Constructing network codes using Möbius transformations

Andreas-Stephan Elsenhans[*] and Axel Kohnert[†]

November 26, 2010

Abstract

We study error correcting constant dimension subspace codes for network coding. The codewords are $\mathbb{F}_2$-subspaces of $\mathbb{F}_{2^n}$, having at most 1-dimensional intersections. $\mathbb{F}_{2^n}^*$ is contained in the automorphism group of the code.

We give a lower bound for the size of such codes, being constructed by a greedy method. Further we describe a code in $\mathbb{F}_{2^{2m}}$ with 2^{3m} codewords, for which we finally present a construction, an encoding and a decoding algorithm. This is based on the geometry of $\mathbb{F}_{2^{2m}}$ as a 2-dimensional $\mathbb{F}_{2^m}$ space and the operation of fractional linear transformations.

1 Introduction

In [KK], Kötter and Kschischang developed the theory of subspace codes for applications in network coding. In the following years different methods for the construction of such codes have been investigated [EV, KK, KoKu, MGR, S, SKK, TK]. In [EKW], Wassermann and the authors showed how to get a practical (there is a good decoding algorithm) code with many codewords. The code consists of 3-dimensional subspaces in $V = \mathbb{F}_2^n$. Identifying V with the field $\mathbb{F}_{2^n}$, we get an action of the multiplicative group $\mathbb{F}_{2^n}^*$. This enables us to construct $\mathbb{F}_{2^n}^*$-orbits of 3-dimensional subspaces, containing $2^n - 1$ elements in general. A randomly chosen 3-dimensional $\mathbb{F}_2$-subspace in $\mathbb{F}_{2^n}$ leads to a good orbit of this length. Good

[*] Mathematisches Institut, Universität Bayreuth, D-95440 Bayreuth, Germany
Mail: `Stephan.Elsenhans@uni-bayreuth.de`
Website: `http://www.staff.uni-bayreuth.de/∼btm216`

[†] Mathematisches Institut, Universität Bayreuth, D-95440 Bayreuth, Germany
Mail: `kohnert@uni-bayreuth.de`
Website: `http://www.algorithm.uni-bayreuth.de/de/team/kohnert_axel`

means here that two different subspaces of the orbit have an at most 1-dimensional intersection. This leads to a code with minimum distance 4.

In [EKW], an error correcting decoding algorithm is presented for such codes. Finally, a heuristic argument for the possibilities of combining many such $\mathbb{F}_{2^n}^*$-orbits to a bigger code is given.

In this note, we study the conditions for a k-dimensional subspace to lead to a good orbit. More precisely, we construct a finite list of equations (L1) such that being good is equivalent to the fact that none of the equations is satisfied.

Further, we construct a second finite list of equations (L2) such that none of them is satisfied if and only if two different subspaces lead to two orbits which can be used in common. I. e., the pairwise intersections of the subspaces of the orbits are at most one-dimensional.

By bounding the number of solutions of the equations in (L1) and (L2), we get a lower bound for the number of subspaces that can form a code. For fixed k and $n \to \infty$, our lower bound and a trivial upper bound differ only by a constant factor.

In the second part of this article, we explain how to handle a code in $\mathbb{F}_{2^{2n}}$ having 2^n or more $\mathbb{F}_{2^{2n}}^*$-orbits of subspaces. For that, we give algorithms for construction, coding, and decoding.

2 Existence and equations for good orbits

We intend to use $\mathbb{F}_{2^n}^*$-orbits of k-dimensional $\mathbb{F}_2$-subspaces in $\mathbb{F}_{2^n}$ as a code. The following lemma will show that such an approach is possible.

2.1. Lemma. —— *If n is sufficiently large then there exists a k-dimensional subspace U such that the equality $z_1U = z_2U$, $z_1, z_2 \in \mathbb{F}_{2^n}^*$, implies $z_1 = z_2$.*

Proof: As $\frac{z_1}{z_2}$ is an element of the stabilizer of U, we have to show that there exists a k-dimensional subspace with trivial stabilizer in $\mathbb{F}_{2^n}^*$.

First, we analyze the possible stabilizers S. As each element in $U \setminus \{0\}$ has trivial stabilizer, this set decomposes into orbits all having length $\#S$. Therefore, we get $\#S \mid 2^k - 1$. Thus, each element in the stabilizer is a solution of the equation $z^{2^k-1} - 1 = 0$. We denote the set of all solutions of the last equation by Z.

To construct U, we start with an arbitrary element $b_1 \in \mathbb{F}_{2^n}^*$. If we can ensure that the only element of the form zb_1 $(z \in Z)$ in U is b_1 then we are done.

The elements zb_1 $(z \in Z)$ generate a subspace of dimension at most $2^k - 1$. Completing a basis of this space to a basis of $\mathbb{F}_{2^n}$, we can construct the desired subspace U as soon as $n \geq 2^k - 1 + (k - 1)$. □

2.2. Remark. —— The lemma shows that, for n large enough, a $\mathbb{F}_{2^n}^*$-orbit of length $2^n - 1$ exits. The inequality $n \geq 2^k - 2 + k$ is very weak. We will prove stronger existence statements below.

Such an orbit gives rise to a constant dimension subspace code with $2^n - 1$ codewords.

For error correcting purposes, we have to control the distance d between two codewords. The distance is given by the distance in the Hasse diagram of the subspace lattice. It can be computed by $d(U,V) = \dim(U+V) - \dim(U \cap V)$.

In the following, we will focus on $\mathbb{F}_{2^n}^*$-orbits of k-dimensional subspaces such that $\dim(U \cap V) \leq 1$ for all U, V in the orbits. Such an orbit is a code with minimum distance $2(k-1)$.

2.3. Definition. —— i) We denote by $\mathrm{Gr}_{k,n}(\mathbb{F}_q)$ the set of all k-dimensional subspaces in $\mathbb{F}_q^n$.

ii) We denote the cardinality of the Graßmannian $\mathrm{Gr}_{k,n}(\mathbb{F}_q)$ by the Gaußian coefficient

$$\begin{bmatrix} n \\ k \end{bmatrix} := \frac{(q^n - q^0)\cdots(q^n - q^{k-1})}{(q^k - q^0)\cdots(q^k - q^{k-1})}\,.$$

iii) We denote by $\mathcal{A}_q(n, 2\delta, k)$ the maximal number of k-dimensional subspaces in $\mathbb{F}_q^n$ such that the pairwise intersection has at most dimension $k - \delta$.

2.4. Remark. —— In this note, q will always be 2. The vector space $\mathbb{F}_2^n$ will be identified with the field $\mathbb{F}_{2^n}$. Thus, we get an action of $\mathbb{F}_{2^n}^*$ on $\mathrm{Gr}_{k,n}(\mathbb{F}_2)$.

2.5. Definition. —— i) For a subset U of $\mathbb{F}_{2^n}$, we define its quotient set by

$$Q(U) := \left\{ \frac{u_1}{u_2} \,\middle|\, u_1, u_2 \in U \setminus \{0\}, u_1 \neq u_2 \right\}\,.$$

ii) We call the subspace $U \in \mathrm{Gr}_{k,n}(\mathbb{F}_2)$ good if $\dim(U \cap tU) \leq 1$ for all $t \in \mathbb{F}_{2^n}^* \setminus \{1\}$.

iii) We call a subspace bad if it is not good. Further, a basis is bad if the generated subspace is bad.

iv) We call two subspaces $U, V \in \mathrm{Gr}_{k,n}(\mathbb{F}_2)$ *combinable* if $\dim(U \cap tV) \leq 1$ for all $t \in \mathbb{F}_{2^n}^*$.

v) A subset of $\mathrm{Gr}_{k,n}(\mathbb{F}_2)$ is called good and combinable if all elements are good and pairwise combinable.

2.6. Remark. —— Note that $Q(U) = Q(zU)$ for all $z \in \mathbb{F}_{2^n}^*$.

Each good subspace leads to an $\mathbb{F}_{2^n}^*$-orbit of length $2^n - 1$. The minimum distance of the corresponding code is $2(k-1)$. A good and combinable subset of $\mathrm{Gr}_{k,n}(\mathbb{F}_2)$ with m elements leads to a code with $m(2^n - 1)$ codewords and minimum distance $2(k-1)$.

The goal of this section is to prove a lower bound for the size of a maximal good and combinable subset in $\mathrm{Gr}_{k,n}(\mathbb{F}_2)$.

2.7. Lemma. —— *Let $k \geq 2$ and $U, V \in \mathrm{Gr}_{k,n}(\mathbb{F}_2)$.*

i) *The subspace U is good if and only if the quotients $\frac{u_1}{u_2}$ for $u_1, u_2 \in U \setminus \{0\}$, $u_1 \neq u_2$ are pairwise different.*

ii) *The subspaces U and V are combinable if and only if the quotient sets $Q(U)$ and $Q(V)$ are disjoint.*

Proof:

i) Assume the intersection of U and zU has dimension at least 2. Then, we get $u_1, u_2 \in (U \cap zU) \setminus \{0\}$ with $u_1 \neq u_2$ and $u_1 =: zu_3$ and $u_2 =: zu_4$. This leads to $\frac{u_1}{u_2} = \frac{u_3}{u_4}$.
On the other hand, when $\frac{u_1}{u_2} = \frac{u_3}{u_4}$ with $u_1 \neq u_2$ is satisfied, we set $z := \frac{u_1}{u_3}$. Then, we get $u_1 = zu_3$ and $u_2 = zu_4$. Thus, the intersection has two non-zero elements and its dimension is at least 2.

ii) Is analogous. □

2.8. Remark. —— To show the existence of good subspaces, we analyze the condition given in the first part of the lemma above. More precisely, we bound the number of bad bases.

Let $b_1, \dots, b_k$ be a basis of $U \in \mathrm{Gr}_{k,n}(\mathbb{F}_2)$. We think of the b_i as variables. The non-zero elements of the generated subspace are represented by $2^k - 1$ non zero linear forms in $\mathbb{F}_2[b_1, \dots, b_k]$.

Thus, for each 4-tuple $l_1, l_2, l_3, l_4 \in \mathbb{F}_2[b_1, \dots, b_k]$ of linear forms $l_1 \neq l_2$, $l_3 \neq l_4$, $(l_1, l_2) \neq (l_3, l_4)$, we get the algebraic equation

$$l_1 l_4 = l_2 l_3 \,, \tag{1}$$

which encodes that the corresponding quotients $\frac{l_1}{l_2}$, $\frac{l_3}{l_4}$ are equal. Note that we never get the zero-equation. Assuming $l_1 l_4 = l_2 l_3$ as an equality of polynomials, we get $\{l_1, l_4\} = \{l_2, l_3\}$ as the polynomial ring has unique factorization. This is a contradiction.

Note that a basis is bad if and only if it solves at least one of the equations of (1).

An obvious upper bound for the number of equations is $(2^k - 1)^4$. As each equation has degree two, it has at most $2(2^n - 1)^{k-1}$ solutions in $\mathbb{F}_2$-linearly independent k-tuples in $\mathbb{F}_{2^n}$. Thus, we have at most $(2^k - 1)^4 2(2^n - 1)^{k-1}$ bad bases. In total there are at least $2^{k(n-1)}$ bases of k-dimensional subspaces.

This observation already leads to an interesting existence theorem for good subspaces as, for fixed k and $n \to \infty$, the total number of bases growth faster than the number of bad bases. We give better estimates in the following. The next lemma reduces the number of equations.

2.9. Remark. —— The linear forms $l_1, \dots, l_4$ lead to the equation

$$l_1 l_4 - l_2 l_3 = 0.$$

To sort the equations, we inspect the $\mathbb{F}_2$-subspace generated by $l_1, \dots, l_4$. This is of dimension 2, 3, or 4. In the following lemma, we will analyze each of the cases.

2.10. Lemma. —— *We fix the subspace L of linear forms in $\mathbb{F}_2[b_1, \dots, b_k]$.*

i) *If L is of dimension two then exactly one equation (which may be solvable by linearly independent elements in $\mathbb{F}_{2^n}$) will correspond to L.*

ii) *If L is of dimension three then exactly 28 equations (which may be solvable by linearly independent elements in $\mathbb{F}_{2^n}$) will correspond to L.*

iii) *If L is of dimension four then exactly 280 equations (which may be solvable by linearly independent elements in $\mathbb{F}_{2^n}$) will correspond to L.*

Proof: We use `magma` [M]. We enumerate all the equations in an iterated loop and eliminate insolvable equations and repetitions. □

2.11. Remark. —— For each of the cases of the lemma above, we give a few examples to explain why some equations are insolvable and why so many repetitions occur.

a) Let l_1, l_2 be two linearly independent linear forms. The equation $l_1 l_2 = (l_1 + l_2) l_2$ is insolvable, as it implies $0 = l_2^2$. Note here that a non-zero linear form never evaluates to zero on linearly independent vectors. Thus, this equation eliminates no linearly independent k-tuple as a basis for a good subspace.

Further, the equations $l_1 l_1 = (l_1 + l_2) l_2$ and $(l_1 + l_2) l_1 = l_2 l_2$ are equivalent.

b) Let l_1, l_2, l_3 be three linearly independent linear forms. Then, the equation $l_1 l_2 = l_3 l_2$ is insolvable. It implies that the non-zero linear form $l_1 - l_3$ evaluates to zero, which is impossible on a basis.

Further, $l_1 l_3 = l_2^2$, $(l_1 + l_2) l_3 = (l_2 + l_3) l_2$, and $(l_1 + l_2) l_2 = (l_2 + l_3) l_1$ are equivalent.

c) If l_1, l_2, l_3, l_4 generate a 4-dimensional subspace then we get many equivalent equations. E.g., $l_1 l_4 = l_2 l_3$, $(l_1 + l_2) l_3 = (l_3 + l_4) l_1$.

Summarizing, we have a set (L1) of

$$G(k) := \begin{bmatrix} k \\ 2 \end{bmatrix} + 28 \begin{bmatrix} k \\ 3 \end{bmatrix} + 280 \begin{bmatrix} k \\ 4 \end{bmatrix} \tag{L1}$$

equations, which is far below the trivial estimate $(2^k - 1)^4$.

2.12. Remark. —— To simplify the following formulas, we denote by $N := (2^n - 1) \cdot \ldots \cdot (2^n - 2^{k-1})$ the number of linearly independent k-tuples in $\mathbb{F}_{2^n}$. Then, we get the following upper bounds for the number of solutions of the equations with $\mathbb{F}_2$-linearly independent k-tuples in $\mathbb{F}_{2^n}$.

i) The only relevant equation in the 2-dimensional case is $\frac{l_1}{l_2} = \frac{l_2}{l_1+l_2}$. It is equivalent to $l_1^2 + l_1l_2 + l_2^2 = 0$. For each value of l_1, at most two values of l_2 lead to a solution. Thus, at most $\frac{2}{2^n-2}N$ linearly independent k-tuples are solutions.

ii) In the 3-dimensional case, each of the 28 equations is equivalent to an equation of the form $\frac{l_1}{l_2} = \frac{l_3}{al_1+bl_2+cl_3}$ for $a, b, c \in \mathbb{F}_2$. Here, l_1, l_2, l_3 is a basis of the 3-dimensional subspace. This is equivalent to $l_1(al_1+bl_2)+(cl_1+l_2)l_3 = 0$. As cl_1+l_2 is a non-zero linear form, we get $l_3 = \frac{l_1(al_1+bl_2)}{cl_1+l_2}$. For each pair of linearly independent values for l_1, l_2, we get one solution for l_3. Thus, at most $\frac{1}{2^n-4}N$ linearly independent k-tuples are solutions.

iii) Let the linear forms l_1, l_2, l_3, l_4 be linearly independent. The equation $\frac{l_1}{l_2} = \frac{l_3}{l_4}$ is equivalent to $l_1l_4 = l_2l_3$. For fixed non-zero values of l_1, l_2, and l_3, we get one solution for l_4. Thus, at most $\frac{1}{2^n-8}N$ linearly independent k-tuples are solutions.

2.13. Proposition. —— *Assume $n \geq 4k-6$. Then, there exists a good element in $\mathrm{Gr}_{k,n}(\mathbb{F}_2)$.*

Proof: We have to show that the number of linearly independent k-tuples is bigger than the number of bad bases that solve at least one of the equations in (1). Again, we denote by $N := (2^n-1)(2^n-2)\cdots(2^n-2^{k-1})$ the number of linearly independent k-tuples in $\mathbb{F}_{2^n}$.

In the case $k = 3$, the counting arguments above exclude at most

$$N \cdot \left(\begin{bmatrix} 3 \\ 2 \end{bmatrix} \frac{2}{2^n-2} + \begin{bmatrix} 3 \\ 3 \end{bmatrix} 28 \frac{1}{2^n-4} \right)$$

bad bases. Thus, in the case $\frac{14}{2^n-2} + \frac{28}{2^n-4} < 1$, a good subspace exists. This inequality is satisfied for all $n \geq 6$.

In the case $k \geq 4$, the counting arguments above exclude at most

$$N \cdot \left(\begin{bmatrix} k \\ 2 \end{bmatrix} \frac{2}{2^n-2} + \begin{bmatrix} k \\ 3 \end{bmatrix} 28 \frac{1}{2^n-4} + \begin{bmatrix} k \\ 4 \end{bmatrix} 280 \frac{1}{2^n-8} \right)$$

bad bases. We have to check that the bracket is less than 1. It is a staight forward computation to show that $n \geq 4k-6$ implies this. □

2.14. Remark. —— The equations encoding that two good k-dimensional subspaces U and V can be combined are simpler to count. When we fix a two-dimensional subspace of U and a two-dimensional subspace of V, we get 6 equations. In total, we get a set (L2) of

$$C(k) := 6 \begin{bmatrix} k \\ 2 \end{bmatrix}^2 = (2^k-1)^2(2^k-2)^2/6 \tag{L2}$$

bilinear equations.

For fixed U, each equation excludes at most $\frac{1}{2^n-2}N$ linearly independent k-tuples of vectors for a basis of V. Thus, many orbits can be combined, forming a code with a huge number of codewords.

2.15. Example. —— i) In the special case that U has dimension 3, the good orbit conditions lead to 35 equations (see Remark 2.11). Thus, at least $(2^n-1)(2^n-2)(2^n-4)-14(2^n-1)(2^n-4)-28(2^n-1)(2^n-2)$ linearly independent triples (b_1, b_2, b_3) exist, which lead to a good orbit. The combination of two orbits is controlled by $6\left[\begin{smallmatrix}3\\2\end{smallmatrix}\right]^2 = 294$ equations. Thus choosing one orbit excludes at most $294(2^n-1)(2^n-4)$ triples of vectors for a basis of subspaces for a combination. This shows that a greedy strategy can combine at least

$$\begin{aligned}&\frac{(2^n-1)(2^n-2)(2^n-4)-14(2^n-1)(2^n-4)-28(2^n-1)(2^n-2)}{294(2^n-1)(2^n-4)}\\=&\frac{(2^n-2)(2^n-4)-14(2^n-4)-28(2^n-2)}{294(2^n-4)}\end{aligned}$$

orbits. For $n = 64$, we can combine more than 10^{16} orbits.

ii) In the special case that U has dimension 4, we get 735 equations. The three groups of equations exclude at most $35\cdot 2(2^n-1)(2^n-4)(2^n-8)$, $28\cdot 15(2^n-1)(2^n-2)(2^n-8)$, and $280(2^n-1)(2^n-2)(2^n-4)$ of the $(2^n-1)(2^n-2)(2^n-4)(2^n-8)$ linearly independent 4-tuples (b_1, b_2, b_3, b_4) for a basis of a good subspace. The combinability of two orbits is controlled by 7350 equations. Thus, choosing one orbit excludes at most $7350(2^n-1)(2^n-4)(2^n-8)$ possible 4-tuples for a combination. This shows that a greedy strategy can combine at least

$$\frac{(2^n-2)\left(1-\frac{35\cdot 2}{2^n-2}-\frac{28\cdot 15}{2^n-4}-\frac{280}{2^n-8}\right)}{7350}$$

orbits. For $n = 64$, we can combine more than 10^{15} orbits. For $n = 128$, we can combine more than 10^{34} orbits.

2.16. Proposition. —— *For all n and k, at least*

$$\frac{(2^n-2)\left(1-\begin{bmatrix}k\\2\end{bmatrix}\frac{2}{2^n-2}-\begin{bmatrix}k\\3\end{bmatrix}\frac{28}{2^n-4}-\begin{bmatrix}k\\4\end{bmatrix}\frac{280}{2^n-8}\right)}{6\begin{bmatrix}k\\2\end{bmatrix}\cdot\begin{bmatrix}k\\2\end{bmatrix}}$$

combinable good subspaces exist.

Proof: We use the greedy method to construct a code. I.e., we enumerate all linearly independent k-tuples. Then, we eliminate all that do not lead to a good subspace. Then, we choose one of them for our code. Now, we eliminate all that are not combinable with the one chosen. We repeat the last two steps until the enumerated list is exhausted. The estimates given above show that the result will contain at least the given number of good combinable subspaces. □

2.17. Corollary. ——

$$\mathcal{A}_2(n, 2(k-1), k) \geq 6\frac{(2^n-1)(2^n-2)}{[(2^k-1)(2^k-2)]^2} \left(1 - \begin{bmatrix} k \\ 2 \end{bmatrix} \frac{2}{2^n-2} - \begin{bmatrix} k \\ 3 \end{bmatrix} \frac{28}{2^n-4} - \begin{bmatrix} k \\ 4 \end{bmatrix} \frac{280}{2^n-8}\right).$$

Proof: This is the lower bound of 2.16 multiplied with the length of the orbits. □

2.18. Remark. —— There is a trivial upper bound for the maximal possible number of codewords in a constant dimension subspace code where two codewords have at most 1-dimensional intersection. For this, note that each 2-dimensional subspace is contained in at most one of the subspaces used in the code. And each subspace contains several of them. This leads to the known [WXS] upper bound

$$\begin{bmatrix} n \\ 2 \end{bmatrix} \Big/ \begin{bmatrix} k \\ 2 \end{bmatrix} = \frac{(2^n-1)(2^n-2)}{(2^k-1)(2^k-2)} \geq \mathcal{A}_2(n, 2(k-1), k).$$

for the size of the code.

2.19. Remark. —— Note that the lower and the upper bounds for the possible size of a code are similar. The quotient of the upper and the lower bound is asymptotically equal to $\begin{bmatrix} k \\ 2 \end{bmatrix}$ for $n \to \infty$ as long as we have $k \leq n/5$.

2.20. Remark. —— Following [EKW], the coding and the decoding algorithms for such codes are easy and fast. But one disadvantage is obvious. A representative of each orbit has to be stored. We will solve this problem in the next section by choosing a transversal of the orbits with more structure.

3 The new code

3.1. Recall. —— It is well known that the Riemann sphere $\mathbb{C} \cup \{\infty\}$ is a 2-dimensional $\mathbb{R}$-manifold. The automorphism group preserving the complex structure is given by fractional linear transformations $z \mapsto \frac{az+b}{cz+d}$. These are called *Möbius transformations.* Every fractional-linear map is constant or bijective $\mathbb{C} \cup \{\infty\} \to \mathbb{C} \cup \{\infty\}$. The inverse of such a maps is again fractional linear. It can be computed by inverting the companion matrix. In fact, the automorphism group is $\mathrm{PGL}_2(\mathbb{C})$.

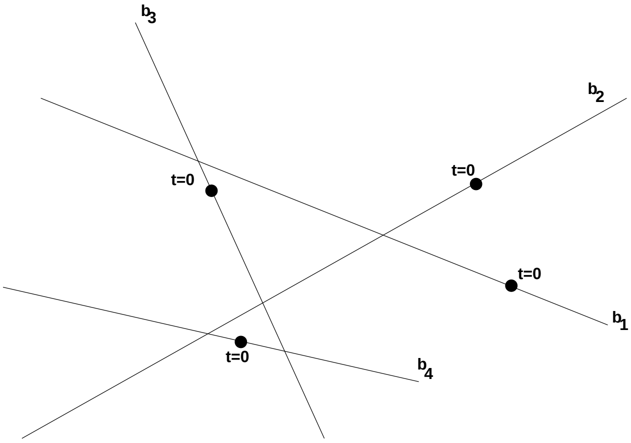

Figure 1: Initial set of lines

We get a geometry of lines and circles. $\mathrm{PGL}_2(\mathbb{C})$ acts transitively on these curves. Two different curves have at most two points in common.

All this carries over to $\mathbb{F}_{q^2} \cup \{\infty\}$ as a 2-dimensional space over $\mathbb{F}_q$. In the finite case, it is the well known construction of a *Miquelian inversive plane* starting from a quadratic field extension [De, p. 257]. One way to define a circle is as the image of a line. Removing one point from this inversive plane leads to the affine plane $\mathbb{F}_q^2$ [De, p. 253].

3.2. Algorithm for Construction. —— To construct the new code of k-dimensional subspaces in $\mathbb{F}_{2^{2n}}$, we start with k arbitrary affine lines in $\mathbb{F}_{2^{2n}}$. I.e., we choose k maps $b_i \colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^{2n}}$ by setting $b_i(t) := a_i + s_i t$ for $a_i, s_i \in \mathbb{F}_{2^{2n}}$. We intend to use $\Gamma \colon t \mapsto \langle b_1(t), \dots, b_k(t) \rangle$ as a transversal with 2^n elements.

In a first step, we have to check that all subspaces in the transversal are good. Here, we have to check that the $b_i(t)$ are linearly independent for all t. Further, the equations (1) for bad subspaces will lead to equations for bad values of t. If none of them is solvable, all subspaces in the transversal are good. Otherwise, we have to exclude the solutions as values for t. Usually, these equations will have no solutions.

In a second step, we have to check that $\Gamma(t_1)$ and $\Gamma(t_2)$ are combinable for each pair $t_1 \neq t_2$ in $\mathbb{F}_{2^n}$. Here, the equations from 2.14 will lead to equations for t_1, t_2. On the average, each equation will exclude one value for t.

3.3. Lemma. —— *Assume the $a_i \in \mathbb{F}_{2^{2n}}$ are $\mathbb{F}_2$-linearly independent. Then, for at most $2^k - 1$ values of t, the subspace generated by $b_i(t)$ has dimension less then k.*

Proof: We assume that a non-trivial linear combination leads to zero. I.e., we have $c_1(a_1 + s_1 t) + \dots + c_k(a_k + s_k t) = 0$ for $c_i \in \{0, 1\}$ not all zero. As the a_i are assumed to be $\mathbb{F}_2$ linearly independent, a non-zero equation for t results. This has at most one solution in $\mathbb{F}_{2^{2n}}$. □

3.4. Remark. —— We have proven that $\langle b_1(t), \ldots, b_k(t)\rangle$ has dimension k for at least $2^n - 2^k + 1$ values of t.

3.5. Remark. —— The quotient sets $Q(< b_1, \ldots, b_k >)$ were used in section 2 to describe good and combinable subspaces. Now, we get $(2^k - 1)(2^k - 2)$ parametrized quotients $t \mapsto \frac{l_i(b_1(t),\ldots,b_k(t))}{l_j(b_1(t),\ldots,b_k(t))}$. These are circles in the inversive plane. The circles might degenerate to a point.

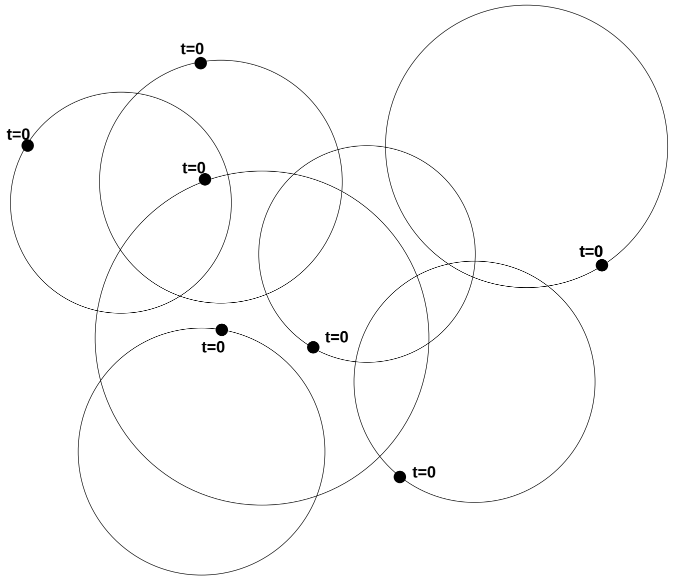

Figure 2: Quotient circles

3.6. Lemma. —— *Assume that $t = 0$ leads to a good orbit. Then, each of the equations in 2.8 for good orbits excludes at most two values of t.*

Proof: All equations are of degree 2. Plugging the $b_i(t)$ into one of them leads to a quadratic equation for t. This is not the zero equation as $t = 0$ is not a solution. As a consequence, each equation excludes at most 2 values of t. □

3.7. Remark. —— For randomly chosen a_i, s_i, we expect that no value of $t \in \mathbb{F}_{2^n}$ is excluded by the conditions of Lemma 3.3 and Lemma 3.6. To explain this, note that the equations constructed are defined over $\mathbb{F}_{2^{2n}}$. Usually, they have no solution in the smaller field $\mathbb{F}_{2^n}$. All computed examples confirm this expectation.

Now we turn to the second part of the construction. I.e., we analyze whether the subspaces are combinable.

3.8. Lemma. —— *Let Γ be as above. Each equation of Remark 2.14(encoding pairwise different quotients) for $\Gamma(t_1)$ and $\Gamma(t_2)$ excludes at most 2 values for t or it excludes all possible values.*

Proof: The quotient map is given by

$$t \mapsto \frac{\sigma_1 t + \alpha_1}{\sigma_2 t + \alpha_2}\,.$$

The σ_i and α_i are linear combinations of s_i and a_i. The image of each quotient map is a circle or a point. Two circles have at most two points in common or they coincide. □

3.9. Remark. —— It would be nice to have a criterion that excludes the degeneration of a quotient circle to a point or the case of two identical circles.

These degenerations seem to be very unlikely. We never observed them in a computed example.

3.10. Definition. —— In the following, we call the parameters that lead to intersection points of quotient circles (Lemma 3.8) and the parameters that are excluded by the other conditions (Lemma 3.3 and Lemma 3.6) the *exceptional set* of the code.

3.11. Remark. —— The arguments above show that, in a non-degenerated case, the exceptional set has at most $\epsilon := 2^k - 1 + 2(2^k - 1)(2^k - 2)[(2^k - 1)(2^k - 2) - 1]$ elements. Therefore, we can construct a code with $(2^{2n} - 1)(2^n - \epsilon)$ codewords.

4 Experiments

4.1. Example —— [A 3-dimensional subspaces code in $\mathbb{F}_{2^{64}}$] We chose a_i, c_i for $i = 1, 2, 3$ by a random number generator. We repeated this 100 times.

We had to exclude about 300 parameter values, each time. The smallest number of exclusions was 234. The exclusions were always caused by intersections of quotient-circles (Lemma 3.8). The resulting code has $(2^{64} - 1)(2^{32} - 234)$ codewords.

4.2. Example —— [A 4-dimensional subspaces code in $\mathbb{F}_{2^{128}}$] We chose the values of a_i, b_i by a random number generator. We repeated this 100 times.

We had to exclude about 7250 parameter values in each step. The smallest number of exclusions was 7044. The exclusions were always caused by the conditions given in Lemma 3.8. The resulting code has $(2^{128} - 1)(2^{64} - 7044)$ codewords.

4.3. Remark. —— The excluded parameters are given by the intersections of the quotient circles. These are $C(k)$ independent conditions. Each of them leads to a quadratic equation for the parameter. On average, it has one solution. Therefore the number of excluded parameters depends only on k.

Thus, a 3-dimensional subspace code in a 128- or 256-dimensional ambient space also has about 300 exceptions and about 2^{192}, resp. 2^{384}, codewords.

5 Construction, encoding, and decoding

The numbers of codewords of the constructed codes are slightly below an exact power of 2. It is easy to fill the gap by iterating the construction. In this section, we will first give an algorithm for the construction of such a code. After that we will give two algorithms for encoding and decoding of such a code.

5.1. Algorithm for Construction. —— We suggest the following strategy to construct a code with 2^{3n} codewords.
1) Choose the values of a_i and c_i $(i = 1, \ldots n)$, two times. I.e., prepare two codes C_1 and C_2.

2) Compute the exceptional set of parameters for the two codes.

3) Check that the exceptional sets are disjoint. If not, restart at the first step.

4) Define the fill-set to be the exceptional set of C_1 and one additional value t_0 which is in none of the two exceptional sets.

5) For each parameter in the fill-set, compute all quotients generated by C_2 and this parameter.

6) Test for all quotients, computed in step 5), whether they occur as quotients generated by C_1.

7) If these two quotient sets are disjoint we are done. Otherwise, we have to restart in step 1).

5.2. Remark. —— The algorithm will give us $2^n + 1$ combinable subspaces of $\mathbb{F}_{2^{2n}}$, giving a code with $(2^n + 1)\,(2^{2n} - 1) = 2^{3n} + 2^{2n} - 2^n - 1$ codewords. The parameters a_i and c_i describe the code and are known to sender and receiver. We used this construction method in the experiments above to get a code of 3-dimensional and a code of 4-dimensional subspaces in $\mathbb{F}_{2^{64}}$. In both cases, the first trial was successful.

5.3. Algorithm for Encoding. —— For practical encoding, we explain how to transform an element $(t, z) \in \mathbb{F}_{2^n} \times \mathbb{F}_{2^{2n}}$ (which is more or less a sequence of $3n$ bits) into a subspace.
1) Use t as a parameter to get an orbit of the first code.

2) In the improbable case that t is in the fill-set, switch to the second code.

3) In each case, a $\mathbb{F}_2$-subspace $\Gamma(t)$ of $\mathbb{F}_{2^{2n}}$ results. This is a canonical representative of a Singer orbit.

4) Translate this subspace inside the orbit by multiplying with z.

5) In the improbable case that z is zero, use the the additional parameter t_0 introduced in step 4). Code $(t, 0)$ by an element of the $\mathbb{F}^*_{2^{2n}}$-orbit of the subspace with parameter t_0 in the second code.

6) To do this, we need an injective map $\mathbb{F}_{2^n} \to \mathbb{F}^*_{2^{2n}}$. This is given by addition of an arbitrary element $r_0 \in \mathbb{F}_{2^{2n}} \setminus \mathbb{F}_{2^n}$.

5.4. Remark. —— In the algorithm above the second code C_2 works as an backup, which is used in the case that the first part t of the encoded word was from the exceptional set of the code C_1 or if the second part z is zero which is a non-valid element from the Singer cycle.

5.5. Remark. —— In principle, the decoding algorithm of [EKW] for subspace codes with minimal distance $2(k-1)$ applies to this code. The basic idea of the decoding algorithm is to start with the quotients generated by the vector space received. We give a decoding algorithm below, which allows the decoding including the correction of up to $(k-2)$ errors (resp. erasures).

5.6. Algorithm for Decoding. —— Input is a received subspace $U \subset \mathbb{F}_{2^{2n}}$.

1) Loop through all 2-dimensional subspaces in U.

2) For each subspace, compute one quotient of a pair of different non-zero elements.

3) For each pair of $\mathbb{F}_2$-linear combinations of the $b_i(t)$ of the first code C_1, compute the value of $t \in \mathbb{F}_{2^{2n}}$ which leads to the quotient computed in 2).

4) If t is in the subfield $\mathbb{F}_{2^n}$ and not in the exceptional set of the code C_1 then the parameter found is possible.

5) For each possible parameter t found, reconstruct the translation factor z by a division in $\mathbb{F}_{2^{2n}}$.

6) For each pair (t, z), compute the distance between the space received and the space encoding (t, z).

7) If the distance is at most $(k-2)$ then return (t, z) as the decoding result.

8) Redo the last 5 steps with the second code C_2 instead of C_1. Continue in step 4) only if the parameter is in the fill-set. Take special care of t_0 in step 7).

9) Return an error-message. There are too many errors for decoding.

5.7. Remarks. —— i) As the subspace distance between the received space U and the original codeword is at most $(k-2)$, there is only one possible parameter t finishing the algorithm in step 7).

ii) If the received subspace has a bigger dimension than k then one can restrict to a lower dimensional part [EKW]. This reduces the number of pairs to be handled in step 1). The number of 2-dimensional subspaces to be checked is in the worst case $\begin{bmatrix} k \\ 2 \end{bmatrix} = \frac{(2^k-1)(2^k-2)}{6}$.

iii) As the quotient map is fractional linear, its inverse is fractional linear, too. In practice, we suggest to precompute the inverse maps in an initialization routine in order to speed up step 3).

iv) For the computation of the distance in step 6), we suggest to use the algorithm in [SE].

6 Generalizations

Let us mention a few generalizations of the construction given above.

i) It is possible to do the same for $\mathbb{F}_q$ instead of $\mathbb{F}_2$.

ii) As there is still a large gap between the size of the constructed code and the lower bound for the possible size of the code, one could try to combine several such $\mathbb{F}_{2^n}$-families of $\mathbb{F}^*_{2^{2n}}$-orbits. This would lead to significantly bigger exceptional sets that have to be managed.

iii) A less obvious generalization is the following. Think of $\mathbb{F}_{2^{mn}}$ as a m-dimensional $\mathbb{F}_{2^n}$-space. Then, one replaces the parameter t by $m-1$ parameters in $\mathbb{F}_{2^n}$.

For example, one could choose $n = 64$ and $m = 4$. Then, the resulting code has about $2^{4 \cdot 64 - 64}$ codewords in $\mathbb{F}_{2^{256}}$. The initial construction, i.e. $n = 128$ and $m = 2$, would give us only $2^{4 \cdot 64 - 128}$ codewords in the same ambient space.

Doing this, the exceptional set becomes by far more complicated. Each equation will lead to one $(m-2)$-dimensional variety of exceptional parameters.

As above, one could try to modify the code in the exceptional cases, using an other code, constructed in the same way. This will only work in most of the cases. Finitely many $(m-3)$-dimensional varieties in the parameter space will still lead to problems.

One could repeat this exception handling process. One expects that each iteration will reduce the dimension of the remaining exceptional set by one.

In practice, one could try to use Gröbner bases for an explicit description of the varieties of exceptional parameters. It is not clear whether this is suitable for applications.

References

[Be] W. Benz: Vorlesungen über Geometrie der Algebren, Springer 1973.

[M] W. Bosma, J. Cannon, and C. Playoust: The Magma algebra system. I. The user language. J. Symbolic Comput., 24(3-4):235-265, 1997.

[De] P. Dembowski: Finite Geometries, Springer 1986.

[EKW] A.-S. Elsenhans, A. Kohnert, and A. Wassermann: Construction of Codes for Network Coding, in Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems – MTNS, Budapest, Hungary, 2010.

[ES] T. Etzion and N. Silberstein: Error-correcting codes in projective spaces via rank-metric codes and Ferrers diagrams, IEEE Transactions on Information Theory, vol. 55, no. 7, pp. 2909–2919, July 2009.

[EV] T. Etzion and A. Vardy, "Error-correcting codes in projective space," in Proceedings of ISIT 2008. IEEE International Symposium on Information Theory, July 2008, pp. 871–875

[KoKu] A. Kohnert and S. Kurz: Construction of large constant dimension codes with a prescribed minimum distance, in Proceedings of MMICS, Lecture Notes in Computer Science, J. Calmet, W. Geiselmann, and J. Müller-Quade, Eds., vol. 5393. Springer, 2008, pp. 31–42.

[KK] R. Kötter, and F. R. Kschischang: Coding for errors and erasures in random network coding. IEEE Transactions on Information Theory, 54(8):3579-3591, July 2008.

[MGR] F. Manganiello, E. Gorla, and J. Rosenthal, "Spread codes and spread decoding in network coding," in Proceedings of the 2008 IEEE International Symposium on Information Theory, Toronto, Canada, 2008, pp. 851–855.

[SE] N. Silberstein, and T. Etzion: Coding Theory in Projective Space, `arXiv:0805.3528,` 2008.

[SKK] D. Silva, F. Kschischang, and R. Koetter, A rank-metric approach to error control in random network coding, IEEE Transactions on Information Theory, vol. 54, no. 9, pp. 3951–3967, Sept. 2008.

[S] V. Skachek: Recursive code construction for random networks, `arXiv:0806.3650`, 2008.

[TK] A.-L. Trautmann and J. Rosenthal, New improvements on the Echelon-Ferrers construction, in Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems – MTNS, Budapest, Hungary, 2010, pp. 405–408.

[WXS] H. Wang, C. Xing, and R.M. Safavi-Naini: Linear authentication codes: bounds and constructions, IEEE Transactions on Information Theory, 49: 866-872, April 2003.