# Locally Recoverable and Constrained Codes

Kevin Shu, Non-Member, IEEE

***Abstract*—Recently, there has been interest in extending classical coding theory in order to apply it to modern distributed storage systems. Two models for this problem are locally recoverable codes and constrained codes. We present additional work on these topics and possible ideas for later research, including new algorithms and constructions. We develop a class of constrained codes based on trees that have good systematic constructions.**

## I. Introduction

MODERN storage and communications systems are growing larger and more complex. A set of servers might need to recover from single server failures efficiently or a server might only know a small subset of the total data stored on the servers. Classical coding theory mostly focuses on the worst-case number of errors that a code can correct, which is often insufficient for distributed storage, because such codes require knowing the entire codeword in order to correct any erasures. Modern applications often require codes that have more diverse and highly structured properties.

Two classes of codes that have been devised to resolve some of these problems are the Locally Recoverable Codes (LRC codes) and the Constrained Codes. In LRC Codes, each individual codeword symbol can be recovered recovered from a ‘small’ number of other codeword symbols. Constrained Codes have the property that the codeword symbols may only depend on a ‘small’ number of message symbols. This paper will present constructions and algorithms that make the implementation of these ideas more effective.

We will define an $(n, k, d)$ code to be a code that maps a length $k$ message to a length $n$ codeword with distance $d$. A code has locality $r$ if, for each codeword symbol $c$, there is a set $S$ of $r$ other codeword symbols, such that the value of $c$ can be determined from the values of $S$.

For example, the simple parity check code $\{c \in \mathbb{F}_q^k : \sum_{i=0}^{k-1} c_i = 0\}$, is a $(k+1, k, 2)$ code, with locality $k$. A $(n, k)$ Reed Solomon code has distance $n-k+1$ and locality $k-1$, as do all maximum distance separable (MDS) codes. [1]

We can also consider constraints on the codeword symbols by only allowing them to depend on certain message symbols. This models a system in which the code symbols are being generated by different sources, where each source can only access a certain set of message symbols. For example, in the simple parity check code, a mapping can be chosen such that the first $i^{th}$ code symbol only depends on the value of the $i^{th}$ message symbol for $0 \leq i \leq k$, and the $k+1^{th}$ depends on all $k$ message symbols.

Singleton-like bounds on the distance achievable with linear locally recoverable codes were established in [3], and Reed-Solomon-like codes which achieved that bound were found in [4].

Codes with constraints on their generator matrices have studied in many contexts, such as [5] and [6]. Locally repairable codes with constraints on the recovering sets were considered in [7]. The first paper that addressed the issue in full generality was [2].

We expand these results, giving constructions for constrained codes, and bounds on the possible parameters associated with these codes.

## II. Preliminaries on Locally Recoverable and Constrained Codes

It was shown in [3] that the distance on a linear locally recoverable code with locality $r$ obeys

$$(1)\ d \leq n - k - \lfloor \frac{k}{r} \rfloor + 1$$

[8] generalizes this bound to non-linear codes.

In [2], the constraints are represented as a bipartite graph, $G = (\mathcal{M}, \mathcal{C}, \mathcal{E})$, where $\mathcal{M}$ is the set of message symbols, $\mathcal{C}$ is the set of code symbols, and $(m_i, c_j) \in \mathcal{E}$ if and only if $c_j$ is allowed to depend on $m_i$ when encoding. For example, in the graph in figure 1, the code symbol $c3$ may depend on the values of message symbols $m1$ and $m2$, but not $m3$. $c1$ can only depend on $m1$. (See figure 1)

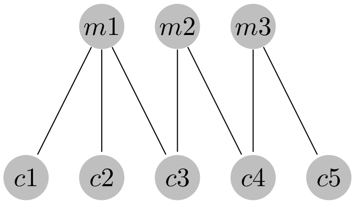

Fig. 1. An example of a constraint graph

If the code is linear, this translates into a a set of constraints on the generator matrix of the code, in particular, if we have that the adjacency matrix of this graph is zero at entry $(i, j)$, then the generator matrix must also have a zero in entry $(i, j)$.

The neighborhood of a set $\mathcal{M}$ of message nodes is denoted $n_{\mathcal{M}}$, and it was shown that

$$(2)\ d \leq |n_{\mathcal{M}'}| - |\mathcal{M}'| + 1$$

for any $\mathcal{M}' \subseteq \mathcal{M}$. [2].

[2] gives a construction that finds a subcode of a Reed-Solomon code that is valid for an adjacency matrix $A$. A Reed-Solomon code that is defined over an evaluation set $E = \{a_i\}$ has an adjacency matrix of the form $V = (e_{ij})$, where $e_{ij} = a_i^j$. One can find a matrix $TV$ that has a zero in entry $(i, j)$ if

B. Hassibi and W. Halbawi are with Caltech University

$A$ has a zero in entry $(i, j)$. The rows of $T$ can be viewed as the coefficients of a polynomial, i.e. let $t_i$ be the $i^{th}$ row of $T$, then $t_i A$ is the evaluation mapping of $t_i$ on the evaluation set of the Reed-Solomon code. Let $\{a_j\}$ be the evaluation set of the Reed-Solomon code. $t_i A$ has a zero in entry $j$ iff $t_i$ has a root at $a_j$. Therefore, we have that if $T$ is a matrix whose rows are the coefficients of polynomials with certain roots, then $A$ will be a valid code.

We have explicit constructions for both LRC and constrained codes that generalize Reed-Solomon codes, presented in [2], [4].

## III. Development of Algorithms

The original problem was to develop an algorithm to find the value associated with a constrained code,

$$k_{sys} = \min_{\overline{\mathcal{E}}} \max_{\rho} z_{\overline{\mathcal{E}},\rho}$$

The minimization is over all of systematic subgraphs of a constraint graph, and the maximization is over all of the rows of the corresponding systematic adjacency matrix, and where $z_{\overline{\mathcal{E}},\rho}$ is the number of zeros in a row $\rho$ of the adjacency matrix.

The problem of finding this value for general graphs is suspected to be hard, but it was found that there is a simple algorithm that solves this problem if the constraint graph has the form of a tree. A tree structured distributed storage system was considered in [9].

The formulation of this algorithm also shows that in the case of a tree, the value of $k_{sys}$ is at most $n - \min_{v \in \mathcal{M}} \deg v + 2$.

**Algorithm 1** Find $k_{min}$ for a tree $T$

```
Choose an arbitrary node $v$
Find the depth first tree of $T$ starting at $v$
for all depth in the depth first search tree do
    for all message node $m$ in this depth do
        for all code symbol node $c$ adjacent to $m$ do
            if $c$ is not adjacent to a message vertex that is lower
            in the tree and of minimal degree then
                match $m$ to $c$
            end if
        end for
        if all code symbol nodes adjacent to $m$ are adjacent to
        a message node of minimal degree then
            Match $m$ to any code symbol lower than $m$ in the
            tree
        end if
    end for
end for
```

*Proof of Correctness for Algorithm 1:* Intuitively, this algorithm works because changes that occur at lower depths in the tree do not propagate upwards, so the only set of nodes that are of concern are those at the current depth.

We have that this value is optimal, because if there were some better matching that did not decrease the minimal degree of any message vertex, then it would be found by this algorithm.

One possible way to generalize this algorithm would be to use the block-cut graph of the constraint graph, and apply the above algorithm on that graph, using exhaustive search to find the matching within each block.

## IV. Constrained Locally Repairable Codes

We also attempted to unify the results of [4] and [2] by finding subcodes of the codes found in [4] that satisfy a given constrain graph. These subcodes would have both low locality and satisfy the constraint graph, and therefore would be very useful in storage applications.

Firstly, we generalize bounds (1) and (2) above.

**Theorem 1.** *Let $C$ be a $(n, k, d)$ code that is valid for some constraint graph $G$. If $C$ has locality $r$, then*

$$(3)\ d \leq |n_M| - |M| - \frac{|M|}{r} + 2$$

*for any $M$ which is a subset of the message symbols.*

*Proof.* Let $M$ be any subset of the message symbols of $C$. Let $\phi$ be the encoding function that maps message vectors to code vectors in $C$. Without loss of generality, we can assume that the message vectors are of the form $(m_1, \dots m_{|M|}, m_{|M+1|}, \dots m_k)$, where $\{m_1, \dots m_{|M|}\} = M$, and $\{m_{|M+1|}, \dots m_k\} = M^c$, where $M^c$ denotes the set of message symbols not in $M$. We can restrict the domain of $\phi$ to those vectors of the form $(m_1, \dots m_{|M|}, 0, \dots 0)$, i.e. we let all of the message symbols not in $M$ be 0, and consider the code $S = \phi(\{x | x|_{M^c} = 0\})$, where $x|_{M^c}$ denotes the restriction of $x$ to those message symbols not in $M$. We have that $S$ is a subcode of $C$.

If $c$ is a code symbol in $n_M^c$, i.e. a code symbol which is not allowed to depend on the message symbols in $M$. The value of $c$ is constant in $S$, as the constraints imply that the values of $c \in n_M^c$ are not allowed to depend on the values of $m \in M$, and all code symbols not in $M$ are constant. Therefore, the code vector is required to have that all of the code symbols not in $n_M$ are constant. We can identify $S$ with a $(|n_M|, |M|, d')$ code by removing the message symbols and code symbols which are constant.

The locality of $S$ is at most $r$, because each code symbol in $C$ can be recovered from a set of code symbols of size no more than $r$, so code symbols in $S$ can be recovered from the same set of at most $r$ code symbols, where some of those code symbols may already be known to be 0 in $S$.

Therefore, we have from (2) that $d' \leq |n_M| - |M| - \frac{|M|}{r} + 2$.

Moreover, because $S$ is a subcode of $C$, if $x, y \in S$, then $d_H(x, y) \geq d$, from the definition of the Hamming distance of $C$. Therefore, in particular, $\min_{x,y \in S}\{d_H(x, y)\} \geq d$. Therefore, we have that $d \leq d'$, so $d \leq |n_M| - |M| - \frac{|M|}{r} + 2$. □

### A. Construction

To begin the construction, we reiterate the construction given in [4],

**Definition 1.** *Tamo-Barg Code*

Let $A$ be a subset of $\mathbb{F}_q$ of size $n$, that is partitioned into $\frac{n}{r+1}$ subsets, $\mathcal{A}_i$, of size $r+1$. We define a *good polynomial*, $g(x)$ to be a degree $r+1$ polynomial that is constant over each $\mathcal{A}_i$. An encoding polynomial is defined as $\sum_{i=0}^{r-1}\sum_{j=0}^{\frac{k}{r}-1} a_{ij}x^i g(x)^j$, where $g(x)$ is a good polynomial, and $(a_{ij})$ is a message vector. If we let $T_{\mathcal{A}}(f)$ be the evaluation map, mapping a polynomial $f$ to the vector $(f(\alpha)|\alpha \in A)$, then a Tamo-Barg code is

$$\mathcal{C}_{\mathcal{TB}} = \{T_{\mathcal{A}}(f)|f \text{ is an encoding polynomial}\}$$

This code has locality $r$, and has distance that meets $(1)$.

The generator matrix of such a code has the following form:

$$G = \Gamma V$$

where $V$ is a $k + \frac{k}{r} - 2 \times n$ Vandermonde-like, and $\Gamma$ is a matrix whose $(i, (r-1)j + l)$ entry (where $l < (r+1)$) is the $i^{th}$ coefficient of $g(x)^l x^j$.

[2] shows a similar construction that finds a subcode of a Reed-Solomon code that is valid for an adjacency matrix $A$, that is to say that if $V$ is the generator matrix for a Reed-Solomon code, one can find a matrix $TV$ that has a zero in entry $(i, j)$ if $A$ has a zero in entry $(i, j)$. The rows of $T$ can be viewed as the coefficients of a polynomial, i.e. let $t_i$ be the $i^{th}$ row of $T$, then $t_iA$ is the evaluation mapping of $t_i$ on the evaluation set of the Reed-Solomon code. Let $\{a_j\}$ be the evaluation set of the Reed-Solomon code. $t_iA$ has a zero in entry $j$ iff $t_i$ has a root at $a_j$. Therefore, we have that if $T$ is a matrix whose rows are the coefficients of polynomials with certain roots, then $A$ will be a valid code.

In order to generalize this idea to the Tamo-Barg codes, one finds that a generator matrix for a subcode of a Tamo-Barg code has the form $\mathcal{T}(\Gamma V)$. Letting $\mathcal{T}\Gamma = T$, then this generator matrix has the form $TV$. As in the case of Reed-Solomon codes, the rows of $T$ must be the coefficients of polynomials with certain roots, but in addition, one requires that the rows of $T$ must be in the row space of $\Gamma$. This translates into the constraint that each $t_i$ be of the form $\sum_{i=0}^{r-1}\sum_{j=0}^{\frac{k}{r}-1} a_{ij}x^i g(x)^j$, for some set of $a_{ij}$.

There is one special case in which there are simple constructions for such codes.

Suppose we are given a graph $G$, and are asked to construct a subcode of a $(n, k, d)$ Tamo-Barg code, $\mathcal{C}_{\mathcal{TB}}$, with locality $r$ and good polynomial $g(x)$, that is valid for $G$. Let $Z_i$ be the complement of the neighborhood of the message symbol $m_i$. If $G$ has the property that there is some set, $S$, of size $\frac{k}{r} - 1$ code symbols such that for each $i$, $|Z_i/S| \leq r-1$, then we can construct a subcode of the Tamo-Barg code as follows.

Firstly, let $\mathcal{A}_i$ be some partition of $\mathcal{A}$ with an element $a$, then consider $g'(x) = g(x) - g(a)$, which is 0 uniformly on $A_i$. Then, construct a new Tamo-Barg code, $\mathcal{G}‘_{\mathcal{TB}}$ with the same partition, and with $g'(x)$ as the good polynomial. Then, let $t_i(x) = g'(x)\Pi_{z\in Z_i/S}(x-z)$. Because $|Z_i/S| \leq r-1$, we have that $t_i(x)$ is a valid evaluation polynomial in the Tamo-Barg basis, and that each $t_i$ has zeros in the appropriate locations.

## V. Conclusion

We have devised a new algorithm for finding the smallest possible dimension of a Reed-Solomon code that can be used to form a systematic generator matrix that is valid for a given constraint graph and new constructions that naturally extend the work done in [2]. In the future, we would like to generalize these ideas to larger classes of constraint graphs.

## References

[1] R. J. McEliece, *The theory of information and coding : a mathematical framework for communication*. Reading, Mass.: Addison-Wesley Pub. Co., Advanced Book Program, 1977.

[2] W. Halbawi, M. Thill, and B. Hassibi, "Coding with Constraints: Minimum Distance Bounds and Systematic Constructions," *CoRR*, vol. abs/1501.07556, 2015. [Online]. Available: http://arxiv.org/abs/1501.07556

[3] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," *IEEE Trans. Inform. Theory*, pp. 6925–6934, 2012.

[4] I. Tamo and A. Barg, "A Family of Optimal Locally Recoverable Codes," *Information Theory, IEEE Transactions on*, vol. 60, no. 8, pp. 4661–4676, 2014.

[5] S. H. Dau, W. Song, Z. Dong, and C. Yuen, "Balanced sparsest generator matrices for MDS codes," *CoRR*, vol. abs/1301.5108, 2013. [Online]. Available: http://arxiv.org/abs/1301.5108

[6] W. Halbawi, T. Ho, H. Yao, and I. M. Duursma, "Distributed reed-solomon codes for simple multiple access networks," *CoRR*, vol. abs/1310.5187, 2013. [Online]. Available: http://arxiv.org/abs/1310.5187

[7] A. Mazumdar, "Storage capacity of repairable networks," *CoRR*, vol. abs/1408.4862, 2014. [Online]. Available: http://arxiv.org/abs/1408.4862

[8] M. Forbes and S. Yekhanin, "On the locality of codeword symbols in non-linear codes," *Discrete Mathematics*, vol. 324, pp. 78 – 84, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0012365X14000284

[9] J. Li, S. Yang, X. Wang, X. Xue, and B. Li, "Tree-structured data regeneration with network coding in distributed storage systems."