Alphabet-Dependent Bounds for Locally Repairable Codes With Joint Information Availability

Jung-Hyun Kim, *Student Member, IEEE*, and Hong-Yeop Song, *Senior Member, IEEE*

***Abstract*— In this letter, we investigate a class of codes with the following property: any decodable set of erased symbols can be repaired from any single set of several disjoint symbol sets with small cardinality. We refer such codes to locally repairable codes (LRCs) with joint availability. In particular, if information symbols of a code have this property, then we refer the code to an LRC with joint information availability. We propose two alphabet-dependent bounds for LRCs with joint information availability. From the bounds, we rederive some well-known bounds for LRCs. Based on the relation between LRCs and batch codes, we also present an alternative proof of an existing bound for batch codes. Finally, we show the achievability and tightness of the proposed bounds using graph-based codes.**

***Index Terms*— Distributed storage systems, locally repairable codes, locality, availability, alphabet-dependent bounds.**

I. Introduction

Locally repairable codes (LRCs) [1] are a class of codes designed for the local correction of erasures. They have attracted a lot of interest in recent years due to their applications in distributed storage systems. A code is called an LRC with locality r_1 if every symbol is recoverable from at most r_1 other symbols. If information symbols of a code have this property then the code is called an LRC with information locality r_1. Lots of researchers have studied the locality and its general notions in [2]–[5], [8], and [9]. In particular, Rawat *et al.* [3] generalized the locality for one symbol to the locality for $l \geqslant 1$ symbols. In this letter, we call this l-locality r_l. Kim *et al.* [4], [5], [8] suggested joint locality which consider multiple values of l instead of a single value of l.

In addition to the locality, another important property of LRCs is availability [6]. A symbol of a code is said to have availability (r_1, t_1) if it can be recovered from t_1 disjoint repair sets of other symbols, each set of size at most r_1. LRCs with availability (r_1, t_1) ensure $t_1 + 1$ parallel reads for each symbol, which is appealing in distributed storage systems containing so-called *hot data* that is frequently and simultaneously accessed by many users.

In this letter, we extend the availability for one symbol into the availability for multiple symbols, and then, also define joint availability of codes. Based on the new notion of availability, we propose two alphabet-dependent bounds. Interestingly, it is possible to deduce some well-known bounds for LRCs from the proposed bounds. Moreover, using our bound, we present an alternative proof of an existing bound for batch codes [11] which were proposed for load balancing in distributed server systems. Finally, we show the achievability and tightness of the proposed bounds using graph-based codes.

The organization of the letter is as follows. In Section II, we define joint availability for LRCs and propose alphabet-dependent bounds for LRCs with joint information availability. In Section III, we present a connection to batch codes and rederive a well-known bound for batch codes using the proposed bound. In Section IV, we show the achievability and tightness of our bounds. In Section V, we conclude the letter.

Manuscript received March 8, 2017; revised April 10, 2017; accepted April 25, 2017. Date of publication April 27, 2017; date of current version August 10, 2017. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2013R1A1A2062061). The associate editor coordinating the review of this letter and approving it for publication was J. Li. *(Corresponding author: Hong-Yeop Song.)*

The authors are with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, South Korea (e-mail: hysong@yonsei.ac.kr).

Digital Object Identifier 10.1109/LCOMM.2017.2699968

II. Alphabet-Dependent Bounds for LRCs With Joint Information Availability

We first define the availability for multiple symbols, joint availability, and joint information availability. Then, we propose alphabet-dependent bounds for LRCs with joint information availability. From the bounds, we rederive some well-known alphabet-dependent bounds and Singleton-like bounds without the alphabet constraint.

In the rest of this letter, we use the notation $(n, k, d)_q$ to denote parameters of a code of length n, cardinality q^k, i.e., dimension k, minimum distance d over alphabet $\mathcal{Q}$, $|\mathcal{Q}| = q$, for both linear and non-linear cases. For a positive integer $z \in \mathbb{Z}^+$, we denote $[z] \triangleq \{1, 2, \ldots, z\}$.

Definition 1: Let $\mathcal{C}$ be an $(n, k, d)_q$ code. For a positive integer $l \in [d-1]$, a symbol set $\mathcal{E}$ of $\mathcal{C}$, $|\mathcal{E}| = l$, is said to have availability (r_l, t_l) if $\mathcal{E}$ can be recovered from any single set of t_l disjoint symbol sets indexed by $\mathcal{R}_1(\mathcal{E}), \mathcal{R}_2(\mathcal{E}), \ldots, \mathcal{R}_{t_l}(\mathcal{E}) \subseteq [n] \setminus \mathcal{E}$ such that $|\mathcal{R}_j(\mathcal{E})| \leqslant r_l$, for all $j \in [t_l]$. □

Definition 2: Let $\mathcal{C}$ be an $(n, k, d)_q$ code. A set of availabilities $\{(r_l, t_l) : l \in [d-1]\}$ is called joint availability of $\mathcal{C}$ if every symbol set of cardinality l of $\mathcal{C}$ has the availability (r_l, t_l). □

Definition 3: Let $\mathcal{C}$ be an $(n, k, d)_q$ code. A set of availabilities $\{(r_l, t_l) : l \in [d-1]\}$ is called joint information availability of $\mathcal{C}$ if every information symbol set of cardinality l of $\mathcal{C}$ has the availability (r_l, t_l). □

Now, we introduce our bounds for LRCs with joint information availability. We note that, since an LRC with joint availability is also an LRC with joint information availability, the proposed bounds also hold for LRCs with joint availability.

Theorem 1: For an $(n, k, d)_q$ code $\mathcal{C}$ that has joint information availability $\{(r_l, t_l) : l \in [d-1]\}$, we have

$$k \leqslant \min_{\substack{z \in \mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j \in [d-1], 1 \leqslant j \leqslant z\} \\ \boldsymbol{y}=\{y_j \in [t_{l_j}], 1 \leqslant j \leqslant z\} \\ A(\boldsymbol{l},\boldsymbol{y})<k}} \left[A(\boldsymbol{l}, \boldsymbol{y}) + k_{opt}^{(q)}(n - B(\boldsymbol{l}, \boldsymbol{y}),\ d) \right],$$

where

$$A(\boldsymbol{l},\boldsymbol{y}) = \sum_{j=1}^{z}((r_{l_j}-1)y_j+1), \quad B(\boldsymbol{l},\boldsymbol{y}) = \sum_{j=1}^{z}(r_{l_j}y_j+l_j),$$

and $k_{opt}^{(q)}(n',d')$ *is the largest possible dimension of a* q*-ary code of length* n' *and minimum distance* d'*.*

Proof: We follow similar steps to the proofs in [8] and [9]. We define $H(\mathbf{Y}_I) = \log_q |\{\mathbf{Y}_I : \mathbf{Y} \in \mathcal{C}\}|$ where $\mathbf{Y}_I$ is the restriction of $\mathbf{Y}$ to a subset of coordinates $I \subset [n]$. First of all, we show that, for any positive integer z satisfying $A(\boldsymbol{l},\boldsymbol{y}) < k$, there exists a set I such that $|I| = B(\boldsymbol{l},\boldsymbol{y})$ and $H(\mathbf{Y}_I) \leqslant A(\boldsymbol{l},\boldsymbol{y})$. To prove this, we construct the set I using the following algorithm. In the algorithm, without loss of generality, we assume that the first k symbols of the code $\mathcal{C}$ form an information symbol set.

Algorithm 1 Construction of I

1: Let $j=1$ and $I_0=\emptyset$.
2: **while** $H(\mathbf{Y}_{I_{j-1}}) < k$ **do**
3: Pick $\mathcal{E}_j \subset [k] \setminus I_{j-1}$, $|\mathcal{E}_j| = l_j$ such that $H(\mathbf{Y}_{I_{j-1}}) < H(\mathbf{Y}_{I_{j-1}\cup\mathcal{E}_j})$.
4: Pick $y_j \leqslant t_{l_j}$ disjoint repair sets of $\mathcal{E}_j$, $\mathcal{R}_1(\mathcal{E}_j)$, $\mathcal{R}_2(\mathcal{E}_j),\ldots,\mathcal{R}_{y_j}(\mathcal{E}_j)$, such that the cardinality of each repair set is at most r_{l_j}.
5: Set $\mathcal{S}(\mathcal{E}_j,y_j) = \mathcal{E}_j \cup \mathcal{R}_1(\mathcal{E}_j) \cup \mathcal{R}_2(\mathcal{E}_j) \cup \cdots \cup \mathcal{R}_{y_j}(\mathcal{E}_j)$.
6: **if** $H(\mathbf{Y}_{I_{j-1}\cup\mathcal{S}(\mathcal{E}_j,y_j)}) < k$ **then**
7: $I_j = I_{j-1} \cup \mathcal{S}(\mathcal{E}_j,y_j)$.
8: **end if**
9: Increment j.
10: **end while**
11: Set $z=j-1$.
12: Pick $\mathcal{T}_z \subset [n]\setminus I_z$ such that $|\mathcal{T}_z| = \sum_{m=1}^{z}(r_{l_m}y_m+l_m)-|I_z|$ and $H(\mathbf{Y}_{I_z\cup\mathcal{T}_z}) < k$.
13: Set $I = I_z \cup \mathcal{T}_z$.

As a result of the algorithm, the size of I finally becomes $\sum_{j=1}^{z}(r_{l_j}y_j+l_j)$. To show $H(\mathbf{Y}_I) \leqslant A(\boldsymbol{l},\boldsymbol{y})$, we first define a set Ψ_j as follows:

$$\Psi_j = \cup_{m=1}^{y_j-1}\psi_m \cup \mathcal{E}_j,$$

where $\psi_m \in \mathcal{R}_m(\mathcal{E}_j) \setminus I_{j-1}$ for $1 \leqslant j \leqslant z$. We can always find such element ψ_m by the selection rule in step 3-4. Then, we have

$$\begin{aligned} H(\mathbf{Y}_I) &= H(\mathbf{Y}_{I\setminus\cup_{j=1}^{z}\Psi_j}) \leqslant \left|I \setminus \cup_{j=1}^{z}\Psi_j\right| \overset{(a)}{=} |I| - \sum_{j=1}^{z}|\Psi_j| \\ &= \sum_{j=1}^{z}(r_{l_j}y_j+l_j) - \sum_{j=1}^{z}(l_j+(y_j-1)) = A(\boldsymbol{l},\boldsymbol{y}) < k. \end{aligned} \tag{1}$$

We note that (a) follows from $\Psi_j \cap I_{j-1} = \emptyset$ for $1 \leqslant j \leqslant z$, and thus, $\Psi_1, \Psi_2, \ldots, \Psi_z$ are disjoint.

Now, it is sufficient to show that, for such a set I, there always exists an $(n-|I|, k-A(\boldsymbol{l},\boldsymbol{y}), d)_q$ code. Consider a codeword $\mathbf{Z} \in \mathcal{Z} \triangleq \{\mathbf{Y}_I : \mathbf{Y} \in \mathcal{C}\}$. We denote a codebook $\tilde{\mathcal{C}}(\mathbf{Z}) = \{\mathbf{Y}_{[n]\setminus I} : \mathbf{Y}_I = \mathbf{Z}\}$. The length of a codeword in $\tilde{\mathcal{C}}(\mathbf{Z})$ is $n-|I|$. It is not difficult to show that codewords in $\tilde{\mathcal{C}}(\mathbf{Z})$ have the minimum distance at least d. We first show that there exists at least one $\mathbf{Z} \in \mathcal{Z}$ such that $|\tilde{\mathcal{C}}(\mathbf{Z})| \geqslant q^{k-A(\boldsymbol{l},\boldsymbol{y})}$. Assume on the contrary that $|\tilde{\mathcal{C}}(\mathbf{Z})| < q^{k-A(\boldsymbol{l},\boldsymbol{y})}$ for every $\mathbf{Z} \in \mathcal{Z}$. Then, we have

$$\begin{aligned} \sum_{\mathbf{Z}\in\mathcal{Z}} |\tilde{\mathcal{C}}(\mathbf{Z})| &= |\mathcal{C}| = q^k < |\mathcal{Z}| \cdot q^{k-A(\boldsymbol{l},\boldsymbol{y})} \\ &= q^{H(\mathbf{Y}_I)} \cdot q^{k-A(\boldsymbol{l},\boldsymbol{y})} = q^{k+(H(\mathbf{Y}_I)-A(\boldsymbol{l},\boldsymbol{y}))}. \end{aligned} \tag{2}$$

Since $H(\mathbf{Y}_I) - A(\boldsymbol{l},\boldsymbol{y}) \leqslant 0$ by (1), the equation (2) violates the assumption. Thus, there is at least one $\mathbf{Z} \in \mathcal{Z}$ such that $|\tilde{\mathcal{C}}(\mathbf{Z})| \geqslant q^{k-A(\boldsymbol{l},\boldsymbol{y})}$, thereby resulting in an $(n-|I|, k-A(\boldsymbol{l},\boldsymbol{y}), d)_q$ code. Since, for all the possible $\boldsymbol{l}$ and $\boldsymbol{y}$, $k-A(\boldsymbol{l},\boldsymbol{y})$ is less than or equal to $k_{opt}^{(q)}(n-|I|,d)$, we obtain

$$k \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ \boldsymbol{y}=\{y_j\in[t_{l_j}],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l},\boldsymbol{y})<k}} \left[A(\boldsymbol{l},\boldsymbol{y}) + k_{opt}^{(q)}(n-|I|,\ d)\right].$$

By defining $B(\boldsymbol{l},\boldsymbol{y}) = |I|$, we complete the proof. ■

Theorem 2: For an $(n,k,d)_q$ *code* $\mathcal{C}$ *that has joint information availability* $\{(r_l,t_l) : l \in [d-1]\}$*, we have*

$$d \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ \boldsymbol{y}=\{y_j\in[t_{l_j}],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l},\boldsymbol{y})<k}} \left[d_{opt}^{(q)}(n-B(\boldsymbol{l},\boldsymbol{y}), k-A(\boldsymbol{l},\boldsymbol{y}))\right],$$

where

$$A(\boldsymbol{l},\boldsymbol{y}) = \sum_{j=1}^{z}((r_{l_j}-1)y_j+1), \quad B(\boldsymbol{l},\boldsymbol{y}) = \sum_{j=1}^{z}(r_{l_j}y_j+l_j),$$

and $d_{opt}^{(q)}(n',k')$ *is the largest possible minimum distance of a* q*-ary code of length* n' *and dimension* k'*.*

Proof: Recall the proof of Theorem 1. For any positive integer z satisfying $A(\boldsymbol{l},\boldsymbol{y}) < k$, there always exists a set I such that $|I| = B(\boldsymbol{l},\boldsymbol{y})$ and $H(\mathbf{Y}_I) \leqslant A(\boldsymbol{l},\boldsymbol{y})$. For such a set I, consider a codeword $\mathbf{Z} \in \mathcal{Z} \triangleq \{\mathbf{Y}_I : \mathbf{Y} \in \mathcal{C}\}$ and the corresponding codebook $\tilde{\mathcal{C}}(\mathbf{Z}) = \{\mathbf{Y}_{[n]\setminus I} : \mathbf{Y}_I = \mathbf{Z}\}$. Let $\tilde{d}(\mathbf{Z})$ be the minimum distance of codewords in $\tilde{\mathcal{C}}(\mathbf{Z})$. Then, we have

$$d \leqslant \tilde{d}(\mathbf{Z}) \leqslant d_{opt}^{(q)}(n-|I|,\ k-H(\mathbf{Y}_I)).$$

Since, for all the possible $\boldsymbol{l}$ and $\boldsymbol{y}$, $H(\mathbf{Y}_I) \leqslant A(\boldsymbol{l},\boldsymbol{y})$, we obtain

$$d \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ \boldsymbol{y}=\{y_j\in[t_{l_j}],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l},\boldsymbol{y})<k}} \left[d_{opt}^{(q)}(n-|I|,\ k-A(\boldsymbol{l},\boldsymbol{y}))\right],$$

By defining $B(\boldsymbol{l},\boldsymbol{y}) = |I|$, we complete the proof. ■

Corollary 1: For an $(n,k,d)_q$ *code* $\mathcal{C}$ *that has information availability* (r_1,t_1)*, we have*

$$k \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{y}=\{y_j\in[t_1],1\leqslant j\leqslant z\} \\ A(\boldsymbol{y})<k}} \left[A(\boldsymbol{y}) + k_{opt}^{(q)}(n-B(\boldsymbol{y}),\ d)\right], \tag{3}$$

$$d \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{y}=\{y_j\in[t_1],1\leqslant j\leqslant z\} \\ A(\boldsymbol{y})<k}} \left[d_{opt}^{(q)}(n-B(\boldsymbol{y}), k-A(\boldsymbol{y}))\right], \tag{4}$$

where

$$A(\boldsymbol{y}) = \sum_{j=1}^{z}((r_1-1)y_j+1), \quad B(\boldsymbol{y}) = \sum_{j=1}^{z}(r_1 y_j+1).$$

Proof: Since it is the case of $\boldsymbol{l} = (1,1,\ldots,1)$ in Theorem 1-2, we obtain (3) and (4) straightforwardly. ■

Corollary 2: For an $(n,k,d)_q$ code $\mathcal{C}$ that has joint information locality $\{r_l : l \in [d-1]\}$, we have

$$k \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l})<k}} \left[A(\boldsymbol{l}) + k_{opt}^{(q)}(n-B(\boldsymbol{l}),\ d)\right], \quad (5)$$

$$d \leqslant \min_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l})<k}} \left[d_{opt}^{(q)}(n-B(\boldsymbol{l}), k-A(\boldsymbol{l}))\right], \quad (6)$$

where

$$A(\boldsymbol{l}) = \sum_{j=1}^{z} r_{l_j}, \quad B(\boldsymbol{l}) = \sum_{j=1}^{z}(r_{l_j}+l_j).$$

Proof: Since it is the case of $\boldsymbol{y} = (1,1,\ldots,1)$ in Theorem 1-2, we obtain (5) and (6) straightforwardly. ■

Corollary 3: For an $(n,k,d)_q$ code $\mathcal{C}$ that has 1-information locality r_1, we have

$$k \leqslant \min_{z\in[\lceil \frac{k}{r_1}\rceil-1]} \left[zr_1 + k_{opt}^{(q)}(n-z(r_1+1),\ d)\right], \quad (7)$$

$$d \leqslant \min_{z\in[\lceil \frac{k}{r_1}\rceil-1]} \left[d_{opt}^{(q)}(n-z(r_1+1), k-zr_1)\right]. \quad (8)$$

Proof: It is the case of $\boldsymbol{l} = (1,1,\ldots,1)$ and $\boldsymbol{y} = (1,1,\ldots,1)$ in Theorem 1-2. From the condition $A(\boldsymbol{l},\boldsymbol{y}) < k$, we have $z \leqslant \lceil \frac{k}{r_1}\rceil - 1$. Then, we obtain (7) and (8). ■

We note that the bounds in Corollary 1-3 were also derived in [7]–[9] respectively. However, Huang *et al.* [7] considered only linear codes, and Kim *et al.* [8] and Cadambe and Mazumdar [9] considered only LRCs with all symbol locality.

Now, we extend the bound in Theorem 2 into a bound without the alphabet constraint and also some well-known Singleton-like bounds for LRCs.

Theorem 3: For an $(n,k,d)_q$ code $\mathcal{C}$ that has joint information availability $\{(r_l,t_l) : l \in [d-1]\}$, we have

$$d \leqslant n-k+1 - \max_{\substack{z\in\mathbb{Z}^+ \\ \boldsymbol{l}=\{l_j\in[d-1],1\leqslant j\leqslant z\} \\ \boldsymbol{y}=\{y_j\in[t_{l_j}],1\leqslant j\leqslant z\} \\ A(\boldsymbol{l},\boldsymbol{y})<k}} \sum_{j=1}^{z}(l_j+(y_j-1)), \quad (9)$$

where $A(\boldsymbol{l},\boldsymbol{y}) = \sum_{j=1}^{z}((r_{l_j}-1)y_j+1)$.

Proof: By using Singleton bound [12] for $d_{opt}^{(q)}(\cdot)$ in Theorem 2, we obtain the bound (9) straightforwardly. ■

Corollary 4: For an $(n,k,d)_q$ code $\mathcal{C}$ that has information availability (r_1,t_1), we have

$$d \leqslant n-k-\left\lceil \frac{(k-1)t_1+1}{(r_1-1)t_1+1}\right\rceil + 2. \quad (10)$$

Proof: We validate the bound by means of some suitable choices of z, $\boldsymbol{l}$, and $\boldsymbol{y}$ in Theorem 3. For $r_1 = 1$, we use $z = k-1$, $\boldsymbol{l} = (1,1,\ldots,1)$, and $\boldsymbol{y} = (t_1,t_1,\ldots,t_1)$. For $t_1 = 1$, we use $z = \lceil \frac{k}{r_1}\rceil - 1$, $\boldsymbol{l} = (1,1,\ldots,1)$, and $\boldsymbol{y} = (1,1,\ldots,1)$. For $r_1,t_1 \geqslant 2$, we follow similar steps to the proof in [7]. For two non-negative integers x and α, let $k = x((r_1-1)t_1+1)+\alpha$, for $1 \leqslant \alpha \leqslant (r_1-1)t_1+1$. We consider two cases: (1) $1 \leqslant \alpha \leqslant r_1$. (2) $r_1 < \alpha \leqslant (r_1-1)t_1+1$. For the first case, we choose $z = \lceil \frac{k}{(r_1-1)t_1+1}\rceil - 1$, $\boldsymbol{l} = (1,1,\ldots,1)$, and $\boldsymbol{y} = (t_1,t_1,\ldots,t_1)$. For the second case, we choose $z = \lceil \frac{k}{(r_1-1)t_1+1}\rceil$, $\boldsymbol{l} = (1,1,\ldots,1)$, and $\boldsymbol{y} = (y_1,y_2,\ldots,y_{z-1},y_z) = (t_1,t_1,\ldots,t_1,\lceil\frac{\alpha-1}{r_1-1}\rceil - 1)$. In both cases, it is easy to check $A(\boldsymbol{l},\boldsymbol{y}) < k$. Now, it is enough to show that

$$\sum_{j=1}^{z}(l_j+(y_j-1)) \geqslant \left\lceil \frac{(k-1)t_1+1}{(r_1-1)t_1+1}\right\rceil - 1.$$

Case (1): For $1 \leqslant \alpha \leqslant r_1$, with the chosen z, $\boldsymbol{l}$, and $\boldsymbol{y}$, we have $k = z((r_1-1)t_1+1)+\alpha$. Then,

$$\sum_{j=1}^{z}(l_j+(y_j-1)) = zt_1 = \frac{(k-\alpha)t_1}{(r_1-1)t_1+1}$$
$$= \frac{(k-1)t_1+1+(r_1-\alpha)t_1}{(r_1-1)t_1+1} - 1 \geqslant \left\lceil \frac{(k-1)t_1+1}{(r_1-1)t_1+1}\right\rceil - 1.$$

Case (2): For $r_1 < \alpha \leqslant (r_1-1)t_1+1$, with the chosen z, $\boldsymbol{l}$, and $\boldsymbol{y}$, we have $k = (z-1)((r_1-1)t_1+1)+\alpha$. Then,

$$\sum_{j=1}^{z}(l_j+(y_j-1)) = (z-1)t_1 + \left\lceil\frac{\alpha-1}{r_1-1}\right\rceil - 1$$
$$\geqslant (z-1)t_1 + \left\lceil\frac{(\alpha-1)t_1+1}{(r_1-1)t_1+1}\right\rceil - 1 = \left\lceil\frac{(k-1)t_1+1}{(r_1-1)t_1+1}\right\rceil - 1.$$

■

The bound (10) was also derived in [7], [10] for only linear codes and in [6] for both linear and non-linear codes, but with an alternative proof.

Corollary 5: For an $(n,k,d)_q$ code $\mathcal{C}$ that has l-information locality r_l, we have

$$d \leqslant n-k+1-l\left(\left\lceil\frac{k}{r_l}\right\rceil - 1\right). \quad (11)$$

Proof: It is the case of $\boldsymbol{l} = (l,l,\ldots,l)$ and $\boldsymbol{y} = (1,1,\ldots,1)$ in Theorem 3. From the condition $A(\boldsymbol{l},\boldsymbol{y}) < k$, we have $z \leqslant \lceil\frac{k}{r_l}\rceil - 1$. Then, we obtain the bound (11). ■

The bound (11) was also derived in [3] and [8] for LRCs with all symbol locality.

III. A Connection to Batch Codes

LRCs with information availability can be viewed as systematic batch codes with restricted query size. Based on the connection between such batch codes and LRCs, we rederive a well-known bound in [11] for batch codes using our bound in Theorem 3. First, recall the definition of the batch codes.

Definition 4 [11]: A primitive (k,n,γ,τ) batch code $\mathcal{C}$ with restricted query size over alphabet $\mathcal{Q}$ encodes a string $\boldsymbol{X} \in \mathcal{Q}^k$ into a string $\boldsymbol{Y} \in \mathcal{Q}^n$, such that for all multisets of indices $i_1,i_2,\ldots,i_\tau$, where all $i_j \in [k]$, each of the entries $X_{i_1},X_{i_2},\ldots,X_{i_\tau}$ can be retrieved independently of each other by reading at most γ symbols of $\boldsymbol{Y}$. It is assumed that the symbols used to retrieve each of the variables X_{i_j}, for $1 \leqslant j \leqslant \tau$, are all disjoint. □

TABLE I
OPTIMALITY OF GRAPH-BASED LRCS WITH JOINT INFORMATION AVAILABILITY

Graph-based LRCs	Joint information availability	Optimality
$\left(\frac{k(k+1)}{2},k,k\right)_2$ complete graph code [4]	$r_l=l+1$, $t_l=k-l$	Optimal cases: $k=p=3,4$, Almost opt. case: $k=p=5$
$\left(\frac{k(k-\frac{k}{p}+2)}{2},k,k-\frac{k}{p}+1\right)_2$ complete multipartite graph code [4]	$r_l=l+1$, $t_1=k-\frac{k}{p}$, $t_2=k-\frac{2k}{p}+1$, $t_3=\begin{cases}k-\frac{3k}{p}+3, & \text{for } p\geqslant 3,\\ 2, & \text{for } p=2.\end{cases}$	Optimal case: $k=4,p=2$, Almost opt. case: $k=6,p=3$
$(3k-3,k,3)_2$ tiara code [8]	$(r_1,t_1)=(2,2)$, $(r_2,t_2)=(3,2)$	Optimal case: $k=4$
$(3k-5,k,3)_2$ crown code [5]	$(r_1,t_1)=(2,2)$, $(r_2,t_2)=(3,1)$	Optimal case: $k=5$
$(2k,k,3)_2$ ring code [5]	$(r_1,t_1)=(2,2)$, $(r_2,t_2)=(4,1)$	Optimal cases: $k=6,7$, Almost opt. case: $k=8$

Corollary 6: Let $\mathcal{C}$ be a systematic primitive (k,n,γ,τ) batch code. Then, the minimum distance d of $\mathcal{C}$ satisfies

$$d\leqslant n-k-(\tau-1)\left(\left\lceil\frac{k}{\gamma\tau-\tau+1}\right\rceil-1\right)+1. \quad (12)$$

Proof: In Theorem 3, we use $z=\lceil\frac{k}{(r_1-1)t_1+r_1}\rceil-1$, $\boldsymbol{l}=(1,1,\ldots,1)$, and $\boldsymbol{y}=(t_1,t_1,\ldots,t_1)$. Then,

$$\begin{aligned}A(\boldsymbol{l},\boldsymbol{y})&=\left(\left\lceil\frac{k}{(r_1-1)t_1+r_1}\right\rceil-1\right)((r_1-1)t_1+1)\\&<\left(\frac{k}{(r_1-1)t_1+1}\right)((r_1-1)t_1+1)=k.\end{aligned}$$

Since $\sum_{j=1}^{z}(l_j+(y_j-1))=zt_1=t_1\left(\lceil\frac{k}{(r_1-1)t_1+r_1}\rceil-1\right)$,

$$d\leqslant n-k-t_1\left(\left\lceil\frac{k}{r_1(t_1+1)-(t_1+1)+1}\right\rceil-1\right)+1.$$

From the relation between an $(n,k,d)_q$ LRC with information availability (r_1,t_1) and a systematic (k,n,γ,τ) batch code, we finally obtain (12) by setting $r_1=\gamma$ and $t_1=\tau-1$. ■

We note that an alternative proof of the bound (12) was shown in [11] for both systematic and non-systematic codes (but only for linear codes).

IV. ACHIEVABILITY AND TIGHTNESS OF THE BOUNDS

Since existing bounds in [7]–[9] are special cases of the bounds in Theorem 1-2, all the optimal codes in terms of the bounds [7]–[9] also achieve the bounds in Theorem 1-2 with equality. Due to the page restrictions, we only show the achievability and tightness of the bound in Theorem 1 using graph-based LRCs in [4], [5], and [8]. Based on the graph representation of the codes, we can easily obtain their joint information availability. With certain choice of parameters, we have some optimal and almost optimal LRCs in terms of our bound. Here, the almost optimality refers to having the value one less than the optimal case. The result is summarized in Table I. To show the tightness, we use the parameters of $(32,8,7)_2$ complete multipartite graph code with joint information availability $\{(r_1,t_1),(r_2,t_2),(r_3,t_3)\}=\{(2,6),(3,5),(4,5)\}$. With the parameters, we compare the optimal dimension $k_{b\text{-}opt}$ of our bound with those of the bounds in [7]–[9]. The bound in [9] gives $k_{b\text{-}opt}=16$. The bounds in [7] and [8] give $k_{b\text{-}opt}=14$. Our bound gives $k_{b\text{-}opt}=13$, and thus, it is the tightest.

V. CONCLUSION

We proposed two alphabet-dependent upper bounds for LRCs with joint information availability. Some existing bounds can be deduced from our bounds by restriction on parameters. We showed the achievability and tightness of our bounds by using graph-based LRCs. As a future work, one may consider the analysis for joint availability of existing codes and new constructions of codes with joint availability.

REFERENCES

[1] D. S. Papailiopoulos and A. G. Dimakis, “Locally repairable codes,” *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5843–5855, Oct. 2014.

[2] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, “On the locality of codeword symbols,” *IEEE Trans. Inf. Theory*, vol. 58, no. 11, pp. 6925–6934, Nov. 2012.

[3] A. S. Rawat, A. Mazumdar, and S. Vishwanath, “Cooperative local repair in distributed storage,” *EURASIP J. Adv. Signal Process.*, vol. 2015, no. 1, p. 107, Dec. 2015.

[4] J.-H. Kim, M.-Y. Nam, and H.-Y. Song, “Binary locally repairable codes from complete multipartite graphs,” in *Proc. Int. Conf. ICT Converg. (ICTC)*, Oct. 2015, pp. 1093–1095.

[5] J.-H. Kim, M.-Y. Nam, and H.-Y. Song, “Optimal binary locally repairable codes with joint information locality,” in *Proc. IEEE Inf. Theory Workshop (ITW)*, Oct. 2015, pp. 54–58.

[6] A. S. Rawat, D. S. Papailiopoulos, A. G. Dimakis, and S. Vishwanath, “Locality and availability in distributed storage,” *IEEE Trans. Inf. Theory*, vol. 62, no. 8, pp. 4481–4493, Aug. 2016.

[7] P. Huang, E. Yaakobi, H. Uchikawa, and P. H. Siegel, “Binary linear locally repairable codes,” *IEEE Trans. Inf. Theory*, vol. 62, no. 11, pp. 6268–6283, Nov. 2016.

[8] J.-H. Kim, M.-Y. Nam, and H.-Y. Song, “Alphabet-dependent upper bounds for locally repairable codes with joint locality,” in *Proc. Int. Symp. Inf. Theory Appl. (ISITA)*, Oct./Nov. 2016, pp. 36–40.

[9] V. R. Cadambe and A. Mazumdar, “Bounds on the size of locally recoverable codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 11, pp. 5787–5794, Nov. 2015.

[10] A. Wang and Z. Zhang, “Repair locality with multiple erasure tolerance,” *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 6979–6987, Nov. 2014.

[11] H. Zhang and V. Skachek, “Bounds for batch codes with restricted query size,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2016, pp. 1192–1196.

[12] W. C. Huffman and V. Pless, *Fundamentals of Error Correcting Codes*. Cambridge, U.K.: Cambridge Univ. Press, 2003.