# Recovering a Parent Code for Subcodes of Maximal Rank Distance Codes

Alexei V. Ourivski*

ourivski@mail.ru

**Abstract**

Several attempts have been made to strengthen the security of the GPT public key cryptosystem which is based on maximal rank distance codes, the Gabidulin codes. One of them is to publish a subcode instead of a full code in the hope that the subcode does not expose the structure of the code. In this paper we present an algorithm that recovers a parent Gabidulin code for a random subcode of it. When the difference between dimensions of the subcode and code itself is not too high the algorithm is of polynomial complexity. Consequently we show that publishing a pure subcode (without any distortion) makes the cryptosystem either insecure or the system falls outside the region of practical interest.

## 1 Maximal Rank Distance Codes — the Gabidulin Codes

Let $\mathbf{F}_N$ be a finite field with $q^N$ elements, and let $\mathbf{F}_1$ be the base field of $q$ elements; $q$ is a power of a prime. Let $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbf{F}_N^n$. The *rank weight*, or simply *rank*, $r(\mathbf{x}|\mathbf{F}_1)$ of $\mathbf{x}$ over $\mathbf{F}_1$ is defined as the maximal number of $x_i$ that are linearly independent over $\mathbf{F}_1$.

The *rank distance* $d_{\mathrm{r}}(\mathbf{x}, \mathbf{y})$ between two vectors $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}, \mathbf{y} \in \mathbf{F}_N^n$, is the rank of the difference $\mathbf{x} - \mathbf{y}$: $d_{\mathrm{r}}(\mathbf{x}, \mathbf{y}) = r(\mathbf{x} - \mathbf{y}|\mathbf{F}_1)$. For any linear $(n, k)$ code $\mathcal{C}$ the *rank distance* $d_r$ is defined by $d_r = \min\{r(\mathbf{x}|\mathbf{F}_1) \mid \mathbf{x} \in \mathcal{C}, \mathbf{x} \neq \mathbf{0}\}$.

In this paper we are concerned with subcodes of Gabidulin rank codes. A $(n, k)$ Gabidulin code $\mathcal{C}_g$ over $\mathbf{F}_N$ is defined by its generator matrix

$$\mathbf{G} = \begin{bmatrix} g_1 & g_2 & \cdots & g_n \\ g_1^{[1]} & g_2^{[1]} & \cdots & g_n^{[1]} \\ g_1^{[2]} & g_2^{[2]} & \cdots & g_n^{[2]} \\ \vdots & \vdots & \ddots & \vdots \\ g_1^{[k-1]} & g_2^{[k-1]} & \cdots & g_n^{[k-1]} \end{bmatrix}, \tag{1}$$

where $g_j \in \mathbf{F}_N$ are all linear independent over $\mathbf{F}_1$, $j = 1, \ldots, n$; $g^{[i]} = g^{q^i}$ means the $i$-th Frobenius power of $g$.

*The author is with Samsung Research Center, Moscow, Russia.

Any matrix of the form (1) is called a Frobenius matrix induced by the generating vector $\mathbf{g} = (g_1, g_2, \ldots, g_n)$. A parity check matrix of the code $\mathcal{C}_g$ is also a Frobenius matrix

$$\mathbf{H} = \begin{bmatrix} h_1 & h_2 & \cdots & h_n \\ h_1^{[1]} & h_2^{[1]} & \cdots & h_n^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ h_1^{[n-k-1]} & h_2^{[n-k-1]} & \cdots & h_n^{[n-k-1]} \end{bmatrix} \tag{2}$$

with generating vector $\mathbf{h} = (h_1, h_2, \ldots, h_n)$, $r(\mathbf{h}|\mathbf{F}_1) = n$.

The code $\mathcal{C}_g$ has the minimum rank distance $d_r = n - k + 1$ and it reaches the upper bound for the rank distance [1], hence this is a *maximal rank distance* code. For $\mathcal{C}_g$ there exists *fast decoding algorithms* correcting any errors of rank up to $t_r = \lfloor (d_r - 1)/2 \rfloor$.

## 2 Cryptosystems Using Subcodes of Gabidulin Codes

A public key cryptosystem of a McEliece-type that uses Gabidulin codes is the GPT cryptosystem. The public key in this system is a matrix

$$\mathbf{G}_{\text{pub}} = \mathbf{SG} + \mathbf{X}, \tag{3}$$

where $\mathbf{G}$ is given by (1), $\mathbf{S}$ is a $k \times k$ non-singular *scramble* matrix, and $\mathbf{X}$ is a randomly chosen $k \times n$ *distortion* matrix such that $r(\mathbf{X}|\mathbf{F}_1) = t_1 < t_r$, where $t_1$ is a *design* parameter. Here $r(\mathbf{X}|\mathbf{F}_1)$ is the *column rank* of $\mathbf{X}$ over the field $\mathbf{F}_1$ defined as the maximal number of columns that are linearly independent over $\mathbf{F}_1$.

A vector of a plaintext $\mathbf{m} \in \mathbf{F}_N^k$ is encrypted as

$$\mathbf{c} = \mathbf{mG}_{\text{pub}} + \mathbf{e} = \mathbf{mSG} + (\mathbf{mX} + \mathbf{e}), \tag{4}$$

where $\mathbf{e}$ is a randomly chose *artificial* vector of errors of rank $r = t_r - t_1$ or less. Decryption is performed by decoding the vector $\mathbf{c}$ to obtain $\mathbf{mS}$ and then $\mathbf{m}$.

Two attacks against GPT PKC were invented by Gibson. They recover a decoder for the published code, or equivalently they find a representation of the public key

$$\mathbf{G}_{\text{pub}} = \mathbf{S}'\mathbf{G}' + \mathbf{X}', \tag{5}$$

where $\mathbf{S}'$ is non-singular, $\mathbf{G}'$ is of the form (1), and $r(\mathbf{X}'|\mathbf{F}_1) \le r(\mathbf{X}|\mathbf{F}_1)$.

The first attack on average requires $O(n^3 q^{sN})$ arithmetical operations in $\mathbf{F}_N$, where $s = \min r(\mathbf{X}'|\mathbf{F}_N)$, and the minimum is taken over all decompositions of the form (5).

The complexity of the second attack is estimated as $O(k^3 + (k+t_1+2) f q^{f(k+2)})$ operations, here $t_1 = \min r(\mathbf{X}'|\mathbf{F}_1)$, and the minimum is taken over all decompositions of the form (5), $s$ is defined above. Gibson claims that in almost all cases *in practice* $f = \max(0, t_1 - 2s)$, and it is known how to choose $\mathbf{X}$ so that the expected value of $f = t_1 - s$.

In the light of these attacks, several modifications of the GPT PKC were introduced. One of them is to use a subcode of Gabidulin code instead of the code itself. Originally this idea was implemented in paper [2] by replacing a square matrix $\mathbf{S}$ in (3) for a rectangular one:

$$\widetilde{\mathbf{G}}_{\text{pub}} = \mathbf{S}_p\mathbf{G} + \mathbf{X}, \tag{6}$$

where $\mathbf{S}_p$ is $(k-p) \times k$ matrix over $\mathbf{F}_N$ of rank $k-p$ picking out a subcode $\mathcal{C}_p$ from the code $\mathcal{C}_g$.

Another implementation was proposed in [5], where authors used a Niederreiter-type system publishing a parity-check matrix of the subcode $\mathcal{C}_p$

$$\mathbf{H}_p = \mathbf{T} \begin{bmatrix} \mathbf{H} \\ \mathbf{A} \end{bmatrix}, \tag{7}$$

where $\mathbf{T}$ is a non-singular $(n-k+p) \times (n-k+p)$ matrix, $\mathbf{H}$ is given by (2), $\mathbf{A}$ — some $p \times n$ matrix defining the subcode.

However, as cryptanalysis is concerned the system with key (7) can be regarded as extremely simplified version of the system with key (6): Put in (6) $\mathbf{X} = \mathbf{0}$ and compute $\mathbf{S}_p$ from the equation $\mathbf{S}_p \mathbf{G} \mathbf{A}^T = \mathbf{0}$.

The rationale behind the design of the system assumed that a subcode is not a Gabidulin code itself, and what is more that the subcode has no obvious algebraic structure that enables recovering a fast decoding algorithm for the subcode both for a cryptographer and cryptanalyst. It is also supposed that the minimum rank distance of the subcode is strictly less than $n-k+p+1$ and it is taken to be equal to $d_r = n-k+1$, thus defined by the parent code. Therefore, for decrypting messages in both systems the parent code (either given by matrix $\mathbf{G}$ or $\mathbf{H}$) is only needed. Thus, to break both systems it is enough to recover *any* parent Gabidulin code in canonical form (1) or (2) for the given subcode.

## 3 Recovering a Parent Code for a Subcode

In this paper, we show that a system with public key $\mathbf{H}_p$ is insecure for any interesting parameters in practice. To do this we present an algorithm that computes a parent code for a random subcode of a Gabidulin code in polynomial time.

Split the matrix $\mathbf{H}$ into three parts $\mathbf{H} = [\mathbf{H}_1 \ \mathbf{H}_2 \ \mathbf{H}_3]$, where $\mathbf{H}_1$ — the first $n-k$ columns of $\mathbf{H}$, $\mathbf{H}_2$ — next $p$ columns, $\mathbf{H}_3$ is the last $k-p$ columns of $\mathbf{H}$. Similarly, split the generating vector $\mathbf{h} = (\mathbf{h}_1 \ \mathbf{h}_2 \ \mathbf{h}_3)$.

Denote $\ell = n-k+p$.

Without loss of generality, assume that the first $\ell$ columns of $\mathbf{H}_p$ forms a non-singular matrix. Since $\mathbf{H}_1$ is non-singular, $\mathbf{H}_p$ can be rewritten as

$$\mathbf{H}_p = \mathbf{T}^* \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 \\ \mathbf{O} & \mathbf{E}_p & \mathbf{B} \end{bmatrix} \tag{8}$$

for some non-singular $\mathbf{T}^*$, where $\mathbf{O}$ — $p \times (n-k)$ all-zero matrix, $\mathbf{E}_p$ is the identity matrix of order $p$.

Reduce $\mathbf{H}_p$ to a systematic form

$$\mathbf{H}_p^{syst} = [\mathbf{E}_\ell \ \mathbf{R}].$$

It is readily shown that

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_{13} - \mathbf{R}_{12}\mathbf{B} \\ \mathbf{B} \end{bmatrix},$$

where $\mathbf{R}_{13} = \mathbf{H}_1^{-1}\mathbf{H}_3$, $\mathbf{R}_{12} = \mathbf{H}_1^{-1}\mathbf{H}_2$.

Thus, the matrix $\mathbf{A}$ may equivalently be represented in the form $\mathbf{A} = [\mathbf{O}\ \mathbf{E}_p\ \mathbf{B}]$ and in fact is known.

We are going to show how the matrix $\mathbf{H}$ can be reconstructed on the basis of $\mathbf{H}_p$: Using elements of the matrix $\mathbf{R}$ we derive and solve a system of linear equations in the unknown components of $\mathbf{h}$. Hence a parent code for the given subcode will be found.

Notice that $[\mathbf{H}_1\ \mathbf{H}_2]\,\mathbf{R} = \mathbf{H}_3$, in other words

$$[\mathbf{H}_1\ \mathbf{H}_2\ \mathbf{H}_3]\begin{bmatrix}\mathbf{R}\\ -\mathbf{E}_{k-p}\end{bmatrix} = \mathbf{0}. \tag{9}$$

Let the matrix $\mathbf{B}$ have exactly $v$ linearly independent over $\mathbf{F}_1$ rows, $0 \le v \le p$. Then $\mathbf{B}$ can be written as

$$\mathbf{B} = \mathbf{P}_B\mathbf{B}_{\text{base}}, \tag{10}$$

where $\mathbf{B}_{\text{base}}$ — a $v \times (k-p)$ matrix, all rows of which are linearly independent over $\mathbf{F}_1$, and $\mathbf{P}_B$ is some $p \times v$ matrix over $\mathbf{F}_1$ of rank $v$. Equation (9) turns into

$$[\mathbf{H}_1\ \mathbf{H}_2^*\ \mathbf{H}_3]\begin{bmatrix}\mathbf{R}_{13} - \mathbf{R}_{12}^*\mathbf{B}_{\text{base}}\\ \mathbf{B}_{\text{base}}\\ -\mathbf{E}_{k-p}\end{bmatrix} = \mathbf{0}, \tag{11}$$

where $\mathbf{H}_2^* = \mathbf{H}_2\mathbf{P}_B$, $\mathbf{R}_{12}^* = \mathbf{R}_{12}\mathbf{P}_B = \mathbf{H}_1^{-1}\mathbf{H}_2^*$.

The matrix $\mathbf{H}^* = [\mathbf{H}_1\ \mathbf{H}_2^*\ \mathbf{H}_3]$ is a Frobenius $(k-p) \times (n-p+v)$ matrix with generating vector $\mathbf{h}^* = (\mathbf{h}_1\ \mathbf{h}_2^*\ \mathbf{h}_3)$, $\mathbf{h}_2^* = \mathbf{h}_2\mathbf{P}_B$, and $r(\mathbf{h}^*|q) = n-p+v$.

Let $\mathbf{R}_j$ be the $j$-th column of the matrix

$$\mathbf{R}^* = \begin{bmatrix}\mathbf{R}_{13} - \mathbf{R}_{12}^*\mathbf{B}_{\text{base}}\\ \mathbf{B}_{\text{base}}\\ -\mathbf{E}_{k-p}\end{bmatrix}. \tag{12}$$

Denote by $\mathbf{R}_j^{[i]}$ the column $\mathbf{R}_j$, each component of which is raised to the $q^i$-th power. Form a matrix

$$\mathbf{G}_R^T = \left[\mathbf{R}_1^{[N]}\mathbf{R}_1^{[N-1]}\cdots\mathbf{R}_1^{[N-n+k+1]}\mathbf{R}_2^{[N]}\mathbf{R}_2^{[N-1]}\cdots\mathbf{R}_2^{[N-n+k+1]}\cdots\mathbf{R}_{k-p}^{[N]}\cdots\mathbf{R}_{k-p}^{[N-n+k+1]}\right]. \tag{13}$$

By some obvious manipulations equation (11) is transformed into the following linear equation

$$\mathbf{h}^*\mathbf{G}_R^T = \mathbf{0}. \tag{14}$$

The matrix $\mathbf{G}_R$ consists of $(k-p)(n-k)$ rows and $n-p+v$ columns. Assume that there are more rows than columns in this matrix, i.e., $(k-p)(n-k) \ge n-p+v$. Otherwise the rate $R_p$ of the subcode will be too low: $R_p < 1/(n-k) = 1/(d_r-1)$.

**Lemma 1.** *Let the rank of* $\mathbf{B}_{\text{base}}$ *is* $b$. *Then the rank* $r_G$ *of* $\mathbf{G}_R$ *satisfies the following bounds*

$$n-p-1 \le r_G \le n-p-1+\min(v, b(n-k)).$$

*Proof.* (Sketch.) Any matrix $\mathbf{B}_{\text{base}}$ of rank $b$ may be represented as a product $\mathbf{B}_{\text{base}} = \mathbf{B}_1\mathbf{B}_2$, where $\mathbf{B}_1$ and $\mathbf{B}_2$ are $v \times b$ and $b \times (k-p)$ matrices both of rank $b$. It can be shown that rows of $\mathbf{G}_R^T$ corresponding to the matrix $\mathbf{B}_{\text{base}}$ are a matrix $\mathbf{C} = \mathbf{C}_1\mathbf{C}_2$, where $\mathbf{C}_1$ is a $v \times b(n-k)$ matrix. Therefore, the rank of $\mathbf{C}$ does not exceed $\min(v, b(n-k))$.

It is easy to prove that among the remaining rows of $\mathbf{G}_R^T$ there are exactly $n-p-1$ linearly independent ones. Thus, the rank of $\mathbf{G}_R$ upper-bounded by the value $n-p-1+\min(v, b(n-k))$ and lower-bounded by $n-p-1$. □

A more accurate lower bound for $r_G$ other than given by Lemma has not been established. However, a lot of simulations were conducted using the computer algebra system MAGMA. The matrix $\mathbf{B}$ was chosen randomly (using MAGMA's built-in pseudo-random number generator), the values of $v$ lay within the range 1 to $p$, and values of $b$ were in the range from 1 through $\min(v, k-p)$. For different fields of characteristic $q = 2$ and extension degree from $N = 24$ through $N = 64$, code lengths $n \leq N$, and for different $p = 1, \ldots, k-2$ and $d_r \geqslant 2$, in every examined case the rank of $\mathbf{G}_R$ was exactly on the upper bound $n-p-1+\min(v, b(n-k))$.

Apparently there exist matrices $\mathbf{B}$ such that $r_G$ is strictly less than $n-p-1+\min(v, b(n-k))$. But most likely a fraction of these matrices among all possible $v \times (k-p)$ matrices $\mathbf{B}$ is extremely low, and we did not encounter them in our simulations.

If the rank of $\mathbf{G}_R$ is exactly $r_G = n-p+v-1$, then by solving equation (14) we compute the vector $\mathbf{h}^*$ (to be more correct, we compute a multiple of $\mathbf{h}^*$ which defines the same code). If $r_G < n-p+v-1$, then a space of solutions of dimension

$$m = n - p + v - r_G \tag{15}$$

will be found. We are interested in solutions (vectors $\mathbf{h}^*$) whose rank is exactly $n-p+v$. If $m$ is a small value, then it is expected that only a few trials will be needed to find $\mathbf{h}^*$ with independent coordinates, and the total cost is estimated by $O(nm)$ operations. If $m$ is large enough, then solving (14) for $\mathbf{h}^*$ requires $O(nmq^{N(m-1)})$ operations in $\mathbf{F}_N$.

Once $\mathbf{h}^*$ is found, the matrices $\mathbf{H}_1$, $\mathbf{H}_2^*$, and $\mathbf{H}_3$ are known.

To find a full $\mathbf{H}$ the matrix $\mathbf{H}_2$ is needed. Since

$$\mathbf{h}_2\mathbf{P}_B = \mathbf{h}_2^*, \tag{16}$$

and $\mathbf{h}_2^*$ with $\mathbf{P}_B$ are known, solve this equation for $\mathbf{h}_2$ as follows. Let some $v$ rows of $\mathbf{P}_B$, say the upper ones, are independent. All components of $\mathbf{h}$ must be independent over $\mathbf{F}_1$, so choose the last $p-v$ components of $\mathbf{h}_2$ to be linearly independent of each other and of components of $\mathbf{h}_1$, $\mathbf{h}_3$, and $\mathbf{h}_2^*$. Then the first $v$ components of $\mathbf{h}_2$, corresponding to a non-singular $v \times v$ submatrix of $\mathbf{P}_B$ are easily calculated from (16).

Thus, a complete algorithm of recovering a parent Gabidulin code $\mathcal{C}_g$ for a given subcode $\mathcal{C}_p$ and computing decomposition (8) is as follows.

**Algorithm.**

1. Calculate a systematic form of the parity-check matrix $\mathbf{H}_p$ of the subcode: $\mathbf{H}_p^{syst} = [\mathbf{E}_m\ \mathbf{R}]$. Put the matrix $\mathbf{A}$ to be the lower $p$ rows of $\mathbf{H}_p^{syst}$: $\mathbf{A} = [\mathbf{O}\ \mathbf{E}_p\ \mathbf{B}]$.
2. Calculate the row rank $v$ of $\mathbf{B}$ over $\mathbf{F}_1$. Represent $\mathbf{B} = \mathbf{P}_B\mathbf{B}_{\text{base}}$, where $\mathbf{B}_{\text{base}}$ is a $v \times (k-p)$ matrix containing all independent rows of $\mathbf{B}$, $\mathbf{P}_B$ is some $p \times v$ $q$-ary matrix.

3. Calculate matrices $\mathbf{R}^*$ and $\mathbf{G}_R$ according (12) and (13).
4. Solve equation (14) for $\mathbf{h}^*$. Put $\mathbf{h}^* = (\mathbf{h}_1\ \mathbf{h}_2^*\ \mathbf{h}_3)$.
5. Choose randomly $p-v$ components of $\mathbf{h}_2$ so that they would be linearly independent of each other and of components of $\mathbf{h}^*$. The rest $v$ components of $\mathbf{h}_2$ compute from (16) as described earlier.
6. Using $\mathbf{h} = (\mathbf{h}_1\ \mathbf{h}_2\ \mathbf{h}_3)$, compute $\mathbf{H} = [\mathbf{H}_1\ \mathbf{H}_2\ \mathbf{H}_3]$.
7. Solve equation $\mathbf{H}_p = \mathbf{T}\left[\begin{smallmatrix}\mathbf{H}\\ \mathbf{A}\end{smallmatrix}\right]$ for a matrix $\mathbf{T}$.

The complexity of the algorithm is estimated by

$$W_{parent} = O(3N^3 + nmq^{N(m-1)}) \tag{17}$$

operations in $\mathbf{F}_1$, where $m$ is given by (15). For an arbitrary subcode (by choosing arbitrary $\mathbf{S}_p$) and subject to $p \sim n-k$ with high probability $m \sim 1$, and the algorithm is polynomial.

Depending on a given subcode the algorithm can recover

$$N_{\text{parent}}(n,p,v) \geq (q^N - q^{n-p+v})(q^N - q^{n-p+v-1}) \dots (q^N - q^{n-1}) \sim q^{N(p-v)}$$

different parent Gabidulin codes in canonical form.

## 4 The security of the system based on subcodes

Resistance of a Niederreiter-type system with key (7) to the presented algorithm substantially depends on the balance between $p$ and $n-k$. If $p \leq b(n-k)$, then the algorithm is polynomial. To make system secure the value $b$ must be small enough. However, we have to choose $b > 3$, otherwise it becomes possible to apply the first Gibson attack to $\mathbf{R}_{13} - \mathbf{R}_{12}\mathbf{B}_{\text{base}}$ to recover $\mathbf{H}_1$ and $\mathbf{H}_3$, and then $\mathbf{H}_2$ with complexity $O(N^3 q^{bN})$ operations.

So $p$ has to be chosen several times greater than $n-k$. Since $p < k$, the choice of parameters for a secure system will be restricted to $n \sim k$ and $k \sim p$. This means that the parent code has a very small rank distance, and the subcode is of very small dimension. Remember that we can only use correcting ability of the parent code when encrypting/decrypting messages. Thus a selection $p > b(n-k)$ making system secure to structural attacks will make it vulnerable to direct (decoding) attacks.

For instance the example of the system given in [5] with $q = 2,\ N = n = 32,\ k = 24,$ $p = 4$ and size of the public key (in systematic form) of 7680 bits will be broken on a PC within less than a second. A secure system can be built for $n = N = 44,\ k = 36,\ p = v = 18,$ $b = 2$ with $W_{parent} \approx 2^{95}$ and public key of 20592 bits. The published code can then be decoded as a random one in $2^{79}$ operations in $\mathbf{F}_1$ [6]. It is easy to see that this system does not reveal any advantages over the original GPT PKC: for $n = N = 44,\ k = 18,\ r = 5,$ $t_1 = 8,\ s = 2$ the best known structural attack requires $2^{87}$ operations in $\mathbf{F}_N$ and the code can be decoded in $2^{100}$ operations in $\mathbf{F}_1$ with the same 20592 published bits. Moreover the GPT PKC has twice as higher the information rate: $0,620$ versus $0,294$ (see also notes on the information rate in [2]).

## 5 Conclusion

We presented an algorithm recovering a parent Gabidulin code for any its subcode. When the difference $p$ between dimensions of the code and the subcode is not to great the algorithm has polynomial complexity. This algorithm fully breaks the system presented at ISIT'2002 [5] for any interesting in practice parameters. When the algorithm becomes computationally infeasible ($p$ is close to $k$) then that system turns out to be inferior to the GPT PKC (in security and information rate) let alone other applications of codes in rank metric [3, 4].

Moreover, since the matrix $\mathbf{B}$ is known it may well happen that there is an improvement that keeps the algorithm polynomial for any values of $p$.

Still on subcodes of Gabidulin codes a secure cryptosystem can be built. It was the lack of an explicit distortion in the public key (7) that made the system vulnerable to the presented algorithm. Careful choice of a subcode and the distortion matrix $\mathbf{X}$ in (6) could prevent cryptanalysis.

## References


[1] **E. M. Gabidulin** Theory of Codes with Maximum Rank Distance// *Probl. Inform. Transm.*, 1985. — Vol. 21, No. 1. — P. 1–12.

[2] **E. M. Gabidulin, A. V. Ourivski** Improved GPT Public Key Cryptosystems. — In: Coding, Communications and Broadcasting // Ed. by P. Farrell, M. Darnell, B. Honary. — Baldock, Hertfordshire, England: Research Studies Press Ltd., 2000. — P. 73–102.

[3] **E. M. Gabidulin, A. V. Ourivski** Modified GPT PKC with Right Scrambler. — In: Proceedings of the International Workshop on Coding and Cryptography — WCC'01 / Ed. by D. Augot, C. Carlet. — Paris, France, January 2001. — P. 233–242.

[4] **E. Gabidulin, A. Ourivski, B. Honary, B. Ammar** Reducible Rank Codes and Applications to Cryptography. — In: Information, Coding and Mathematics / Ed. by M. Blaum, P. G. Farrell, H. C. A. van Tilborg. — Boston: Kluwer Academic Publishers, 2002. — P. 121–132.

[5] **T. Berger, P. Loidreau** Security of the Niederreiter Form of the GPT Public-Key Cryptosystem. — In: Proceedings of the 2002 IEEE International Symposium on Information Theory — ISIT'02. — Lausanne, Switzerland, 2002. — P. 267.

[6] **A. Ourivski, T. Johansson** New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications// *Probl. Inform. Transm.*, 2002. — Vol. 38, No. 3. — P. 237–246.