On cardinality of network subspace codes

Alexander Shishkin, Ernst Gabidulin, Nina Pilipchuk

Moscow Institute of Physics and Technology

ACCT-2014, Kaliningrad, 07-13.09.2014

Outline

Subspace distance

- $\mathbb{K}_q^n$ - n-dimensional vector space over $GF(q)$.
- $\mathcal{A}(n)$ - the set of all subspaces in $\mathbb{K}_q^n$.
- For $\mathcal{U}, \mathcal{V} \in \mathcal{A}(n)$ Grassmanian distance is defined as:

$$d_{sub}(\mathcal{U}, \mathcal{V}) = \dim(\mathcal{U} \cup \mathcal{V}) - \dim(\mathcal{U} \cap \mathcal{V}).$$

Subspace codes

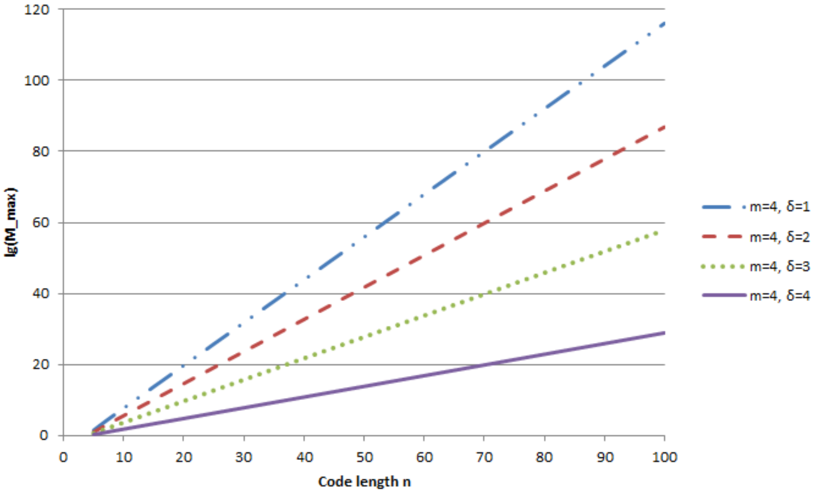
Subspace $[n, M, d_{sub}, k]$-code is the set of k-dimensional subspaces in $\mathcal{A}(n)$, which cardinality equals M and the Grassmanian distance between any pair of subspaces is not less than d_{sub}.

There is an upper bound for subspace codes cardinality (Wang, 2003):

$$M_{max} = \frac{(q^n - 1)(q^{n-1} - 1) \dots (q^{n-k+\delta} - 1)}{(q^k - 1)(q^{k-1} - 1) \dots (q^\delta - 1)},$$

where $\delta = \frac{d_{sub}}{2}$.

Upper bound as a function of code length



Silva-Koetter-Kschishang codes

SKK-code is a set of $k \times n$ matrices over the base field $GF(q)$:

$$\mathcal{C} = \left\{ \begin{bmatrix} I_k & M \end{bmatrix} \right\},$$

where I_k is the identity matrix of order k, submatrix M is the $k \times (n-k)$ rank code matrix over the field $GF(q)$. Notice that $d_{sub}(\mathcal{C}) = 2d_r$, where d_r is the rank distance of this subcode.

Cardinality of this code equals:

$$M_{SKK} = \begin{cases} q^{(n-k)(k-\delta+1)}, n \geq 2k; \\ q^{k(n-k-\delta+1)}, n < 2k. \end{cases}$$

Multicomponent codes with zero prefix

Multicomponent codes with zero prefix (Gabidulin-Bossert, 2008) consist of several components:

$$\mathcal{C}_0 = \left\{ \begin{bmatrix} I_k & M_0 \end{bmatrix} \right\},$$

$$\mathcal{C}_1 = \left\{ \begin{bmatrix} O_k^\delta & I_k & M_1 \end{bmatrix} \right\},$$

$$\mathcal{C}_2 = \left\{ \begin{bmatrix} O_k^\delta & O_k^\delta & I_k & M_2 \end{bmatrix} \right\},$$

$$\dots$$

$$\mathcal{C}_r = \left\{ \begin{bmatrix} O_k^\delta & \dots & O_k^\delta & I_k & M_r \end{bmatrix} \right\},$$

where I_k is the identity matrix of order k, M_i denotes rank code submatrices, O_k^δ is $k \times \delta$ zero matrix.

Multicomponent codes with zero prefix

Cardinality of multicomponent codes with zero prefix can be found as:

$$M_0 = M_{SKK} + S_1 + S_2 + 1,$$

where

$$S_1 = \sum_{i=1}^{s1} 2^{k(n-m-i\delta)},$$

$$S_2 = \sum_{i=1}^{s2} 2^{k_i m}.$$

Multicomponent codes based on greedy search algorithm

- On the first step for the set of all binary vectors-indices of length n and Hamming weight k the cardinality of corresponding rank subcodes is defined.
- Lexicographically the first vector-index corresponds to the SKK-code and it is used as the first code component.
- Then greedy search starts for the code component with the biggest cardinality among the remaining.
- If its subspace distance to all already added to code subspaces is not less than d_{sub} it is included into the code, and so on.

Disadvantage: there is no equation for M_{greedy}.

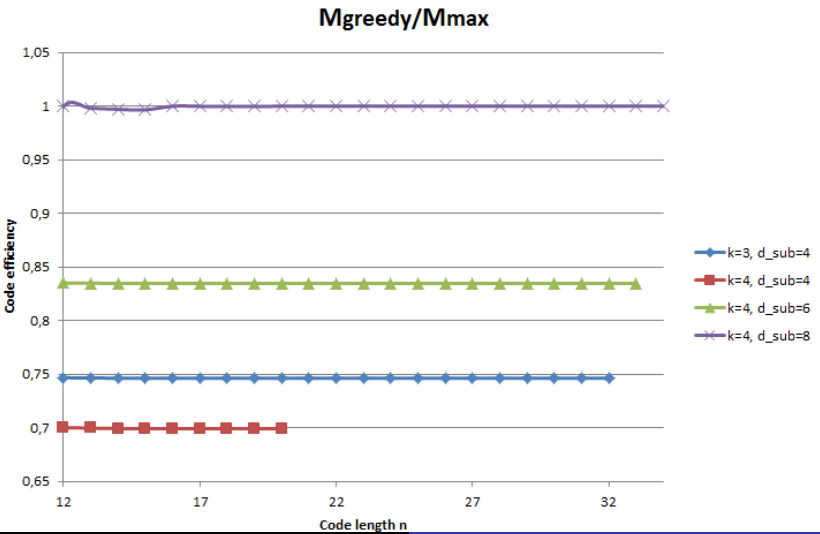
Definition of efficiency

The code efficiency is defined as the ratio of its actual cardinality to the upper bound for fixed parameters:

$$\eta = \frac{M}{M_{max}}.$$

We investigate η_{SKK}, η_0 and η_{greedy} for different code parameters.

Efficiency dependence on code length



Examples of codes efficiency

$$n = 16, \delta = 5$$

k	2	3	4	5	6	7	8
η_{skk}	-	-	-	0,969	0,954	0,946	0,942
η_0	-	-	-	0,999	0,954	0,946	0,942
η_{greedy}	-	-	-	0,999	0,954	0,946	0,942

$$n = 16, \delta = 4$$

k	2	3	4	5	6	7	8
η_{skk}	-	-	0,938	0,908	0,894	0,887	0,884
η_0	-	-	1	0,912	0,894	0,887	0,884
η_{greedy}	-	-	1	0,912	0,894	0,887	0,884

Examples of codes efficiency

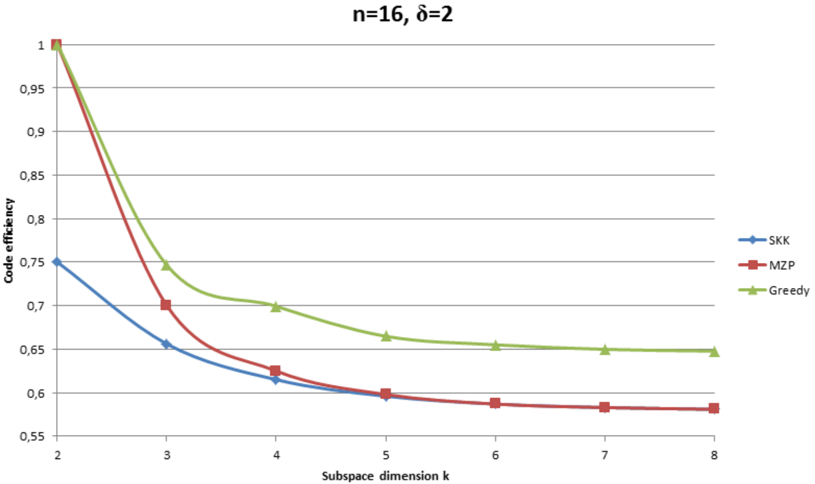
$$n = 16, \delta = 3$$

k	2	3	4	5	6	7	8
η_{skk}	-	0,875	0,82	0,794	0,782	0,777	0,774
η_0	-	1	0,823	0,796	0,782	0,777	0,774
η_{greedy}	-	1	0,835	0,798	0,785	0,778	0,775

$$n = 16, \delta = 2$$

k	2	3	4	5	6	7	8
η_{skk}	0,75	0,656	0,615	0,596	0,587	0,583	0,581
η_0	1	0,7	0,625	0,598	0,587	0,583	0,581
η_{greedy}	1	0,746	0,699	0,665	0,655	0,650	0,648

Efficiency comparison



Conclusion

- Upper bound for codes cardinality rises as a power of code length.
- Code efficiency weakly depends on code length.
- For all the considered codes efficiency clearly depends on the subspace distance: it rises with the growth of the subspace distance at the fixed values of other parameters.
- For codes with zero prefix the upper bound of cardinality attains in the case of maximum code distance. Thus the cardinality of codes with zero prefix can be used as a lower bound for subspace codes cardinality.

Conclusion

Thank you!