A Public Key Exchange Cryptosystem Based on Ideal Secrecy

Vamshi Krishna Kammadanam
IITB-Monash Resarch Academy
IIT-Bombay
Mumbai, Maharashtra, India 400076
vkkam1@student.monash.edu

Virendra R. Sule
Department of Electrical Engineering
IIT-Bombay
Mumbai, Maharashtra, India 400076
vrs@ee.iitb.ac.in

Yi Hong
Department of Electrical and Computer Science Engineering
Monash University
Melbourne, Victoria, Australia
yi.hong@monash.edu

November 21, 2018

Abstract

This paper proposes two closely related asymmetric key (or a public key) schemes for key exchange whose security is based on the notion of *ideal secrecy*. In the first scheme, the private key consists of two singular matrices, a polar code matrix and a random permutation matrix all over the binary field. The sender transmits addition of two messages over a public channel using the public key of the receiver. The receiver can decrypt individual messages using the private key. An adversary, without the knowledge of the private key, can only compute multiple equiprobable solutions in a space of sufficiently large size related to the dimension of the kernel of the singular matrices. This achieves security in the sense of ideal secrecy. The next scheme extends over general matrices. The two schemes are cryptanalyzed against various attacks.

1 Introduction

Symmetric key cryptography arose to satisfy the major need of the civilisation, that of confidentiality of information. It served at least in principle, two functions, that of *confidentiality* and *data integrity* of communicating large quantity (bulk) of information or data over an insecure channel tapped by a passive adversary. However, a symmetric key cryptographic scheme presumes a secure channel for key exchange for its own functioning. This requirement was fulfilled by several ingenious asymmetric (public) key schemes. The well known Diffie Hellman key exchange protocol in 1976 was followed by RSA [1] and Rabin [2] based on integer factorisation problem, El-Gammal[3] based on discrete logarithm problem, Paillier [4], McEliece [5] and NTRU [6]. These have lead to an assurance of security of information transaction over internet and have provided a practically feasible infrastructure for secure E-commerce since the 1990s. Public key cryptography has now provided the functionalities of small scale encryption, key exchange over public channels and authentication (signature) of entities. These in turn have shown a vast scope of application to practice of information storage, communication and transaction affecting the civilisation from secure personal level transactions over internet, E-commerce, secure networks for communication to conduct of elections affecting the nations and democracies.

Unfortunately, despite several new and powerful proposals in asymmetric key cryptography after the original schemes, it turns out that a very small number of public key proposals have survived to offer security for practical use. Security of most of these primitives have been based on computational hardness of the schemes. Although Shannon's original paper [7] proposed the concept of *perfect secrecy* proving perfect secrecy of the one time pad, this concept did not percolate into public key schemes which were dominated by computational hardness (or complexity) for security measure in contrast to information theoretic security. For security all of these public key schemes utilize unproved computational hardness of one of the problems such as the Elliptic curve discrete log computation, prime factorization or sometimes proved hypotheses such as NP completeness of a computational problem, square root computation modulo numbers without prime factorization and shortest vector problem in Lattices. In recent times information theoretic security has turned out to be the basis of *secrecy capacity* for wiretapped channels [8], [9] and forms an alternative to the traditional public key cryptography.

1.1 Public key scheme based on ideal secrecy

The purpose of this paper is to propose two related *asymmetric* (hereafter called *public*) key schemes whose security is not based purely on computational hardness but is based also on the notion of *ideal secrecy* which seems to have appeared first in a paper by Geffe [10]. This shift to ideal secrecy offers a new direction to search for public key primitives. In this paper we provide concrete realizations of such schemes. The notion of ideal secrecy is briefly equivalent to guessing a correct string in a large set. This can be explained as follows. If a function having the trapdoor information (private key) has a large and equiprobably number of arguments (inputs) which give rise to the same output without using the key, then even if the problem of constructing the input given the output is not computationally hard guessing the correct input is secured by the knowledge of the trapdoor. We say that function provides an ideal secrecy if the size of the set of inputs where guess is equiprobable is large enough. In the special cases of primitives when this set is itself as large as the key space, this secrecy is analogous to perfect secrecy [10].

From the point of view of computational security, the ideal secrecy can be systematically explored in terms of *one way functions* (OWFs) which form the basic foundation of both symmetric as well as public key cryptography. Our notion of security of the proposed public key scheme re-interprets this notion of one way-ness in terms of ideal secrecy. This differentiation is explained next.

1.1.1 One way, trapdoor one way and one way functions based on ideal secrecy

Both the symmetric and public key primitives depend on functions with practically feasible computation, called *one way function* (OWF) and *trapdoor one way function* (TOWF). Public key schemes based on computational hardness for security have required three types of functions as follows. For asymmetric encryption traditionally one requires three functions $F : \mathcal{U} \times \mathcal{X} \to \mathcal{Y}$, $H : \mathcal{Y} \times \mathcal{R} \to \mathcal{X}$ and $G : \mathcal{R} \to \mathcal{U}$ where $\mathcal{U}$ denotes the set of public keys while $\mathcal{R}$ is the set of private keys. The functions $\mathcal{F}$, $\mathcal{H}$ and $\mathcal{G}$ are required to have following properties.

1. $F(u, \cdot) : \mathcal{X} \to \mathcal{Y}$ is a OWF for every $u = G(r)$ for any r in $\mathcal{R}$.
2. $H(r, \cdot) : \mathcal{Y} \to \mathcal{X}$ is a TOWF with trapdoor r.
3. $G(\cdot) : \mathcal{R} \to \mathcal{U}$ is a OWF.

Here the OWFs are considered in the sense of computational hardness described at the start of Section 2.1 on the following page. In the asymmetric scheme we consider a notion of asymmetric encryption based on ideal OWFs as follows.

An asymmetric encryption based on ideal secrecy is defined by three functions $F : \mathcal{U} \times \mathcal{X} \to \mathcal{Y}$, $H : \mathcal{R} \times \mathcal{X} \to \mathcal{X}$ and $G : \mathcal{R} \to \mathcal{U}$ such that, for $u = G(r)$ for any r in $\mathcal{R}$ the function $F(u, .) : \mathcal{X} \to \mathcal{Y}$ is an ideal OWF, $H(r, .) : \mathcal{Y} \to \mathcal{X}$ is an ideal TOWF while G is a OWF. The definitions of an ideal OWF (IOWF) and an ideal TWOF (ITWOF) are given in the next section. This briefly presents the nature of the public key scheme we develop in this paper and its difference with respect to traditional schemes based on computational security.

1.2 Previous work on public key primitives

The literature has two kinds of public key primitives. One is used for encryption and the second is for key exchange. Although if you have a public key scheme available you can exchange something securely over a public channel, but there is no need to exchange information beforehand if you want to encrypt information by public key and send. At the same time, if you have a key exchange scheme over a public channel, since you are able to exchange a key securely you can also do encryption. Both solve the problems of key exchange and encryption over a public channel. Originally, Diffie Hellman [11] was proposed as exchange of key over public channel and RSA [1] and [2] was proposed as encryption scheme over a public channel. After RSA, El-Gammal [3] scheme was proposed to show that encryption could be done using Diffie-Hellman key exchange. All the three schemes are later used for development of signature, which solved the problem of authentication of entities. After elliptic curve based Diffie Hellman scheme (ECC) came on the scene several possibilities of protocols and signature schemes based on public key exchange using pairing were proposed and form foundation of public key cryptography. The literature on ECC is too vast but comprehensively available in [12], [13]. Other successful public key cryptographic schemes are McEliece [5] and NTRU [6]. All of these schemes offer security on the computational hardness of certain problems such as elliptic curve discrete log, prime factorization, shortest vector in a lattice [6] or NP-completeness of decoding [5]. The central theme of the cryptographic scheme we propose in this paper is to construct ITWOFs with properties stated above required for public key schemes. We first construct the public key primitive based on polar code matrix and study its security. Then propose a primitive using general matrices.

2 Definitions of ideal secrecy, OWF, TOWF

In this section we discuss definitions of OWF and TOWF and their variants based on ideal secrecy. Let U_B, R_B be public and private keys of an entity Bob. In a public key cryptography, suppose Alice wants to send a message m to Bob. Consider public private key pair being related by a function G,

$$U_B = G(R_B)$$

where G is required to be a OWF. The encrypted ciphertext c of the message m is,

$$c = E(U_B, m)$$

Hence E also has to be a OWF of m since U_B is known to public. The decryption function with trapdoor R_B and ciphertext c is

$$m = D(R_B, c)$$

Here D has to be a TOWF with trapdoor R_B. The ideal secrecy variants of such a scheme can be considered with the help of ideal OWF and ideal TWOF which are defined next.

2.1 Definitions of IOWF and ITOWF

Traditionally a OWF is a function $F : \mathcal{X} \to \mathcal{Y}$ between sets $\mathcal{X}$, $\mathcal{Y}$ which allows "easy" (i.e. in practically feasible time) computation of y in $\mathcal{Y}$ given any member

x in $\mathcal{X}$ but given y in $\mathcal{Y}$ computation of an x in $\mathcal{X}$ such that $y = F(x)$ is "difficult" (i.e. there is no known algorithm to compute x in practically feasible time). A OWF with the variant of ideal secrecy is defined as follows.

Definition 2.1 (Ideal One Way Function) *A function $F : \mathcal{X} \to \mathcal{Y}$ from a set $\mathcal{X}$ to $\mathcal{Y}$ is said to be* Ideal One Way Function *(IOWF) of* order n *if*

1. *Computation of $y = F(x)$ for any x in $\mathcal{X}$ is "easy".*
2. *Given y there is a set $S_y \subset \mathcal{X}$ of size at least n such that any known algorithm results into computation of an equiprobable number of x in S_y which satisfy $y = F(x)$.*

To define a decryption function using private key, traditionally one needs the concept of a Trapdoor One Way Function (TOWF) where the trapdoor represents the private key. A function $F : \mathcal{T} \times \mathcal{X} \to \mathcal{Y}$ with $\mathcal{T}$ the set of keys or trapdoors is called a TOWF if for each t in $\mathcal{T}$ the function $F_t : \mathcal{X} \to \mathcal{Y}$ where $F_t(x) = F(t, x)$ is a OWF and given a pair (x, y) in $\mathcal{X} \times \mathcal{Y}$ such that $y = F_t(x)$ for some t, computation of t such that $y = F(t, x)$ is not known to be feasible. From the notion of IOWF above we can define a notion of Ideal TOWF (ITWOF) as follows.

Definition 2.2 (Ideal Trapdoor One Way Function) *A function $F : \mathcal{T} \times \mathcal{X} \to \mathcal{Y}$ is said to be an* Ideal TOWF *(ITOWF) of order n with trapdoor set $\mathcal{T}$ if*

1. *for any t in $\mathcal{T}$ the function $F_t : \mathcal{X} \to \mathcal{Y}$ where $y = F_t(x) = F(t, x)$ is an IOWF of order n.*
2. *Given the pair (x, y) computation of t in $\mathcal{T}$ such that $y = F(t, x)$ is not known to be feasible.*

Using these ideal variants of OWF, TOWF we construct a public key exchange scheme in next section using the algebraic properties of the polar code matrix.

3 Key exchange based on the public key primitive using polar code matrix

In this section, we discuss properties of polar code matrix and present an IOWF and an ITOWF for a public key scheme with examples. Polar codes, introduced by Arıkan in [14], are the linear block error correction codes with provable capacity-achieving capability over certain types of discrete memoryless channels, and uses only $\frac{N}{2} \log N$ XOR operations at each encoding/decoding of a length-N polar code. In this paper, we only focus on the polar code matrix. Given a polar code, denoted by $(N, K, \mathcal{A})$, where $N = 2^n$ and K denote the code length and the message length, n is an integer, and $\mathcal{A}$ is the set of information bit indices with cardinality $|\mathcal{A}| = K$, the polar code matrix is defined as

$$\mathbf{G} \triangleq \mathbf{F}^{\otimes n}$$

where $\mathbf{G}$ is an n fold Kronecker product of

$$\mathbf{F} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \tag{1}$$

which is a 2×2 binary kernel matrix of a polar code. We have the following property of the tensor product.

Proposition 3.1 *Let* $\mathbf{A}$*,* $\mathbf{B}$*,* $\mathbf{C}$ *and* $\mathbf{D}$ *are matrices of size* $m \times n$*,* $p \times q$*,* $n \times k$ *and* $q \times r$ *respectively. Then*

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD})$$

See [15] for this identity is known as mixed-product property. The next proposition is also well known in [16].

Proposition 3.2

$$\mathbf{F}^{\otimes n}\mathbf{F}^{\otimes n} = \mathbf{I}_N \tag{2}$$

where the operation between the matrices is binary matrix multiplication operation modular 2 *and* $\mathbf{I}_N$ *denotes an identity matrix of size* $N \times N$*.*

Proof. For $n = 1$, it is straightforward, i.e., $\mathbf{F} \cdot \mathbf{F} = \mathbf{I}_2$. For $n \geq 2$, we prove it by induction using the following recursion. Suppose Proposition 3.2 holds for all values of n up to k, where $k \geq 1$. When $n = k + 1$, we obtain

$$\begin{aligned} \mathbf{F}^{\otimes n}\mathbf{F}^{\otimes n} &= (\mathbf{F} \otimes \mathbf{F}^{\otimes k})(\mathbf{F} \otimes \mathbf{F}^{\otimes k}) \\ &= (\mathbf{FF}) \otimes (\mathbf{F}^{\otimes k}\mathbf{F}^{\otimes k}) \\ &= \mathbf{I}_2 \otimes \mathbf{I}_{2^k} \\ &= \mathbf{I}_{2^n} \end{aligned} \tag{3}$$

Eq. (3) is due the identity in Proposition 3.1. □

Proposition 3.3 *The multiplication of any binary vector* $\mathbf{v}$ *of length* n *and* $\mathbf{G}$ *can be treated as a non-systematic polar encoder with the input vector* $\mathbf{v}$*, which has the exact* $\frac{N}{2}\log N$ *XOR operations [17].*

The row (or column) index set of a polar code generator matrix $\mathbf{G} = \mathbf{F}^{\otimes n}$, where $\mathbf{F}$ is shown in Eq. (1), is $\mathcal{I}_N \triangleq \{1, 2, ..., N\}$. Consider the set of indices $\mathcal{A} \triangleq \{i_1, ..., i_K\} \subset \mathcal{I}_N$, $(i_1 < i_2 < ... < i_K)$, and $\mathcal{A}^c \triangleq \mathcal{I}_N \setminus \mathcal{A} = \{i_{K+1}, ..., i_N\}$, $(i_{K+1} < i_{K+2} < ... < i_N)$. Then $\mathbf{G}$ has submatrices $\mathbf{G}_{\mathcal{A}}$ and $\mathbf{G}_{\mathcal{A}^c}$, according to the sets $\{\mathcal{A}\}$ and $\{\mathcal{A}^c\}$ as row indices. Similarly, using the sets $\{\mathcal{A}\}$ and $\{\mathcal{A}^c\}$ as column indices, the matrix $\mathbf{G}$ has submatrices $\mathbf{H}_{\mathcal{A}}$ and $\mathbf{H}_{\mathcal{A}^c}$. Hence there is a row permutation $\mathbf{Q}$ such that

$$\mathbf{QG} = \begin{bmatrix} \mathbf{G}_{\mathcal{A}} \\ \mathbf{G}_{\mathcal{A}^c} \end{bmatrix} \tag{4}$$

$$\mathbf{GQ}^T = \begin{bmatrix} \mathbf{H}_{\mathcal{A}} & \mathbf{H}_{\mathcal{A}^c} \end{bmatrix} \tag{5}$$

We have the following property.

Property 3.4

$$\mathbf{G}_{\mathcal{A}}\mathbf{H}_{\mathcal{A}} = \mathbf{I}_K \tag{6}$$

$$\mathbf{G}_{\mathcal{A}}\mathbf{H}_{\mathcal{A}^c} = \mathbf{0}_{K\times(N-K)} \tag{7}$$

$$\mathbf{G}_{\mathcal{A}^c}\mathbf{H}_{\mathcal{A}} = \mathbf{0}_{(N-K)\times K} \tag{8}$$

$$\mathbf{G}_{\mathcal{A}^c}\mathbf{H}_{\mathcal{A}^c} = \mathbf{I}_{N-K} \tag{9}$$

Proof. From the definitions of $\mathbf{G}_{\mathcal{A}}$, $\mathbf{G}_{\mathcal{A}^c}$, $\mathbf{H}_{\mathcal{A}}$ and $\mathbf{H}_{\mathcal{A}^c}$. There is a permutation matrix $\mathbf{Q}$ of size $N \times N$, such that

$$\mathbf{Q}\mathbf{F}^{\otimes n} = \begin{bmatrix}\mathbf{G}_{\mathcal{A}}\\ \mathbf{G}_{\mathcal{A}^c}\end{bmatrix} = \mathbf{G}$$

$$\mathbf{F}^{\otimes n}\mathbf{Q}^T = \begin{bmatrix}\mathbf{H}_{\mathcal{A}} & \mathbf{H}_{\mathcal{A}^c}\end{bmatrix} = \mathbf{H}$$

and

$$\mathbf{G}\mathbf{H} = \mathbf{Q}\mathbf{F}^{\otimes n}\mathbf{F}^{\otimes n}\mathbf{Q}^T = \mathbf{I}$$

Hence the proof follows $\mathbf{F}^{\otimes n}\mathbf{F}^{\otimes n} = \mathbf{I}$ and $\mathbf{Q}$ being a permutation matrix. □

Based on the polar code matrix properties in Section 3 on page 5, we present the following asymmetric key cryptosystem.

3.1 Public Key Exchange Scheme Based On Polar Codes

Consider Alice initiating a key exchange session with Bob whose private and public keys are defined as follows.

3.1.1 Private and public keys

Recall the row index set $\mathcal{I}_N = \{1, \ldots, N\}$ and the set $\{\mathcal{A}\} \subset \mathcal{I}_N$. Let $\mathbf{S}_1$ and $\mathbf{S}_2$ be $K \times K$ and $(N-K)\times(N-K)$ singular matrices respectively with ranks r_1, r_2 and $\mathbf{P}$ be the $N \times N$ binary permutation matrix. Define

$$\mathbb{1}_{\mathcal{A}}(i) \triangleq \begin{cases} 1 & \text{if } \ i \in \mathcal{A} \\ 0 & \text{if } \ i \in \mathcal{A}^c \end{cases}$$

We then define the private key $\mathcal{T}^{priv} \triangleq \{\mathbb{1}_{\mathcal{A}}, \mathbf{S}_1, \mathbf{S}_2, \mathbf{P}\}$ and the public key

$$\mathcal{T}^{pub} \triangleq \{\mathbf{G}^{pub}_{\mathcal{A}} = \mathbf{S}_1\mathbf{G}_{\mathcal{A}}\mathbf{P}, \mathbf{G}^{pub}_{\mathcal{A}^c} = \mathbf{S}_2\mathbf{G}_{\mathcal{A}^c}\mathbf{P}\} \tag{10}$$

As discussed earlier for public key primitive we require an IOWF for encryption and an ITOWF for decryption which are constructed below.

3.1.2 Encryption and decryption

Alice has two randomly generated input vectors x_1 of length K-bits and x_2 of length $N-K$-bits and the public key $\mathcal{T}^{pub}$, the encryption of x_1, x_2 is given by

$$c = x_1\mathbf{G}_{\mathcal{A}}^{pub} + x_2\mathbf{G}_{\mathcal{A}^c}^{pub} \tag{11}$$

Given the private key $\mathcal{T}^{priv}$, the cipher-text c, Bob decrypts as

$$\begin{aligned} c\mathbf{P}^T\mathbf{H}_{\mathcal{A}} &= (x_1\mathbf{S}_1\mathbf{G}_{\mathcal{A}}\mathbf{P} + x_2\mathbf{S}_2\mathbf{G}_{\mathcal{A}^c}\mathbf{P})\mathbf{P}^T\mathbf{H}_{\mathcal{A}} \\ &= (x_1\mathbf{S}_1\mathbf{G}_{\mathcal{A}} + x_2\mathbf{S_2}\mathbf{G}_{\mathcal{A}^c})\mathbf{H}_{\mathcal{A}} \\ &= x_1\mathbf{S}_1 \end{aligned} \tag{12}$$

The simplification to Eq. (12) is due to $\mathbf{P}\mathbf{P}^T = \mathbf{I}$ and Eq. (7) on page 7 in Property 3.4 on page 7 . Then finally Bob multiplies by $\mathbf{G}_{\mathcal{A}}\mathbf{P}$ in Eq. (12) to get $x_1\mathbf{S}_1\mathbf{G}_{\mathcal{A}}\mathbf{P} = x_1\mathbf{G}_{\mathcal{A}}^{pub}$, which is the shared common information between Alice and Bob.

3.2 OWF, IOWF, ITOWF based on polar code matrix

We now describe one wayness of various functions above based on polar code matrix.

1. Public private key pair function

$$\begin{aligned} \mathbf{G}_{\mathcal{A}}^{pub} &= \mathbf{S}_1\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ \mathbf{G}_{\mathcal{A}^c}^{pub} &= \mathbf{S}_2\mathbf{G}_{\mathcal{A}^c}\mathbf{P} \end{aligned}$$

2. Encryption function

$$c = x_1\mathbf{G}_{\mathcal{A}}^{pub} + x_2\mathbf{G}_{\mathcal{A}^c}^{pub}$$

3. Decryption function Computing $x_1\mathbf{G}_{\mathcal{A}}^{pub}$, $x_2\mathbf{G}_{\mathcal{A}}^{pub}$

$$\begin{aligned} x_1\mathbf{G}_{\mathcal{A}}^{pub} &= c\mathbf{P}^T\mathbf{H_A}\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ x_2\mathbf{G}_{\mathcal{A}^c}^{pub} &= c\mathbf{P}^T\mathbf{H}_{\mathbf{A}^c}\mathbf{G}_{\mathcal{A}^c}\mathbf{P} \end{aligned}$$

$\mathbf{G}_{\mathcal{A}}^{pub}$ is of size $K \times N$ and rank r_1 and $\mathbf{G}_{\mathcal{A}^c}^{pub}$ is of size $(N-K) \times N$ and rank r_2. $\mathbf{S}_1$ of rank r_1 and of size $K \times K$. $\mathbf{S}_2$ of rank r_2 and of size $(N-K) \times (N-K)$ respectively. $\mathbf{G}_{\mathcal{A}}$ and $\mathbf{G}_{\mathcal{A}^c}$ are matrices of size $K \times N$ and $(N-K) \times N$ and P is a permutation matrix of size $N \times N$.

3.2.1 Encryption function is IOWF

Consider $x = \{x_1, x_2\} \in \mathcal{X}$ is a N-bit random input vector of which x_1, x_2 are K-bit and $N-K$-bit vectors respectively. $u = \{\mathbf{G}_1, \mathbf{G}_2\}$ where $\mathbf{G}_1$ is of rank r_1 and of size $K \times N$ and $\mathbf{G}_2$ is of rank r_2 and of size $N - K \times N$.

Theorem 3.5 *The encryption function $F : \mathcal{U} \times \mathcal{X} \to \mathcal{Y}$, defined as*

$$y = F(u, x) \triangleq x_1\mathbf{G}_1 + x_2\mathbf{G}_2$$

where $y \in \mathcal{Y}$ is a N-bit output vector, is an IOWF of order $2^{N-r_1-r_2}$

Proof. For the function F to be an IOWF, we need to show that given $y \in \mathcal{Y}$, computing for $x \in \mathcal{X}$ results in a set $S_y \in \mathcal{X} \times \mathcal{E}$ of order n all of which are equiprobable. So, given y and public keys $\mathbf{G}_1$ and $\mathbf{G}_2$

$$y = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \mathbf{G}^{pub}$$

The rank of matrix $\mathbf{G}_1$ is r_1 and the rank of matrix $\mathbf{G}_2$ is r_2. Hence rank $\mathbf{G}^{pub} = r_1 + r_2$. Hence the underdetermined system of linear equations above has $N - r_1 - r_2$ free variables. So the number of solutions of $x = \{x_1, x_2\} \in \mathcal{X}$ is $2^{N-r_1-r_2}$. So the encryption function $F(u, x)$ is IOWF of order $2^{N-r_1-r_2}$. □

3.2.2 Decryption function is ITOWF

Consider a trapdoor $t = \{\mathbb{1}_{\mathcal{A}}, \mathbf{S}_1, \mathbf{S}_2, \mathbf{P}\}$ and we have

Theorem 3.6 *The decryption function $F : \mathcal{T} \times \mathcal{Y} \to \mathcal{M}$, defined as,*

$$\begin{aligned} F(r, y) &\triangleq m = \{m_1, m_2\} \\ m_1 &= x_1\mathbf{G}_{\mathcal{A}}^{pub} = y\mathbf{P}^T\mathbf{H}_{\mathbf{A}}\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ m_2 &= x_2\mathbf{G}_{\mathcal{A}^c}^{pub} = y\mathbf{P}^T\mathbf{H}_{\mathbf{A}^c}\mathbf{G}_{\mathcal{A}^c}\mathbf{P} \end{aligned}$$

$m = \{m_1, m_2\}$ denotes a string of which m_1 and m_2 are symbols of size N, is a ITOWF of order $2^{N-r_1-r_2}$ with private key $\mathcal{T}$ as trapdoor.

Proof. For the function F to be an ITOWF of order n, we need to show

1. it is an IOWF of order n for each private key parameters t
2. given y and x, computation of t is not known to be feasible.

For the function F to be IOWF, we need to show that, given m, computing y results in the set $S_m \in \mathcal{Y}$ of order n. For arbitrarily chosen $\mathcal{A}, \mathbf{P}$ we obtain

$$\begin{aligned} m &= y\mathbf{P}^T\mathbf{H}_{\mathcal{A}}\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ &= y\mathbf{R} \end{aligned}$$

where $\mathbf{R} = \mathbf{P}^T\mathbf{H}_{\mathcal{A}}\mathbf{G}_{\mathcal{A}}\mathbf{P}$. The rank of $\mathbf{P}$, $\mathbf{H}_{\mathcal{A}}$ and $\mathbf{G}_{\mathcal{A}}$ are N, K and K and are all part of private key. For any chosen $\mathbf{R}$, of rank less than K, is product of these matrices. We can see from the above equation this is of the form $\mathbf{A}x = b$, where $\mathbf{A}$ is not of full rank. Hence the least number of free variables in above equation is $N-(r_1+r_2)$ and number of solutions for y are at least $2^{N-(r_1+r_2)}$. Hence, for any chosen R, the function F is a IOWF of order $2^{N-(r_1+r_2)}$.

For the second part

$$\begin{aligned} m &= y\mathbf{P}^T\mathbf{H}_{\mathcal{A}}\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ &= y\mathbf{R} \end{aligned}$$

To solve for the private keys $\mathcal{T}$ from the above equation requires solving a quadratic system of equations. This is an expected hard problem when the number of variables is sufficiently. There is no known polynomial time algorithm for solving such systems. Hence the the function F is an ITOWF of order $2^{N-(r_1+r_2)}$. □

3.2.3 Public private key pair function is OWF

Consider $t=\{\mathbb{1}_{\mathcal{A}},\mathbf{S}_1,\mathbf{S}_2,\mathbf{P}\}$. $\mathbf{S}_1$ is of rank r_1 and of size $K\times K$. $\mathbf{S}_2$ is of rank r_2 and of size $(N-K)\times N$. $\mathbf{G}_{\mathcal{A}}$, $\mathbf{G}_{\mathcal{A}^c}$ are matrices of size $K\times N$ and $(N-K)\times(N-K)$ respectively, as defined Section 3 on page 5. $\mathbf{P}$ is a random permutation matrix of size $N\times N$. We use this notation to propose the below theorem

Theorem 3.7 *The function $G:\mathcal{T}\to\mathcal{U}$, $G(t)=u$, defined as*

$$\begin{aligned} G(t) &\triangleq u=\{\mathbf{G}_1^{pub},\mathbf{G}_2^{pub}\} \\ \mathbf{G}_1^{pub} &= \mathbf{S}_1\mathbf{G}_{\mathcal{A}}\mathbf{P} \\ \mathbf{G}_2^{pub} &= \mathbf{S}_2\mathbf{G}_{\mathcal{A}^c}\mathbf{P} \end{aligned}$$

where $u=\{\mathbf{G}_1^{pub},\mathbf{G}_2^{pub}\}$. $\mathbf{G}_1^{pub}$, $\mathbf{G}_2^{pub}$ are matrices of size $K\times N$ and $(N-K)\times(N-K)$ respectively, is a OWF.

Proof. To prove that it is OWF, given $\mathbf{G}_1^{pub}$, we need to show that computation of $t=\{\mathcal{A},\mathbf{P}\}$ is not known to be feasible. Consider the following equation

$$\begin{bmatrix}\mathbf{G}_1^{pub}\\ \mathbf{G}_2^{pub}\end{bmatrix}=\begin{bmatrix}\mathbf{S}_1 & 0\\ 0 & \mathbf{S}_2\end{bmatrix}\mathbf{Q}\mathbf{F}^{\otimes n}\mathbf{P}$$

this is a quadratic system of equations with additional constraint of $\mathbf{Q}$, $\mathbf{P}$ being permutation for solving such systems. This is an expected hard problem for sufficiently large number of variables and no general polynomial time algorithm is known for solving suhc systems. □

3.3 Example

Let $GF(p)$ be the finite field where all the computations are performed, where $p=2$. Let

$$\mathbf{S}_1=\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&1&1&1\\1&0&1&0\end{bmatrix}$$

and

$$\mathbf{S}_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

with dimension of kernel of $\mathbf{S}_2, \mathbf{S}_1$ being 2 each. Let

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

be a randomly generated permutation matrix. $\mathcal{A} = \{2, 4, 6, 7\}$ be the indices for the matrix $\mathbf{G}$.

$$\mathbf{G}_{\mathcal{A}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and

$$\mathbf{G}_{\mathcal{A}^c} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

.

1. Private Key

$$\mathcal{T}^{priv} = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{P}, \mathcal{A}\}$$

2. Public Key

$$\mathcal{T}^{pub} = \big\{\mathbf{G}^{pub}_{\mathcal{A}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix},$$

$$\mathbf{G}^{pub}_{\mathcal{A}^c} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}\}$$

3. Input $x = \{x_1, x_1\}$ randomly generated vectors

$$x_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}, x_2 = \begin{bmatrix} 1 & 1 & 0 & 1 \end{bmatrix}$$

4. Encryption Eq. (11) on page 8

$$c = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

5. Decryption Eq. (12) on page 8

$$\tilde{c} = c\mathbf{P}^T\mathbf{H}_\mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}$$

6. we multiply by $\tilde{c}\mathbf{G}_\mathcal{A}\mathbf{P}$ to obtain the shared information between the users

$$x_1\mathbf{G}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

3.4 Cryptanalysis

For any cryptosystem, cryptanalysis is carried out to test its security against standard attacks.

1. Brute force attack: Searching key space for private key with the known public key as input.

2. Chosen cipher text attack: If the decryption of chosen ciphertext is available to the attacker, the attacker tries to find the private key used in the decryption.

3.4.1 Brute force attack

In the brute force attack the attacker has $\mathcal{K}^{priv}$ to retrieve the private keys. In the proposed cryptosystem the private key is $\mathcal{T}^{priv} = \{\mathbb{1}_\mathcal{A}, \mathbf{S}_1, \mathbf{S}_2, \mathbf{P}\}$. By stirling's approximation, for $N, K > 1$, the first part of the private key $\mathbb{1}_\mathcal{A}$ has

$$\binom{N}{K} \approx 2^{\{N*H(K/N)\}}$$

where

$$H\left(\frac{K}{N}\right) \triangleq \frac{K}{N}\log\frac{N}{K} + \left(1 - \frac{K}{N}\right)\log\frac{1}{(1-\frac{K}{N})}$$

is the binary entropy function. For the permutation matrix $\mathbf{P}$ of size $N \times N$ yields $N! \approx \sqrt{2\pi N}\left(\frac{N}{e}\right)^N$ possible combinations. For a singular matrix of rank r and size $K \times K$ they are $\prod_{i=0}^{r-1}(2^K - 2^i)2^r(K-r)$ combinations. As a result, search of the secret key is not feasible in polynomial time.

3.4.2 Chosen ciphertext attack (CCA)

In CCA, the adversary uses the knowledge of decryption function and provides chosen ciphertexts to be deciphered to get plaintexts and use the ciphertext and plaintext pair to retrieve information about the secret key or future plaintext. From decryption function Eq. (12) on page 8 we get

$$c\mathbf{P}^T\mathbf{H}_\mathcal{A}\mathbf{G}_\mathcal{A}\mathbf{P} = x_1$$

So, the adversary can deduce equations in the parameters of P and indices $\mathcal{A}$ and $\mathcal{A}^c$. These result into a quadtaric system of equations in parameters of P and G. For suffciently large N and $N-K$ this is an expected hard problem.

3.5 Construction of the Scheme

The construction requires generation of the secret key parameters $\mathcal{T}^{priv} = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{P}, \mathcal{A}\}$. Let the the initial seed be denoted by $\mathcal{T}$ used to generate the secret key $\mathcal{T}^{priv}$.

The indices $\mathcal{A}$ and the permutation matrix $\mathbf{P}$ can be generated using the RC4 key expansion algorithm, using the initial seed $\mathcal{T}$. An RC4 key scheduling algorithm is is a permutation where in it permutes the indices of the N size array. For generation of a singular matrix $\mathbf{S}$ of size $m \times m$ and rank r, we have the following steps The algorithm 1 can be used generate matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ of sizes

1. Consider identity matrix $\mathbf{I}_r$ of size $r \times r$
2. Construct a matrix $\mathbf{L} = \begin{bmatrix} \mathbf{I}_r & \mathbf{0}_{r\times m-r} \\ \mathbf{0}_{m-r\times r} & \mathbf{0}_{m-r\times m-r} \end{bmatrix}$;
3. Generate two random permutations $\mathbf{Q}_1$, $\mathbf{Q}_2$ of size $m \times m$ using the initial seed $\mathcal{T}$ as an input to RC4 key expansion algorithm
4. Matrix $\mathbf{S} \triangleq \mathbf{Q}_1\mathbf{L}\mathbf{Q}_2$

Algorithm 1: Steps to generate matrix $\mathbf{S}$ of rank r

$K \times K$ and $(N-K) \times (N-K)$ of ranks r_1 and r_2 respectively. For example to achieve an ideal secrecy of order 2^{128}, we can choose $N = 512$, $r_1 = r_2 = 192$ and $K = 256$.

4 Public key primitive based on general matrices

This section extends the scheme of previous section based on polar code matrices to a public key primitive using matrices over finite fields. Let $\mathbb{F}$ be a finite field. Consider two matrices $\mathbf{V}_1 = [\alpha_1^T \dots \alpha_K^T]^T$, $\mathbf{V}_2 = [\alpha_{K+1}^T \cdots \alpha_N^T]^T$, where $\alpha_1, \dots, \alpha_N$ are linearly independent vectors in $\mathbb{F}^n$ and let there exist a matrix $\mathbf{V}_3$ such that $\mathbf{V}_2\mathbf{V}_3 = 0$ and $\mathbf{V}_1\mathbf{V}_3$ is non-singular. We shall show why such a matrix exists. Let $\mathbf{S}_1$, $\mathbf{S}_2$ be two matrices, with entries in $\mathbb{F}$, with rank of $\mathbf{S}_1$, $\mathbf{S}_2$ be r_1 and r_2 respectively, are two matrices to hide rows of $\mathbf{V}_1$, $\mathbf{V}_2$. Let $\mathbf{P}$ be some random permutation matrix of size $N \times N$

4.1 Private Key And Public Keys

Matrices $\mathbf{V}_1$, $\mathbf{V}_2$, $\mathbf{S}_1$, $\mathbf{S}_2$, $\mathbf{P}$, of sizes $K\times N$, $(N-K)\times N$, $K\times K$, $(N-K)\times(N-K)$, $N\times N$ respectively, form part of private key, with above following properties.

There exists a matrix $\mathbf{V}_3$ such that $\mathbf{V}_2\mathbf{V}_3 = 0$ and $\mathbf{V}_1\mathbf{V}_3$ is non-singular

$$\mathcal{T}^{priv} = \{\mathbf{V}_1, \mathbf{V}_2, \mathbf{S}_1, \mathbf{S}_2, \mathbf{P}\} \tag{13}$$

Let $\mathbf{G}_1^{pub} = \mathbf{S}_1\mathbf{V}_1\mathbf{P}$ and $\mathbf{G}_2^{pub} = \mathbf{S}_2\mathbf{V}_2\mathbf{P}$. Then the public key is

$$\mathcal{T}^{pub} = \{\mathbf{G}_1^{pub}, \mathbf{G}_2^{pub}\} \tag{14}$$

The existence of matrix $\mathbf{V}_3$ is given by

$$\mathbf{V}_3 = \mathbf{V}^{-1}\mathbf{E} \tag{15}$$

where $\mathbf{E} = \begin{bmatrix} \mathbf{I}_K \\ \mathbf{0}_{(N-K)\times N} \end{bmatrix}$.

4.2 Public Key Exchange Scheme

Let x_1, of length of K, x_2, of length of $(N-K)$, be two randomly chosen texts by the transmitter. For the users to share common information, given the public key of the receiver, $\mathcal{T}^{pub}$, the encryption function by transmitter, based on the IOWF described above, is as follows

$$c = x_1\mathbf{G}_1^{pub} + x_2\mathbf{G}_2^{pub} \tag{16}$$

After receiving c, the receiver computes/decrypts using the ITOWF function as described above, using the secret key, $\mathcal{T}^{priv}$, as following,

$$\begin{aligned} c\mathbf{P}^T\mathbf{V}_3 &= x_1\mathbf{S}_1\mathbf{V}_1\mathbf{P}\mathbf{P}^T\mathbf{V}_3 + x_2\mathbf{S}_2\mathbf{V}_2\mathbf{P}\mathbf{P}^T\mathbf{V}_3 \\ &= x_1\mathbf{S}_1\mathbf{V}_1\mathbf{V}_3 + \mathbf{0} \\ c\mathbf{P}^T\mathbf{V}_3 &= x_1\mathbf{S}_1\mathbf{V}_1\mathbf{V}_3 = x_1\mathbf{S}_1 \end{aligned} \tag{17}$$

$x_1\mathbf{S}_1$ is obtained and the receiver computes the vector $x_1\mathbf{S}_1\mathbf{V}_1\mathbf{P} = x_1\mathbf{G}_1^{pub}$, which is the common information between the receiver and transmitter. The above equations can rewritten as below

$$c = [x_1 \;\; x_2] \begin{bmatrix} \mathbf{S}_1\mathbf{V}_1\mathbf{P} \\ \mathbf{S}_2\mathbf{V}_2\mathbf{P} \end{bmatrix} = [x_1 \;\; x_2] \begin{bmatrix} \mathbf{G}_1^{pub} \\ \mathbf{G}_2^{pub} \end{bmatrix}$$

To make the inversion non-unique and result in multiple solutions of large set, the row spans of $\mathbf{G}_1^{pub}$ and $\mathbf{G}_2^{pub}$ to have an intersection of dimension d, which can be achieved using the singularity of $\mathbf{S}_1$, $\mathbf{S}_2$. The number of solutions of the equation is p^d for the field of $\mathbb{F}_p$. We show in the next subsection that this degree of ideal secrecy is achievable in the proofs of IOWFs. Then in the a later subsection discuss a realisation or construction of the public key primitive for specific d in terms of matrices.

4.3 Development Of OWF, IOWF, ITOWF

Similar to OWF for public private key pair, IOWF for encryption and ITOWF for decryption based on polar codes, we develop the same for generalised matrices.

1. Public private key pair function

$$\mathbf{G}_1^{pub} = \mathbf{S}_1\mathbf{V}_1\mathbf{P}$$
$$\mathbf{G}_2^{pub} = \mathbf{S}_2\mathbf{V}_2\mathbf{P}$$

2. Encryption function

$$c = x_1\mathbf{G}_1^{pub} + x_2\mathbf{G}_2^{pub}$$

3. Decryption function

$$x_1\mathbf{G}_1^{pub} = c\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P}$$

where c is N-bits vector, $x = \{x_1, x_2\}$ where x_1, x_2 are of K-bit and $(N-K)$-bit random vector respectively. $\mathbf{G}_1^{pub}$, $\mathbf{G}_2^{pub}$ are matrices of size $K \times N$ and $(N-K) \times N$ respectively. $\mathbf{S}_1$ of rank r_1 and of size $K \times K$. $\mathbf{S}_2$ of rank r_2 and of size $(N-K) \times (N-K)$ respectively. $\mathbf{V}_1$ and $\mathbf{V}_2$ are matrices of size $K \times N$ and $(N-K) \times N$ and P is a permutation matrix of size $N \times N$.

4.3.1 Encryption function is IOWF

Consider $x = \{x_1, x_2\} \in \mathcal{X}$ is a N symbols random input vector of which x_1, x_2 are K symbol and $(N-K)$ symbol vectors respectively. $u = \{\mathbf{G}_1, \mathbf{G}_2\}$ where $\mathbf{G}_1$ is of rank r_1 and of size $K \times N$ and $\mathbf{G}_2$ is of rank r_2 and of size $(N-K) \times N$.

Theorem 4.1 *The encryption function $F : \mathcal{U} \times \mathcal{X} \to \mathcal{Y}$, defined as*

$$y = F(u, x) \triangleq x_1\mathbf{G}_1 + x_2\mathbf{G}_2$$

where $y \in \mathcal{Y}$ is a N symbol output vector, is an IOWF of order $p^{N-r_1-r_2}$

Proof. For the function F to be an IOWF, we need to show that given $y \in \mathcal{Y}$, computing for $x \in \mathcal{X}$ results in a set $S_y \in \mathcal{X} \times \mathcal{E}$ of order n all of which are equiprobable. So, given y and public keys $\mathbf{G}_1$ and $\mathbf{G}_2$

$$y = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \mathbf{G}^{pub}$$

The rank of matrix $\mathbf{G}_1$ is r_1 and the rank of matrix $\mathbf{G}_2$ is r_2. Hence rank $\mathbf{G} \leq r_1 + r_2$. Hence the underdetermined system of linear equations above has at least $N - r_1 - r_2$ free variables. So the number of solutions of $x = \{x_1, x_2\} \in \mathcal{X}$ is at least $p^{N-r_1-r_2}$. So the encryption function $F(u, x)$ is IOWF of order $p^{N-r_1-r_2}$. □

4.3.2 Decryption function is ITOWF

Consider a trapdoor $t = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{V}_1, \mathbf{V}_2, \mathbf{P}\}$ where $\mathbf{S}_1$ is of rank r_1 and of size $K \times K$ and $\mathbf{S}_2$ is of rank r_2 and of size $(N-K) \times (N-K)$. $x = \{x_1,\ x_2\}$ denotes a random string of which x_1, x_2 are K symbols and $N-K$ symbols respectively. y is vector string of N symbols vector. $\mathbf{P}$ is a random permutation matrix of size $N \times N$. With this notation we have

Theorem 4.2 *The function $F : \mathcal{T} \times \mathcal{Y} \rightarrow \mathcal{M}$, defined as,*

$$F(t, y) \triangleq M = x\mathbf{G}_1 = y\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P}$$

where m denotes a vector string of N symbols, is a ITOWF of order p^{N-K} with the private key $\mathcal{T} = \mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P}$ as trapdoor.

Proof. For the function F to be an ITOWF of order n, we need to show the following

1. is an IOWF of order n for each private key parameters t
2. given y and x, computation of t is not known to be feasible.

For the function F to be IOWF, we need to show that, given m, computing y results in the set $S_m \in \mathcal{Y}$ of order n. For a arbitrarily chosen $\mathbf{V}_1, \mathbf{P}$ we get the following equations

$$m = y\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P} = y\mathbf{R}$$

The rank of $\mathbf{P}$, $\mathbf{V}_1$ and $\mathbf{V}_3$ are N, K and K and are all part of private key. For any chosen $\mathbf{R}$, of rank less than K, is product of these matrices. We can see from the above equation this is of the form $\mathbf{A}x = b$, where $\mathbf{A}$ is not of full rank. Hence the least number of free variables in above equation is $N-K$ and number of solutions for y are at least p^{N-K}. Hence, for any chosen $\mathbf{R}$, the function F is a IOWF of order p^{N-K}.

For the second part of the group, we form the equations as

$$m = y\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P} = y\mathbf{R}$$

This equation is a quadratic system in private key parameters along with constraint equations for P to be permutation and relations between V_1 and V_3. With sufficiently large N and the difference $N-K$ this is an expected hard problem as no known general algorithm is known for solving quadratic systems. Hence the function F is a ITOWF of order p^{N-K}. □

4.3.3 Public private key pair function is OWF

Consider $t = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{V}_1, \mathbf{V}_2, \mathbf{P}\}$. $\mathbf{S}_1$ is of rank r_1 and of size $K \times K$. $\mathbf{S}_2$ is of rank r_2 and of size $N-K \times N$. $\mathbf{V}_1$, $\mathbf{V}_2$ are matrices of size $K \times N$ and $N-K \times N$ respectively. $\mathbf{P}$ is a random permutation matrix of size $N \times N$. We use this notation to propose the below theorem

Theorem 4.3 *The function $G : \mathcal{T} \to \mathcal{U}$, defined as,*

$$\begin{aligned} G(t) &\triangleq u = \{\mathbf{G}_1^{pub}, \mathbf{G}_2^{pub}\} \\ \mathbf{G}_1^{pub} &= \mathbf{S}_1\mathbf{V}_1\mathbf{P} \\ \mathbf{G}_2^{pub} &= \mathbf{S}_2\mathbf{V}_2\mathbf{P} \end{aligned}$$

$\mathbf{G}_1^{pub}$, $\mathbf{G}_2^{pub}$ are matrices of size $K \times N$ and $N - K \times N - K$ respectively, is a OWF.

Proof. To prove that it is IOWF, given $\mathbf{G}_1^{pub}$, we need to show that computation of $r = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{V}_1, \mathbf{V}_2, \mathbf{P}\}$ is not known to be feasible. Consider the following equation

$$\begin{bmatrix} \mathbf{G}_1^{pub} \\ \mathbf{G}_2^{pub} \end{bmatrix} = \begin{bmatrix} \mathbf{S}_1 & 0 \\ 0 & \mathbf{S}_2 \end{bmatrix} \mathbf{Q}\mathbf{F}^{\otimes n}\mathbf{P}$$

this is a quadratic system of equations in parameters of the private key with a additional constraints of $\mathbf{Q}$, $\mathbf{P}$ being permutation matrices. With N and $N - K$ sufficiently large this is an expected hard problem and no general algorithm is known for solving quadratic systems. □

4.4 Example

Let $GF(p)$ be the finite field where all the computations are performed, where $p = 7$. Let

$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

and

$$\mathbf{S}_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

with rank of $\mathbf{S}_2 = 2$ and rank of $\mathbf{S}_1 = 3$. Let

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

be a randomly generated permutation matrix. We choose N linearly independent vectors in $\mathbf{F}^N$ to form the matrix $\mathbf{V}$, a full rank matrix, which is partitioned as $\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix}$. The matrix $\mathbf{V}_1$ is of size $K \times N$ and $\mathbf{V}_2$ is of size $(N-K) \times N$.

$$\mathbf{V}_1 = \begin{bmatrix} 2 & 2 & 2 & 3 & 5 & 4 & 3 & 5 \\ 5 & 1 & 2 & 3 & 1 & 1 & 4 & 1 \\ 1 & 3 & 5 & 5 & 1 & 3 & 3 & 2 \\ 1 & 1 & 6 & 3 & 3 & 4 & 5 & 1 \end{bmatrix}$$

and

$$\mathbf{V}_2 = \begin{bmatrix} 0 & 0 & 6 & 3 & 1 & 1 & 3 & 5 \\ 5 & 2 & 1 & 0 & 3 & 6 & 6 & 5 \\ 4 & 6 & 5 & 1 & 1 & 3 & 2 & 0 \\ 4 & 2 & 0 & 2 & 3 & 2 & 0 & 4 \end{bmatrix}$$

Then there exists a matrix $\mathbf{V}_3$ satisfying $\mathbf{V}_1\mathbf{V}_3 = \mathbf{I}_K$ and $\mathbf{V}_2\mathbf{V}_3 = \mathbf{0}_{(N-K)\times N}$. Then the existence of matrix $\mathbf{V}_3$ is given by $\mathbf{V}_3 = \mathbf{V}^{-1}\mathbf{E}$, where $\mathbf{E} = \begin{bmatrix} \mathbf{I}_K \\ \mathbf{0}_{(N-K)\times N} \end{bmatrix}$. Using matrix $\mathbf{V}$ and above conditions we compute the matrix $\mathbf{V}_3$

$$\mathbf{V}_3 = \begin{bmatrix} 0 & 5 & 2 & 5 \\ 2 & 2 & 0 & 4 \\ 5 & 6 & 1 & 0 \\ 0 & 4 & 5 & 0 \\ 6 & 3 & 2 & 1 \\ 0 & 4 & 2 & 2 \\ 3 & 5 & 1 & 6 \\ 5 & 0 & 0 & 0 \end{bmatrix}$$

1. Private Key
$$\mathcal{T}^{priv} = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{P}, \mathbf{V}_1, \mathbf{V}_2\}$$

2. Public Key

$$\mathcal{T}^{pub} = \{\mathbf{G}_1^{pub} = \begin{bmatrix} 6 & 1 & 6 & 0 & 3 & 5 & 0 & 0 \\ 4 & 6 & 2 & 5 & 6 & 2 & 2 & 1 \\ 3 & 0 & 1 & 5 & 2 & 0 & 2 & 1 \\ 6 & 1 & 6 & 0 & 3 & 5 & 0 & 0 \end{bmatrix},$$

$$\mathbf{G}_2^{pub} = \begin{bmatrix} 4 & 5 & 3 & 3 & 4 & 2 & 2 & 6 \\ 4 & 3 & 2 & 5 & 1 & 1 & 4 & 5 \\ 2 & 4 & 5 & 4 & 4 & 6 & 5 & 4 \\ 4 & 5 & 3 & 3 & 4 & 2 & 2 & 6 \end{bmatrix}\}$$

3. Input $x = \{x_1, x_2\}$ randomly generated vectors

$$x_1 = \begin{bmatrix} 2 & 1 & 5 & 6 \end{bmatrix}, x_2 = \begin{bmatrix} 3 & 6 & 1 & 4 \end{bmatrix}$$

4. Encryption Eq. (16) on page 14

$$c = \begin{bmatrix} 2 & 1 & 2 & 1 & 1 & 5 & 6 & 5 \end{bmatrix}$$

5. Decryption Eq. (17) on page 14,

$$\tilde{c} = c\mathbf{P}^T\mathbf{V}_3 = \begin{bmatrix} 6 & 6 & 6 & 6 \end{bmatrix}$$

6. we multiply by $\tilde{c}\mathbf{V}_1\mathbf{P}$ to obtain the shared information between the users

$$\tilde{c}\mathbf{V}_1\mathbf{P} = x_1\mathbf{G}_1 = \begin{bmatrix} 4 & 0 & 6 & 2 & 5 & 0 & 5 & 6 \end{bmatrix}$$

Let the matrix $\mathbf{G} = \begin{bmatrix} \mathbf{G}_1^{pub} \\ \mathbf{G}_2^{pub} \end{bmatrix}$

$$\mathbf{G} = \begin{bmatrix} 6 & 1 & 6 & 0 & 3 & 5 & 0 & 0 \\ 4 & 6 & 2 & 5 & 6 & 2 & 2 & 1 \\ 3 & 0 & 1 & 5 & 2 & 0 & 2 & 1 \\ 6 & 1 & 6 & 0 & 3 & 5 & 0 & 0 \\ 4 & 5 & 3 & 3 & 4 & 2 & 2 & 6 \\ 4 & 3 & 2 & 5 & 1 & 1 & 4 & 5 \\ 2 & 4 & 5 & 4 & 4 & 6 & 5 & 4 \\ 4 & 5 & 3 & 3 & 4 & 2 & 2 & 6 \end{bmatrix}$$

The echelon form matrix $\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 3 & 4 & 3 \\ 0 & 1 & 0 & 0 & 0 & 0 & 2 & 4 \\ 0 & 0 & 1 & 0 & 0 & 1 & 3 & 4 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 3 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$. The above echelon form shows the $\mathbf{G}_1^{pub}$ and $\mathbf{G}_2^{pub}$ have an intersection of $d = 3$ resulting in 7^3 solutions to solve the ciphertext plaintext equation.

4.5 Cryptanalysis

We again consider the two attacks as discussed above.

4.5.1 Brute force attack

In the brute force attack the adversary performs an exhaustive search of the key space, $\mathcal{T}^{priv}$, to retrieve the keys. For the attack to be unsuccessful, the key space has to be sufficiently large. In the proposed cryptosystem the private key $\mathcal{T}^{priv} = \{\mathbf{V}_1, \mathbf{V}_2, \mathbf{S}_1, \mathbf{S}_2, \mathbf{P}\}$. For the permutation matrix $\mathbf{P}$ of size $N \times N$ yields $N! \approx \sqrt{2\pi N}\left(\frac{N}{e}\right)^N$ possible combinations. For a singular matrices of rank r and size $K \times K$ they are $\prod_{i=0}^{r-1}(p^K - p^i)p^r(K - r)$ combinations. Since $\mathbf{V}_1, \mathbf{V}_2$ form N linearly independent vectors they are $\prod_{i=0}^{N-1}(p^N - p^i)$. As a result, search of the secret key is not feasible in polynomial time.

4.5.2 Chosen ciphertext attack (CCA)

Here the adversary can provide ciphertexts c for decryption. With many such chosen ciphertexts, from decryption function Eq. (17) on page 14 we get

$$c\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P} = x_1$$

So, the adversary can get equations in parameters of the private key

$$\mathbf{P}^T\mathbf{V}_3\mathbf{V}_1\mathbf{P}$$

However again these can be organised as a system of quadratic equations in private key parameters which is an expected hard problem with sufficiently large N and $N - K$. Hence the scheme is secure against CCA.

4.6 Construction of the Scheme

The construction requires generation of the secret key parameters $\mathcal{T}^{priv} = \{\mathbf{S}_1, \mathbf{S}_2, \mathbf{V}_1, \mathbf{V}_2, \mathbf{P}\}$ and show existence of $\mathbf{V}_3$. Let the initial seed be denoted by $\mathcal{T}$ used to generate the secret key $\mathcal{T}^{priv}$. Refer to Eq. (15) on page 14 for construction of matrices $\mathbf{V}_1$, $\mathbf{V}_2$ and $\mathbf{V}_3$. The matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ can be constructed using the algorithm 1 in Section 3.5 on page 13. For a choice of N, K and d the $rank(\mathbf{S}_1) = r_1$ is the floor of $(N - d)/2$ and the $rank(\mathbf{S}_2) = r_2$ is $N - d - r_1$. For example to achieve an ideal secrecy of order $2^{128} \approx 5^{55}$, we can choose $N = 128$, $r_1 = 36$ and $r_2 = 37$ and $K = 64$

5 Conclusion

A key exchange scheme over a public channel is proposed using the polar code matrix and also its extension over matrices over finite fields. The schemes are shown to be secured in terms of the notions of ideal secrecy as well as hardness of computation of a large system of quadratic equations. The encryption and decryption are computationally efficient. Ideal secrecy based public key schemes are not previously known. Due to simplicity of computation in encryption and decryption involving only linear algebra the scheme is believed to be practically useful for key exchange as well as small scale encryption. Applications of the scheme for construction of signatures may involve new problems which shall be explored in future work.

References

[1] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[2] Michael O Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR COMPUTER SCIENCE, 1979.

[3] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4):469–472, 1985.

[4] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, pages 223–238, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[5] Robert J McEliece. A public-key cryptosystem based on algebraic coding theory. *DSN progress report*, 42(44):114–116, 1978.

[6] Jeff Hoffstein, Daniel Lieman, Jill Pipher, and Joseph H Silverman. Ntru: A public key cryptosystem. *Submissions and Contributions to IEEE P*, 1363, 1999.

[7] Claude E Shannon. Communication theory of secrecy systems*. *Bell system technical journal*, 28(4):656–715, 1949.

[8] Aaron D Wyner. The wire-tap channel. *Bell System Technical Journal, The*, 54(8):1355–1387, 1975.

[9] Hessam Mahdavifar and Alexander Vardy. Achieving the secrecy capacity of wiretap channels using polar codes. *IEEE Transactions on Information Theory*, pages 6428–6443, 2011.

[10] Ph R Geffe. Secrecy systems approximating perfect and ideal secrecy. *Proceedings of the IEEE*, 53(9):1229–1230, 1965.

[11] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654, 1976.

[12] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography.* CRC press, 2005.

[13] Ian F Blake, Gadiel Seroussi, Nigel P Smart, et al. *Advances in elliptic curve cryptography*, volume 317. Cambridge University Press, 2005.

[14] Erdal Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *CoRR*, pages –1–1, 2008.

[15] R.A. Horn, R.A. Horn, and C.R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, 1994.

[16] Marc P. C. Fossorier. Polar codes: Graph representation and duality. *CoRR*, abs/1312.0372, 2013.

[17] Harish Vangala, Yi Hong, and Emanuele Viterbo. Efficient algorithms for systematic polar encoding. *IEEE communications letters*, 20(1):17–20, 2016.