care of other undesired source symbols as $K > 1$. Interestingly, a decomposition based approach turns out to be effective, i.e., we first separately design a secure storage code for each source symbol and then combine each separate code to produce a joint code that works for all source symbols.

Our second main result is a generalization of Theorem 1 from $D = 1$ to any $D > 1$, but under an additional condition to ensure that each coded symbol must be fully covered by noise symbols (then $C \leq 1/D$, equivalently, $L_v \geq D \times L_w$). Under such a condition, we characterize all extremal graphs whose capacity is $C = 1/D$ (refer to Theorem 2). Compared to Theorem 1 where each edge may recover $D = 1$ source symbol, Theorem 2 considers the case where each edge may recover $D > 1$ source symbols and this introduces some technical difficulty. While the same decomposition based approach continues to apply, ensuring the simultaneous recovery of multiple source symbols is more involved. As a consequence of such difficulty, the code construction in Theorem 1 is explicit while in Theorem 2 we are only able to provide an existence proof that relies on randomized code constructions over higher dimensions. So the code construction and associated (achievability and converse) proofs go much beyond those in [3].

Finally, noting that there exist graphs whose secure storage code rates are strictly larger than $C = 1/D$, we study the extremal rate of $2/D$, which is the highest possible symbol rate among all graphs. This graph class turns out to be fairly straightforward and is stated mainly for completeness. Here any pair of nodes connected by an edge have a total storage size of $2L_v = D \times L_w$, i.e., all storage space is occupied by the desired D source symbols. The extremal rate of $2/D$ places very strict constraints on the graph G. Our third main result is a full characterization of all extremal graphs G whose capacity is $C = 2/D$ (refer to Theorem 3). Notably, linear coding (storing linear combinations of different source symbols) is necessary to achieve the capacity of $2/D$, i.e., storing the source symbols directly is not sufficient.

II. Problem Statement and Definitions

Consider K independent source symbols $W_1, \cdots, W_K$, each of L_w bits.

$$H(W_1, \cdots, W_K) = H(W_1) + \cdots + H(W_K),$$
$$L_w = H(W_1) = \cdots = H(W_K). \quad (1)$$

Consider N coded symbols $V_1, \cdots, V_N$, each of L_v bits. Note that L_w, L_v are not necessarily integer values. For example, if W_k are uniformly random $\mathbb{F}_3$ symbols, then $L_w = \log_2 3$ bits. Furthermore, since we are interested in their relative size (see (4)), L_w, L_v are allowed to take arbitrarily large values.

The constraints on the coded symbols are specified by a graph $G = (\mathcal{V}, \mathcal{E})$, where the node[1] set $\mathcal{V} = \{V_1, \cdots, V_N\}$ and the edge set $\mathcal{E}$ is a set of unordered pairs from $\mathcal{V}$. Each edge $\{V_i, V_j\} \in \mathcal{E}$ is associated with a subset $\mathcal{D}$ of

[1]Note that we abuse the notation by using V_n to denote both a coded symbol and a node of the graph, for the sake of simplicity. The context will make its meaning clear.

$\{1, 2, \cdots, K\} \triangleq [K]$, which either has D elements or is an empty set, i.e., $|\mathcal{D}| = D$ or $\mathcal{D} = \emptyset$. The edge association is described by a function $t : t(\{V_i, V_j\}) = \mathcal{D}$. For each edge $\{V_i, V_j\}$, it is required that from (V_i, V_j), we can decode and only decode the messages $(W_k)_{k\in\mathcal{D}}$. That is, $\forall\{V_i, V_j\} \in \mathcal{E}$ such that $t(\{V_i, V_j\}) = \mathcal{D}$, we have

$$\text{(Correctness)} \quad H\left((W_k)_{k\in\mathcal{D}} \mid V_i, V_j\right) = 0, \tag{2}$$

$$\text{(Security)} \quad I\left(V_i, V_j; (W_k)_{k\in[K]\backslash\mathcal{D}} \mid (W_k)_{k\in\mathcal{D}}\right) = 0 \tag{3}$$

where for two sets $\mathcal{A}, \mathcal{B}$, $\mathcal{A}\backslash\mathcal{B}$ denotes the set of elements that belong to $\mathcal{A}$ but not to $\mathcal{B}$. To understand the security constraint, we may interpret the threat model as the existence of an external eavesdropper, who may observe any edge of the graph but cannot obtain any additional information. An isolated node V, i.e., a node connected to no edges, is trivial as it has no constraint. Without loss of generality, we assume that any graph G considered in this work contains no isolated nodes.

A mapping from the source symbols $W_1, \cdots, W_K$ to the coded symbols $V_1, \cdots, V_N$ that satisfies the correctness and security constraints (2), (3) specified by a graph $G = (\mathcal{V}, \mathcal{E})$ is called a secure storage code. The (achievable) symbol rate of a secure storage code is defined as

$$R \triangleq \frac{L_w}{L_v} \tag{4}$$

and the supremum of symbol rates is called the capacity, C. Note that supremum includes limits, so $R = \lim_{L_w\to\infty} L_w/L_v$ is also (asymptotically) achievable.

A. Graph Definitions

To facilitate the presentation of our results, we introduce some graph definitions in this section.

For a graph $G = (\mathcal{V}, \mathcal{E})$, we wish to separately consider each source symbol W_k and see if each edge is associated with W_k (i.e., can recover W_k). This leads us to the definition of $G^{[k]}$.

Definition 1 (Characteristic Graph $G^{[k]}$ of W_k): For a graph $G = (\mathcal{V}, \mathcal{E})$, define $\forall k \in [K]$

$$\begin{aligned} G^{[k]} &= (\mathcal{V}^{[k]}, \mathcal{E}^{[k]}) \text{ such that } \mathcal{V}^{[k]} \\ &= \{V_1^{[k]}, \cdots, V_N^{[k]}\}, \\ \{V_i^{[k]}, V_j^{[k]}\} &\in \mathcal{E}^{[k]} \text{ if and only if } \{V_i, V_j\} \in \mathcal{E}, \\ t^{[k]}(\{V_i^{[k]}, V_j^{[k]}\}) &= \begin{cases} \{k\} & \text{if } k \in t(\{V_i, V_j\}) \\ \emptyset & \text{else if } k \notin t(\{V_i, V_j\}). \end{cases} \end{aligned} \tag{5}$$

Fig. 2 shows an example of G and its $G^{[1]}$ of W_1.

For a node V of a graph $G = (\mathcal{V}, \mathcal{E})$, the common elements of the source symbols associated with each of its connected edges are relevant in stating our results, then we make them explicit in the following definition.

Definition 2 (Common Sources $\mathcal{C}(V)$): Consider a node $V \in \mathcal{V}$ of a graph $G = (\mathcal{V}, \mathcal{E})$, define

$$\mathcal{C}(V) = \bigcap_{i:\{V,V_i\}\in\mathcal{E}} t(\{V, V_i\}). \tag{6}$$

For example, consider node V_1 in Fig. 2, $\mathcal{C}(V_1) = \{1, 2\} \cap \{2, 3\} \cap \{1, 3\} = \emptyset$.

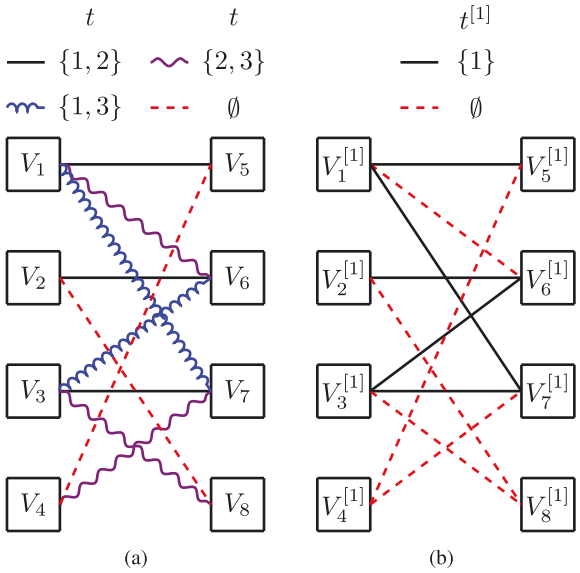


Fig. 2. (a) An example graph G with $K = 3$ source symbols, $N = 8$ coded symbols, and $D = 2$ (each edge may be associated with 2 source symbols) and (b) its characteristic graph $G^{[1]}$ of W_1.

For an edge of a graph $G = (\mathcal{V}, \mathcal{E})$, it is important if the edge is associated with some source symbol or no source symbol. Depending on this, an edge is called either qualified or unqualified and we have similar definitions for paths and components.

Definition 3 (Qualified/Unqualified Edge/Path/Component): Consider a graph $G = (\mathcal{V}, \mathcal{E})$. An edge $E \in \mathcal{E}$ is called qualified if $t(E) \neq \emptyset$ and unqualified if $t(E) = \emptyset$. A sequence of connecting qualified/unqualified edges is called a qualified/unqualified path. A qualified edge that connects two nodes in an unqualified path is said to be internal. A qualified/unqualified component is a maximal induced subgraph of G wherein any two nodes are connected by a qualified/unqualified path.

Note that the above definition applies to both G and $G^{[k]}$. For example, in Fig. 2, $\{V_1, V_5\}$ is a qualified edge, $\{V_1^{[1]}, V_6^{[1]}\}$ is an unqualified edge, $(\{V_5^{[1]}, V_4^{[1]}\}, \{V_4^{[1]}, V_7^{[1]}\})$ is an unqualified path, G is a qualified component, and $G^{[1]}$ contains no internal qualified edges.

Finally, a node V of a graph $G = (\mathcal{V}, \mathcal{E})$ whose all connected edges are associated with the same set of source symbols, is degenerate (because all constraints of V can be satisfied by storing the same set of source symbols in V). It is convenient to remove all degenerate nodes when the results are presented (note that this is only to simplify the presentation and our results and proofs hold with degenerate nodes) and we have the following definition.

Definition 4 (Non-degenerate Subgraph $\widetilde{G}$ of G): For a graph $G = (\mathcal{V}, \mathcal{E})$, denote the set of degenerate nodes by $\mathcal{V}_d$, i.e.,

$$\mathcal{V}_d \triangleq \bigcup \left\{V \in \mathcal{V} \mid t(\{V, V_i\}) = \mathcal{C}(V), \forall\{V, V_i\} \in \mathcal{E}\right\}. \tag{7}$$

The subgraph of G induced by the non-degenerate nodes $\mathcal{V}\backslash\mathcal{V}_d$ is defined as $\widetilde{G}$, i.e., $\widetilde{G} \triangleq G[\mathcal{V}\backslash\mathcal{V}_d]$.

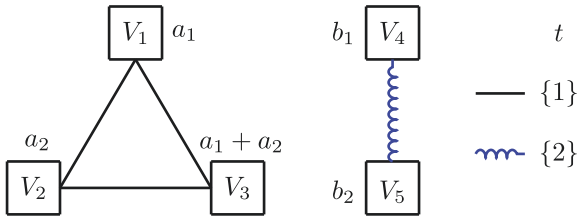


Fig. 3. An example graph G with $K = 2, N = 5, D = 1$ and $\widetilde{G} = \emptyset$. $W_1 = (a_1, a_2)$, $W_2 = (b_1, b_2)$ and $C = 2$.

III. Results

In this section, we present our main results along with illustrative examples and observations.

A. $D = 1$ and Extremal Graphs With $C = 1$

We start with the setting of $D = 1$. All extremal graphs whose secure storage capacity is $C = 1$ are characterized in the following theorem.

Theorem 1: The capacity of a secure storage code over a graph G with $D = 1$ is $C = 1$ if and only if the non-degenerate subgraph $\widetilde{G}$ of G is not empty and for every qualified component Q of $\widetilde{G}$, the characteristic graph $Q^{[k]}$ of each coded symbol $W_k, k \in [K]$ contains no internal qualified edge.

Remark 1: When the non-degenerate subgraph $\widetilde{G}$ of G is empty, each node of G is connected to edges that are associated with the same set of source symbols. If there exists a qualified edge in G, the secure storage capacity is 2 (this trivial case will be covered in Theorem 3, see Fig. 3 for an example).

The proof of Theorem 1 is presented in Section IV. Here to illustrate the idea, we give two examples. The first example (see Fig. 4) is used to explain the 'if' part, i.e., the graph G satisfies the condition in Theorem 1 and the secure storage capacity is $C = 1$.

Example 1: Consider the secure storage problem instance in Fig. 4. Each node has security constraint such that $L_v \geq L_w$ and $R \leq 1$. An optimal code with $R = 1$ is constructed as follows. Suppose each coded symbol W_k is from $\mathbb{F}_5$. First, G is a qualified component so that the same independent noise variable Z (uniform over $\mathbb{F}_5$) must be used (coined noise alignment, refer to Lemma 3). Second, the coded symbols are designed by considering each W_k and $G^{[k]}$ separately. For example, consider W_1 and $G^{[1]}$ in Fig. 4.(b), wherein an unqualified component cannot reveal anything about W_1 so that the same coded symbol must be assigned (coined coded symbol alignment, refer to Lemma 4). We then assign a generic combination to each unqualified component, e.g., $V_1^{[1]} = V_4^{[1]} = W_1 + Z, V_2^{[1]} = V_3^{[1]} = 2W_1 + Z, V_5^{[1]} = V_6^{[1]} = 3W_1 + Z$ (colored differently in Fig. 4.(b)). As there is no internal qualified edge, all qualified edges span different unqualified components and contain linearly independent combinations of the source symbol and the noise, from which the desired source symbol can be obtained (e.g., see $(V_1^{[1]}, V_5^{[1]})$ in Fig. 4.(b)). Finally, we combine (add) the source symbol assignment in each $G^{[k]}$ to produce the coded symbol assignment in G so that for any edge, the desired source symbol has different coefficients (thus correct) and undesired source symbols (and noise) are aligned (thus secure).

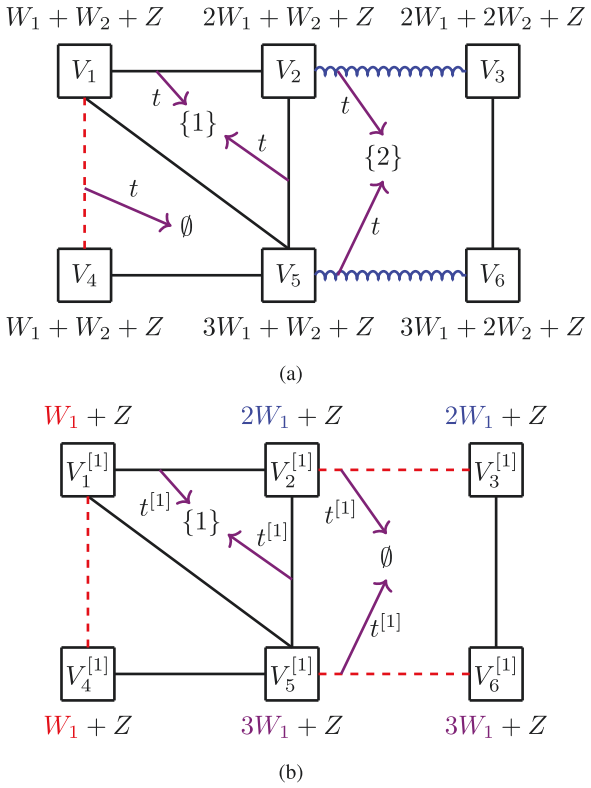


Fig. 4. (a) An example graph G with $K = 2, N = 6, D = 1$ and (b) its $G^{[1]}$ of W_1. The secure storage capacity over G is 1 and a capacity achieving code construction is shown.

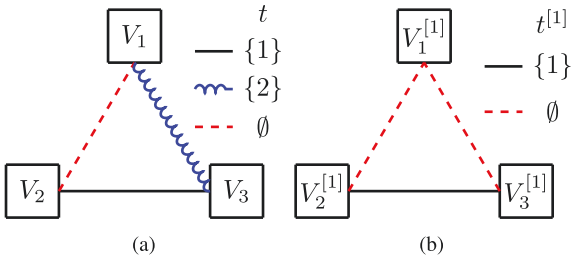


Fig. 5. (a) An example graph G with $K = 2, N = 3, D = 1$ and (b) its $G^{[1]}$ of W_1. The secure storage capacity over G cannot be 1.

The second example (see Fig. 5) is used to explain the 'only if' part, i.e., when $\widetilde{G} \neq \emptyset$ and the graph G does not satisfy the condition in Theorem 1, then the secure storage capacity $C < 1$.

Example 2: Consider the secure storage problem instance in Fig. 5. $G^{[1]}$ contains an internal qualified edge $\{V_2^{[1]}, V_3^{[1]}\}$ inside the unqualified path $\left(\{V_2^{[1]}, V_1^{[1]}\}, \{V_1^{[1]}, V_3^{[1]}\}\right)$. The intuition that $C \neq 1$, i.e., $L_v \neq L_w$ is as follows (ignoring $o(L_w)$ terms). When $L_v = L_w$, in Fig. 5.(a), G is a qualified component so that the same noise must be used in each of V_1, V_2, V_3 (called noise alignment, refer to Lemma 3); in Fig. 5.(b), $\left(\{V_2^{[1]}, V_1^{[1]}\}, \{V_1^{[1]}, V_3^{[1]}\}\right)$ is an unqualified path so that the same coded symbol about W_1 must be stored in V_1, V_2, V_3 (called coded symbol alignment, which can be captured by conditioned entropy. Refer to Lemma 4). It then

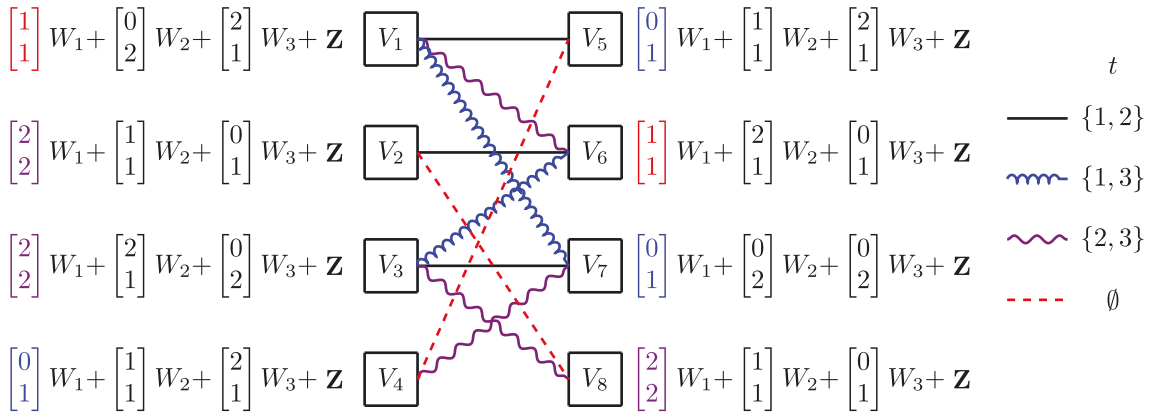


Fig. 6. An example graph G with $K=3, N=8, D=2$ and a code construction that achieves the capacity $1/2$. Note that the precoding coefficients for each source symbol are vectors while only scalars are needed in [3].

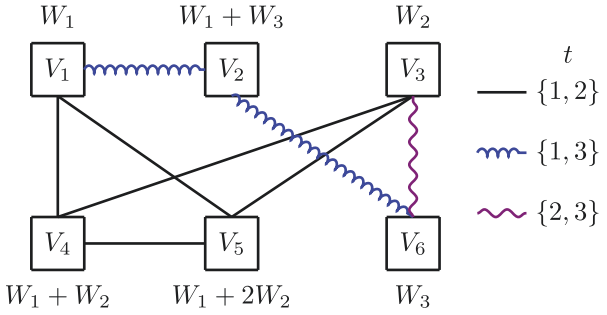


Fig. 7. An example graph G with $K=3, N=6, D=2$ and a code construction that achieves the capacity 1.

follows that V_2, V_3 must store the same information about W_1, which contradicts the fact that from (V_2, V_3), we can decode W_1. The above discussion can be translated to entropy manipulations and the details are presented in Section V-A.

B. Arbitrary D and Extremal Graphs With $C = 1/D$

Next, we extend Theorem 1 to the setting of arbitrary D. Under the condition that every non-degenerate node has no common source, all extremal graphs whose secure storage capacity is $C = 1/D$ are characterized in the following theorem.

Theorem 2: Consider the class of graph $G = (\mathcal{V}, \mathcal{E})$ where the non-degenerate subgraph $\widetilde{G}$ of G is not empty and $\mathcal{C}(V) = \emptyset, \forall V \in \mathcal{V}\backslash\mathcal{V}_d$. For this class of graph G, the capacity of a secure storage code is $C = 1/D$ if and only if for every qualified component Q of $\widetilde{G}$, the characteristic graph $Q^{[k]}$ of each coded symbol $W_k, k \in [K]$ contains no internal qualified edge.

Remark 2: Theorem 2 includes Theorem 1 as a special case because when $D = 1$, any non-degenerate node must have no common source (note that non-degenerate nodes are connected to edges associated with different source symbols).

The proof of Theorem 2 is presented in Section V. An example is given in Fig. 6 to explain the code construction of the 'if' part.

Example 3: Consider the graph G in Fig. 6. The code construction is based on a similar idea as that of Example 1, i.e., each W_k and $G^{[k]}$ is considered separately and generic combinations are assigned to each unqualified component of $G^{[k]}$ (colored differently in Fig. 6 for W_1); then the overall assignment is obtained as the sum of each assignment in $G^{[k]}$. In Fig. 6, each W_k is from $\mathbb{F}_3$ and $\mathbf{Z} \in \mathbb{F}_3^{2\times 1}$ is an independent uniform noise. The main difference between this example where $D = 2$ and Example 1 where $D = 1$ is that to ensure correctness, $D = 1$ only requires the coefficients of the desired source symbol to be different while $D > 1$ needs the coefficient matrix to be full rank (for which an explicit design is not obvious). An explicit solution is provided in Fig. 6 for this small example while in general, the proof in Section V-B relies on randomized construction.

C. Arbitrary D and Extremal Graphs With $C = 2/D$

Finally, we consider the extremal rate of $2/D$. All extremal graphs whose secure storage capacity is $C = 2/D$ are characterized in the following theorem.

Theorem 3: The capacity of a secure storage code over a graph $G = (\mathcal{V}, \mathcal{E})$ is $C = 2/D$ if and only if the following two conditions are satisfied.

1) For any $V \in \mathcal{V}$, $|\mathcal{C}(V)| \geq D/2$.
2) For any $\{V_i, V_j\} \in \mathcal{E}$, $\mathcal{C}(V_i) \cup \mathcal{C}(V_j) = t(\{V_i, V_j\})$.

In words, the conditions in Theorem 3 are 1). for each node, there are at least $D/2$ common sources and 2). for each qualified edge, the union of the common sources of both nodes must be the set of D desired source symbols. The intuition is fairly straightforward as the total storage of any qualified edge is exactly $2L_v = 2 \times 1/R \times L_w = D \times L_w$ (ignoring $o(L_w)$ terms), which must be fully occupied by the desired D source symbols and there is absolutely no room for anything else. As a consequence, we can show that each coded symbol must be a deterministic function of its common sources (refer to Lemma 5). Then the two conditions in Theorem 3 follow as necessary conditions as otherwise we do not have sufficient information from the desired source symbols to fill a node and a qualified edge. The two conditions also turn out to be sufficient by random linear coding, i.e., storing a sufficient number of generic combinations of the common sources guarantees the successful recovery of the desired source symbols (see Fig. 7 for an example). The detailed proof of Theorem 3 is deferred to Section VI.

IV. Proof of Theorem 1

Theorem 1 is recovered as a special case of Theorem 2, so the proof of Theorem 1 can also be recovered from the proof of Theorem 2, presented in Section V. However, in this section we still provide a proof of the code construction for the 'if' part because here $D = 1$, the code can be made explicit while in Theorem 2 where D can be arbitrary, the code is randomized.

A. Code Construction of the 'if' Part

We show that if G satisfies the condition in Theorem 1, then we can construct a secure storage code of symbol rate $R = 1$. Suppose $L_w = \log_2(q)$ bits and each source symbol W_k is one symbol from finite field $\mathbb{F}_q$, where the field size q will be specified later in the proof. Each coded symbol V_n will be set as one symbol from $\mathbb{F}_q$, i.e., $L_v = \log_2(q)$ bits and $R = L_w/L_v = 1$, as desired.

A degenerate node $V \in \mathcal{V}_d$, i.e., all connected edges are associated with the same coded symbol W_i (or all connected edges are unqualified, in this case set V to contain an independent noise variable), is trivial, and set $V = W_i$. Henceforth, we focus on the non-degenerate subgraph $\widetilde{G}$ of G, i.e., all non-degenerate nodes $\mathcal{V}\backslash\mathcal{V}_d$. Suppose $\widetilde{G}$ has M qualified components, $Q_1, \cdots, Q_M$. To assign the coded symbols in $Q_m, m \in [M]$, we will first consider the characteristic graph $Q_m^{[k]}, m \in [M], k \in [K]$ of each coded symbol W_k separately and then combine the separated assignments.

Consider $Q_m^{[k]}, \forall m \in [M], k \in [K]$. Suppose $Q_m^{[k]}$ contains $U_m^{[k]}$ unqualified components and set the field size q as a prime number such that $q > \max_{m,k} U_m^{[k]}$. The nodes in $Q_m^{[k]}$ are set as follows.

$$\text{For each node } V^{[k]} \text{ in the } u\text{-th unqualified component of } Q_m^{[k]} \text{ where } u \in [U_m^{[k]}], \text{ set } V^{[k]} = u \times W_k + Z_m^{[k]} \tag{8}$$

where $Z_m^{[k]}, \forall m, k$ are i.i.d. uniform noise symbols from $\mathbb{F}_q$ and are independent of W_k. As the condition of Theorem 1 is satisfied, i.e., $Q_m^{[k]}$ contains no internal qualified edge, the assignment (8) satisfies the following two properties.

$$\text{For each qualified edge } \{V_i^{[k]}, V_j^{[k]}\} \text{ in } Q_m^{[k]}, \text{ from } V_i^{[k]} - V_j^{[k]} \text{ we can obtain } W_k. \tag{9}$$

$$\text{For each unqualified edge } \{V_i^{[k]}, V_j^{[k]}\} \text{ in } Q_m^{[k]}, \; V_i^{[k]} = V_j^{[k]}. \tag{10}$$

(9) follows from the observation that as there is no internal edge, $V_i^{[k]}$ and $V_j^{[k]}$ belong to different unqualified components such that the coefficients before W_k are different (see (8)). (10) follows from the fact that any unqualified edge belongs to the same unqualified component and (8).

Consider $Q_m, \forall m \in [M]$: Note that the nodes in Q_m and $Q_m^{[k]}$ have a one-to-one mapping. Each node in Q_m is simply set as the sum of each corresponding node in $Q_m^{[k]}$ for all $k \in [K]$.

$$\text{For each node } V \text{ in } Q_m, \text{ set } V = \textstyle\sum_{k\in[K]} V^{[k]}. \tag{11}$$

We Show That the Code Construction (8), (11) is Correct and Secure: Edges connected to degenerate nodes are trivial and we only need to consider the remaining edges. Consider any qualified edge $\{V_i, V_j\}$, i.e., $t(\{V_i, V_j\}) = \{l\}$.

$$V_i - V_j \overset{(11)}{=} \sum_{k\in[K]} \left(V_i^{[k]} - V_j^{[k]}\right) \tag{12}$$

$$\overset{(10)}{=} V_i^{[l]} - V_j^{[l]} \tag{13}$$

where the last step follows from the fact that $\{V_i^{[k]}, V_j^{[k]}\}, k \neq l$ is an unqualified edge as $t(\{V_i, V_j\}) = \{l\}$ (see Definition 1). Further, $\{V_i^{[l]}, V_j^{[l]}\}$ is a qualified edge, so by (9), $V_i^{[l]} - V_j^{[l]}$ can recover W_l and correctness is guaranteed. To verify security, note that (V_i, V_j) is invertible to $(V_i - V_j, V_i)$, which is further invertible to (W_l, V_i). From (8), (11), V_i is fully covered by uniform noise variables such that nothing about source symbols other than W_l is revealed and security follows. Finally, any unqualified edge $\{V_i, V_j\}$ is easily seen to be secure, because if V_i, V_j belong to the same qualified component, then $V_i = V_j$ and V_i is independent of all source symbols; otherwise V_i, V_j belong to two qualified components, then V_i, V_j are covered by independent noise variables.

V. Proof of Theorem 2

This section contains the proof of Theorem 2. We first prove the 'only if' part in Section V-A and then prove the 'if' part in Section V-B.

A. Only If Part

We start with a useful lemma that holds for any symbol rate and any graph. This lemma will be used in the proof of Theorem 3 as well.

Lemma 1 (Independence of Non-common Sources): A coded symbol V must be independent of its non-common source symbols (with and without conditioning on the common source symbols),

$$I\left(V; (W_k)_{k\in[K]\backslash\mathcal{C}(V)} \mid (W_k)_{k\in\mathcal{C}(V)}\right) = 0, \tag{14}$$

$$I\left(V; (W_k)_{k\in[K]\backslash\mathcal{C}(V)}\right) = 0. \tag{15}$$

Proof: First, we prove (14). Consider any non-common source symbol W_i of the node V, i.e., $i \in [K]\backslash\mathcal{C}(V)$. As W_i is not a common source symbol of V, there must exist an edge $\{V, V_j\}$ such that $i \notin t(\{V, V_j\}) = \mathcal{D}$, for which from the security constraint (3) we have

$$0 \overset{(3)}{=} I\left(V, V_j; (W_k)_{k\in[K]\backslash\mathcal{D}} \mid (W_k)_{k\in\mathcal{D}}\right) \tag{16}$$

$$\geq I\left(V; W_i \mid (W_k)_{k\in[K]\backslash\{i\}}\right). \tag{17}$$

Consider any subset $\mathcal{J}$ of $[K]\backslash(\{i\} \cup \mathcal{C}(V))$. As the source symbols W_k are independent (refer to (1)), from (17) we have

$$0 \overset{(17)}{\geq} I\left(V; W_i \mid (W_k)_{k\in[K]\backslash\{i\}}\right) \tag{18}$$

$$\overset{(1)}{=} I\left(V, (W_k)_{k\in[K]\backslash(\{i\}\cup\mathcal{C}(V)\cup\mathcal{J})}; W_i \mid \ldots \right.$$
$$\left. \ldots (W_k)_{k\in\mathcal{J}}, (W_k)_{k\in\mathcal{C}(V)}\right) \tag{19}$$

$$\geq I\left(V; W_i \mid (W_k)_{k\in\mathcal{J}}, (W_k)_{k\in\mathcal{C}(V)}\right). \tag{20}$$

The desired identity (14) can now be obtained by adding (20) for a proper sequence of $\mathcal{J}$ that is consistent with the chain rule expansion of (14).

Second, we prove (15), as a simple consequence of (14).

$$0 \overset{(14)}{=} I\left(V; (W_k)_{k\in[K]\setminus\mathcal{C}(V)} \mid (W_k)_{k\in\mathcal{C}(V)}\right) \tag{21}$$
$$\overset{(1)}{=} I\left(V, (W_k)_{k\in\mathcal{C}(V)}; (W_k)_{k\in[K]\setminus\mathcal{C}(V)}\right) \tag{22}$$
$$\geq I\left(V; (W_k)_{k\in[K]\setminus\mathcal{C}(V)}\right). \tag{23}$$

■

We now proceed to the proof of the 'only if' part. We show that symbol rate $R = 1/D$ is not achievable if a graph G does not satisfy the condition in Theorem 2, i.e., there exists a qualified component Q of the non-degenerate subgraph $\widetilde{G}$ of G such that the characteristic graph $Q^{[k]}$ of some coded symbol W_k contains an internal qualified edge. Without loss of generality, suppose $k = 1$ and suppose the internal qualified edge is $\{V_1^{[1]}, V_P^{[1]}\}$, which is inside the sequence of unqualified edges $\left(\{V_1^{[1]}, V_2^{[1]}\}, \cdots, \{V_{P-1}^{[1]}, V_P^{[1]}\}\right)$. To set up the proof by contradiction, let us assume that $R = \frac{1}{D} = \lim_{L_w\to\infty} \frac{L_w}{L_v}$ is asymptotically achievable, i.e., $L_v = DL_w + o(L_w)$. Note that when the rate is exactly achievable, the $o(L_w)$ term is zero and the following proof continues to hold.

We show that when $R = 1/D$, each non-degenerate coded symbol that has no common source must be fully covered by noise. This property is stated in the following lemma.

Lemma 2 (Noise Size): When $R = 1/D$, for a non-degenerate graph $\widetilde{G} = (\mathcal{V}, \mathcal{E})$ such that every node $V \in \mathcal{V}$ satisfies $\mathcal{C}(V) = \emptyset$, we have

$$H(V) = H\left(V \mid (W_k)_{k\in[K]\setminus\{1\}}\right)$$
$$= H\left(V \mid (W_k)_{k\in[K]}\right) = DL_w + o(L_w). \tag{24}$$

Proof: From (15) and $\mathcal{C}(V) = \emptyset$, we have $H(V) = H\left(V \mid (W_k)_{k\in[K]\setminus\{1\}}\right) = H\left(V \mid (W_k)_{k\in[K]}\right)$. Noting that $H(V) \leq L_v = DL_w + o(L_w)$, we only need to prove $H(V) \geq DL_w + o(L_w)$ and this is presented next.

As V is non-degenerate, there exists a qualified edge $\{V, V_i\} \in \mathcal{E}$ and $t(\{V, V_i\}) = \mathcal{D}, |\mathcal{D}| = D$. From the correctness constraint (2), we have

$$DL_w \overset{(1)}{=} H\left((W_k)_{k\in\mathcal{D}}\right) \tag{25}$$
$$\overset{(2)}{=} I\left(V, V_i; (W_k)_{k\in\mathcal{D}}\right) \tag{26}$$
$$\overset{(15)}{=} I\left(V; (W_k)_{k\in\mathcal{D}} \mid V_i\right) \tag{27}$$
$$\leq H(V) \tag{28}$$

where in (27), we use the fact that $V_i \in \mathcal{V}\setminus\mathcal{V}_d$ such that $\mathcal{C}(V_i) = \emptyset$ and from Lemma 1, V_i is independent of the source symbols W_k. ■

Next we show that when $R = 1/D$, all nodes in a qualified component must use the same noise, i.e., noise must align. This property is stated in the following lemma.

Lemma 3 (Noise Alignment): When $R = 1/D$, for a non-degenerate graph $\widetilde{G} = (\mathcal{V}, \mathcal{E})$ such that every node $V \in \mathcal{V}$ satisfies $\mathcal{C}(V) = \emptyset$, we have

$$\forall\{V_i, V_j\} \in \mathcal{E} \text{ such that } t(\{V_i, V_j\}) = \mathcal{D}, |\mathcal{D}| = D,$$
$$H\left(V_i, V_j \mid (W_k)_{k\in[K]}\right) = DL_w + o(L_w), \tag{29}$$
and for any qualified component Q of $\widetilde{G}$
with node index set $\mathcal{Q}$,
$$H\left((V_i)_{i\in\mathcal{Q}} \mid (W_k)_{k\in[K]}\right) = DL_w + o(L_w). \tag{30}$$

Proof: First, consider (29). On the one hand, we have

$$H\left(V_i, V_j \mid (W_k)_{k\in[K]}\right)$$
$$= H\left(V_i, V_j\right) - I\left(V_i, V_j; (W_k)_{k\in[K]}\right) \tag{31}$$
$$\overset{(2)}{=} H\left(V_i, V_j\right) - I\left(V_i, V_j, (W_k)_{k\in\mathcal{D}}; (W_k)_{k\in[K]}\right) \tag{32}$$
$$\leq H\left(V_i, V_j\right) - H\left((W_k)_{k\in\mathcal{D}}\right) \tag{33}$$
$$\overset{(1)}{\leq} 2L_v - DL_w = DL_w + o(L_w). \tag{34}$$

On the other hand, we have

$$H\left(V_i, V_j \mid (W_k)_{k\in[K]}\right)$$
$$\geq H\left(V_i \mid (W_k)_{k\in[K]}\right) \overset{(24)}{=} DL_w + o(L_w). \tag{35}$$

Second, consider (30). The "$\geq$" direction is obvious, because for any $j \in \mathcal{Q}$

$$H\left((V_i)_{i\in\mathcal{Q}} \mid (W_k)_{k\in[K]}\right)$$
$$\geq H\left(V_j \mid (W_k)_{k\in[K]}\right) \overset{(24)}{=} DL_w + o(L_w) \tag{36}$$

and we only need to prove the "$\leq$" direction. Start with any qualified edge $\{V_{i_1}, V_{i_2}\}, i_1, i_2 \in \mathcal{Q}$ in the qualified component Q, inside which there must exist a node V_{i_3} and a node from V_{i_1}, V_{i_2} (suppose it is V_{i_2} without loss of generality) such that $\{V_{i_2}, V_{i_3}\}$ is a qualified edge. From the sub-modularity property of entropy functions, we have

$$H\left(V_{i_1}, V_{i_2} \mid (W_k)_{k\in[K]}\right) + H\left(V_{i_2}, V_{i_3} \mid (W_k)_{k\in[K]}\right)$$
$$\geq H\left(V_{i_1}, V_{i_2}, V_{i_3} \mid (W_k)_{k\in[K]}\right) + H\left(V_{i_2} \mid (W_k)_{k\in[K]}\right) \tag{37}$$
$$\overset{(29)(24)}{\Longrightarrow} DL_w + DL_w \geq H\left(V_{i_1}, V_{i_2}, V_{i_3} \mid (W_k)_{k\in[K]}\right)$$
$$+ DL_w + o(L_w) \tag{38}$$
$$\Rightarrow \quad H\left(V_{i_1}, V_{i_2}, V_{i_3} \mid (W_k)_{k\in[K]}\right) \leq DL_w + o(L_w). \tag{39}$$

Then we can similarly proceed to include all nodes in Q. As Q is a qualified component, there must exist a vertex $V_{i_4}, i_4 \in \mathcal{Q}$ such that $\{V, V_{i_4}\}$ is a qualified edge, where V is one vertex from $V_{i_1}, V_{i_2}, V_{i_3}$. Similarly, we have

$$H\left(V_{i_1}, V_{i_2}, V_{i_3}, V_{i_4} \mid (W_k)_{k\in[K]}\right) \leq DL_w + o(L_w), \cdots,$$
$$H\left((V_i)_{i\in\mathcal{Q}} \mid (W_k)_{k\in[K]}\right) \leq DL_w + o(L_w). \tag{40}$$

■

Consider the nodes $V_1, \cdots, V_P$ that violate the condition in Theorem 2, i.e., from each one of $(V_1, V_2), \cdots, (V_{P-1}, V_P)$, we cannot learn anything about W_1; from (V_1, V_P), we can decode W_1. In the following lemma, we show that the coded symbols $V_1, \cdots, V_P$ must contain the same information of W_1 and noise, i.e., the coded symbols must align.

Lemma 4 (Coded Symbol Alignment): When $R = 1/D$, for the nodes $V_1, \cdots, V_P$ as specified above, we have

$$\forall p \in [P-1],$$
$$H\left(V_p, V_{p+1} \mid (W_k)_{k\in[K]\setminus\{1\}}\right) = DL_w + o(L_w), \tag{41}$$
$$H\left(V_1, V_P \mid (W_k)_{k\in[K]\setminus\{1\}}\right) = DL_w + o(L_w). \tag{42}$$

Proof: For both (41) and (42), the "$\geq$" direction follows from (24) and we only need to prove the "$\leq$" direction.

First, consider (41). From (V_p, V_{p+1}) we cannot decode W_1, so the edge $\{V_p, V_{p+1}\}$ is either an unqualified edge or a qualified edge but $1 \notin t(\{V_p, V_{p+1}\})$. In the former case, from the security constraint (3) where $t(\{V_p, V_{p+1}\}) = \emptyset$, we have

$$\begin{aligned} &H\left(V_p, V_{p+1} \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \\ &\overset{(3)(1)}{=} H\left(V_p, V_{p+1} \mid (W_k)_{k\in[K]}\right) && (43)\\ &\leq H\left((V_i)_{i\in\mathcal{Q}} \mid (W_k)_{k\in[K]}\right) && (44)\\ &\overset{(30)}{=} DL_w + o(L_w) && (45)\end{aligned}$$

where (44) follows from the fact that $V_1, \cdots, V_P$ belong to a qualified component Q with node index set $\mathcal{Q}$. In the latter case, from the correctness constraint (2) where $1 \notin t(\{V_p, V_{p+1}\}) = \mathcal{D}$, we have

$$\begin{aligned} &H\left(V_p, V_{p+1} \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \\ &\overset{(2)}{=} H\left(V_p, V_{p+1}\right) - I\left(V_p, V_{p+1}, (W_k)_{k\in\mathcal{D}}; \ldots \right. \\ &\quad \left. \ldots (W_k)_{k\in[K]\setminus\{1\}}\right) && (46)\\ &\overset{(1)}{\leq} 2L_v - DL_w = DL_w + o(L_w). && (47)\end{aligned}$$

Second, consider (42). From the sub-modularity property of entropy functions, we have

$$\begin{aligned} &(P-1)DL_w + o(L_w) \\ &\overset{(41)}{=} \sum_{p\in[P-1]} H\left(V_p, V_{p+1} \mid (W_k)_{k\in[K]\setminus\{1\}}\right) && (48)\\ &\geq H\left(V_1, \cdots, V_P \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \\ &\quad + \sum_{p=2}^{P-1} H\left(V_p \mid (W_k)_{k\in[K]\setminus\{1\}}\right) && (49)\\ &\overset{(24)}{\geq} H\left(V_1, V_P \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \\ &\quad + (P-2)DL_w + o(L_w) && (50)\\ &\Rightarrow H\left(V_1, V_P \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \leq DL_w + o(L_w). && (51)\end{aligned}$$

■

After establishing the above lemmas, we are ready to demonstrate the contradiction as follows. Recall that from (V_1, V_P), we can recover W_1, i.e., $1 \in t(\{V_1, V_P\})$.

$$\begin{aligned} &DL_w + o(L_w) \\ &\overset{(42)}{=} H\left(V_1, V_P \mid (W_k)_{k\in[K]\setminus\{1\}}\right) && (52)\\ &\overset{(2)}{=} H\left(V_1, V_P, W_1 \mid (W_k)_{k\in[K]\setminus\{1\}}\right) && (53)\\ &= H\left(W_1 \mid (W_k)_{k\in[K]\setminus\{1\}}\right) \\ &\quad + H\left(V_1, V_P \mid (W_k)_{k\in[K]}\right) && (54)\\ &\overset{(1)(29)}{=} L_w + DL_w + o(L_w). && (55)\end{aligned}$$

Normalizing (55) by L_w and letting L_w approach infinity, we have $D = 1 + D$, and the contradiction is arrived. The proof of the only if part is thus complete.

B. If Part

We show that if the condition in Theorem 2 is satisfied, then the secure storage capacity is $1/D$. We first prove that $R \leq 1/D$ and then show that $R = 1/D$ is achievable.

The proof of $R \leq 1/D$ is immediate. As $\widetilde{G}$ is not empty, there must exist a qualified edge $\{V_i, V_j\}$ such that $t(\{V_i, V_j\}) = \mathcal{D}, |\mathcal{D}| = D$. From the correctness constraint (2), we have

$$\begin{aligned} DL_w &\overset{(1)}{=} H\left((W_k)_{k\in\mathcal{D}}\right) && (56)\\ &\overset{(2)}{=} I\left(V_i, V_j; (W_k)_{k\in\mathcal{D}}\right) && (57)\\ &\overset{(15)}{=} I\left(V_j; (W_k)_{k\in\mathcal{D}} \mid V_i\right) && (58)\\ &\leq H(V_j) \leq L_v && (59)\\ \Rightarrow \quad R &\overset{(4)}{=} L_w/L_v \leq 1/D && (60)\end{aligned}$$

where (58) follows from the condition that $\mathcal{C}(V_i) = \emptyset, \forall V_i \in \mathcal{V}\setminus\mathcal{V}_d$ and (15).

We now present a secure storage code construction that achieves symbol rate $R = 1/D$ if $G = (\mathcal{V}, \mathcal{E})$ satisfies the condition in Theorem 2. The scheme is a generalization of that presented in Section IV-A. Suppose $L_w = \log_2(q)$ bits and each source symbol W_k is one symbol from finite field $\mathbb{F}_q$, where $q > D|\mathcal{E}|$. Each coded symbol V_n will be set as D symbols from $\mathbb{F}_q$, i.e., $L_v = D\log_2(q)$ bits and $R = L_w/L_v = 1/D$, as desired.

Degenerate nodes $\mathcal{V}_d$ (and their connected edges) are trivial and we only need to consider the non-degenerate subgraph $\widetilde{G}$ of G. Suppose $\widetilde{G}$ has M qualified components, $Q_1, \cdots, Q_M$.

Consider $Q_m^{[k]}, \forall m \in [M], k \in [K]$. Suppose $Q_m^{[k]}$ contains $U_m^{[k]}$ unqualified components.

For each node $V^{[k]}$ in the u-th unqualified component of $Q_m^{[k]}$ where $u \in [U_m^{[k]}]$,
$$\text{set } V^{[k]} = \mathbf{h}_{m,u}^{[k]} \times W_k + \mathbf{Z}_m^{[k]} \tag{61}$$

where $\mathbf{h}_{m,u}^{[k]} \in \mathbb{F}_q^{D\times 1}$; $\mathbf{Z}_m^{[k]} \in \mathbb{F}_q^{D\times 1}, \forall m, k$ are i.i.d. uniform noise symbols that are independent of W_k.

Consider $Q_m, \forall m \in [M]$.

For each node V in Q_m, set $V = \sum_{k\in[K]} V^{[k]}$. (62)

We show that there exists a choice of $\mathbf{h}_{m,u}^{[k]}, k \in [K], m \in [M], u \in [U_m^{[k]}]$ such that the code construction (61), (62) is correct and secure. To this end, choose every entry of $\mathbf{h}_{m,u}^{[k]}$ independently and uniformly from $\mathbb{F}_q$. Consider correctness. For any qualified edge $\{V_i, V_j\}$, i.e., $t(\{V_i, V_j\}) = \mathcal{D}$, $|\mathcal{D}| = D$, we have

$$\begin{aligned} V_i - V_j &\overset{(62)}{=} \sum_{k\in[K]} \left(V_i^{[k]} - V_j^{[k]}\right) && (63)\\ &\overset{(61)}{=} \sum_{k\in\mathcal{D}} \left(V_i^{[k]} - V_j^{[k]}\right) && (64)\\ &= \mathbf{H}_{ij} \times (W_k)_{k\in\mathcal{D}} && (65)\end{aligned}$$

where (64) from the fact that $\{V_i^{[k]}, V_j^{[k]}\}, k \notin \mathcal{D}$ is an unqualified edge such that $V_i^{[k]}, V_j^{[k]}$ belong to the same unqualified component and from (61), $V_i^{[k]} = V_j^{[k]}, k \notin \mathcal{D}$. (65) is obtained because $\{V_i^{[k]}, V_j^{[k]}\}, k \in \mathcal{D}$ is a qualified edge that is not internal, i.e., spans different unqualified components. In addition, $\mathbf{H}_{ij}$ is a $D \times D$ matrix over $\mathbb{F}_q$, whose entries can be obtained from $\mathbf{h}_{m,u}^{[k]}$ and we require the matrix $\mathbf{H}_{ij}$ to have full rank while the corresponding place in the code of [3] is an obviously non-zero scalar. View the determinant of $\mathbf{H}_{ij}$, $|\mathbf{H}_{ij}|$ as a polynomial in variables $\mathbf{h}_{m,u}^{[k]}, k \in [K], m \in [M], u \in [U_m^{[k]}]$. This determinant polynomial has degree D and is not a zero polynomial as there exists a realization of $\mathbf{h}_{m,u}^{[k]}$ such that the determinant is not zero. Consider the product of the determinant polynomials for all $|\mathcal{E}|$ qualified edges,

$$\text{poly} \triangleq \prod_{i,j:\{V_i,V_j\}\in\mathcal{E}} |\mathbf{H}_{ij}| \tag{66}$$

which is a non-zero polynomial and has degree at most $D|\mathcal{E}|$. By the Schwartz–Zippel lemma [21], [22], [23], a uniform choice of $\mathbf{h}_{m,u}^{[k]}, k \in [K], m \in [M], u \in [U_m^{[k]}]$ over $\mathbb{F}_q$ where $q > D|\mathcal{E}|$ (the degree of poly) guarantees poly is not always zero. It follows that there exists some realization of $\mathbf{h}_{m,u}^{[k]}, k \in [K], m \in [M], u \in [U_m^{[k]}]$ such that poly $\neq 0$. Then each $|\mathbf{H}_{ij}|$ is not zero and from each qualified edge, we can recover the D desired source symbols, i.e., correctness is guaranteed.

Finally consider security. For any qualified edge $\{V_i, V_j\}$, security is guaranteed by noting that (V_i, V_j) is invertible to $\left((W_k)_{k\in\mathcal{D}}, V_i\right)$ and V_i is fully covered by uniform noise variables. For any unqualified edge $\{V_i, V_j\}$, security holds no matter whether V_i, V_j belong to the same qualified component (same coded symbol assignment, i.e., $V_i = V_j$) or two qualified components (then V_i, V_j are protected by independent noise variables).

VI. Proof of Theorem 3

This section contains the proof of Theorem 3. We first prove the 'only if' part in Section VI-A and then prove the 'if' part in Section VI-B.

A. Only If Part

We start with a useful property for any secure storage code of symbol rate $R = 2/D$, stated in the following lemma. Note that when $R = \frac{2}{D} = \lim_{L_w\to\infty} \frac{L_w}{L_v}$, we have[2]

$$2L_v = DL_w + o(L_w). \tag{67}$$

Lemma 5 (Deterministic of Common Sources): When $R = 2/D$, a coded symbol V that is connected to a qualified edge is asymptotically deterministic given its common source symbols,

$$H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}\right) = o(L_w). \tag{68}$$

[2]The same proof holds when the $o(L_w)$ term is 0, i.e., when the rate is exactly achievable.

Proof: Consider any qualified edge $\{V, V_i\}$ such that $t(\{V, V_i\}) = \mathcal{D}, |\mathcal{D}| = D$. From the correctness constraint (2), we have

$$\begin{aligned}
2L_v &\geq H(V, V_i) && (69)\\
&\overset{(2)}{=} H\left(V, V_i, (W_k)_{k\in\mathcal{D}}\right) && (70)\\
&= H\left((W_k)_{k\in\mathcal{D}}\right) + H\left(V, V_i \mid (W_k)_{k\in\mathcal{D}}\right) && (71)\\
&\overset{(1)}{\geq} DL_w + H\left(V \mid (W_k)_{k\in\mathcal{D}}\right) && (72)\\
&\geq DL_w && (73)\\
&\overset{(67)}{=} 2L_v + o(L_w). && (74)
\end{aligned}$$

The above sequence of inequalities starts and ends both with $2L_v$ (ignoring $o(L_w)$ terms), then all the inequalities must be equalities within the distortion of $o(L_w)$. In particular,

$$H(V, V_i) = 2L_v + o(L_w), \quad H(V) = L_v + o(L_w) \tag{75}$$

and

$$\begin{aligned}
o(L_w) &= H\left(V \mid (W_k)_{k\in\mathcal{D}}\right) && (76)\\
&= H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}, (W_k)_{k\in\mathcal{D}\setminus\mathcal{C}(V)}\right) && (77)\\
&= H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}\right) - I\left(V; \ldots\right.\\
&\quad \left.\ldots (W_k)_{k\in\mathcal{D}\setminus\mathcal{C}(V)} \mid (W_k)_{k\in\mathcal{C}(V)}\right) && (78)\\
&\geq H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}\right) - I\left(V; \ldots\right.\\
&\quad \left.\ldots (W_k)_{k\in[K]\setminus\mathcal{C}(V)} \mid (W_k)_{k\in\mathcal{C}(V)}\right) && (79)\\
&\overset{(14)}{=} H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}\right). && (80)
\end{aligned}$$

■

Equipped with Lemma 5, we are ready to present the proof of the 'only if' part. We show that if either of the two conditions in Theorem 3 is violated, then the symbol rate R cannot be $2/D$. We will prove this by contradiction, so suppose there exists a secure storage code of symbol rate $R = 2/D$.

Suppose condition 1 is violated, i.e., there exists a node V such that $|\mathcal{C}(V)| < D/2$. Then

$$\begin{aligned}
&L_v + o(L_w)\\
&\overset{(75)}{=} H(V) && (81)\\
&= \underbrace{H\left(V \mid (W_k)_{k\in\mathcal{C}(V)}\right)}_{\overset{(68)}{=} o(L_w)} + I\left(V; (W_k)_{k\in\mathcal{C}(V)}\right) && (82)\\
&\leq H\left((W_k)_{k\in\mathcal{C}(V)}\right) + o(L_w)\\
&\overset{(1)}{=} |\mathcal{C}(V)| \times L_w + o(L_w) && (83)\\
&< D/2 \times L_w + o(L_w) && (84)\\
&\Rightarrow R = \lim_{L_w\to\infty} L_w/L_v > 2/D && (85)
\end{aligned}$$

which contradicts the assumption that $R = 2/D$.

Suppose condition 1 is satisfied while condition 2 is violated, i.e., there exists a qualified edge $\{V_i, V_j\}$ such that $\mathcal{C}(V_i) \cup \mathcal{C}(V_j)$ is a strict subset of $t(\{V_i, V_j\})$.

Then $|\mathcal{C}(V_i) \cup \mathcal{C}(V_j)| < D$ and

$$\begin{aligned} & 2L_v + o(L_w) \\ & \overset{(75)}{=} H(V_i, V_j) && (86) \\ & = \underbrace{H\left(V_i, V_j \mid (W_k)_{k\in\mathcal{C}(V_i)\cup\mathcal{C}(V_j)}\right)}_{\overset{(68)}{=} o(L_w)} \\ & \quad + I\left(V_i, V_j; (W_k)_{k\in\mathcal{C}(V_i)\cup\mathcal{C}(V_j)}\right) && (87) \\ & \le H\left((W_k)_{k\in\mathcal{C}(V_i)\cup\mathcal{C}(V_j)}\right) + o(L_w) \\ & \overset{(1)}{=} |\mathcal{C}(V_i) \cup \mathcal{C}(V_j)| \times L_w + o(L_w) && (88) \\ & < D \times L_w + o(L_w) && (89) \\ & \Rightarrow R = \lim_{L_w\to\infty} L_w/L_v \ > \ 2/D && (90) \end{aligned}$$

which contradicts the assumption that $R = 2/D$.

B. If Part

We show that if the two conditions in Theorem 3 are satisfied, then the secure storage capacity is $2/D$. We first prove that $R \le 2/D$ and then show that $R = 2/D$ is achievable.

The proof of $R \le 2/D$ is immediate. As condition 1 is satisfied, all edges are qualified. Pick any one, say $\{V_i, V_j\}$ such that $t(\{V_i, V_j\}) = \mathcal{D}, |\mathcal{D}| = D$. From the correctness constraint (2), we have

$$\begin{aligned} 2L_v & \ge H(V_i, V_j) && (91) \\ & \overset{(2)}{=} H\left(V_i, V_j, (W_k)_{k\in\mathcal{D}}\right) && (92) \\ & \ge H\left((W_k)_{k\in\mathcal{D}}\right) && (93) \\ & \overset{(1)}{=} D \times L_w && (94) \\ \Rightarrow \quad R & \overset{(4)}{=} L_w/L_v \ \le \ 2/D. && (95) \end{aligned}$$

We now present a secure storage code construction that achieves symbol rate $R = 2/D$. Consider any graph $G = (\mathcal{V}, \mathcal{E})$ that satisfies the two conditions in Theorem 3. Set $L_w = 2\log_2(q)$ bits, where $q > 2D|\mathcal{E}|$. Suppose each W_k consists of 2 i.i.d. uniform symbols from $\mathbb{F}_q$, i.e., $W_k \in \mathbb{F}_q^{2\times 1}$. We set each coded symbol $V_1, \cdots, V_N$ as follows so that $V_n \in \mathbb{F}_q^{D\times 1}, \forall n \in [N]$, i.e., $L_v = D\log_2(q)$ bits and the symbol rate achieved is $R = L_w/L_v = 2/D$, as desired.

$$\text{Set } V_n = \mathbf{H}_n \times (W_k)_{k\in\mathcal{C}(V_n)}, \forall n \in [N] \tag{96}$$

where $(W_k)_{k\in\mathcal{C}(V_n)} \in \mathbb{F}_q^{2|\mathcal{C}(V_n)|\times 1}$ is a column vector that stacks each W_k and $\mathbf{H}_n \in \mathbb{F}_q^{D\times 2|\mathcal{C}(V_n)|}$.

Next we show that there exists a choice of $\mathbf{H}_n, n \in [N]$ so that the constructed code satisfies the correctness and security constraints (2), (3). To prove the existence, we generate $\mathbf{H}_n$, $n \in [N]$ randomly by choosing each element of $\mathbf{H}_n, n \in [N]$ independently and uniformly from $\mathbb{F}_q$.

Note that condition 2 in Theorem 3 is satisfied, i.e., for any qualified edge $\{V_i, V_j\}$ such that $t(\{V_i, V_j\}) = \mathcal{D}, |\mathcal{D}| = D$, we have $\mathcal{C}(V_i) \cup \mathcal{C}(V_j) = \mathcal{D}$. Then from the code construction (96), the coded symbols (V_i, V_j) do not contain any undesired source symbols $(W_k)_{k\in[K]\setminus\mathcal{D}}$ so that nothing is revealed about the undesired source symbols (note that the source symbols are independent) and security is guaranteed. Regarding correctness, for any qualified edge, from the coded symbols (V_i, V_j) we have $2D$ linear combinations in the $2D$ desired source symbols. That is, the row stack of V_i, V_j produces

$$[V_i; V_j] = \mathbf{H}_{ij} \times (W_k)_{k\in\mathcal{D}} \tag{97}$$

where $\mathbf{H}_{ij} \in \mathbb{F}_q^{2D\times 2D}$ can be obtained from $\mathbf{H}_i$, $\mathbf{H}_j$, $\mathcal{C}(V_i)$, $\mathcal{C}(V_j)$. View the determinant of $\mathbf{H}_{ij}$, $|\mathbf{H}_{ij}|$ as a polynomial in variables $\mathbf{H}_n, n \in [N]$. This determinant polynomial has degree $2D$ and is not a zero polynomial as there exists a realization of $\mathbf{H}_n, n \in [N]$ such that the determinant is not zero. Consider the product of the determinant polynomials for all $|\mathcal{E}|$ qualified edges,

$$\text{poly} \ \triangleq \prod_{i,j:\{V_i,V_j\}\in\mathcal{E}} |\mathbf{H}_{ij}| \tag{98}$$

which is a non-zero polynomial and has degree at most $2D|\mathcal{E}|$. By the Schwartz–Zippel lemma [21], [22], [23], a uniform choice of $\mathbf{H}_n, n \in [N]$ over the finite field $\mathbb{F}_q$ where $q > 2D|\mathcal{E}|$ (the degree of poly) guarantees poly is not always zero. It follows that there exists some realization of $\mathbf{H}_n, n \in [N]$ such that poly $\neq 0$. Then each $|\mathbf{H}_{ij}|$ is not zero and from each qualified edge, we can recover all desired source symbols, i.e., correctness is guaranteed.

VII. Discussion

In this work we have formulated a problem on secure storage under data access and security constraints specified by graphs and considered the maximum storage efficiency - capacity, as the performance metric. We have focused on extremal graphs where the capacity takes extremal values (e.g., maximum with non-trivial security constraints). The extremal graph characterizations obtained in this work are guided by an alignment view that is effective for both code constructions and impossibility claims. For the extremal rates considered, a crucial graphical structure turns out to be 'internal qualified edges', which capture the tension between using the same noise and storing the same coded symbol for security, and diversifying the coded symbols for correctness. While we have focused exclusively on the symmetric rate where each source/coded symbol has the same size, generalizing to sum rate or rate region might reveal additional insights and the characterization might involve more parameters such as the number of nodes, representing an interesting research avenue. Other open problems include relaxing the assumption in Theorem 2 of no common sources to allow common sources and relaxing the symmetric assumption that each edge is associated with the same number of sources to possibly different numbers.

Similar to many challenging open problems in network information theory, allowing arbitrary network topologies often includes intractable problem instances. The perspective we take in this work is to concentrate on extremal networks and study the consequences of the extremal structures. While we have exclusively focuses on networks with extremal rates (and special extremal values), many other choices appear

promising along this line, e.g., shortest/sparest codes under smoothness/locality constraints [24], [25] and might lead to new interesting questions and solutions.

REFERENCES

[1] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, "Protecting data privacy in private information retrieval schemes," in *Proc. 13th Annu. ACM Symp. Theory Comput. (STOC)*, 1998, pp. 151–160.

[2] B. Applebaum, B. Arkis, P. Raykov, and P. N. Vasudevan, "Conditional disclosure of secrets: Amplification, closure, amortization, lower-bounds, and separations," in *Proc. Annu. Int. Cryptol. Conf.* Cham, Switzerland: Springer, 2017, pp. 727–757.

[3] Z. Li and H. Sun, "Conditional disclosure of secrets: A noise and signal alignment approach," *IEEE Trans. Commun.*, vol. 70, no. 6, pp. 4052–4062, Jun. 2022.

[4] Z. Li and H. Sun, "On the linear capacity of conditional disclosure of secrets," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2021, pp. 3202–3207.

[5] Z. Wang and S. Ulukus, "Communication cost of two-database symmetric private information retrieval: A conditional disclosure of multiple secrets perspective," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2022, pp. 402–407.

[6] A. Beimel, "Secret-sharing schemes: A survey," in *Proc. Int. Conf. Coding Cryptol.* Cham, Switzerland: Springer, 2011, pp. 11–46.

[7] H.-M. Sun and S.-P. Shieh, "Secret sharing in graph-based prohibited structures," in *Proc. INFOCOM*, Apr. 1997, pp. 718–724.

[8] M. Soleymani and H. Mahdavifar, "Distributed multi-user secret sharing," *IEEE Trans. Inf. Theory*, vol. 67, no. 1, pp. 164–178, Jan. 2021.

[9] A. Khalesi, M. Mirmohseni, and M. A. Maddah-Ali, "The capacity region of distributed multi-user secret sharing," *IEEE J. Sel. Areas Inf. Theory*, vol. 2, no. 3, pp. 1057–1071, Sep. 2021.

[10] J. Wu, N. Liu, and W. Kang, "The capacity region of distributed multi-user secret sharing under the perfect privacy condition," 2023, *arXiv:2302.03920*.

[11] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[12] N. Cai and R. W. Yeung, "Secure network coding," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2002, p. 323.

[13] N. Cai and T. Chan, "Theory of secure network coding," *Proc. IEEE*, vol. 99, no. 3, pp. 421–437, Mar. 2011.

[14] C. K. Ngai and R. W. Yeung, "Network coding gain of combination networks," in *Proc. Inf. Theory Workshop*, Oct. 2004, pp. 283–287.

[15] S. Maheshwar, Z. Li, and B. Li, "Bounding the coding advantage of combination network coding in undirected networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 570–584, Feb. 2012.

[16] S. Saeedi Bidokhti, V. M. Prabhakaran, and S. N. Diggavi, "Capacity results for multicasting nested message sets over combination networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 4968–4992, Sep. 2016.

[17] A. Salimi, T. Liu, and S. Cui, "Generalized cut-set bounds for broadcast networks," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 2983–2996, Jun. 2015.

[18] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol, "Index coding with side information," *IEEE Trans. Inf. Theory*, vol. 57, no. 3, pp. 1479–1494, Mar. 2011.

[19] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[20] S. Sahraei and M. Gastpar, "GDSP: A graphical perspective on the distributed storage systems," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 2218–2222.

[21] R. A. Demillo and R. J. Lipton, "A probabilistic remark on algebraic program testing," *Inf. Process. Lett.*, vol. 7, no. 4, pp. 193–195, Jun. 1978.

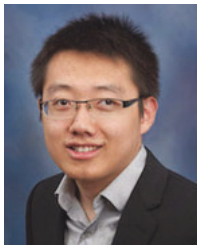
[22] J. T. Schwartz, "Fast probabilistic algorithms for verification of polynomial identities," *J. ACM*, vol. 27, no. 4, pp. 701–717, Oct. 1980.

[23] R. Zippel, "Probabilistic algorithms for sparse polynomials," in *Proc. Int. Symp. Symbolic Algebr. Manipulation*. Cham, Switzerland: Springer, 1979, pp. 216–226.

[24] H. Sun and S. A. Jafar, "On the capacity of locally decodable codes," *IEEE Trans. Inf. Theory*, vol. 66, no. 10, pp. 6566–6579, Oct. 2020.

[25] K. Kazama, A. Kamatsuka, T. Yoshida, and T. Matsushima, "A note on a relationship between smooth locally decodable codes and private information retrieval," in *Proc. Int. Symp. Inf. Theory Appl. (ISITA)*, Oct. 2020, pp. 259–263.

Zhou Li (Graduate Student Member, IEEE) received the B.E. degree in communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently pursuing the Ph.D. degree with the University of North Texas. His current research interests include information theory, network coding, security, and privacy.



Hua Sun (Member, IEEE) received the B.E. degree in communications engineering from the Beijing University of Posts and Telecommunications, China, in 2011, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at Irvine, USA, in 2013 and 2017, respectively.

He is currently an Assistant Professor with the Department of Electrical Engineering, University of North Texas, USA. His current research interests include information theory and its applications to communications, privacy, security, and storage. He was a recipient of the NSF CAREER Award in 2021, the UNT College of Engineering Junior Faculty Research Award in 2021, and the UNT College of Engineering Distinguished Faculty Fellowship in 2023. His coauthored papers received the IEEE Jack Keil Wolf ISIT Student Paper Award in 2016, an IEEE GLOBECOM Best Paper Award in 2016, and the 2020–2021 IEEE Data Storage Best Student Paper Award.