Generic Secure Repair for Distributed Storage

Wentao Huang and **Jehoshua Bruck**
California Institute of Technology, Pasadena, USA
{whuang,bruck}@caltech.edu

Abstract

This paper studies the problem of repairing secret sharing schemes, i.e., schemes that encode a message into n shares, assigned to n nodes, so that any $n-r$ nodes can decode the message but any colluding z nodes cannot infer any information about the message. In the event of node failures so that shares held by the failed nodes are lost, the system needs to be repaired by reconstructing and reassigning the lost shares to the failed (or replacement) nodes. This can be achieved trivially by a trustworthy third-party that receives the shares of the available nodes, recompute and reassign the lost shares. The interesting question, studied in the paper, is how to repair without a trustworthy third-party. The main issue that arises is repair security: how to maintain the requirement that any colluding z nodes, including the failed nodes, cannot learn any information about the message, during and after the repair process? We solve this secure repair problem from the perspective of secure multi-party computation. Specifically, we design generic repair schemes that can securely repair *any* (scalar or vector) linear secret sharing schemes. We prove a lower bound on the repair bandwidth of secure repair schemes and show that the proposed secure repair schemes achieve the optimal repair bandwidth up to a small constant factor when n dominates z, or when the secret sharing scheme being repaired has optimal rate. We adopt a formal information-theoretic approach in our analysis and bounds. A main idea in our schemes is to allow a more flexible repair model than the straightforward one-round repair model implicitly assumed by existing secure regenerating codes. Particularly, the proposed secure repair schemes are simple and efficient two-round protocols.

I. Introduction

The problem of repairing secret sharing schemes has attracted significant interests recently. Specifically, a secret sharing scheme encodes a message into n shares, such that the message can be decoded from any $n-r$ shares (reliability), and that any z shares are independent of the message (security). In the setting of distributed storage, a system consists of n nodes and one share is assigned to each node. Therefore a secret sharing scheme can tolerate r node failures (erasures) as well as z colluding adversarial nodes trying to infer information about the message. In the event of node failures, the shares held by the failed nodes are lost and in order to maintain the same level of reliability, the system needs to repair the failures by reconstructing the lost shares and reassigning them to the failed (or replacement) nodes. Two problems arise during the repair process, namely, 1) bandwidth efficiency: it is desirable to minimize the amount of communication induced by the repair process; 2) repair security: the system needs to maintain the security requirement that any colluding z nodes, including the failed (or replacement) nodes, cannot infer any information about the message, during and after the repair process.

Secure regenerating codes, e.g., [11], [13], [12], [10], [9], [16], [4], are a class of secret sharing schemes that are carefully designed to address the above problems. We classify secure regenerating codes into two categories: codes that only address the bandwidth efficiency problem (i.e., codes with non-secure repair), and codes that address both the bandwidth efficiency and the repair security problems (i.e., codes with secure repair). Specifically, codes with non-secure repair focus on reducing the repair bandwidth without worrying about the security of the repair process. For example, the codes that tolerate Type-I adversary in [16] and the codes in [6] belong to this category. For this case one can think of having a trustworthy repair dealer that will receive information from the available helper nodes, reconstruct the lost share and then forward it to the failed node. The repair dealer may receive enough information to gain knowledge of the message, and therefore has to be trustworthy. In comparison, regenerating codes with secure repair guarantee by code design that such a dealer will not learn any information about the message. This in fact removes the need for the dealer to be trustworthy and the failed node can act as the dealer. Unfortunately, the guarantee that the dealer cannot learn any information about the message is shown to come at a high cost in rate [11], [16], because more independent randomness (keys) is required in order to protect the message from the dealer, resulting in increased overhead. Therefore, codes with non-secure repair in general have a significantly better rate and repair bandwidth (when normalized by rate) than codes with secure repair.

In this paper we address the problem of repair security from a different perspective, without needing to take the heavy penalty in rate and other aspects of efficiency as in the case of secure regenerating codes. The key idea is that we allow a more flexible repair protocol: secure regenerating codes implicitly assume a simple "one-round" repair protocol, in which the helper nodes transmit information to the failed nodes but they themselves do not receive information from other nodes. This implicit "one-round" assumption is expensive in terms of efficiency. We show that, just by slightly relaxing this assumption and allowing a "two-round" protocol, it becomes possible to securely repair *any* secret sharing scheme in a black-box manner, in the sense that the proposed repair protocol is generic and there is no need to design or modify the secret sharing scheme. Refer to Figure 1 for a simple example of the two-round secure repair protocol.

We remark that a two-round protocol is advantageous in that more nodes are allowed to receive information rather than only the failed node. This is intuitively beneficial because, if $d > z$ nodes can receive information, then we can take advantage of the gap between d and z in the following way. During the repair process, let the information received by any z nodes be independent randomness (so that the security requirement is met), and let the information received by all d nodes reveal useful information on the lost share. We then use an extra round of communication to transmit the information on and only on the lost share from the d nodes to the failed node so that the lost share can be reconstructed. Loosely speaking, we can think of the repair process as letting the failed node "compute" its share securely, so that it only learns the share but nothing else. This is naturally related to the problem of secure multi-party computation and the ideas in [1], [2] play an important role in our repair schemes. We remark that we adopt a formal information-theoretic approach in our analysis and bounds, which differs from many existing works on secure multi-party computation. We also note that relaxing the repair process to involve more than one round is practical. For example, POTSHARDS [15] employs a heuristic multi-round repair scheme to improve the security of the repair process.

Our generic secure repair schemes have two important advantages over secure regenerating codes with secure repair. First, the generic nature implies that there is no need to compromise the efficiency of the secret sharing scheme for secure repair. Here, aspects of efficiency at stake are not limited to the rate and repair bandwidth discussed earlier, but also include, for example, computational complexity [5] and decoding bandwidth [8], as it is not clear how to construct secure regenerating codes with optimal computation or decoding bandwidth. Second, most secure regenerating codes focus on secure repair by a fixed number of helper nodes. In the case that not enough helper nodes are available due to multiple node failures, it is not clear how secure repair can be achieved.

We briefly summarize the contributions of the paper. In Section II, we present a generic two-round secure repair scheme based on the ideas in [1], [2]. Specifically, in the first round each helper node encodes its share into $z+1$ pieces using a secret sharing scheme, so that any z pieces reveal no information about the share and that the share can be decoded from $z+1$ pieces. The $z+1$ pieces are sent to $z+1$ receiver nodes, and each receiver node receives a piece from each helper node (if the helper node and the receiver node are the same node, then the corresponding piece needs not be transmitted). For example, in Figure 1-(b), the helper nodes and receiver nodes are both Nodes 2 and 3. The set of pieces received by all receiver nodes contains enough information to decode the shares of all helper nodes and the lost share. We then need to communicate the information about the lost share, but no extra information about the shares of the helper nodes, to the failed node. To achieve this, each receiver node locally computes a function that takes the pieces received by the node as inputs, and outputs a "distilled" piece such that the set of "distilled" pieces only contains information about the lost share. This set is then transmitted from the receiver nodes to the failed node. Refer to Figure 1-(c) for an example.

The generic repair scheme in Section II requires a relatively large repair bandwidth. In Section III, we reduce the repair bandwidth of the scheme significantly by adopting the idea of parallelism in [3]. Instead of repairing one single share at a time, we repair multiple shares together in parallel, therefore amortizing the communication overhead over the multiple shares. This is achieved by letting all n nodes be receiver nodes (instead of $z+1$ nodes) and by using a secret sharing scheme of a higher rate in the first round. The larger gap between the number of receiver nodes and z implies that we can encode more information in the secret sharing scheme (so that it has a higher rate) and can repair more shares in parallel.

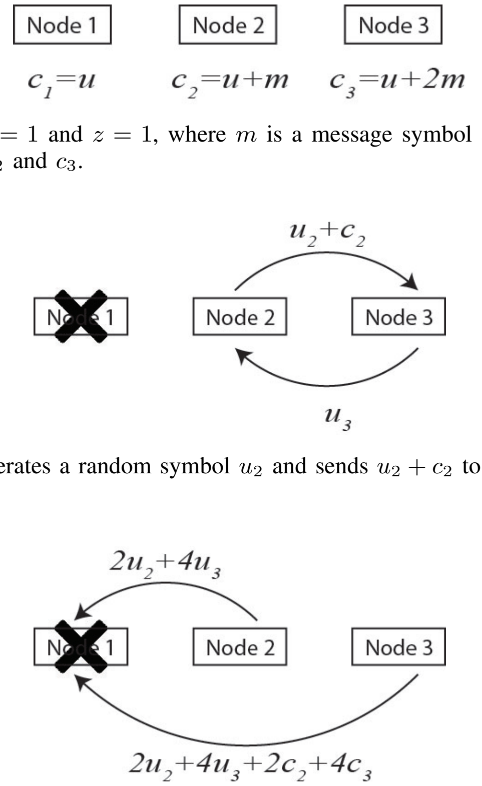
The generic repair schemes in Sections II and III can securely repair any *scalar* linear secret sharing schemes. A more general class of schemes are *vector* linear secret sharing schemes. For a vector linear scheme over a field, each node stores multiple elements of the field instead of a single element as in the scalar linear case. Many efficient secret sharing schemes, e.g., schemes with efficient decoding bandwidth [8], [6], schemes with efficient computation [5], [7], and schemes with efficient repair bandwidth [13], [6], are intrinsically vector linear. In Section IV we generalize our secure repair schemes to generically repair any vector linear schemes. In particular, this generalization allows us to leverage the property of secret sharing schemes with efficient (non-secure) repair bandwidth, i.e., secure regenerating codes with non-secure repair, to further reduce the (secure) repair bandwidth.

Finally, in Section V we prove an information-theoretic lower bound on the repair bandwidth of secure repair schemes. The bound implies that the secure repair schemes in Sections III and IV achieve the optimal repair bandwidth within a small constant factor when n dominates z, or when the secret sharing scheme being repaired has optimal rate.

II. Generic secure repair

Secret sharing schemes address the problem of storing a secret message securely and reliably. Specifically, an (n, k, r, z) secret sharing scheme over $\mathbb{F}_q$ is a randomized function that maps (encodes) a message $\boldsymbol{m} = (m_1, \cdots, m_k)$ of k symbols over $\mathbb{F}_q$ to n shares $\boldsymbol{c} = (c_1, \cdots, c_n)$ over $\mathbb{F}_q$, such that 1) $\boldsymbol{m}$ can be decoded from any subset of $n-r$ shares; 2) any subset of z shares do not reveal information on $\boldsymbol{m}$. Shamir's scheme is a well known secret sharing scheme with $k=1$.

Construction 1. (Shamir's scheme [14]) *For any n, and $z < n$, let $k = 1$, $r = n - z - 1$ and $\mathbb{F}_q$ be a finite field of size $q > n$. Let u_i, $i \in [z]$ be i.i.d. uniformly distributed over $\mathbb{F}_q$ (also referred to as keys) and let α_i, $i \in [n]$ be arbitrary distinct*



(a) A secret sharing scheme over $\mathbb{F}_5$ with $r = 1$ and $z = 1$, where m is a message symbol and u is a random key uniformly distributed over $\mathbb{F}_5$. We denote the three shares by c_1, c_2 and c_3.

(b) Repairing Node 1 (round 1): Node 2 generates a random symbol u_2 and sends $u_2 + c_2$ to Node 3. Node 3 generates a random symbol u_3 and sends it to Node 2.

(c) Repairing Node 1 (round 2): Node 2, having access to u_2 and u_3, computes and sends $2u_2 + 4u_3$ to Node 1. Node 3, having access to $u_2 + c_2$, u_3 and c_3, computes and sends $2u_2 + 4u_3 + 2c_2 + 4c_3$ to Node 1. Node 1 can reconstruct its share since $c_1 = 2c_2 + 4c_3$.

Fig. 1: Securely repairing a secret sharing scheme. Note that it is impossible to securely repair any node failure under the one-round repair model of regenerating codes, because for the failed node to reconstruct its share it has to collect the shares from the other two nodes, which will violate the security requirement. However, any node failure can be securely repaired by the two-round scheme shown above. To see that the scheme is secure, note that after the repair process Node 1 has access to c_1 and $2u_2 + 4u_3$; Node 2 has access to c_2, u_2 and u_3; Node 3 has access to c_3, u_3 and $u_2 + c_2$. Therefore, any single node has access to only one share as well as some random symbols that are independent of the shares. Therefore no single node can learn any information about the message m.

non-zero elements of $\mathbb{F}_q$. The shares corresponding to message m_1 are

$$(c_1, c_2, \cdots, c_n) = (m_1, u_1, u_2, \cdots, u_z) \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^z & \alpha_2^z & \cdots & \alpha_n^z \end{bmatrix}. \tag{1}$$

Lemma 1. *Let c_i, $i \in [n]$ be the shares of Shamir's scheme (1) on message m_1 and keys u_i, $i \in [z]$, and let c'_i, $i \in [n]$ be the shares of the scheme on message m'_1 and keys u'_i, $i \in [z]$. Then for arbitrary linear function $f : \mathbb{F}_q^2 \to \mathbb{F}_q$, $f(c_i, c'_i)$, $i \in [n]$ are the shares of the scheme on message $f(m_1, m'_1)$ and keys $f(u_i, u'_i)$, $i \in [z]$.*

Proof. Follows from the linearity of (1). □

A secret sharing scheme allows secure and reliable storage of information, i.e., it can tolerate the loss of any r shares as well as the exposure of any z shares to an adversary. However, the problem of *repair* is not addressed. Consider the situation that one or more shares are lost or unavailable, and so in order to maintain the same level of reliability one needs to reconstruct the lost shares. For example, in the application of storage, the n shares are assigned to n storage nodes, and in the situation of node failures, one wishes to repair the failures by reconstructing the shares originally assigned to the failed nodes. The repair problem can be easily solved if there is a trusted dealer, who can collect the available shares, recompute the lost shares and reassign them to the failed or replacement nodes. However, the assumption of a trusted dealer responsible for centralized repair may not be practical in many applications.

In this paper we study the situation that a trusted repair dealer is not available and the nodes holding the shares are responsible for carrying out the repair by themselves. A naive approach is to transmit the available shares to the failed node so that it can recompute its share. By doing so, however, information about the message may be leaked to the failed node. Therefore, the main question of interest is *how to repair securely without a trusted dealer*. Below we utilize the ideas developed in secure multi-party computation [1], [2], [3], to design mechanisms to securely repair secret sharing schemes. We first formalize the notion of secure repair. Throughout the paper, for a vector such as $(m_1, \cdots, m_k)$ and an index set I, we denote $\{m_i : i \in I\}$ by m_I.

Definition 1. (Secure repair scheme) *Consider an (n, k, r, z) secret sharing scheme and n nodes such that node i holds the share c_i. For any $e \in [n]$, and $I \subset [n]$, suppose that node e fails and nodes in I are available to help repairing node e. A secure repair scheme is a protocol of communication between the nodes, such that 1) the information sent by a node to other nodes is a function of the share it holds, its local coin flips, and the information it received from other nodes; 2) denote by d_i all the information received by node i from the protocol and denote by u_i the result of coin flips at node i, then*

- *(Repairability) $H(c_e|d_e) = 0$.*
- *(Security) $I(\boldsymbol{m}; c_A, u_A, d_A) = 0$, for all $A \subset [n]$, $|A| = z$.*

Note that Definition 1 naturally extends the threat model of secret sharing, e.g., it maintains the security requirement that any z nodes cannot learn any information about the message, during and after the repair process.

Construction 2. (Generic secure repair) *Consider any (n, k, r, z) secret sharing scheme, any $e \in [n]$, and any $I = \{i_1, \cdots, i_{|I|}\} \subset [n]$, $e \notin I$ such that there exists a linear function f so that $f(c_{i_1}, c_{i_2}, \cdots, c_{i_{|I|}}) = c_e$. Let $J = \{j_1, \cdots, j_{z+1}\}$ be an arbitrary subset of $[n]$ of size $z + 1$. The secure repair process involves three steps:*

1) *For each node $i \in I$, encode c_i into $c_{i,1}, c_{i,2}, \cdots, c_{i,z+1}$ by a $(z+1, 1, 0, z)$ Shamir's scheme (in Construction 1 all nodes choose the same α_i's) and send $c_{i,k}$ to node $j_k \in J$.*
2) *For each node $j \in J$, compute $c'_j = f(c_{i_1,j}, c_{i_2,j}, \cdots, c_{i_{|I|},j})$, and send c'_j to node e.*
3) *Node e obtains c_e by decoding the $(z + 1, 1, 0, z)$ Shamir's scheme, regarding $c'_{j_1}, c'_{j_2}, \cdots, c'_{j_{z+1}}$ as the $z + 1$ shares.*

Theorem 1. *Construction 2 is a secure repair scheme.*

Proof. We need to show that Construction 2 meets the repairability and security requirements in Definition 1. By Lemma 1, $c'_{j_1}, c'_{j_2}, \cdots, c'_{j_{z+1}}$ are the shares of a $(z+1, 1, 0, z)$ Shamir's scheme that encodes $f(c_{i_1}, c_{i_2}, \cdots, c_{i_{|I|}}) = c_e$ as message. This proves repairability. We now focus on security. Let A be an arbitrary set of nodes controlled by the adversary, with $|A| = z$. We consider two cases.

Case 1: $e \notin A$. In this case $d_j = (c_{i_1,j}, c_{i_2,j}, \cdots, c_{i_{|I|},j})$ if $j \in J$, and $d_j = 0$ if $j \notin J$. Denote $c_{A,B} = \{c_{i,j} : i \in A, j \in B\}$, we have

$$\begin{aligned}
I(\boldsymbol{m}; c_A, u_A, d_A) &= I(\boldsymbol{m}; c_A, u_A, c_{I,J\cap A}) && (2)\\
&\overset{(a)}{=} I(\boldsymbol{m}; c_A, u_A, c_{I\backslash A,J\cap A})\\
&\overset{(b)}{=} I(\boldsymbol{m}; c_A, u_A|c_{I\backslash A,J\cap A}) + I(\boldsymbol{m}; c_{I\backslash A,J\cap A})\\
&\overset{(c)}{\leq} I(\boldsymbol{m}; c_A, u_A|c_{I\backslash A,J\cap A}) + I(\boldsymbol{c}; c_{I\backslash A,J\cap A})\\
&\overset{(d)}{=} I(\boldsymbol{m}; c_A, u_A|c_{I\backslash A,J\cap A})\\
&\overset{(e)}{=} I(\boldsymbol{m}; c_A, u_A)\\
&\overset{(f)}{=} I(\boldsymbol{m}; c_A)\\
&\overset{(g)}{=} 0. && (3)
\end{aligned}$$

Here (a) is due to the fact that $c_{I\cap A,J}$ is a function of c_A and $\boldsymbol{u}_A$; (b) follows from the chain rule; (c) follows from the data processing inequality and the Markov chain $\boldsymbol{m} \to \boldsymbol{c} \to c_{I\backslash A,J\cap A}$, i.e., $c_{I\backslash A,J\cap A}$ can be dependent on $\boldsymbol{m}$ only via $\boldsymbol{c}$; (d) follows from the fact that $c_{I\backslash A,J\cap A}$ are the shares of $|I\backslash A|$ independent $(z + 1, 1, 0, z)$ secret sharing schemes and that for each scheme at most $|J \cap A| \leq z$ of its shares are included; (e) follows from the fact that $(\boldsymbol{m}, c_A, u_A) \perp c_{I\backslash A,J\cap A}$, implied by (d); (f) follows from $(\boldsymbol{m}, c_A) \perp u_A$; and (g) follows from security of the secret sharing scheme being repaired.

Case 2: $e \in A$. Since $|A| = z$ and $|J| = z+1$, $J\backslash A$ is not empty. Assume with out loss of generality that $j_1 \in J\backslash A$. Because $c'_{j_1}, c'_{j_2}, \cdots, c'_{j_{z+1}}$ are the shares of a $(z+1, 1, 0, z)$ Shamir's scheme that encodes c_e, it follows that $I(c_e; c'_{j_2}, c'_{j_3}, \cdots, c'_{j_{z+1}}) = 0$ and that there exists a linear function g such that $g(c'_{j_1}, c'_{j_2}, \cdots, c'_{j_{z+1}}) = \sum_{k=1}^{z+1} g_k c'_{j_k} = c_e$. This implies that $g_1 \neq 0$ and so $c'_{j_1} = (c_e - \sum_{k=2}^{z+1} g_k c'_{j_k})g_1^{-1}$, namely,

$$H(c'_{j_1}|c_e, c'_{J\backslash\{j_1\}}) = 0. \qquad (4)$$

We have,

$$\begin{aligned} I(\boldsymbol{m}; c_A, u_A, d_A) &= I(\boldsymbol{m}; c_A, u_A, c'_J, c_{I,A\cap J}) \\ &\overset{(h)}{\leq} I(\boldsymbol{m}; c_A, u_A, c'_J, c_{I,J\backslash\{j_1\}}) \\ &\overset{(i)}{=} I(\boldsymbol{m}; c_A, u_A, c'_{J\backslash\{j_1\}}, c_{I,J\backslash\{j_1\}}) \\ &\overset{(j)}{=} I(\boldsymbol{m}; c_A, u_A, c_{I,J\backslash\{j_1\}}). \end{aligned} \tag{5}$$

Here (h) follows from $A \cap J \subset J\backslash\{j_1\}$; (i) follows from (4); and (j) follows from the fact that $c'_{J\backslash\{j_1\}}$ is a function of $c_{I,J\backslash\{j_1\}}$. We continue the chain of inequality by treating (5) in a similar way as Case 1. Namely, applying an argument similar to that of (2) - (3), we have

$$\begin{aligned} I(\boldsymbol{m}; c_A, u_A, d_A) &\leq I(\boldsymbol{m}; c_A, u_A, c_{I,J\backslash\{j_1\}}) \\ &= I(\boldsymbol{m}; c_A, u_A, c_{I\backslash A,J\backslash\{j_1\}}) \\ &= I(\boldsymbol{m}; c_A, u_A | c_{I\backslash A,J\backslash\{j_1\}}) + I(\boldsymbol{m}; c_{I\backslash A,J\backslash\{j_1\}}) \\ &\leq I(\boldsymbol{m}; c_A, u_A | c_{I\backslash A,J\backslash\{j_1\}}) + I(\boldsymbol{c}; c_{I\backslash A,J\backslash\{j_1\}}) \\ &= I(\boldsymbol{m}; c_A, u_A | c_{I\backslash A,J\backslash\{j_1\}}) \\ &= I(\boldsymbol{m}; c_A, u_A) \\ &= I(\boldsymbol{m}; c_A) \\ &= 0. \end{aligned}$$

The proof is complete. □

We remark that Construction 2 is a generic scheme that can securely repair any linear secret sharing scheme. Particularly, it does not require modifying the secret sharing scheme. In a sense this suggests that secure repair "comes for free" without needing to compromise other aspects of efficiency of the scheme. In comparison, the secure regenerating codes in [11], [13], [12], [16] allow secure repair at the cost of reducing rate. We also remark that multiple failures can be repaired securely by invoking Construction 2 multiple times.

We analyze the repair bandwidth, i.e., the total amount of information that is communicated during the repair process. In Step 1, at most $|I|(z+1)$ symbols are transmitted and in Step 2, at most $z+1$ symbols are transmitted. Therefore the total repair bandwidth is at most $(|I|+1)(z+1)$ symbols, which is approximately $z+1$ times of the non-secure repair bandwidth $|I|$.

III. Reducing the secure repair bandwidth

While Construction 2 provides a generic approach to repair secret sharing schemes securely, it incurs a large overhead in the repair bandwidth. In this section we propose an improved generic secure repair scheme with a significantly better repair bandwidth. The main idea is that, instead of repairing one single share/symbol at a time, we repair multiple shares together in parallel, and therefore amortizing the communication overhead over the multiple shares. For this to work we need every node to store multiple shares, which is typically the case because the amount of information to be stored (e.g., a file) usually exceeds the amount of information that can be stored by a single secret sharing scheme. Therefore the file will be split and stored by multiple independent instances of a secret sharing scheme, resulting in multiple shares to be assigned to a node. In the reminder of the paper we assume that there are enough shares in the failed node to be repaired. Then, the main improvement is that in the first round of the repair scheme, rather than using a low rate $(z+1, 1, 0, z)$ secret sharing scheme, we use a high rate $(n, n-z, 0, z)$ scheme. This allows one to repair $n-z$ shares in parallel and reduce the amortized overhead in the repair bandwidth (which are the z keys in the secret sharing schemes of the first round) by $n-z$ times.

Formally, we assume that each node stores $n-z$ shares from $n-z$ independent instances of a secret sharing scheme. We use superscripts to index instances, e.g., $\boldsymbol{m}^{(i)} = (m_1^{(i)}, \cdots, m_k^{(i)})$ is the message encoded by the i-th instance. In the first round of repair we use a high rate secret sharing scheme defined in Construction 3, which is a generalization of Shamir's scheme to the case of $k > 1$.

Construction 3. (Ramp version of Shamir's scheme [3], [8]) *For any n, r, z such that $n > r + z$, let $k = n - r - z$ and $\mathbb{F}_q$ be a finite field of size $q > n$. Let u_i, $i \in [z]$ be i.i.d. uniformly distributed over $\mathbb{F}_q$ and let α_i, $i \in [n]$ be arbitrary distinct non-zero elements of $\mathbb{F}_q$. The shares corresponding to message $\boldsymbol{m} = (m_1, m_2, \cdots, m_k)$ are*

$$(c_1, c_2, \cdots, c_n) = (m_1, \cdots, m_k, u_1, \cdots, u_z) \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{z+k-1} & \alpha_2^{z+k-1} & \cdots & \alpha_n^{z+k-1} \end{bmatrix}.$$

Construction 3 is an $(n, k = n - r - z, r, z)$ secret sharing scheme [8].

Construction 4. (Bandwidth-efficient secure repair) *Consider any (n, k, r, z) secret sharing scheme, any $e \in [n]$, and any $I = \{i_1, \cdots, i_{|I|}\} \subset [n]$, $e \notin I$ such that there exists a linear function f so that $f(c_{i_1}, c_{i_2}, \cdots, c_{i_{|I|}}) = c_e$. The secure repair process involves three steps:*

1) *For each node $i \in I$, encode $c_i^{(1)}, c_i^{(2)}, \cdots, c_i^{(n-z)}$ into $c_{i,1}, c_{i,2}, \cdots, c_{i,n}$ by a $(n, n-z, 0, z)$ scheme according to Construction 3 (all nodes should choose the same α_i's) and send $c_{i,j}$ to node j.*
2) *For each node $j \in [n]$, compute $c'_j = f(c_{i_1,j}, c_{i_2,j}, \cdots, c_{i_{|I|},j})$, and send c'_j to node e.*
3) *Node e obtains $c_e^{(1)}, c_e^{(2)}, \cdots, c_e^{(n-z)}$ by decoding the $(n, n-z, 0, z)$ scheme, regarding $c'_1, c'_2, \cdots, c'_n$ as the n shares.*

Theorem 2. *Construction 4 is a secure repair scheme.*

Proof. Similar to Theorem 1, repairability follows from the linearity of Construction 3, which implies that $c'_{[n]}$ are the shares of a $(n, n-z, 0, z)$ secret sharing scheme that encodes $(f(c_{i_1}^{(1)}, c_{i_2}^{(1)}, \cdots, c_{i_{|I|}}^{(1)}), \cdots, f(c_{i_1}^{(n-z)}, c_{i_2}^{(n-z)}, \cdots, c_{i_{|I|}}^{(n-z)})) = (c_e^{(1)}, \cdots, c_e^{(n-z)})$ as message. Focusing on security, let $A \subset [n]$, $|A| = z$ be an arbitrary set of nodes controlled by the adversary, then by the property of Construction 3 it follows that $c_e^{[n-z]} \perp c'_A$ and $H(c'_A) = z$. We have

$$\begin{aligned}
H(\boldsymbol{c}'_{[n]\setminus A}|c_e^{[n-z]}, c'_A) &\overset{(a)}{=} H(c_e^{[n-z]}, c'_{[n]}) - H(c_e^{[n-z]}, c'_A) && (6)\\
&\overset{(b)}{\leq} H(c_e^{[n-z]}) + z - H(c_e^{[n-z]}, c'_A)\\
&\overset{(c)}{=} H(c_e^{[n-z]}) + z - H(c_e^{[n-z]}|c'_A) - H(c'_A)\\
&\overset{(d)}{=} H(c_e^{[n-z]}) + z - H(c_e^{[n-z]}) - H(c'_A)\\
&= z - H(c'_A)\\
&\overset{(e)}{=} 0. && (7)
\end{aligned}$$

Here (a) and (c) follows from the chain rule; (b) follows from the fact that $c'_{[n]}$ is a function of $c_e^{[n-z]}$ and z random keys; (d) follows from $c_e^{[n-z]} \perp c'_A$ and (e) follows from $H(c'_A) = z$.

Consider the case that $e \in A$, we have

$$\begin{aligned}
I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, d_A) &= I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, c'_{[n]}, c_{I,A}) && (8)\\
&\overset{(f)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, c'_A, c_{I,A})\\
&\overset{(g)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, c_{I,A}), && (9)\\
&\overset{(h)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, c_{I\setminus A,A})\\
&\overset{(i)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A|c_{I\setminus A,A}) + I(\boldsymbol{m}^{[n-z]}; c_{I\setminus A,A})\\
&\overset{(j)}{\leq} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A|c_{I\setminus A,A}) + I(\boldsymbol{c}^{[n-z]}; c_{I\setminus A,A})\\
&\overset{(k)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A|c_{I\setminus A,A})\\
&\overset{(l)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A)\\
&\overset{(m)}{=} I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]})\\
&\overset{(n)}{=} 0, && (10)
\end{aligned}$$

where (f) follows from (7); (g) follows from the fact that c'_A is a function of $c_{I,A}$; (h) follows from the fact that $c_{A,A}$ is a function of $c_A^{[n-z]}$ and u_A; (i) follows from the chain rule; (j) follows form the Markov chain $\boldsymbol{m}^{[n-z]} \to \boldsymbol{c}^{[n-z]} \to c_{I\setminus A,A}$ and the data processing inequality; (k) follows from the fact that $c_{I\setminus A,A}$ are the shares of $|I\setminus A|$ independent $(n, n-z, 0, z)$ secret sharing schemes and that for each scheme only $|A| = z$ of its shares are included; (l) follows from the fact that $(\boldsymbol{m}^{[n-z]}, c_A^{[n-z]}, u_A) \perp c_{I\setminus A,A}$, implied by (k); (m) follows from $(\boldsymbol{m}^{[n-z]}, c_A^{[n-z]}) \perp u_A$; and (n) follows from security of the secret sharing scheme being repaired.

For the case that $e \notin A$, we have $I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, d_A) = I(\boldsymbol{m}^{[n-z]}; c_A^{[n-z]}, u_A, c_{I,A})$, which can be treated in the same way as (9) - (10). The proof is complete. □

In Step 1, at most $|I|n$ symbols are communicated and in Step 2, at most n symbols are communicated. Therefore the total repair bandwidth is at most $(|I| + 1)n$ symbols, for repairing $n - z$ symbols. The normalized repair bandwidth to repair each symbol is at most $\frac{(|I|+1)n}{n-z}$ symbols. In the case that n dominates z, the normalized repair bandwidth approaches $|I| + 1$

symbols. Note that $|I|$ is the non-secure repair bandwidth and a trivial lower bound on the secure repair bandwidth. Therefore when n dominates z (e.g., the high rate case), the secure repair bandwidth of Construction 4 is essentially optimal. Specifically, it is essentially the same as the non-secure repair bandwidth, implying that we can have secure repair essentially for free, even in terms of repair bandwidth.

IV. Vector linear secure repair

The secure repair schemes in Constructions 2 and 4 deal with scalar secret sharing schemes, i.e., schemes that are linear over a finite field and such that each share is an element of the field. A more general class of secret sharing schemes are vector linear secret sharing schemes, also referred to as array schemes. A vector linear (n, k, r, z) secret sharing scheme over $\mathbb{F}_q^t$ is a randomized function that maps (encodes) a message $\boldsymbol{m} = (m_1, \cdots, m_k)$ of k symbols over $\mathbb{F}_q^t$ to n shares $\boldsymbol{c} = (c_1, \cdots, c_n)$ over $\mathbb{F}_q^t$, such that the encoding function is linear over $\mathbb{F}_q$ and that the same reliability and security requirements as before are met. We denote $m_i = (m_{i,1}, m_{i,2}, \cdots, m_{i,t})$, where $m_{i,j} \in \mathbb{F}_q$, for $i \in [k], j \in [t]$. Similarly we denote $c_i = (c_{i,1}, c_{i,2}, \cdots, c_{i,t})$, for $i \in [n]$. Note that scalar schemes are special cases of vector schemes with $t = 1$.

Many efficient secret sharing schemes, e.g., schemes with efficient decoding bandwidth [8], [6], schemes with efficient computation [5], [7], and schemes with efficient repair bandwidth [13], [6], are intrinsically vector linear. In this section we extend our secure repair framework to vector linear schemes. This is especially interesting because it allows us to leverage the property of secret sharing schemes with efficient (non-secure) repair bandwidth, i.e., secure regenerating codes, to further reduce the (secure) repair bandwidth.

We remark that existing secure regenerating codes can be classified into two categories: codes with non-secure repair and codes with secure repair. Secure regenerating codes with non-secure repair focus on reducing the repair bandwidth without worrying about the security of the repair process. In this case one can think of having a trustworthy repair dealer that will reconstruct the lost share and forward it to the failed node. As remarked previously, during the repair process the dealer may gain information about the message and therefore has to be trustworthy. In comparison, regenerating codes with secure repair, by code design, guarantee that such a dealer will not learn any information about the message. This in fact removes the need for the dealer to be trustworthy and the failed node can act as the dealer. In this sense, secure regenerating codes with secure repair naturally admit a secure repair scheme that meets Definition 1. Particularly, the secure repair scheme is a simple "one-round" scheme in the sense that the helper nodes will transmit information to the failed node but they themselves do not need to receive information from other nodes. Unfortunately, one-round secure repair comes at a high cost in rate and codes with non-secure repair generally have a much better rate as well as repair bandwidth when normalized by rate than codes with secure repair [11], [16]. Our main result in this section implies that this trade-off between rate and secure repair is not necessary: we can apply our generic approach to secure regenerating codes with non-secure repair to achieve secure repair, a good rate, and a good repair bandwidth. The only cost is that the repair process now involves two rounds instead of one round.

Construction 5. (Vector linear secure repair) *Consider any vector linear (n, k, r, z) secret sharing scheme over $\mathbb{F}_q^t$, any $e \in [n]$, and any $I = \{i_1, \cdots, i_{|I|}\} \subset [n]$, $e \notin I$ such that there exists $J \subset [t]$ and a linear function f over $\mathbb{F}_q$ that takes $c_{i,j}$, $i \in I$, $j \in J$ as input and outputs $c_e = (c_{e,1}, c_{e,2}, \cdots, c_{e,t})$. The secure repair process involves three steps:*

1) *For each node $i \in I$, and $j \in J$, encode $c_{i,j}^{(1)}, c_{i,j}^{(2)}, \cdots, c_{i,j}^{(n-z)}$ into $c_{i,j,1}, c_{i,j,2}, \cdots, c_{i,j,n}$ by a $(n, n-z, 0, z)$ scheme according to Construction 3 (all nodes choosing the same $\alpha_1, \alpha_2 \cdots, \alpha_n$) and send $c_{i,j,k}$ to node k.*
2) *For each node $k \in [n]$, compute $(c'_{k,1}, c'_{k,2}, \cdots, c'_{k,t}) = f(c_{i,j,k})_{i \in I, j \in J}$, and send $c'_{k,j}$, $j \in [t]$ to node e.*
3) *For $j \in [t]$, node e obtains $c_{e,j}^{(1)}, c_{e,j}^{(2)}, \cdots, c_{e,j}^{(n-z)}$ by decoding the $(n, n-z, 0, z)$ scheme, regarding $c'_{1,j}, c'_{2,j}, \cdots, c'_{n,j}$ as the n shares.*

Theorem 3. *Construction 5 is a secure repair scheme.*

Proof. As in Theorem 2, repairability follows from the linearity of Construction 3, which implies that $c'_{1,j}, c'_{2,j}, \cdots, c'_{n,j}$ are the shares of a $(n, n-z, 0, z)$ secret sharing scheme that encodes $c_{e,j}^{(1)}, c_{e,j}^{(2)}, \cdots, c_{e,j}^{(n-z)}$ as message, for $j \in [t]$. We now turn to security, and follow a similar flow as Theorem 2. Let $A \subset [n]$, $|A| = z$ be an arbitrary set of nodes controlled by the adversary, then by the property of Construction 3 it follows that $c_e^{[n-z]} \perp c'_{A,j} = 0$ and $H(c'_{A,j}) = z$, for $j \in [t]$. We have, for $j \in [t]$,

$$\begin{aligned} H(c'_{[n]\backslash A,j} | c_{e,j}^{[n-z]}, c'_{A,j}) &= H(c_{e,j}^{[n-z]}, c'_{[n],j}) - H(c_{e,j}^{[n-z]}, c'_{A,j}) \\ &\leq H(c_{e,j}^{[n-z]}) + z - H(c_{e,j}^{[n-z]}, c'_{A,j}) \\ &= H(c_{e,j}^{[n-z]}) + z - H(c_{e,j}^{[n-z]} | c'_{A,j}) - H(c'_{A,j}) \\ &= H(c_{e,j}^{[n-z]}) + z - H(c_{e,j}^{[n-z]}) - H(c'_{A,j}) \\ &= z - H(c'_{A,j}) \\ &= 0, \end{aligned} \tag{11}$$

where the justification for the steps are similar to that of (6) - (7). (11) implies that

$$H(c'_{[n]\backslash A,[t]} | c^{[n-z]}_{e,[t]}, c'_{A,[t]}) = 0. \tag{12}$$

Now consider the case that $e \in A$, we have

$$\begin{aligned}
I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, d_A) &= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, c'_{[n],[t]}, c_{I,J,A}) \\
&\overset{(a)}{=} I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, c'_{A,[t]}, c_{I,J,A}) \\
&= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, c_{I,J,A}),
\end{aligned} \tag{13}$$

$$\begin{aligned}
&= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, c_{I\backslash A,J,A}) \\
&= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A | c_{I\backslash A,J,A}) + I(\boldsymbol{m}^{[n-z]}; c_{I\backslash A,J,A}) \\
&\leq I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A | c_{I\backslash A,J,A}) + I(\boldsymbol{c}^{[n-z]}; c_{I\backslash A,J,A}) \\
&\overset{(b)}{=} I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A | c_{I\backslash A,J,A}) \\
&= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A) \\
&= I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}) \\
&= 0,
\end{aligned} \tag{14}$$

where (a) follows from (12); (b) follows from the fact that $c_{I\backslash A,J,A}$ are the shares of $|I\backslash A| \cdot |J|$ independent $(n, n-z, 0, z)$ secret sharing schemes and that for each scheme only $|A| = z$ of its shares are included; and the remaining equalities/inequalities are similar to (8) - (10).

For the case that $e \notin A$, we have $I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, d_A) = I(\boldsymbol{m}^{[n-z]}; c^{[n-z]}_{A,[t]}, u_A, c_{I,J,A})$, which can be treated in the same way as (13) - (14). The proof is complete. □

Consider the repair bandwidth of the scheme. In Step 1, at most $n|I||J|$ symbols (over $\mathbb{F}_q$) are transmitted and in Step 2, at most nt symbols are transmitted. Therefore, the total repair bandwidth is at most $(|I||J| + t)n$ symbols, for repairing $(n-z)t$ symbols. The normalized repair bandwidth to repair each symbol is at most $\frac{(|I||J|+t)n}{(n-z)t}$ symbols. In the case that n dominates z, the normalized repair bandwidth approaches $\frac{|I||J|}{t} + 1$. Note that the normalized non-secure repair bandwidth is $\frac{|I||J|}{t}$, and therefore in this case the the secure repair bandwidth of Construction 5 is essentially optimal and is almost the same as the non-secure repair bandwidth.

The MSR secure regenerating codes in [13], [6] have optimal rate as well as optimal non-secure repair bandwidth (among rate-optimal schemes). Specifically, the rate of the scheme is $\frac{n-k-z}{n}$, and that for $|I|$ helper nodes to non-securely repair a failed node, each helper node will transmit $1/(1 + |I| - k - z)$ fraction of the symbols it stores, i.e., $|J| = \frac{t}{1+|I|-k-z}$. By applying Construction 5 to these codes, we obtain schemes with optimal rate and low secure repair bandwidth. In the next section we will show that the secure repair bandwidth is in fact optimal up to a small constant factor.

V. Lower bound on secure repair bandwidth

The bandwidths of Construction 4 and Construction 5 are significantly better than Construction 2. A natural question is whether it is possible to do even better, or in other words, what is a lower bound on the secure repair bandwidth. As we previously remarked, when n dominates z, the bandwidths of Constructions 4 and 5 approach the non-secure repair bandwidth, which is a naive lower bound. Therefore in this case the bandwidths of Constructions 4 and 5 are asymptotically optimal. In this section, we prove a stronger lower bound on the secure repair bandwidth and show that the bandwidths of Constructions 4 and 5 are optimal for all parameters up to a constant factor of 2, as long as the secret sharing scheme being repaired is rate-optimal.

Assume that a trustworthy repair dealer is available. The dealer will receive information from the helper nodes, evaluate a *repair function* that outputs the lost share, and reassign the share. In this case, the repair bandwidth is the size of the input to the repair function plus the size of the lost share. Now consider the situation that a trustworthy dealer is not available and a secure repair scheme is used for repair. The secure repair scheme essentially is a method to evaluate the repair function (e.g., f in Constructions 2, 4 and 5) securely at the failed node, and the repair bandwidth again depends on the size of the input to the repair function. The repair function is an intrinsic component of the secret sharing scheme and the size of the input can be minimized by carefully designing the secret sharing scheme, e.g., [13], [6]. Refer to the size of the input to the repair function as the non-secure repair bandwidth of a secret sharing scheme. Below we prove a lower bound on the repair bandwidth of secure repair schemes, given the non-secure repair bandwidth of the secret sharing scheme.

Theorem 4. *For any rate-optimal $(n, k = n - r - z, r, z)$ secret sharing scheme, let W be the non-secure repair bandwidth of the scheme, then a secure repair scheme requires a bandwidth of at least $\frac{(n-1)W}{2(n-z-1)}$.*

Proof. Note that $k = n - r - z$ implies that the scheme is rate-optimal [8]. Let the message $\boldsymbol{m} = (m_1, m_2, \cdots, m_k)$ be uniformly distributed. Then for any $I \subset [n]$, $|I| = k + z$ and $J \subset I$, $|J| = z$, by the security and the decodability of the scheme we have $I(\boldsymbol{m}; c_J) = 0$ and $I(\boldsymbol{m}; c_I) = H(\boldsymbol{m}) = k$. It follows that

$$\begin{aligned} I(\boldsymbol{m}; c_{I\backslash J}|c_J) &= H(\boldsymbol{m}|c_J) - H(\boldsymbol{m}|c_I) \\ &= H(\boldsymbol{m}) - H(\boldsymbol{m}|c_I) \\ &= I(\boldsymbol{m}; c_I) \\ &= k. \end{aligned} \tag{15}$$

Since $|I\backslash J| = k$, $H(c_{I\backslash J}) \leq k$, and hence (15) implies that $H(c_{I\backslash J}) = k$ and $c_{I\backslash J} \perp c_J$ and that

$$H(c_{I\backslash J}|\boldsymbol{m}, c_J) = 0. \tag{16}$$

Therefore among the n shares of the secret sharing scheme, any $|I\backslash J| = k$ shares are uniformly distributed and that any $|J| = z$ shares are independent of any other k shares. This in turn implies that any $k + z$ shares are uniformly distributed, i.e.,

$$H(c_I) = k + z. \tag{17}$$

Assume that c_e is lost, and for $i \in [n]\backslash\{e\}$, let w_i be the information sent by node i to node e for non-secure repair, namely, the input signal to the repair function from node i (with the convention that $w_i = 0$ if node i does not participate in the repair). Then w_i is a function of c_i and $\sum_{i\in[n]\backslash\{e\}} H(w_i) = W$. Now consider any secure repair protocol, and for $i \in [n]\backslash\{e\}$, $j \in [n]$, let $v_{i,j}$ be the set of signals that are sent to node j by node i or sent to node i by node j during the protocol (with the convention that $v_{i,i} = \emptyset$). Then w_i must be a function of the signals incoming to and outgoing from node i, namely, $H(w_i|v_{i,[n]}) = 0$, implying that

$$I(w_i; v_{i,[n]}) = H(w_i). \tag{18}$$

Let A be an arbitrary set of nodes controlled by the adversary such that $i \notin A$, $|A| = z$, and let B be an arbitrary set of nodes such that $i \in B$, $|B| = k$, $A \cap B = \emptyset$. We have

$$\begin{aligned} I(w_i; v_{i,A}) &= H(w_i) - H(w_i|v_{i,A}) \\ &\overset{(a)}{=} H(w_i|c_A) - H(w_i|v_{i,A}) \\ &\leq H(w_i|c_A) - H(w_i|v_{i,A}, c_A) \\ &= I(w_i; v_{i,A}|c_A) \\ &\overset{(b)}{\leq} I(c_i; v_{i,A}|c_A) \\ &\leq I(c_B; v_{i,A}|c_A) \\ &\overset{(c)}{=} I(\boldsymbol{m}; v_{i,A}|c_A) \\ &\overset{(d)}{=} I(\boldsymbol{m}; v_{i,A}|c_A) + I(\boldsymbol{m}; c_A) \\ &= I(\boldsymbol{m}; v_{i,A}, c_A) \\ &\overset{(e)}{=} 0. \end{aligned} \tag{19}$$

Here (a) follows from the fact that w_i is a function of c_i and by (17), $c_i \perp c_A$; (b) follows from the data processing inequality and the fact that w_i is a function of c_i; (c) follows from the data processing inequality and (16), i.e., c_B is a function of $\boldsymbol{m}$ given c_A; (d) follows from the security of the secret sharing scheme; and (e) follows from the security of the repair scheme. Let

$$A^* = \operatorname*{argmax}_{A\subset[n]\backslash\{i\}, |A|=z} \sum_{l\in A} H(v_{i,l}),$$

and let $\bar{A}^* = [n]\backslash(\{i\} \cup A^*)$, then for $j \in \bar{A}^*$ and $j^* \in A^*$, $H(v_{i,j}) \leq H(v_{i,j^*})$. We have

$$\begin{aligned} H(v_{i,\bar{A}^*}) &\geq I(w_i; v_{i,\bar{A}^*}|v_{i,A}) \\ &\overset{(f)}{=} I(w_i; v_{i,\bar{A}^*}|v_{i,A}) + I(w_i; v_{i,A}) \\ &= I(w_i; v_{i,[n]}) \\ &\overset{(g)}{=} H(w_i), \end{aligned}$$

where (f) follows from (19) and (g) follows from (18). Therefore there exist $j \in \bar{A}^*$ such that $H(v_{i,j}) \geq H(w_i)/|\bar{A}^*|$ and so for $j^* \in A^*$, $H(v_{i,j^*}) \geq H(w_i)/(n-z-1)$. Therefore the amount of information transmitted and received by node i is lower bounded by

$$\begin{aligned}\sum_{j\in[n]} H(v_{i,j}) &\geq H(v_{i,\bar{A}^*}) + |A^*|\frac{H(w_i)}{n-z-1} \\ &= \frac{(n-1)H(w_i)}{n-z-1}. \end{aligned} \tag{20}$$

Summing (20) over all $i \in [n]\backslash\{e\}$, it follows that the amount of information transmitted and received by nodes in $[n]\backslash\{e\}$ is at least $\frac{(n-1)W}{n-z-1}$. Since the amount of communication is counted exactly twice, i.e., when information is transmitted and when it is received, the repair bandwidth of the scheme is lower bounded by $\frac{(n-1)W}{2(n-z-1)}$. This completes the proof. □

The bandwidths of Constructions 4 and 5 are upper bounded by $\frac{(W+1)n}{n-z}$, and therefore are optimal up to a factor of approximately 2 by Theorem 4.

VI. Concluding Remarks

This paper studies the problem of repairing lost shares of a secret sharing scheme without a trustworthy repair dealer. We design generic repair schemes that can securely repair *any* (scalar or vector) linear secret sharing schemes. We prove a lower bound on the repair bandwidth of secure repair schemes and show that the proposed secure repair schemes achieve the optimal repair bandwidth up to a small constant factor when n dominates z, or when the secret sharing scheme being repaired has optimal rate.

An interesting open problem is to study the secure repair bandwidth under the general repair model when the secret sharing scheme being repaired is not rate-optimal. More generally, while the tradeoff between repair bandwidth and rate has attracted significant interests under the one-round repair model, under the general repair model whether a tradeoff exists or not and how to characterize it remain open. Another interesting open problem is to study secure repair in the presence of active adversarial nodes that may deviate from the prescribed repair protocol.

References

[1] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," in *ACM symposium on Theory of computing*, 1988, pp. 1–10.

[2] D. Chaum, C. Crpeau, and I. Damgrd, "Multiparty unconditionally secure protocols," in *ACM symposium on Theory of computing*, 1988, pp. 11–19.

[3] M. Franklin and M. Yung, "Communication complexity of secure computation," in *ACM symposium on Theory of computing*, 1992, pp. 699 – 710.

[4] K. Huang, U. Parampalli, and M. Xian, "On secrecy capacity of minimum storage regenerating codes," *IEEE Transactions on Information Theory*, vol. 63, no. 3, pp. 1510 – 1524, 2017.

[5] W. Huang and J. Bruck, "Secure RAID schemes for distributed storage," in *IEEE International Symposium on Information Theory (ISIT)*, 2016.

[6] W. Huang and J. Bruck, "Secret sharing with optimal decoding and repair bandwidth," in *IEEE International Symposium on Information Theory (ISIT)*, 2017.

[7] W. Huang and J. Bruck, "Secure RAID schemes from EVENODD and STAR codes," in *IEEE International Symposium on Information Theory (ISIT)*, 2017.

[8] W. Huang, M. Langberg, J. Kliewer, and J. Bruck, "Communication efficient secret sharing," *IEEE Transactions on Information Theory*, vol. 62, no. 12, pp. 7195–7206, 2016.

[9] S. Kadhe and A. Sprintson, "Weakly secure regenerating codes for distributed storage," in *International Symposium on Network Coding*, 2014.

[10] O. O. Koyluoglu, A. S. Rawat, and S. Vishwanath, "Secure cooperative regenerating codes for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5228 – 5244, 2014.

[11] S. Pawar, S. E. Rouayheb, and K. Ramchandran, "Securing dynamic distributed storage systems against eavesdropping and adversarial attacks," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6734–6753, 2011.

[12] A. S. Rawat, O. O. Koyluoglu, N. Silberstein, and S. Vishwanath, "Optimal locally repairable and secure codes for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 60, no. 1, pp. 212 – 236, 2014.

[13] N. B. Shah, K. V. Rashmi, and P. V. Kumar, "Information-theoretically secure regenerating codes for distributed storage," in *IEEE GLOBECOM 2011*. IEEE, 2011, pp. 1–5.

[14] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[15] M. W. Storer, K. M. Greenan, E. L. Miller, and K. Voruganti, "Potshards - a secure, recoverable, long-term archival storage system," *ACM Transactions on Storage*, vol. 5, no. 2, pp. 1–35, 2009.

[16] R. Tandon, S. Amuru, T. C. Clancy, and R. M. Buehrer, "Toward optimal secure distributed storage systems with exact repair," *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3477–3492, 2016.