Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090247

A Real-Field Public Key Cryptosystem based on Sparse Recovery

Zaixing HE, Xinyue ZHAO[*] *and Shuyou ZHANG*

Department of Mechanical Engineering, Zhejiang University, Hangzhou, China

Received: 18 Jun. 2014, Revised: 16 Sep. 2014, Accepted: 18 Sep. 2014
Published online: 1 Mar. 2015

Abstract: An efficient and secure real-field public key cryptosystem (PKC) based on sparse recovery is proposed. The security of the proposed cryptosystem depends on the following facts: 1. when the measurement matrix is known, the decryption algorithm, Cross Low-dimensional Pursuit, can efficiently solve the sparse recovery problem, where the sparse vector has a relatively high proportion of nonzeros; 2. without the measurement matrix, it is NP-hard to directly solve the sparse recovery problem. The proposed PKC is novel. First, unlike the traditional PKCs that are defined in finite fields, the proposed PKC is defined in the real field. Second, unlike popular cryptosystems based on number-theoretic problems, the proposed cryptosystem is based on the sparse recovery problem.

Keywords: Sparse recovery, error correction, permuted block diagonal matrix, cross low-dimensional pursuit

1 Introduction

The security of public key cryptosystems (PKCs) is usually based on difficulty of mathematical problems. In widely used PKCs, e.g., RSA [1] and ECC (elliptic curve cryptography) [2,3], most of the involved problems are number-theoretic ones, such as factoring integers and finding discrete logarithms. However, quantum computers can break the RSA, ECC PKCs, since they can efficiently factor integers and extract discrete logarithms. Another PKC, McEliece [4], which is based on coding theory (the error correction problem), is believed to be more secure against quantum attacks. Therefore, the McEliece and the related Niederreiter PKCs [5] has been extensively studied.

The classic error correction is defined over the finite field. In recent years, a new error correction problem that arises in the real-field has attracted a lot of attention [6]. By solving a sparse recovery problem [7,6,8,9], the errors can be corrected. Motived by the McEliece PKC and the sparse recovery problem, we propose a novel PKC defined over the real-field based on sparse recovery. Massive research results in sparse recovery make it become possible for developing efficient and secure real-field PKCs.

The sparse recovery problem, to recover a sparse vector from incomplete linear measurements, is of significant importance. It arrives in many areas such as compressed sensing [8,9], decoding real codes [6], sparse representation [7], and data stream computing [10]. Solving the problem involves finding the original vector from an underdetermined system. Since there exist numerous solutions to the underdetermined system, general recovery is impossible. Fortunately, it has been proved that when the vector is sufficiently sparse and the measurement matrix has low-coherent columns, the recovery is possible and the only way is to find the sparsest solution to the underdetermined system [7]. Directly searching for the sparsest solution is known as an NP-hard problem. However, recent studies have shown that it can be solved in reasonable time when the vector is highly sparse. A lot of numerical methods are available such as Greedy algorithms [11,12,13,14,15,16,17], e.g., Orthogonal Matching Pursuit (OMP) [12,15], ℓ_1-norm minimization algorithms [18,19,20,21], which solve a Basis Pursuit (BP) problem [18], nonconvex optimization algorithms [22,23], and so on. Compared to the direct search, these methods can only recover vectors with much fewer nonzeros in the sparse vector, especially when the nonzeros are with the same absolute value. Therefore, when the proportion of nonzeros is higher than a certain bound, the recovery is still NP-hard.

In our previous work, a special sparse recovery algorithm named Cross Low-dimensional Pursuit (CLP)

[*] Corresponding author e-mail: zhaoxinyue@zju.edu.cn

has been proposed [24]. Unlike other algorithms, CLP depends on a specific measurement matrix, the Permuted Block Diagonal (PBD) matrix [24]. Without knowing the PBD matrix, CLP is unavailable for recovering sparse vectors. Compared to other algorithms, CLP can recover vectors with much more nonzeros. In another word, the bound of the proportion of nonzeros for successful recovery of CLP is much higher than those of other algorithms. The two properties of specific matrix dependence and high nonzero proportion bound provide a new resource for developing PKCs.

In the proposed PKC, the PBD matrix serves as the secret key and the CLP algorithm works as the decryption algorithm. A sparse vector, whose nonzero proportion is between the bound of other algorithms and that of the CLP algorithm, is generated randomly during encryption and destroyed afterwards. To decrypt the ciphertext, a sparse recovery problem has to be solved to recover the destroyed sparse vector. The CLP algorithm can recover it fast with the secret key; while without the secret key, it is NP-hard to recover it since CLP is unavailable and other algorithms cannot recover it because of the high proportion of nonzeros. During both encryption and decryption, no large number calculation is involved and only regular matrix calculation is needed. Therefore, the cryptosystem is efficient.

The remainder of the paper is organized as follows. Section 2 introduces the basic knowledge of sparse recovery and error correction based on it. Then, Section 3 presents the proposed sparse recovery-based PKC as well as the parameter setting and parameter sets, complexity and security analysis. Finally, Section 4 concludes the paper.

2 Sparse recovery and real-field error correction

In this section, we briefly introduce the basic knowledge of parse recovery and its application of error correction. The core mathematical problem of the proposed cryptosystem is sparse recovery, and the main idea of the proposed cryptosystem is motivated by error correction based on sparse recovery.

2.1 Sparse recovery

Assume that a sparse vector $\mathbf{e} \in \mathbb{R}^N$ is measured by a matrix $\mathbf{D} \in \mathbb{R}^{M \times N}$ $(M < N)$:

$$\mathbf{s} = \mathbf{D}\mathbf{e}, \tag{1}$$

obtaining M linear measurements $\mathbf{s}$. The original sparse vector $\mathbf{e}$ needs to be recovered from these incomplete linear measurements. It has been proven that when $\mathbf{e}$ is sufficiently sparse and the columns of $\mathbf{D}$ are low-coherent, the sparsest solution to Eq. (1) equals $\mathbf{e}$ [7]. That is to solve the following optimization problem:

$$(P_0) \ : \ \min \|\mathbf{z}\|_0 \quad \text{subject to} \quad \mathbf{D}\mathbf{z} = \mathbf{s}, \tag{2}$$

where the ℓ_0-norm counts the nonzero elements of the vector. This optimization problem is well-known as an NP-hard problem. Directly solving such a problem in high dimensions is computationally intractable. Therefore, many efficient numerical algorithms, which solve the problem in indirect ways, are proposed to find sparse solutions. These algorithms require stricter conditions than the direct way includes a highly sparse $\mathbf{e}$.

An alternative to Eq. (2) is:

$$(P_p) \ : \ \min \|\mathbf{z}\|_p^p \quad \text{subject to} \quad \mathbf{D}\mathbf{z} = \mathbf{s}, \tag{3}$$

where $0 < p \leq 1$. When $p = 1$, Eq. (3) is convex and can be recast as a Linear Program (LP). The corresponding algorithms are called ℓ_1-min algorithms [18, 19, 20, 21]. When $p < 1$, Eq. (3) is nonconvex, and it can be approximated by an Iteratively Reweighted Least Squares (IRLS) algorithm, e.g., [22, 23].

The greedy algorithms find sparse solutions in a different way: the coordinates and altitudes of nonzeros of $\mathbf{e}$ are determined step by step [11, 12, 13, 14, 15, 16, 17].

These three families of algorithms do not rely on specific matrices ($\mathbf{D}$). There is another family of algorithms that are based on sparse matrices in order to accelerate the solving procedure, e.g., Sequential Sparse Matching Pursuit (SSMP) [25]. In [24], we proposed a new algorithm, CLP, which is also based on a specific matrix, PBD, for recovering sparse vectors from incomplete measurements. It has been reported in [24] that CLP has higher sparse recovery ability and efficiency than other algorithms.

2.2 Real-field Error correction

The sparse recovery can be used for error correction [6]. Consider transmitting a message $\mathbf{x} \in \mathbb{R}^K$ by encoding it with a full *rank* matrix $\mathbf{F} \in \mathbb{R}^{N \times K}(K = N - M)$. For simplification, the columns of $\mathbf{F}$ are assumed to be orthonormal. Furthermore, a small fraction of entries of the codeword are corrupted over the transmitting channel since there is impulsive noise in the channel. Thus, the corrupted output can be written as:

$$\mathbf{y} = \mathbf{F}\mathbf{x} + \mathbf{e}, \tag{4}$$

where $\mathbf{e} \in \mathbb{R}^N$ is a sparse error vector. The final object is to exactly recover $\mathbf{x}$ with knowledge of the corrupted output $\mathbf{y}$ and coding matrix $\mathbf{F}$.

In order to reconstruct $\mathbf{x}$ from $\mathbf{y}$ and $\mathbf{F}$, one can firstly construct a matrix $\mathbf{D} \in \mathbb{R}^{M \times N}$ such that $\mathbf{D}\mathbf{F} = \mathbf{0}$. Obviously, $\mathbf{D}$ is a matrix whose rows span the null space of $\mathbf{F}^T$. The matrix $\mathbf{D}$ can also be viewed as a parity-check matrix. Then one can apply $\mathbf{D}$ to the corrupted output $\mathbf{y}$ and obtain,

$$\begin{aligned} \mathbf{s} &= \mathbf{Dy} \\ &= \mathbf{De}. \end{aligned} \tag{5}$$

Note that reconstructing $\mathbf{e}$ is a sufficient condition for reconstructing $\mathbf{x}$, since

$$\mathbf{x} = \mathbf{F}^T(\mathbf{y} - \mathbf{e}). \tag{6}$$

Therefore, the decoding problem is reduced to the problem of reconstructing a sparse vector $\mathbf{e}$ from an underdetermined system of Eq. (5). Since Eq. (5) is the same as Eq. (1), it can be reconstructed by solving a sparse recovery problem, where the parity-check matrix and sparse error vector in decoding correspond to the measurement matrix and sparse vector in sparse recovery, respectively.

3 Proposed PKC

The main idea of the proposed PKC is choosing a specific sparse recovery algorithm, which depends on a special measurement matrix and has high recovery ability. Thus, the specific measurement matrix can serve as the secret key. And the sparse recovery algorithm can serve as the decryption algorithm. Since the recovery ability of the decryption algorithm is higher than other sparse recovery algorithms, we can choose a proper parameter of the nonzero proportion such that the sparse vector can be recovered by the decryption algorithm but can not be recovered by the other algorithms. In this way, the PKC is secure without knowing the secret key.

3.1 Key generation

The secret and public keys are generated in the similar way to those of the McEliece PKC. We construct a sparse structured matrix, the PBD matrix, which is proposed in [24] for generating the secret key. First, we generate a PBD matrix $\mathbf{D} \in \mathbb{R}^{M\times N}$. Then, we generate a random dense nonsingular matrix $\mathbf{S} \in \mathbb{R}^{K\times K}$ and a random permutation matrix $\mathbf{Q} \in \mathbb{R}^{N\times N}$. The secret key is then generated: $(\mathbf{D},\mathbf{Q},\mathbf{S})$.

The public key consists of two parts: a coding matrix $\mathbf{H}$ and a parameter ρ that stands for the proportion of nonzeros in $\mathbf{e}$. The coding matrix $\mathbf{H}$ is constructed as follows. First, generate a matrix $\mathbf{F}$, whose columns span the null space of $\mathbf{D}$. There are several ways to construct $\mathbf{F}$ satisfying $\mathbf{DF} = \mathbf{0}$, e.g., the QR decomposition that produces orthonormal columns. Finally, the public key $\mathbf{H} \in \mathbb{R}^{N\times K}$ is generated as: $\mathbf{H} = \mathbf{QFS}$. The public key $(\mathbf{H},\rho)$ is transmitted through public channels to the encryption users.

Here we give the brief generation of the PBD matrix; refer to [24] for the details. Suppose that we have L matrices that are block diagonal, $\mathbf{W}_1 \in \mathbb{R}^{M_1\times N} = diag(\mathbf{w}_1,\cdots,\mathbf{w}_1)$, $\cdots$, $\mathbf{W}_L \in \mathbb{R}^{M_L\times N} = diag(\mathbf{w}_L,\cdots,\mathbf{w}_L)$, where $M_1+\cdots+M_L = M$ and $\mathbf{w}_1,\cdots,\mathbf{w}_L \in \mathbb{R}^{m\times n}$, and L different random permutation matrices, namely $\mathbf{p}_1,\cdots,\mathbf{p}_L \in \mathbb{R}^{N\times N}$. We construct a PBD matrix $\mathbf{D} \in \mathbb{R}^{M\times N}$ as follows:

$$\mathbf{D} = \begin{bmatrix} \mathbf{W}_1\mathbf{p}_1 \\ \vdots \\ \mathbf{W}_L\mathbf{p}_L \end{bmatrix}. \tag{7}$$

3.2 Encryption and decryption

For a plaintext $\mathbf{x} \in \mathbb{R}^K$, the encryption includes two stages using the public key $(\mathbf{H},\rho)$. Firstly, code the $\mathbf{x}$ with $\mathbf{H}$, obtaining $\mathbf{Hx}$. Secondly, generate a random $\mathbf{e}$ with ρ proportion nonzero elements and each with the same absolute value; code $\mathbf{Hx}$ with $\mathbf{e}$, obtaining the ciphertext $\mathbf{y}$ $(= \mathbf{Hx}+\mathbf{e})$; destroy $\mathbf{e}$. Mathematically speaking, the encryption is the same as described in Eq. (4), although $\mathbf{e}$ here is generated artificially.

The core decryption problem is the decoding of error correction via sparse recovery. To decrypt the ciphertext $\mathbf{y}$, first compute $\mathbf{y}' = \mathbf{Q}^{-1}y$. Then, the secret key $\mathbf{D}$ can be applied to the $\mathbf{y}'$: $\mathbf{s} = \mathbf{Dy}' = \mathbf{DFSx}+\mathbf{De}'$. We can obtain $\mathbf{s} = \mathbf{De}'$, similar to Eq. (5), since $\mathbf{DF} = \mathbf{0}$. The sparse recovery algorithm, CLP, is used to recover the sparse error vector $\mathbf{e}'$. Then $\mathbf{x}' = \mathbf{Sx}$ is computed as $\mathbf{x}' = \mathbf{F}^T(\mathbf{y}'-\mathbf{e}')$. Finally $\mathbf{x}$ is decrypted as: $\mathbf{x} = \mathbf{S}^{-1}\mathbf{x}'$.

Although the core decryption problem is an error correction problem, they are different. The difference is: in error correction, the easier $\mathbf{e}$ can be recovered, the better; in the cryptosystem, $\mathbf{e}$ should be easy to be recovered with the secret key and hard to be recovered without it. This point make them totally different in designing the frameworks, in which the decoding (decryption) algorithm is of significant importance. In error correction, any powerful sparse recovery algorithm is suitable for decoding. While in the cryptosystem, the sparse recovery algorithm should be the only one that can decrypt the ciphertext, and the secret key should be necessary for the utilization of the algorithm.

The CLP algorithm is suitable for the cryptosystem since it satisfies the two requirements. With the secret key, the ciphertext can be easily decrypted since CLP has linear complexity. The CLP algorithm is based on the PBD matrix. Here we introduce it briefly. Refer to [24] for the details.

In order to simplify the presentation, suppose the PBD matrix is generated from two block diagonal matrices $(L = 2)$. The recovery procedure corresponding to a PBD matrix generated from more block diagonal matrices is similar. CLP solves the sparse recovery problem of Eq. (2), in which $\mathbf{s} = \mathbf{De}$. For the PBD matrix,

$$\mathbf{De} = \begin{bmatrix} (\mathbf{W}_1\mathbf{p}_1)\mathbf{e} \\ (\mathbf{W}_2\mathbf{p}_2)\mathbf{e} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_1(\mathbf{p}_1\mathbf{e}) \\ \mathbf{W}_2(\mathbf{p}_2\mathbf{e}) \end{bmatrix} = \mathbf{s}. \tag{8}$$

Therefore, the elements of $\mathbf{e}$ can be recovered from the following two systems of equations separately,

$$\mathbf{W}_1\mathbf{z} = \mathbf{s}_1, \tag{9}$$

$$\mathbf{W}_2\mathbf{z} = \mathbf{s}_2, \tag{10}$$

where $\mathbf{s}_1$, $\mathbf{s}_2$ denote the first and second halves of $\mathbf{s}$. In each system, since the matrix, $\mathbf{W}_1$ or $\mathbf{W}_2$, is block diagonal, the high dimensional equations can be divided into a group of highly low-dimensional underdetermined equations. Each one corresponds to a segment of $\mathbf{e}$. When a segment is sufficiently sparse, it can be recovered by solving a sparse recovery problem. Since the dimension can be only 2 or 4, the sparsest solution can be found by the direct exhaustive search. Further, the entries of the recovered segments of Eq. (9) can be substituted into Eq. (10) to recover more entries, and vice versa. Such a cross substitution stage continues until no new entry is recovered or all the entries have been recovered. In the case that there are still some entries unrecovered after the cross solving procedure, a simple pseudoinversion process is applied to the residual equations corresponding to the unrecovered entries, in order to recover the remaining entries.

3.3 Complexity

In the proposed cryptosystem, the encryption and decryption are conducted in a matrix style. No large number calculations are involved. In the encryption, a plaintext is simply multiplied by a matrix and added with a sparse vector. The calculation is KN multiplications and KN additions, where K is the length of the plaintext and N is the length of the ciphertext. Note that for each time K numbers in the vector are encrypted. For each number, only N multiplications and N additions are needed. The complexity is only $O(N)$.

The decryption includes several procedures. The first one is to apply the inverse permutation matrix to a vector, the computation needed is N flops. The complexity of applying $\mathbf{D}$ to $\mathbf{y}'$ is proportional to the nonzeros in $\mathbf{D}$, which is $O(N)$. The complexity of applying $\mathbf{F}^T$ to $\mathbf{y}' - \mathbf{e}'$ is $O(KN)$. The complexity of applying $\mathbf{S}^{-1}$ to $\mathbf{x}'$ is $O(K^2)$. The last one is using the CLP algorithm to recover $\mathbf{e}$. In the CLP algorithm, the most time-consuming part is solving the small systems of low-dimension equations. Since the dimension of the low-dimension equations is constant and the number of these equations grows linearly as N increases, the complexity for solve them is linear to N. Solving the remaining residual equation consumes the second most computation time. Since the corresponding residual matrix is highly sparse, the solution can be quickly obtained by a Conjugate Gradient (CG) method, where the corresponding complexity is also $O(N)$. Therefore, the complexity of the CLP algorithm is still $O(N)$. The details of complexity analysis of the CLP algorithm are in [24]. It can be seen that for a K-length plaintext, the total complexity of the decryption is also $O(KN)$, or for each number, the complexity is $O(N)$.

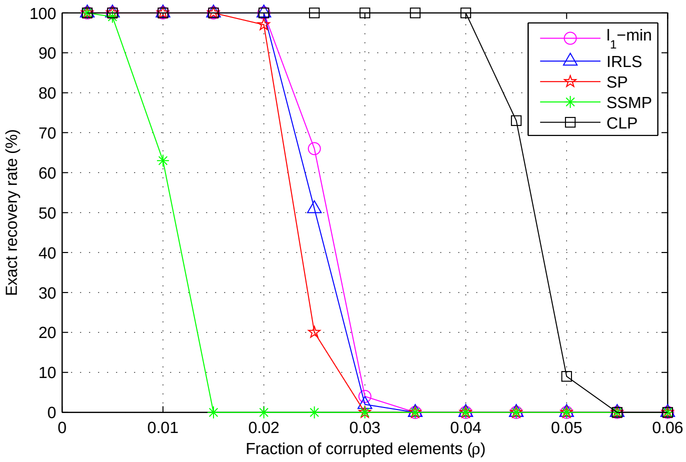


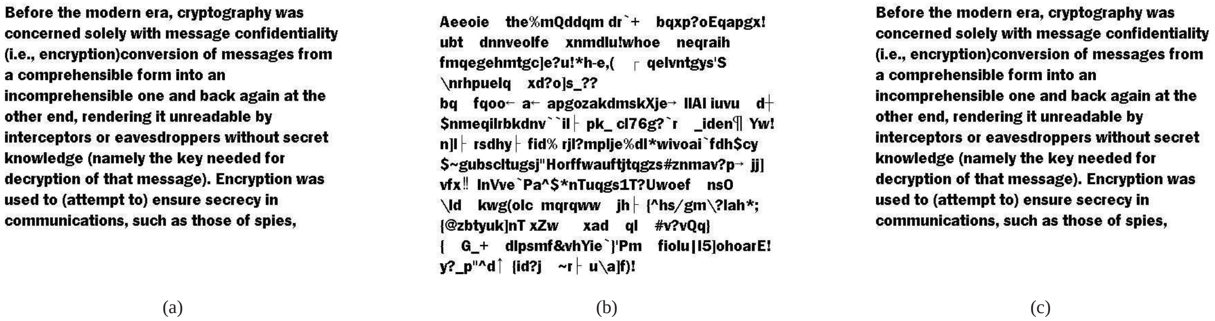
Fig. 2: Exact recovery rate versus fraction of corruption with $K = 1792$, $N = 2048$ ($\frac{K}{N} = \frac{7}{8}$).

3.4 Practical example

Figure 1 shows a example using the proposed cryptosystem. The original message is a text message with 448 characters. Characters are transformed into numbers with the corresponding ASCII codes; the public key $\mathbf{F}$ is of size 512×448 generated by the QR factorization; the public key $\mathbf{e}$ is with 4% nonzero entries, whose locations and signs are randomly chosen and the magnitudes are the mean of the codeword. Figure 1 (b) shows the decrypted text message by the ℓ_1-min algorithm and Figure 1 (c) shows the decrypted text message by the CLP algorithm. It can be observed that the decryption algorithm CLP decrypted the text message successfully, while the ℓ_1-min algorithm failed.

3.5 Setting ρ

The decryption algorithm CLP is efficient and depends on the secret key. Further for the security, it should be satisfied that other sparse recovery algorithms can not decrypt the ciphertext. This can be done by choosing the proportion (ρ) of the nonzero elements of $\mathbf{e}$, since the CLP algorithm can recover a much higher proportion of nonzeros than other algorithms. Choosing a proper ρ is of significant importance for the security and successful decryption. When ρ is too small, the ciphertext can be easily attacked using usual sparse recovery algorithms; while if it is too large, CLP cannot recover $\mathbf{e}$ and the ciphertext cannot be decrypted. Suitable ρ should be beyond the upper recovery bound of other algorithms and below the lower recovery bound of CLP.



Before the modern era, cryptography was
concerned solely with message confidentiality
(i.e., encryption)conversion of messages from
a comprehensible form into an
incomprehensible one and back again at the
other end, rendering it unreadable by
interceptors or eavesdroppers without secret
knowledge (namely the key needed for
decryption of that message). Encryption was
used to (attempt to) ensure secrecy in
communications, such as those of spies,

(a)

Aeeoie the%mQddqm dr`+ bqxp?oEqapgx!
ubt dnnveolfe xnmdlu!whoe neqraih
fmqegehmtgc]e?u!*h-e,(┌ qelvntgys'S
\nrhpuelq xd?o]s_??
bq fqoo← a← apgozakdmskXje→ llAl iuvu d┼
$nmeqilrbkdnv``il├ pk_ cl76g?`r _iden¶ Yw!
n]l├ rsdhy├ fid% rjl?mplje%dl*wivoai`fdh$cy
$~gubscltugsj"Horffwauftjtqgzs#znmav?p→ jj]
vfx‼ lnVve`Pa^$*nTuqgs1T?Uwoef nsO
\ld kwg(olc mqrqww jh├ {^hs/gm\?lah*;
{@zbtyuk]nT xZw xad ql #v?vQq}
{ G_+ dlpsmf&vhYie`}'Pm fiolu|l5]ohoarE!
y?_p"^d↑ {id?j ~r├ u\a]f)!

(b)

Before the modern era, cryptography was
concerned solely with message confidentiality
(i.e., encryption)conversion of messages from
a comprehensible form into an
incomprehensible one and back again at the
other end, rendering it unreadable by
interceptors or eavesdroppers without secret
knowledge (namely the key needed for
decryption of that message). Encryption was
used to (attempt to) ensure secrecy in
communications, such as those of spies,

(c)

Fig. 1: An example of the proposed cryptosystem with a text message ($\frac{K}{N} = \frac{7}{8}$, $\rho = 4\%$). (a) The original text message; (b) The text message decrypted by ℓ_1-min; (c) The text message decrypted by CLP.

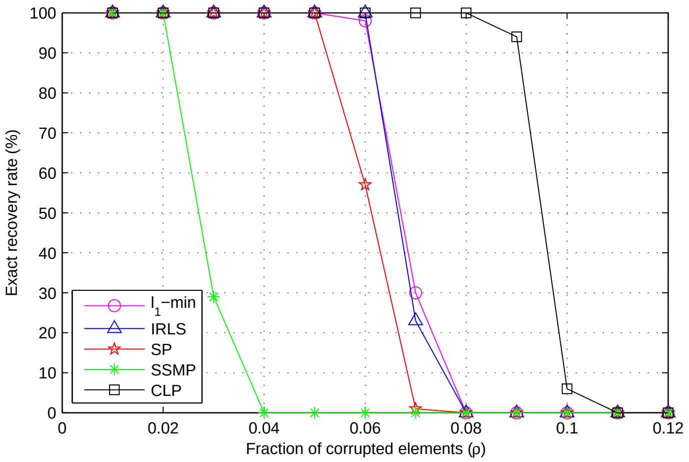


Fig. 3: Exact recovery rate versus fraction of corruption with $K = 1536$, $N = 2048$ ($\frac{K}{N} = \frac{3}{4}$).

In the proposed cryptosystem, recovering $\mathbf{e}$ is necessary and sufficient for decrypting the ciphertext. In the decryption stage, The PBD matrix serves as the secret key and CLP uses it to recover $\mathbf{e}$. However, for the hackers, the secret key is unavailable since it does not need to be transmitted. In this case, a natural way of trying to recover $\mathbf{e}$ is to construct another matrix, whose rows span the null subspace of $\mathbf{H}$. Let $\mathbf{R}$ be such a matrix, and $\mathbf{R}$ satisfies $\mathbf{RH} = \mathbf{0}$. The following steps are similar to the decryption, where another sparse recovery algorithm is used to recover $\mathbf{e}$ instead of CLP. If ρ is beyond the upper recovery bound of the algorithm, it is impossible to recover $\mathbf{e}$. Theoretical research results have proven that the bound of widely-used algorithms, e.g., ℓ_1-min algorithms, is $\frac{M \cdot \alpha}{N}$, where $\alpha = \frac{1}{O(\log(\frac{N}{M}))}$. However, it is a rough bound and impractical for choosing ρ. Therefore, a clearer bound needs to be found through numerical studies.

The theoretical results show that the bound depends on the ratio of $\frac{M}{N}$, or $\frac{K}{N}$ instead. Here we show the numerical study of two cases: $\frac{K}{N} = \frac{7}{8}$ and $\frac{K}{N} = \frac{3}{4}$. We compare the recovery ability of CLP with four well-known approaches: ℓ_1-min, IRLS (for solving the nonconvex optimization problem: (P_p) with $p < 1$), Subspace Pursuit (SP) [17], and SSMP [25]. The concrete solver for ℓ_1-min is PDCO [26], and the concrete IRLS algorithm is proposed by Daubechies *et. al.* [23], where p gradually varies from 1 to 0.5. These algorithms stands for the most powerful sparse recovery algorithms. Given the secret key $(\mathbf{D}, \mathbf{Q}, \mathbf{S})$ and the public key $(\mathbf{H}, \rho)$, the experiment form is as follows:

1.set I with $|I| = \rho N$ uniformly at random, and generate a sparse vector $\mathbf{e}$ with random ± 1 on I;
2.generate a random vector $\mathbf{x}$ as the plaintext and make $\mathbf{Hx} + \mathbf{e}$;
3.generate a nonsigular matrix $\mathbf{R} \in \mathbb{R}^{\mathbf{M \times N}}$ such that $\mathbf{RH} = \mathbf{0}$;
4.recover $\mathbf{x}$ by $\mathbf{D}$ or $\mathbf{R}$ and the corresponding algorithms;
5.for each ρ, repeat 1)-4) steps for 100 times and compute the percentage of exact recovery (an exact recovery is considered to be achieved when $\frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leq 10^{-5}$).

Figures 2 and 3 show the results. We can observe that when $\frac{K}{N} = \frac{7}{8}$, ρ can be set between the range of 3.5% and 4%. While for the case of $\frac{K}{N} = \frac{3}{4}$, ρ can be set around 8%. The large $\frac{K}{N}$ is, the wider gap between the bounds of CLP and the other algorithms it is, and more difficult it is to attack the cryptosystem using these algorithms. This is consistent with our massive numerical studies. However, $\frac{K}{N}$ should not be too small. The number of nonzeros of $\mathbf{e}$ is ρN. When ρN is too small, $\mathbf{e}$ can be easily recovered by direct combinational exhaustive search. For example, when $\rho N = 1$, there is only one nonzero in $\mathbf{e}$; it can be easily recovered by finding a most coherent column in $\mathbf{D}$ to the measurements $\mathbf{s}$. From the viewpoint of security, the choice of $\frac{K}{N}$ is wide. For example, in the experiments, where $N = 2048$, $\frac{K}{N}$ can be set between $\frac{1}{16}$ and $\frac{1}{8}$ to ensure the high security. It demonstrated that the proposed cryptosystem is secure with a proper pair of $\frac{K}{N}$ and ρ that can be easily chosen.

3.6 Security

Although the proposed PKC is defined in the real-field, different from the finite-field defined McEliece PKC, the main structures are similar. The proposed PKC is as secure as the McEliece PKC. We discuss the difficulty of decrypting $\mathbf{x}$ when $\mathbf{H}$ and $\mathbf{y}$ are known. One possible attack way can be trying to recover $\mathbf{D}$ and further use the CLP algorithm to recover $\mathbf{e}$. Another possible attack way is to recover $\mathbf{e}$ directly without the CLP algorithm.

In the first attack, $\mathbf{D}$ needs to be recovered if one wish to use the CLP algorithm for decryption. This is because CLP is a matrix-dependent algorithm. Without knowing $\mathbf{D}$, CLP is unavailable. However, recovering $\mathbf{D}$ from $\mathbf{PFS}$ seems impossible. The only information for recovering $\mathbf{D}$ is that $\mathbf{DF} = \mathbf{0}$. Therefore, $\mathbf{F}$ needs to be recovered first at least. What makes the attack hopeless is that it can not be recovered in reasonable time when N and K are large enough.

To directly recover $\mathbf{e}$ without the CLP algorithm, it is to solve the sparse recovery problem, where the sparse vector is with a high proportion of nonzeros. As the proportion of nonzeros, ρ, is set much beyond the solving bound of the available algorithms, $\mathbf{e}$ can not be recovered by them. The only way is to search for the sparsest solution to an underdetermined system, which is known to be an NP-hard problem. For an N-length vector $\mathbf{e}$, the probability of locating ρN nonzeros is: $\binom{N}{\rho N}$. For example, when $N = 1024$ and $\rho = 4\%$, $\binom{1024}{41}$ $= 3.5 \times 10^{73}$.

4 Conclusion

In this paper, we proposed a novel real-field PKC based on sparse recovery. The practical encryption and decryption algorithms are presented as well as parameter choosing, complexity and security analysis. The proposed cyptosystem has two important advantages: efficient and secure. The complexities of encryption and decryption for each number are low. To attack the cryptosystem, an NP-hard sparse recovery problem has to be solved.

Acknowledgement

This research was supported by NSFC (51405434), the Science Fund for Creative Research Groups of NSFC (51221004), Zhejiang Provincial Natural Science Foundation (LQ13F020002), and the Fundamental Research Funds for the Central Universities.

References

[1] Rivest, A. Shamir R., and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, **21**, pp. 120–126, 1978.

[2] V. Miller, "Use of elliptic curves in cryptography," *CRYPTO 85*, pp. 416–426, 1985.

[3] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, **48**, pp. 203–209, 1987.

[4] R.J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *JPL DSN Progress Report*, **42**, pp. 114–116, 1978.

[5] Niederreiter H., "Knapsack-type cryptosystems and algebraic coding theory," *Probl Contr Inf Theory*, **15**, pp. 157–166, 1986.

[6] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, **51**, pp. 4203–4215, December 2005.

[7] D. L. Donoho and M. Elad, "Optimally sparse representation from overcomplete dictionaries via ℓ^1-norm minimization," *Proc. Nat. Acad. Sci. USA*, **100**, pp. 2197–2002, March 2003.

[8] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, **52**, pp. 489–509, February 2006.

[9] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, **52**, pp. 1289–1306, April 2006.

[10] S. Muthukrishnan, *Data Streams: Algorithms and Applications*, chapter Syntax-directed program modularization, Now Publishers, 2005.

[11] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, **41**, pp. 3397–3415, December 1993.

[12] G. Davis, S. Mallat, and Z. Zhang, "Adaptive time-frequency decompositions," *Opt. Eng.*, **33**, pp. 2183–2191, July 1994.

[13] D. Needell and R. Vershynin, "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit," *Found. Comput. Math.*, **9**, pp. 317–334, 2009.

[14] D. L. Donoho and Y. Tsaig, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," *Preprint*, March 2006.

[15] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inform. Theory*, **53**, pp. 4655–4666, December 2007.

[16] D. Needell and J.A. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, **26**, pp. 301–321, May 2009.

[17] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inform. Theory*, **55**, pp. 2230–2249, May 2009.

[18] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, **20**, pp. 33–61, 1999.

[19] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Statist. Soc. B*, *58*, pp. 267–288, 1996.

[20] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application

to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, **1**, pp. 586–598, December 2007.

[21] D. L. Donoho and Y. Tsaig, "Fast solution of ℓ^1 norm minimization problems when the solution may be sparse," *IEEE Trans. Inform. Theory*, **54**, pp. 4789–4812, November 2008.

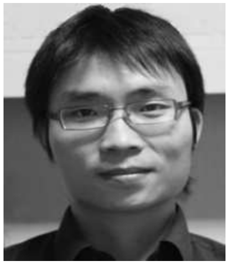
[22] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Processing Letters*, **14**, pp. 707–710, 2007.

[23] Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and C. Sinan Gntrk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, pp. 1–8, January 2010.

[24] Zaixing He, Takahiro Ogawa, and Miki Haseyama, "Cross low-dimension pursuit for sparse signal recovery from incomplete measurements based on permuted block diagonal matrix," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, **E94-A**, pp. 1793–1803. Available at: https://dl.dropboxusercontent.com/u/2786346/CLP.pdf, September 2011.

[25] R. Berinde and P. Indyk, "Sequential sparse matching pursuit," in *Proc. 47th Annu. Allerton Conf. Commun. Control Comput.*, 2009, pp. 36–43.

[26] M. A. Saunders, "Pdco: Primal-dual interior method for convex objectives," http://www.stanford.edu/group/SOL/software/pdco.html.



Zaixing HE received his B.Sc. and M.Sc. degrees in Mechanical Engineering from Zhejiang University, China in 2006 and 2008, respectively. He received his Ph. D. degree in 2012 from the Graduate School of Information Science and Technology, Hokkaido University, Japan. He is currently an assistant professor in the Department of Mechanical Engineering, Zhejiang University. His research interests include sparse recovery, sparse representation, compressed sensing and their applications to image processing, signal processing and pattern recognition.

Xinyue ZHAO received her M.S. degree in Mechanical Engineering from Zhejiang University, China in 2008, and her Ph.D degree in Graduate School of Information Science and Technology from Hokkaido University, Japan in 2012. She is currently an assistant professor in the Department of Mechanical Engineering, Zhejiang University, China. Her research interests include computer vision and image processing.

Shuyou ZHANG received his M.S. degree in Mechanical Engineering and the Ph.D. degree in State Key Lab. Of CAD&CG from Zhejiang University, China, in 1991 and 1999, respectively. He is currently a professor in the Department of Mechanical Engineering, Zhejiang University, China. He is also the vice administer of Institute of Engineering & Computer Graphics in Zhejiang University, assistant director of Computer Graphics Professional Committee for China Engineering Graphic Society, member of Product Digital Design Professional Committee, and chairman of Zhejiang Engineering Graphic Society. His research interests include product digital design, design and stimulation for complex equipments, and engineering and computer graphics.