On the Design of Storage Orbit Codes

Shiqiu Liu and Frédérique Oggier

Abstract We propose the use of orbit codes to design storage codes, that is, subspace codes obtained from orbits of the action of subgroups of $GL_n(\mathbb{F}_q)$ on m-dimensional subspaces of $\mathbb{F}_q^n$. We translate the storage code parameters into those of the algebraic objects involved, and construct a simple family of storage orbit codes.

Key words: Orbit Codes, Storage Codes, Group Action

1 Introduction

Let $\mathbb{F}_q$ be the finite field with q elements, with q a prime power. The set of all subspaces of $\mathbb{F}_q^n$ of dimension m is called *Grassmannian* and is denoted by $G_q(m,n)$:

$$G_q(m,n) = \{\mathscr{U} \text{ subspace of } \mathbb{F}_q^n,\ \dim(\mathscr{U}) = m\}.$$

The subspace distance d between two subspaces $\mathscr{U}, \mathscr{V}$ is

$$d(\mathscr{U}, \mathscr{V}) = \dim(\mathscr{U}) + \dim(\mathscr{V}) - 2\dim(\mathscr{U} \cap \mathscr{V}).$$

We denote by $GL_n(\mathbb{F}_q)$ the set of $n \times n$ invertible matrices with coefficients in $\mathbb{F}_q$. Multiplication by elements of $GL_n(\mathbb{F}_q)$ defines a group action from the right on $G_q(m,n)$ by

Shiqiu Liu
Division of Mathematical Sciences, Nanyang Technological University, Singapore, e-mail: SLIU012@e.ntu.edu.sg

Frédérique Oggier
Division of Mathematical Sciences, Nanyang Technological University, Singapore e-mail: frederique@ntu.edu.sg

$$G_q(m,n) \times GL_n(\mathbb{F}_q) \longrightarrow G_q(m,n)$$
$$(\mathscr{U},A) \longmapsto \mathscr{U}A = \{uA,\ u \in \mathscr{U}\}$$

since any element of $GL_n(\mathbb{F}_q)$ maps an m-dimensional subspace to an m-dimensional subspace. In fact, as pointed out in [5], since any two m-dimensional subspaces can be mapped onto each other by an element of $GL_n(\mathbb{F}_q)$, $GL_n(\mathbb{F}_q)$ acts transitively on $G_q(m,n)$.

Codes whose codewords are elements of $G_q(m,n)$ are popular for error correction in network coding [3], where the distance of interest between codewords is the subspace distance. They are sometimes referred to as constant dimension codes. Constant dimension codes have been obtained from orbits of a cyclic group of $GL_n(\mathbb{F}_q)$ in [5], together with a decoding algorithm for a subclass of such codes.

In this paper, we are interested in storage codes whose codewords still are elements of $G_q(m,n)$, but whose design criterion varies from that of constant dimension codes. In [1], such storage codes were built from cliques within the Grassmannian graph. We will present in Section 2 another approach to the design of storage codes, that of orbit codes, following the terminology of [5]. In Section 3, we propose a family of cyclic orbit codes suitable for collaborative repair (the meaning of collaborative repair will be explain below).

2 Storage Codes

Fix $\mathscr{U} \in G_q(m,n)$, let G be a subgroup of $GL_n(\mathbb{F}_q)$, and let $\mathscr{U}G = \{\mathscr{U}g,\ g \in G\}$ be the orbit of $\mathscr{U}$ under the right action of G. We will refer to $\mathscr{U}G$ as an *orbit code*.

In order to represent an m-dimensional subspace $\mathscr{U}$ of the vector space $\mathbb{F}_q^n$, we fix a basis of $\mathscr{U}$, and use an $m \times n$ matrix U whose row space $\{vU,\ v \in \mathbb{F}_q^m\}$ is $\mathscr{U}$.

A storage code $\mathscr{C}$, in the context of networked distributed storage [2], aims at encoding a data object $\mathbf{o} \in \mathbb{F}_q^n$ into N network nodes, such that the object can be retrieved from a subset of live nodes in case of node failures. In the case of linear codes, every node stores the inner products of $\mathbf{o} \in \mathbb{F}_q^n$ with some (say m, $m \geq 1$) vectors in $\mathbb{F}_q^n$, thus the node may compute linear combinations of these inner products, and thus is seen as storing an m-dimensional vector space, the span of these m vectors which are assumed to be linearly independent, without loss of generality. The main difference between a storage code and an erasure code is that a storage code should be amenable to repair, namely, the code should be such that the data stored at one node can be computed from a (small) subset of other nodes, without (necessarily) having to decode the object first. Alternatively, in the case of collaborative repair, a subset of nodes can be computed from another subset of repair nodes, allowing these repair nodes to exchange data among each other [2].

Let us translate these storage parameters for an orbit code $\mathscr{U}G$:

1. the number of storage nodes is the cardinality $|\mathscr{U}G|$ of the orbit, and it is well known that

$$|\mathscr{U}G| = \frac{|G|}{|Stab_G(\mathscr{U})|},$$

where $Stab_G(\mathscr{U}) = \{g \in G,\ \mathscr{U}g = \mathscr{U}\}$ is a subgroup of G called the stabilizer of $\mathscr{U}$,

2. the storage capacity (or number of stored symbols in $\mathbb{F}_q$) for every node is m,
3. the size of the object to be stored is n.

Remark 1. Note that nodes are storing the inner product of the object $\mathbf{o}$ with basis vectors, thus we associate to a basis vector $\mathbf{v}$ the inner product between $\mathbf{o}$ and $\mathbf{v}$. Consequently, we say that a node stores $\mathbf{v}$ instead of saying that it stores $\mathbf{v}\mathbf{o}^T$, which makes it easier to translate storage codes in terms of group action. However, in terms of storage, $\mathbf{v}\mathbf{o}^T$ is one element of $\mathbb{F}_q$, not n.

Example 1. Let $G = \langle g \rangle$ be the subgroup of $GL_5(\mathbb{F}_2)$ generated by

$$g = \begin{bmatrix} 0&0&1&0&0 \\ 0&0&0&0&1 \\ 1&1&1&1&1 \\ 1&0&0&0&0 \\ 0&1&0&0&0 \end{bmatrix}$$

Consider the orbit code $\mathscr{U}G$, where $\mathscr{U}$ has for basis

$$U = \begin{bmatrix} 1&0&0&0&0 \\ 0&1&0&0&0 \\ 0&0&1&0&0 \end{bmatrix}.$$

Since g has order 4 and $Stab_G(\mathscr{U})$ is trivial, the orbit code $\mathscr{U}G$ is

$$U,\ \begin{bmatrix} 0&0&1&0&0 \\ 0&0&0&0&1 \\ 1&1&1&1&1 \end{bmatrix},\ \begin{bmatrix} 1&1&1&1&1 \\ 0&1&0&0&0 \\ 0&0&0&1&0 \end{bmatrix},\ \begin{bmatrix} 0&0&0&1&0 \\ 1&0&0&0&0 \\ 0&0&0&0&1 \end{bmatrix}.$$

This corresponds to a storage code $\mathscr{C}$ where the size of the object $\mathbf{o}$ is $n = 5$, every storage node stores $m = 3$ symbols (e.g., the first node corresponding to the subspace $\mathscr{U}$ stores $(1,0,0,0,0)\mathbf{o}^T = o_1$, $(0,1,0,0,0)\mathbf{o}^T = o_2$, $(0,0,1,0,0)\mathbf{o}^T = o_3$ for $\mathbf{o} = (o_1,\ldots,o_5) \in \mathbb{F}_q^5$). The data object $\mathbf{o}$ may be retrieved out of any two nodes, since $\dim(\mathscr{U}g^i + \mathscr{U}g^j) = 5$ for all $i \neq j$. In case of one node failure, the subspace $\mathscr{U}g^i$ may be computed from the knowledge of a subspace of dimension 1 from each of the other $\mathscr{U}g^j$, $j \neq i$, or in other words $\dim(\mathscr{U}g^i \cap \mathscr{U}g^j) = 1$ for all $i \neq j$. This code instance has been reported in [4].

3 A Simple Instance of Cyclic Orbit Codes

We next propose a family of cyclic orbit codes and compute the parameters of the corresponding storage codes.

Lemma 1. *Let $G = \langle g \rangle$ be the subgroup of $GL_n(\mathbb{F}_q)$ generated by the $n \times n$ matrix*

$$g = \begin{bmatrix} 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ & & & \ddots & \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix},$$

and let U contain a canonical basis of $G_2(n-1,n)$. Then any two distinct elements $\mathscr{U}g^i$ and $\mathscr{U}g^j$ of the orbit $\mathscr{U}G$ of $\mathscr{U}$ under the right action of G intersect in a subspace of dimension $n-2$. Furthermore, the size of the orbit is $n+1$.

Proof. The size of the orbit $\mathscr{U}G$ is the order of g which is $n+1$ since the stabilizer is trivial. Since

$$\dim(\mathscr{U}g^i + \mathscr{U}g^j) = 2(n-1) - \dim(\mathscr{U}g^i \cap \mathscr{U}g^j) \leq n,$$

it must be that

$$n-2 \leq \dim(\mathscr{U}g^i \cap \mathscr{U}g^j)$$

which concludes the proof, since $\mathscr{U}g^i$ and $\mathscr{U}g^j$ are distinct for $i \neq j$.

Proposition 1. *The orbit $\mathscr{U}G$ forms a code $\mathscr{C}$ such that*

1. *the object is retrieved by contacting any two nodes,*
2. *the repair of two failures is done by downloading $n-1$ elements of $\mathbb{F}_q$ from one node for each failure, and exchanging one element of $\mathbb{F}_q$ between the two repair nodes.*

Recall Remark 1 for counting the size of elements downloaded/repaired.

Proof. Label the $n+1$ storage nodes from 0 to n. By assumption, the ith node stores the element $\mathscr{U}g^i$ of the orbit $\mathscr{U}G$ of $\mathscr{U}$ under the right action of G, $i = 0,1,...,n$.

When contacting any two nodes, we get two elements of the orbit $\mathscr{U}G$, say $\mathscr{U}g^i$ and $\mathscr{U}g^j$, $i \neq j$, and we have

$$\begin{aligned} \dim(\mathscr{U}g^i \cup \mathscr{U}g^j) &= \dim(\mathscr{U}g^i) + \dim(\mathscr{U}g^j) - \dim(\mathscr{U}g^i \cap \mathscr{U}g^j) \\ &= (n-1) + (n-1) - (n-2) = n, \end{aligned}$$

hence we can retrieve the object.

Suppose two nodes have failed, say node s and node t, download from node i the subspace $A = \mathscr{U}g^i \cap \mathscr{U}g^s$ of dimension $n-2$, and from node j the subspace $B = \mathscr{U}g^j \cap \mathscr{U}g^t$ also of dimension $n-2$, by the above lemma. If $\dim(A \cup B) = n$, the

repair process can be completed by collaboration, by exchanging the missing basis vector at each repair node. Indeed, since $\dim(A \cup B) = n$, write the missing basis vector a at node s in a basis $\{v_1, \ldots, v_n\}$ of $A \cup B$ as $a = \sum_{i=1}^{n} a_i v_i$. If $v_1, \ldots, v_l \in A$, ask the symbol $\sum_{i=l+1}^{n} a_i v_i$ from B. Iterate this process for the missing basis vector at node t.

We are left to show that $\dim(A \cup B) = n$, or in fact, since

$$\dim(A \cup B) = \dim(A) + \dim(B) - \dim(A \cap B) = 2(n-2) - \dim(A \cap B),$$

to show that $\dim(A \cap B) = n-4$.

Write $\mathscr{U} g^i = \langle g_1, ..., g_{n-2}, g_{i_0} \rangle$, $\mathscr{U} g^j = \langle g_1, ..., g_{n-2}, g_{j_0} \rangle$. Then $A = \mathscr{U} g^s \cap \mathscr{U} g^i = \langle g_1, ...g_{s_0-1}, g_{s_0+1}, ..., g_{n-2}, g_{i_0} \rangle$, otherwise the nodes i, j and s would intersect in the same subspace of dimension $n-2$, which is not possible by definition of G, and for the same reason $B = \mathscr{U} g^t \cap \mathscr{U} g^j = \langle g_1, ...g_{t_0-1}, g_{t_0+1}, ..., g_{n-2}, g_{j_0} \rangle$. Without loss of generality, suppose that $s_0 < t_0$. Then

$$A \cap B = \langle g_1, ..., g_{s_0-1}, g_{s_0+1}, ..., g_{t_0-1}, g_{t_0+1}, ...g_{n-2} \rangle$$

which has dimension $n-4$.

Example 2. Consider $G_2(3,4)$, and the subspace $\mathscr{U}$, with basis

$$U = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Let $G = \langle g \rangle$ be the cyclic subgroup of $GL_4(\mathbb{F}_q)$ generated by

$$g = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

The order of g is 5. The orbit of $\mathscr{U} \in G_2(3,4)$ is by definition

$$\mathscr{U} G := \{\mathscr{U} g^i,\ 0 \leq i \leq 4\}.$$

The elements of the orbit are explicitly given by

$$U,\ \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix},\ \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix},\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

We store them in node 0 to 4. Assume that node 1 and node 4 failed. To repair node 1, download $\{(0010), (0001)\}$ from node 2, to repair node 4, download $\{(1000), (0100)\}$ from node 0, then during collaboration, the node repairing node 1 can compute (0110) and send it, and the node repairing node 4 sends (0100) in exchange. Note that the strategy is not unique. To repair node 1, download alterna-

tively $\{(0001), (0110)\}$ from node 3, to repair node 4, get $\{(1100), (1111)\}$ from node 2, then the node repairing node 1 can get (0011) and the one repairing node 4 can get (0111) from the collaboration.

4 Future Work

In this paper, we translated the parameters of storage codes into those of orbit codes, illustrated how one known instance of storage code be seen as an orbit code, and constructed a family of storage codes suitable for collaborative repair.

Future research directions naturally include:

- the construction of other storage codes, using other cyclic groups, but also other finite groups,
- the analysis and comparison of the code parameters with both the codes available in the literature, but also the known bounds.

Acknowledgement

The research of S. Liu is supported by the Singapore National Research Foundation under Research Grant NRF-RF2009-07. The research of F. Oggier for this work is supported by the MoE Tier-2 grant "eCODE: Erasure Codes for Datacenter Environments".

References

1. Oggier, F.: Some constructions of storage codes from grassmann graphs. In: Proceedings of the International Zurich Seminar on Communications. Zürich (2014)
2. Oggier, F., Datta, A.: . Foundations and Trends in Information and Communication Theory. Now Publishers
3. R. Koetter, F.K.: Coding for errors and erasures in random network coding. IEEE Transactions on Information Theory **54**(8) (2008)
4. Rashmi, K., Shah, N., Kumar, P., Ramchandran, K.: Explicit construction of optimal exact regenerating codes for distributed storage. In: Proceedings of the Allerton conference on Communication, Control, and Computing, vol. 1-2, pp. 1243–1249. IEEE, Urbana Champaign (2009)
5. Trautmann, A.L., Manganiello, F., Braun, M., Rosenthal, J.: Cyclic orbit codes. IEEE Transactions on Information Theory **59**(11) (2013)