A variant of the McEliece cryptosystem with increased public key security

Marco Baldi, Marco Bianchi, Franco Chiaraluce, Joachim Rosenthal, Davide Schipani

A variant of the McEliece cryptosystem with increased public key security

Marco Baldi[1], Marco Bianchi[1], Franco Chiaraluce[1], Joachim Rosenthal[2], and Davide Schipani[2] *

[1] Università Politecnica delle Marche, Ancona, Italy
{m.baldi,m.bianchi,f.chiaraluce}@univpm.it
[2] University of Zurich, Zurich, Switzerland
{rosenthal,davide.schipani}@math.uzh.ch

Abstract. We propose a new variant of the McEliece cryptosystem which ensures that the code used as the public key is not permutation-equivalent to the secret code. This allows to increase the security of the public key, and to reconsider possible adoption of classical families of codes, like Reed-Solomon codes. A reduction in the key size or, equivalently, an increased level of security against information set decoding are the main advantages of the modified cryptosystem. As a drawback, the domain of intentional error vectors must be restricted, but we show that this has no significant impact on the security level.

Keywords: McEliece cryptosystem, error-correcting codes, public key security.

1 Introduction

The McEliece cryptosystem [8] is one of the most promising public-key cryptosystems able to resist attacks based on quantum computers. In fact, differently from cryptosystems exploiting integer factorization or discrete logarithms, it relies on the hardness of decoding a linear block code without any visible structure [3].

The original McEliece cryptosystem adopts the generator matrix of a binary Goppa code as the private key, and exploits a dense transformation matrix and a permutation matrix to disguise the secret key into the public one. No polynomial-time attack to the system has been devised up to now; however, the increased computing power and the availability of optimized attack procedures have required to update its original parameters [4].

The main advantage of the McEliece cryptosystem consists in its fast encryption and decryption procedures, which require a significantly lower number of operations with respect to alternative solutions (like RSA). However, the original McEliece cryptosystem has two main disadvantages: low encryption rate and large key size, both due to the binary Goppa codes it is based on. When adopting

* The Research was supported in part by the Swiss National Science Foundation under grant No. 132256.

Goppa codes, a first improvement is obtained through the variant proposed by Niederreiter [10], which uses parity-check matrices instead of generator matrices.

A significant improvement in both these aspects would be obtained if other families of codes could be included in the system. In particular, the use of Reed-Solomon (RS) codes could yield significant advantages. In fact, RS codes are maximum distance separable codes, which ensures they achieve maximum error correction capability. In the McEliece system, this translates into shorter keys for the same security level, or a higher security level for the same key size, with respect to binary Goppa codes (having the same code rate).

Many attempts of replacing Goppa codes with other families of codes have exposed the system to security threats [13], [18]. Some recent proposals based on Quasi-Cyclic and Quasi-Dyadic codes have also been broken [17]. Low-Density Parity-Check (LDPC) codes, in principle, should offer high design flexibility and compact keys. However, also the applicability of such a class of codes may expose the system to flaws [9], [11]. Nevertheless, it is still possible to exploit Quasi-Cyclic LDPC codes to design a variant of the system that is immune to any known attack [1].

The idea in [1] is to replace the permutation matrix $\mathbf{P}$, used in the original McEliece cryptosystem, with a dense transformation matrix $\mathbf{Q}$. The matrix $\mathbf{Q}$ used in [1] is a sparse matrix and its density must be chosen as a trade-off between two opposite effects: i) increasing the density of the public code parity-check matrix so that it is too difficult to search for low weight codewords in its dual code and ii) limiting the propagation of the intentional errors so that they are still correctable by the legitimate receiver.

We improve this approach by introducing a more effective class of $\mathbf{Q}$ matrices and by generalizing their form also to the non-binary case. The new proposal is based on the fact that there exist some classes of dense $\mathbf{Q}$ matrices that have a limited propagation effect on the intentional error vectors. The use of these matrices allows to better disguise the private key into the public one, with a controlled error amplification effect. So, we propose a modified cryptosystem that can restore the use of advantageous families of codes, as RS codes, by ensuring increased public key security. In the proposed cryptosystem, the domain of possible error vectors needs to be restricted depending on the choice of $\mathbf{Q}$. However, we will show that this restriction has a limited effect on the system security.

2 Description of the cryptosystem

The main features of the proposed system are as follows. Bob chooses his secret key as the $k \times n$ systematic generator matrix $\mathbf{G}$ of a linear block code over GF(p). He also chooses other two secret matrices: a $k \times k$ non-singular scrambling matrix $\mathbf{S}$ and an $n \times n$ non-singular transformation matrix $\mathbf{Q}$. The public key is:

$$\mathbf{G}' = \mathbf{S}^{-1} \cdot \mathbf{G} \cdot \mathbf{Q}^{-1}. \tag{1}$$

So, in general, differently from the original McEliece cryptosystem, the public code is not permutation-equivalent to the private code.

Alice, after obtaining Bob's public key, applies the following encryption map:

$$\mathbf{x} = \mathbf{u} \cdot \mathbf{G}' + \mathbf{e}, \tag{2}$$

where $\mathbf{x}$ is the ciphertext corresponding to the cleartext $\mathbf{u}$, and $\mathbf{e}$ is a vector of intentional errors. After receiving $\mathbf{x}$, Bob inverts the transformation as follows:

$$\mathbf{x}' = \mathbf{x} \cdot \mathbf{Q} = \mathbf{u} \cdot \mathbf{S}^{-1} \cdot \mathbf{G} + \mathbf{e} \cdot \mathbf{Q}, \tag{3}$$

thus obtaining a codeword of the secret code affected by the error vector $\mathbf{e} \cdot \mathbf{Q}$. Bob shall be able to correct all the errors and get $\mathbf{u} \cdot \mathbf{S}^{-1}$, thanks to the systematic form of $\mathbf{G}$. He can then obtain $\mathbf{u}$ through multiplication by $\mathbf{S}$.

2.1 Choice of Q

In general, the use of a transformation matrix $\mathbf{Q}$ in place of a permutation matrix causes an error propagation effect. However, if $\mathbf{Q}$ is chosen within a given class of matrices, this effect can be controlled or even eliminated, when needed. For the latter purpose, let us consider a first form of $\mathbf{Q}$, called $\mathbf{Q}_1$, obtained as the sum of a permutation matrix $\mathbf{P}_1$ and a rank-1 matrix $\mathbf{R}$, that is:

$$\mathbf{Q}_1 = \mathbf{R} + \mathbf{P}_1, \tag{4}$$

with

$$\mathbf{R} = \mathbf{a}^T \cdot \mathbf{b} = \left[a_1 \; a_2 \cdots a_n \right]^T \cdot \left[b_1 \; b_2 \cdots b_n \right], \tag{5}$$

where $\mathbf{a}$ and $\mathbf{b}$ are two random vectors over GF(p) and ^{T} denotes transposition. If $\mathbf{Q}_1$ is full rank, $\mathbf{Q}_1^{-1}$ can be used to generate the public code.

In the choice of $\mathbf{Q}_1$ it is important to avoid some special cases which could allow an attacker to derive a code that is permutation-equivalent to the secret one, thus bringing security back to that of the classical McEliece system.

Let us suppose the j-th element of $\mathbf{b}$ is zero and that $\mathbf{P}_1$ has a symbol 1 at position (i, j). In this case, the j-th column of $\mathbf{Q}_1$ is completely null, except for its element at row i. Since $\mathbf{Q}_1^{-1} = \widehat{\mathbf{Q}} / |\mathbf{Q}|$, where $\widehat{\mathbf{Q}}$ is the adjoint matrix and $|\mathbf{Q}|$ is the determinant of $\mathbf{Q}_1$, it follows from the definition of $\widehat{\mathbf{Q}}$ that the i-th column of $\mathbf{Q}_1^{-1}$ is completely null, except for its element at row j (that is not necessarily equal to 1, except for the binary case). So, the i-th column of $\mathbf{Q}_1^{-1}$ has the effect of a column permutation (apart from multiplication by a constant), like in the original McEliece cryptosystem.

In order to avoid such a flaw, we impose that all the elements of $\mathbf{b}$ are non-zero. We then restrict the generation of intentional error vectors to vectors $\mathbf{e} = [e_1, e_2, \ldots, e_n]$ such that:

$$\sum_{i=1}^{n} a_i e_i = 0. \tag{6}$$

This requires that $\mathbf{a}$ is disclosed as part of the public key, and ensures that the product $\mathbf{e} \cdot \mathbf{R}$ gives the all-zero vector, so that the legitimate receiver gets:

$$\mathbf{e}' = \mathbf{e} \cdot \mathbf{Q}_1 = \mathbf{e} \cdot \mathbf{R} + \mathbf{e} \cdot \mathbf{P}_1 = \mathbf{e} \cdot \mathbf{P}_1. \tag{7}$$

So, the weight of $\mathbf{e}'$ is exactly coincident with that of $\mathbf{e}$. If we work on GF(p), with $p > 2$, we can replace the permutation matrix with a generalized permutation matrix with non-zero values randomly selected among the $p-1$ non-zero elements of GF(p). More generally, we can consider to use m matrices of the latter type, the ith one being denoted by $\mathbf{\Pi}_i$. So, in place of (4), we have:

$$\mathbf{Q}_m = \mathbf{R} + \mathbf{\Pi}_1 + \mathbf{\Pi}_2 + \ldots + \mathbf{\Pi}_m. \tag{8}$$

Provided that only intentional error vectors that satisfy (6) are used, a matrix $\mathbf{Q}_m$ as in (8) allows to amplify the number of intentional errors (at most) by a factor m. Such controlled error amplification effect can be compensated by using codes with a high error correction capability, as it occurs for LDPC codes [1]. Moreover, the use of $\mathbf{Q}_m$ (through its inverse) allows to disguise the private matrix of a code over GF(p) in a way that, at least in principle, is much stronger than what can be done by using a permutation matrix (as in the original McEliece system). An even more general form of $\mathbf{Q}_m$ can be designed by replacing the rank-1 matrix $\mathbf{R}$ with a rank-z ($z \geq 1$) matrix, thus modifying condition (6) accordingly with a set of z constraints.

2.2 Design issues

As we have seen in Section 2.1, null elements must be avoided in $\mathbf{b}$ to prevent the public code from being (almost) permutation-equivalent to the secret one.

Focusing on the binary case, this imposes that $\mathbf{b}$ is the all-one vector. However, in such a case, further issues exist in the design of $\mathbf{Q}$. For example, let us consider $\mathbf{a}$ as an all-one vector too, so that $\mathbf{R} = \mathbf{1}$, and suppose that only one random permutation matrix is used (as in (4)). It is easy to verify that the public code has the following parity-check matrix:

$$\mathbf{H}' = \mathbf{H} \cdot \mathbf{Q}^T, \tag{9}$$

where $\mathbf{H}$ is the parity-check matrix of the private code. In the special case of $\mathbf{Q}_1 = \mathbf{1} + \mathbf{P}_1$, we have $\mathbf{H}' = \mathbf{H} \cdot \mathbf{1} + \mathbf{H} \cdot \mathbf{P}_1^T$. By assuming a regular $\mathbf{H}$ (i.e. with constant row and column weights), two cases are possible:

- If the rows of $\mathbf{H}$ have even weight, $\mathbf{H} \cdot \mathbf{1} = \mathbf{0}$ and $\mathbf{H}' = \mathbf{H} \cdot \mathbf{P}_1^T$.
- If the rows of $\mathbf{H}$ have odd weight, $\mathbf{H} \cdot \mathbf{1} = \mathbf{1}$ and $\mathbf{H}' = \mathbf{1} + \mathbf{H} \cdot \mathbf{P}_1^T$.

In both cases, the public code has a parity-check matrix that is simply a permuted version of that of the secret code (or its complementary). This reduces the security to that of the original McEliece cryptosystem, that discloses a permuted version of the secret code. Such security level is not sufficient when adopting, for

example, LDPC codes, since the permuted version of the secret $\mathbf{H}$ matrix can be attacked by searching for low weight codewords in the dual of the secret code.

A more general formulation of the flaw follows from the consideration that $\mathbf{Q}_1 = \mathbf{1} + \mathbf{P}_1$ has a very special inverse. First of all, let us consider that $\mathbf{Q}_1$ is invertible only when it has even size. This is obvious since, for odd size, $\mathbf{Q}_1$ has even row/column weight; so, the sum of all its rows is the zero vector. If we restrict ourselves to even size $\mathbf{Q}_1$ matrices, it is easy to show that their inverse has the form $\mathbf{Q}_1^{-1} = \mathbf{1} + \mathbf{P}_1^T$, due to the property of permutation matrices (as orthogonal matrices) to have their inverse coincident with the transpose.

So, $\mathbf{Q}_1^{-1}$ has the same form of $\mathbf{Q}_1$ and, as in the case of $\mathbf{H}$, disclosing $\mathbf{G}' = \mathbf{S}^{-1}\mathbf{G}\mathbf{Q}_1^{-1}$ might imply disclosing a generator matrix of a permuted version of the secret code or its complementary (depending on the parity of its row weight). Therefore, the form $\mathbf{Q}_1 = \mathbf{1} + \mathbf{P}_1$ might reduce the security to that of the permutation used in the original McEliece cryptosystem.

Based on these considerations, one could think that adopting a vector $\mathbf{a}$ different from the all-one vector could avoid the flaw. However, by considering again that $\mathbf{Q}_1^{-1} = \widehat{\mathbf{Q}}/|\mathbf{Q}|$, it is easy to verify that a weight-1 row in $\mathbf{Q}_1$ produces a weight-1 row in $\mathbf{Q}_1^{-1}$ and a weight-$(n-1)$ row in $\mathbf{Q}_1$ produces a weight-$(n-1)$ row in $\mathbf{Q}_1^{-1}$. It follows that $\mathbf{Q}_1^{-1}$ contains couples of columns having Hamming distance 2. Since their sum is a weight-2 vector, the sum of the corresponding columns of the public matrix results in the sum of two columns of $\mathbf{S}^{-1}\mathbf{G}$. Starting from this fact, an attacker could try to solve a system of linear equations with the aim of obtaining a permutation-equivalent representation of the secret code, at least for the existing distance-2 column pairs.

So, our conclusion concerning the binary case is that the choice of $\mathbf{Q}$ as in (4) should be avoided. A safer $\mathbf{Q}$ is obtained by using an $\mathbf{R}$ matrix with rank $z > 1$ and by adding more than one permutation matrices to it (i.e. $m > 1$). This obviously has the drawback of requiring codes with increased error correction capability; so, in this work, we will focus on non-binary codes and $m = 1$.

3 Comparison with previous cryptosystems

Other proposals for increasing key security have been made in the past, such as using a distortion matrix together with rank codes in the GPT cryptosystem [5] and exploiting the properties of subcodes in variants of the McEliece and the GPT cryptosystems [2]. Unfortunately, cryptanalysis has shown that such approaches exhibit security flaws [13], [18].

The idea of using a rank-1 matrix with the same structure we consider can also be found in [6]. However, such a matrix was added to the secret matrix (rather than multiplied) and no selection of the error vectors was performed, so that a completely different solution was implemented.

Instead, the idea of replacing the permutation in the McEliece cryptosystem with a more general transformation matrix is already present in the variant of the GPT cryptosystem adopting a column scrambler [12], [16] and in cryptosystems based on full decoding [7, sec. 8.3]. These proposals are shortly examined next.

3.1 Comparison with the modified GPT cryptosystem

Apart from the code extension and the inclusion of an additive distortion matrix, in the modified GPT cryptosystem the public generator matrix is obtained through right-multiplication by a non-singular matrix that is not necessarily a permutation matrix. So, in principle, it is the same idea of a more general transformation matrix as in the proposed cryptosystem. However, in order to preserve the ability to correct the intentional error vectors, the GPT cryptosystem works in the rank metric domain and adopts rank distance codes, as Gabidulin codes.

Unfortunately, the properties of Gabidulin codes make it possible to exploit the effect of the Frobenius automorphism on the public generator matrix in order to mount a polynomial-time attack [13]. Differently from the GPT cryptosystem, the proposed solution still exploits Hamming distance codes and is able to replace the permutation matrix with a more general transformation matrix by properly selecting the error vectors.

3.2 Comparison with full-decoding cryptosystems

The main idea behind full-decoding cryptosystems in [7] is to let the intentional error vectors have any arbitrary weight. This way, an attacker would be forced to try full-decoding of the public code, that is known to be a NP-complete task. Obviously, the legitimate receiver must be able to decode any intentional error vector with reasonable complexity; so, the problem of full decoding must be transformed from a one-way function to a trapdoor function. For this purpose, the main idea is to use a transformation that maps a set of error vectors with weight $\leq t$ into a set of arbitrary weight intentional error vectors.

If this transformation is represented by the $n \times n$ matrix $\mathbf{M}$, the public code (as proposed first in [7]) would be $\mathbf{G}' = \mathbf{G} \cdot \mathbf{M}$. The basic point for obtaining a trapdoor function is to make Alice use only those error vectors that can be expressed as $\mathbf{e}' = \mathbf{e} \cdot \mathbf{M}$, where $\mathbf{e}$ is a weight-t error vector. This way, when Bob uses the inverse of the secret matrix $\mathbf{M}$ to invert the transformation, he re-maps each arbitrary weight error vector into a correctable error vector. Unauthorized users would instead be forced to try full-decoding over arbitrary weight error vectors; so, the trapdoor is obtained.

In order to exploit the full-decoding problem, Alice must use, for encryption, only those error vectors that can be anti-transformed into correctable error vectors. So, some information on the transformation used to originate them must be disclosed. A solution is that the first $p < n$ rows of $\mathbf{M}$ are made public [7]. However, it has been proved that, this way, the security reduces to that of the original McEliece cryptosystem, and an attacker does not have to attempt full-decoding, but only normal decoding.

Further variants aim at better hiding the secret transformation matrix in its disclosed version [7]. In the last variant, a generator matrix of a maximum distance-t anticode is used to hide the secret transformation. This way, after inverting the secret transformation, the error vector remains correctable for the legitimate receiver. To our knowledge, the latter version has never been proved

to be insecure nor to reduce to the same problem of the original McEliece cryptosystem. However, the construction based on anticodes seems quite unpractical.

Differently from full-decoding cryptosystems, our proposal still relies on the same problem as the original McEliece cryptosystem (that is, normal decoding); so, we need to perform only a selection of the random error vectors (without any transformation). For this reason, the information "leakage" on the secret transformation that is needed in the proposed cryptosystem is considerably lower with respect to what happens in full-decoding cryptosystems.

4 Attacks against the proposed cryptosystem

A first concern about the proposed cryptosystem is to verify that it is actually able to provide increased key security, with respect to previous variants of the McEliece cryptosystem, in such a way as to allow the use of widespread families of codes (as RS and Generalized RS codes) without incurring in the attacks that have prevented their use up to now.

From the comparison with the variants described in Sections 3.1 and 3.2, we infer that previous attacks targeted to those cryptosystems do not succeed against the proposed one, due to the differences in the family of codes used and in the information leakage on the secret transformation. Concerning the latter point, we observe that, even if the whole matrix $\mathbf{R}$ (and not only the vector $\mathbf{a}$) were public, an attacker would not gain much information. In fact, in this case, he could compute $\mathbf{x} \cdot \mathbf{R} = \mathbf{u} \cdot \mathbf{G}' \cdot \mathbf{R}$. However, $\mathbf{R}$ has rank $z \ll n$, so $\mathbf{G}' \cdot \mathbf{R}$ is not invertible. Moreover, multiplication by $\mathbf{G}' \cdot \mathbf{R}$ only provides a dimension-z syndrome of $\mathbf{u}$, whose decoding is known to be a hard problem [3].

The most powerful attack procedures seem to be those techniques that attempt information set decoding (ISD) on the public code; so we estimate the security level of the proposed cryptosystem against them.

4.1 ISD attacks

In [4] the authors have proposed some smart speedup techniques to reduce the Stern algorithm work factor (WF) over the binary field, this way obtaining a theoretical WF close to 2^{60}. Their attack was implemented on a big cluster of computers that was able to break the McEliece cryptosystem with original parameters ($n = 1024$, $k = 524$, $w = 50$). As a consequence, the authors have proposed some new set of system parameters in order to increase the security level. The information set decoding attack is not polynomial in the code dimension, since it aims at decoding a random linear code without exploiting any structural property (even if present) and this task is notoriously non-polynomial. One of the biggest improvements presented in [4] is a smart way to find k independent columns in the public generator matrix at each iteration without performing Gaussian reduction on all such columns. A further improvement consists in the pre-computation of the sum of some rows during the reduction.

In [15], Peters points out that these speedups are efficient on very small fields. As it results from the table available in [14], for $q > 16$ these speedups are not relevant and the algorithm is quite similar to Stern's one. The difference relies on guessing not only p error positions but also p error values in the k independent columns, due to the field cardinality. Finiasz and Sendrier have proposed a further improvement that could yield a slight modification in the WF, resulting in a maximum increase of 2^6 or a maximum decrease close to 2^3.

In Table 1 we report some values of the WF when using RS codes in the variant of the McEliece cryptosystem we propose. They were computed through the PARI/GP script available in [14], that allows the estimation of the security level, although it is not extremely accurate (it can be about 4-8 times higher than the actual value). The reported WF values are the lowest ones obtained for each set of parameters. Based on Table 1, we can compare the proposed cryptosystem with the instances of the McEliece system presented in [4].

Example 1 To reach WF $> 2^{80}$, the (1632, 1269) Goppa code is suggested, resulting in a public-key size of 460647 bits (that is the lowest possible value for this code, obtained by storing the non-systematic part of $\mathbf{H}$, as in the Niederreiter cryptosystem). With the new variant we can consider the RS code with $n = 255$, $k = 195$, $t = 30$, having an estimated WF $\approx 2^{86.06}$ and an actual WF $\approx 2^{84.18}$ (found through the C program available in [14]). The public key size for this code, due to storing the 195×255 matrix $\mathbf{G}'$ and the 1×255 vector $\mathbf{a}$, both with elements over GF(256), is 399840 bits, that is about 13% less than (the minimum size of) that obtained by the revised McEliece cryptosystem [4]. The security level of the two systems remains comparable when the constraint expressed by $\mathbf{a}$ is imposed on the intentional error vectors of the modified cryptosystem. In fact, as it will be shown in the next subsection, the introduction of each constraint results in a decreased WF for the ISD attack of 2^3 at most.

Example 2 As another example, we can consider the Goppa code suggested in [4] to achieve WF $\geq 2^{128}$, which has $n = 2960$, $k = 2288$, yielding a key length of 1537536 bits. An RS code with the same rate (0.77), defined over GF(512), is reported in Table 1 and has $n = 511$, $k = 395$. The corresponding key size for the proposed McEliece system is 1821204 bits (that is slightly bigger than the one in the Niederreiter system proposed in [4]), but the security level grows up to $2^{158.67}$ (more precisely, it is estimated as $2^{155.89}$ with the C program from [14]). This value remains very high even when we consider the presence of the constraint expressed by $\mathbf{a}$ on the intentional error vectors.

4.2 Exploiting the knowledge on error vectors

It is important to assess whether the constraints imposed on the intentional error vectors used in the proposed cryptosystem have consequences on its security.

For this purpose, a conservative approach consists in considering, in the WF computations, a reduced number of intentional errors, that is, $t' = t - z$, where

Table 1. Work factor ($\log_2$) of ISD attacks on RS codes.

RS codes with $n = 127$ defined over GF(128)															
Rate	0.75	0.73	0.72	0.70	0.69	0.67	0.65	0.64	0.62	0.61	0.59	0.57	0.56	0.54	0.53
t	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
WF	49.2	50.1	51.0	51.7	52.3	52.8	53.3	53.7	54.0	54.2	54.3	54.4	54.4	54.4	54.2
RS codes with $n = 255$ defined over GF(256)															
Rate	0.81	0.80	0.78	0.76	0.75	0.73	0.72	0.70	0.69	0.67	0.65	0.64	0.62	0.61	0.59
t	24	26	28	30	32	34	36	38	40	42	44	46	48	50	52
WF	79.0	81.6	83.9	86.1	87.9	89.6	91.1	92.4	93.5	94.4	95.2	95.8	96.2	96.5	96.7
RS codes with $n = 511$ defined over GF(512)															
Rate	0.94	0.93	0.91	0.90	0.89	0.88	0.87	0.86	0.84	0.83	0.82	0.81	0.80	0.78	0.77
t	16	19	22	25	28	31	34	37	40	43	46	49	52	55	58
WF	81.3	90.1	98.1	105.6	112.4	118.8	124.7	130.2	135.3	140.0	144.3	148.4	152.1	155.5	158.7

z is the number of constraints we impose on the intentional error vectors. This approach is conservative in the sense that we assume that the attacker exactly knows both the position and value of z errors, while he actually knows only their values. This has been done in Table 2. As we can observe from the values obtained (and their comparison with those reported in Table 1, corresponding to $z = 0$), we have a WF decrease close to 2^3 when z is increased by 1. So, the security level for the considered parameters does not vary significantly for low values of z.

Table 2. Work factor ($\log_2$) of ISD attacks on RS codes with $n = 255$, defined over GF(256), when $z = 1$ or $z = 2$ constraints are imposed on the error vectors.

Rate	0.81	0.80	0.78	0.76	0.75	0.73	0.72	0.70	0.69	0.67	0.65	0.64	0.62	0.61	0.59
t	24	26	28	30	32	34	36	38	40	42	44	46	48	50	52
WF ($z = 1$)	75.9	78.6	81.1	83.3	85.3	87.0	88.6	90.0	91.2	92.2	93.0	93.7	94.2	94.6	94.8
WF ($z = 2$)	72.8	75.6	78.2	80.5	82.6	84.5	86.1	87.6	88.9	89.9	90.9	91.6	92.2	92.6	92.9

5 Conclusion

We have introduced a variant of the McEliece cryptosystem that, by replacing the secret permutation matrix with a more general transformation matrix, is able to avoid that the public code is permutation-equivalent to the secret code. This allows to prevent attacks against classical families of codes, as RS codes, and to reconsider them as possible good candidates in this framework.

We have assessed the security level of the proposed cryptosystem by considering up-to-date attack procedures, and we have compared it with the classical McEliece cryptosystem and the Niederreiter variant. Our results show that the proposed solution, by exploiting RS codes, is able to guarantee the same security level with reduced key size or, equivalently, an increased security level with a similar key size.

References

1. Baldi, M., Bodrato, M., Chiaraluce, F.: A new analysis of the McEliece cryptosystem based on QC-LDPC codes. In: Security and Cryptography for Networks, Lecture Notes in Computer Science, vol. 5229, pp. 246–262. Springer Berlin / Heidelberg (2008)
2. Berger, T.P., Loidreau, P.: How to mask the structure of codes for a cryptographic use. Designs, Codes and Cryptography 35, 63–79 (2005)
3. Berlekamp, E., McEliece, R., van Tilborg, H.: On the inherent intractability of certain coding problems. IEEE Trans. Inform. Theory 24(3), 384–386 (May 1978)
4. Bernstein, D.J., Lange, T., Peters, C.: Attacking and defending the McEliece cryptosystem. In: Post-Quantum Cryptography, Lecture Notes in Computer Science, vol. 5299/2008, pp. 31–46. Springer Berlin / Heidelberg (2008)
5. Gabidulin, E.M., Paramonov, A.V., Trejakov, O.V.: Ideals over a non-commutative ring and their application in cryptography. D. W. Davies, Ed., Advances in Cryptology - EUROCRYPT 91, Lecture Notes in Computer Science 547, Springer Verlag (1991)
6. Gabidulin, E.M., Kjelsen, O.: How to avoid the Sidel'nikov-Shestakov attack. In: Error Control, Cryptology, and Speech Compression, Lecture Notes in Computer Science, vol. 829, pp. 25–32. Springer Berlin / Heidelberg (1994)
7. Kabatiansky, G., Krouk, E., Semenov, S.: Error Correcting Coding and Security for Data Networks: Analysis of the Superchannel Concept. John Wiley & Sons (2005)
8. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. DSN Progress Report pp. 114–116 (1978)
9. Monico, C., Rosenthal, J., Shokrollahi, A.: Using low density parity check codes in the McEliece cryptosystem. In: Proc. IEEE International Symposium on Information Theory (ISIT 2000). p. 215. Sorrento, Italy (Jun 2000)
10. Niederreiter, H.: Knapsack-type cryptosystems and algebraic coding theory. Probl. Contr. and Inform. Theory 15, 159–166 (1986)
11. Otmani, A., Tillich, J.P., Dallot, L.: Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes. In: Proc. First International Conference on Symbolic Computation and Cryptography (SCC 2008). Beijing, China (Apr 2008)
12. Ourivski, A., Gabidulin, E.: Column scrambler for the GPT cryptosystem. Discrete Applied Mathematics 128, 207–221 (2003)
13. Overbeck, R.: Structural attacks for public key cryptosystems based on Gabidulin codes. Journal of Cryptology 21(2), 280–301 (2008)
14. Peters, C.: (2010), `http://www.win.tue.nl/~cpeters/isdfq.html`
15. Peters, C.: Information-set decoding for linear codes over F_q. In: Sendrier, N. (ed.) Post-Quantum Cryptography, Lecture Notes in Computer Science, vol. 6061, pp. 81–94. Springer Berlin / Heidelberg (2010)
16. Rashwan, H., Gabidulin, E.M., Honary, B.: Security of the GPT cryptosystem and its applications to cryptography. Security Comm. Networks (2010)
17. Umana, V.G., Leander, G.: Practical key recovery attacks on two McEliece variants. In: Cid, C., Faugere, J.C. (eds.) Proc. 2nd Int. Conf. on Symbolic Computation and Cryptography. pp. 27–44. Egham, UK (Jun 2010)
18. Wieschebrink, C.: Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. In: Sendrier, N. (ed.) Post-Quantum Cryptography: PQCrypto 2010, LNCS, vol. 6061, pp. 61–72. Springer (2010)