Efficient reconciliation protocol with polar codes for quantum key distribution

Sunghoon Lee
School of Electrical Engineering
Korea University
Seoul, Korea
sunghoon89@korea.ac.kr

Jun Heo
School of Electrical Engineering
Korea University
Seoul, Korea
junheo@korea.ac.kr

***Abstract*—** Information reconciliation is a critical process of any Quantum Key Distribution (QKD) protocol, where two legitimate parties agree on a secret key. Two parties attempt to eliminate the discrepancies between their correlated keys in the presence of an adversary, while revealing a minimum amount of information. In this paper, we propose a reconciliation protocol using polar codes with a soft-output decoder. We demonstrate that our method is better than a conventional protocol using polar codes in the view of reconciliation efficiency.

Keywords—quantum key distribution; information reconciliation ; polar codes; soft-output decoder

I. INTRODUCTION

Quantum cryptography or more precisely quantum key distribution (QKD) is a new technology which gets a high level of attention today worldwide. The idea of QKD is first proposed by C.H. Bennett, G. Brassard in 1984[1].

QKD is a secure key generation method between two distant parties by wisely exploiting properties of quantum mechanics. In a QKD protocol, two legitimate parties, Alice and Bob, aim at sharing an information-theoretic secret key, even in the presence of an eavesdropper, Eve. In the quantum part of such a protocol, Alice (sender) and Bob (receiver) exchange quantum signals, e.g. single photons, which carry classical information, 0 or 1. After repeating this step several times, Alice and Bob share two n-bit strings, X and Y. Eve has access to a random variable Z, possibly correlated to X and Y.

In any realistic implementation of a QKD protocol, X and Y suffer discrepancies mainly due to losses in the channel and noise in Bob's detectors but which are conservatively attributed to the action of an eavesdropper. Therefore, any QKD protocol must include a classical post-processing step in order to extract a secret key form the correlated strings X and Y. After the post-processing step, Alice and Bob agree on a residual key string.

The paper is organized as follows: in section II, information reconciliation is detailed and the previous work is reviewed. In section III, the use of polar codes to reconciliation in QKD is laid out. Finally, the performance of proposed scheme is demonstrated in section IV.

II. INFORMATION RECONCILIATION WITH LINEAR CODES

A. Previous Work

Information reconciliation is one part of a classical post-processing. It is the process of finding and correcting errors through public discussion [2]. Alice sends Bob additional information that allows him to generate an error-corrected key that is identical to Alice's. This information may be parity bits or syndrome bits. Indeed, information reconciliation is the counterpart of error correction is classical communication. However, there is one important difference: rather than combining information that allows correcting errors directly into the message to be transmitted, additional information is sent after key sifting is complete as it is only at this point that the message to be corrected in known.

The first reconciliation scheme was proposed in [2]. It was a simple scheme using binary search algorithm. Afterwards, CASCADE was proposed in [3], which is very elegant and compact. However, CASCADE has a high interactivity problem, several protocols have been developed that uses linear block codes. A protocol using Hamming codes, called WINNOW, was introduced in 2003 [4]. A reconciliation protocol using low-density parity-check (LDPC) codes was proposed in [5-6]. The use of polar codes in reconciliation was proposed in [7].

The use of polar codes has been previously considered for many scenarios [12-14]. Polar codes have some characteristics that make them fit for QKD error correction. First, they are easily employed in a rateless setup because the rate of codes can be freely changed. Secondly, they enable one-way error correction, similarly to LDPC codes, and contrary to two-way protocols like CASCADE.

The first reconciliation protocol with polar codes was proposed in [15]. However, this reconciliation protocol employed a successive cancellation (SC) decoder. More specific details on polar codes in QKD has been introduced, and specifically designed polar codes for the QKD environment were suggested in [16]. However, the performance of polar codes under the SC decoder falls behind that of state-of-the-art codes.

B. Reconciliation Efficiency

Information reconciliation is a general term to demonstrate any method that can be used to extract shared secret keys [8]. In

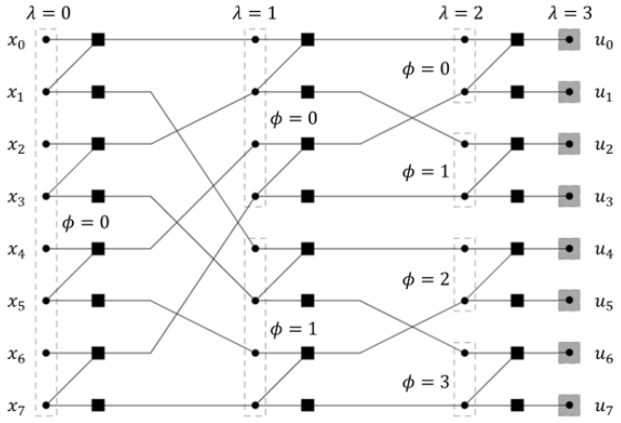

Fig. 1 Factor graph of a polar code with $\boldsymbol{n = 8}$

other words, *X* and *Y* are considered as the input and output of a modeled quantum channel, respectively.

It is usually presumed that Alice and Bob hold sifted keys **x** and **y**, two *n*-bit strings that are outcomes of *X* and *Y*. During reconciliation, Alice and Bob communicate a set of messages *M* to reconcile their keys. At the end of the protocol, they agree on a shared secret key in the presence of Eve while revealing some information. Therefore, *X* and *Y* can be considered correlated random variables, and **y** is given by transition probability $P_W\{\mathbf{x}|\mathbf{y}\}$. It is given as if every symbol was generated by memoryless channel *W*, equivalently. Errors occurred during QKD are commonly considered as uncorrelated and symmetric. Thus, the memoryless channel *W* can be seen as a binary symmetric channel (BSC). The crossover probability *p* of the BSC is generally supposed as given.

The problem is how a message *M* is encoded. This problem of reconciliation can be treated as well-known Slepian-Wolf coding, which is a way of encoding two lossless compressed correlated sources [9]. The Slepian-Wolf bound states that (X^n, Y^n) can be compressed into more than $nH(X|Y)$ bits without information loss. This is the minimum information required to agree on a secret key. In practice, codewords will be encoded with a rate $fH(X|Y)$, where f is the reconciliation efficiency. Therefore, efficiency is defined as

$$f = \frac{|M|}{nH(X|Y)} \tag{1}$$

where *M* is a message or codeword Alice sends through public channel and $|\cdot|$ is a cardinality.

Lower efficiency means fewer information leaks during reconciliation. More information remains in sifted keys, leading to a higher secret key rate. After all, a lower efficiency is directly related to the higher secret key rate, which is why efficiency is a performance indicator of reconciliation protocols.

III. Preliminaries

A. Polar Codes

The development of polar codes by Arikan [10] was a major advance in coding theory. Polar codes have been proven to achieve the capacity of symmetric binary-input discrete memoryless channels (BI-DMC) with an explicit construction. Moreover, polar codes have low encoding and decoding complexity, which makes them suitable for efficient hardware implementation [11].

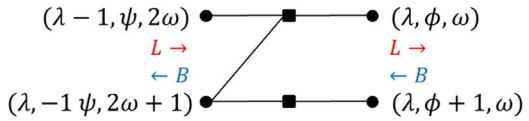

Fig. 2 Basic element of factor graph

With the Arikan's construction method [10], the generation matrix of a polar code is an $n \times n$ matrix $G = B_n F^{\otimes m}$, where $n = 2^m$, B_N is the bit reversal permutation matrix [10], and $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Here $\otimes m$ denotes the mth Kronecker power and $F^{\otimes m} = \mathrm{F} \otimes F^{(m-1)}$. Let $u_0^{n-1} = (u_0, u_1, \cdots, u_{n-1})$ denote uncoded data bits and $x_0^{n-1} = (x_0, x_1, \cdots, x_{n-1})$ encoded bits, then $x_o^{n-1} = u_o^{n-1} G$. An (N, K) polar code is defined by setting $N - K$ bits of u_0^{n-1} to zero, denoted as frozen bits. The information is transmitted by the rest K bits, denoted as information bits.

B. Soft Cancellation Decoding

The belief propagation (BP) decoding of polar codes over factor graph was proposed in [10]. Compared with their SC counterparts, polar BP decoders are easier to achieve low-latency. However, polar BP decoder need a large number of iterations and suffer from high computation complexity. On this account, a soft cancellation (SCAN) decoder is proposed in [17]. The SCAN decoding algorithm makes use of a serial message updating schedule, which is similar to the SC decoder of polar codes. By limiting the soft information propagation schedule in the decoding process, the computational complexity of SCAN decoders is much lower than that of polar BP decoders.

The SCAN decoding is conducted over the factor graph, which is derived from the encoding equations. The corresponding factor graph of a polar code with $n = 8$ is shown in Fig. 1. This factor graph shows relationship between uncoded and encoded bits. In graph, λ denotes layers and ϕ denotes groups. Basic element of factor graph is represented in Fig. 2. Here ω denotes the node number in group. Each node has two associated LLR messages $L_\lambda(\phi, \omega)$ and $B_\lambda(\phi, \omega)$ which are passed to the right and left directions of factor graph. These messages can be computed as follows

$$L_\lambda(\phi, \omega) =$$
$$L_{\lambda-1}(\psi, 2\omega) \boxplus [L_{\lambda-1}(\psi, 2\omega + 1) + B_\lambda(\phi + 1, \omega)] \tag{2}$$
$$L_\lambda(\phi + 1, \omega) =$$
$$L_{\lambda-1}(\psi, 2\omega + 1) + [L_{\lambda-1}(\psi, 2\omega) \boxplus B_\lambda(\phi, \omega)] \tag{3}$$
$$B_{\lambda-1}(\psi, 2\omega) =$$
$$B_\lambda(\phi, \omega) \boxplus [B_\lambda(\phi + 1, \omega) + L_{\lambda-1}(\psi, 2\omega + 1)] \tag{4}$$
$$B_{\lambda-1}(\psi, 2\omega + 1) =$$
$$B_\lambda(\phi + 1, \omega) + [B_\lambda(\phi, \omega) \boxplus L_{\lambda-1}(\psi, 2\omega)] \tag{5}$$

where $a \boxplus b$ is defined as

$$a \boxplus b = 2\tanh^{-1}\left[\tanh\left(\frac{a}{2}\right) \times \tanh\left(\frac{b}{2}\right)\right] \quad (6)$$

At iteration 1, $\{L_0(0,i)\}_{i=0}^{(N-1)}$ is set to LLRs from channel and $\{B_n(i,0)\} = 0$ if i is an index of information bits and $\{B_n(i,0)\} = 1$ if i is that of frozen bits. The other LLRs are set to all zero. After the LLRs of each node have been computed, the iteration 1 ends. From iteration 2, SCAN updates LLRs using previous iteration's LLRs. After all iteration, SCAN decoding decides $\hat{u}_i$ to 0 if $\left(B_n(i,0) + L_n(i,0)\right) \geq 0$. Otherwise, it decides $\hat{u}_i$ to 1.

IV. Polar Codes for QKD

A. Source Coding with Side Information

As demonstrated in section II, reconciliation is a problem of encoding two correlated sequences. From the view of QKD, Alice's string X is a source and Bob's string Y is a side information. Alice sends a message M to reconcile X and Y. It is exactly the same as source coding with side information. Alice's string X is a source and Bob's string Y is a side information, and a message M is a compressed version of X (see Fig. 3). In [19], Liveris proposed how to use LDPC codes to encoding two correlated sources.

In [18], Arikan studied source coding with polar codes, which is called as 'source polarization'. In this paper, we adapt a method of source polarization to reconciliation. Lossless source coding with side information using polar codes can be conducted with the analysis of source polarization.

In source polarization, bits indexed by a high-entropy (index) set $E_{X|Y}$ play the role of frozen bits. A high-entropy set indicates α sub-channel indices of the largest source Bhattacharyya parameters, where α denotes the size of the set. The source Bhattacharyya parameter is defined as

$$Z(X|Y) = 2\sum_{y} P_Y(y)\sqrt{P_{X|Y}(0|y)P_{X|Y}(1|y)} \quad (7)$$

There are differences between channel coding and reconciliation with polar codes. First, it is that frozen bits and a high-entropy set are not the same. It basically originates from channel and source coding therefore they are computed differently. Second, codewords are also different. In channel coding, redundancy is conveyed with information bits but, it is not the case in reconciliation. Additional information is transmitted after X and Y are obtained. Bits indexed by a high-entropy set cannot be placed in X before encoding. Thus, values of bits are not required.

We propose a reconciliation protocol using polar codes under the SCAN decoder. The process of the protocol is not much different from that of source polarization. It is also similar to reconciliation using LDPC codes.

B. Protocol

To encode an arbitrary binary sequence, Alice and Bob prepare (n, k) polar code where they share information about all source Bhattacharyya parameters of a code. They also agree with

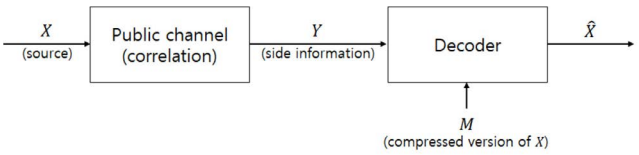

Fig. 3 Source coding with side information

the values of efficiency on each quantum bit error rate (QBER) in advance. Efficiency decides the size of a high-entropy set $E_{X|Y}$ and a code rate. For reconciliation of n-length sifted keys $\mathbf{x}$ and $\mathbf{y}$, Alice generates a codeword $\mathbf{u} = \mathbf{x}G$ where G is a generator matrix of polar codes. She sends $\mathbf{u}_{E_{X|Y}}$ which is a part of $\mathbf{u}$ and bits indexed by a high-entropy set through an error-free public channel.

Bob estimates the n-length sequence $\mathbf{x}$ from its $(n-k)$-length codeword $\mathbf{u}_{E_{X|Y}}$ and the correlated n-length sequence $\mathbf{y}$. Bob calculates all LLRs using (2) - (5) and SCAN scheduling. After all iterations are terminated, Bob builds an estimate $\hat{\mathbf{u}}$ of $\mathbf{u}$ by the rule

$$\hat{u}_i = \begin{cases} u_i & \text{if } i \in E_{X|Y} \\ 0 & \text{if } i \in E_{X|Y}^c \text{ and } \left(B_n(i,0) + L_n(i,0)\right) \geq 0 \\ 1 & else \end{cases}$$

The leaked information of the proposed protocol is information about a codeword $\mathbf{u}_{E_{X|Y}}$ which has $(n-k)$-length. We can easily measure a quantity of transmitted message M as $n-k$, thus efficiency can be calculated as

$$f = \frac{n-k}{nH(X|Y)} = \frac{1-R}{H(X|Y)} \quad (8)$$

where R is a code rate. Conditional entropy $H(X|Y)$ is a binary entropy of a crossover probability p of BSC.

V. Experimental Results

In this section, we discuss the efficiency of the proposed protocol by comparing the results of protocols using polar codes under the SC decoder for block length of 2048. We have implemented polar codes under the SC decoder described in [16]. Through these simulations, we showed that the proposed protocol experiences smaller leakage for several QBER over the BSC.

As explained in Section II, the performance of a reconciliation protocol can be evaluated by measuring the amount of information leaked in this process. For the BSC with a crossover probability p, an ideal reconciliation protocol would disclose $n \cdot h(p)$ bits where $h(\cdot)$ is the binary entropy. However, a real protocol cannot avoid revealing more fraction which is measured by reconciliation efficiency f.

Fig. 4 shows the efficiency, defined in (1), of the proposed protocol as well as polar codes under the SC decoder. In each simulation, the efficiency was divided by a unit of 0.01, and the simulation points that most approximated the frame error rate

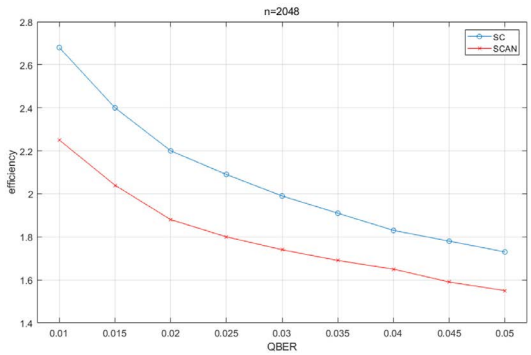

Fig. 4 Performance of the SCAN and SC decoders with block size of 2048

(FER) of 10^{-3} were selected. The efficiency points selected are shown in curves revealing FER as a function of efficiency for two block sizes in the QBER range $[0.010, 0.050]$.

Code construction of polar codes in each QBER was performed using the Monte-Carlo method [10]. Codes are constructed differently for each QBER to obtain better simulations. However, a single code designed for specific QBER can be used for multiple QBERs. We calculated and sorted the source Bhattacharyya parameter of each symbol. The size of the high-entropy set was computed as $\lceil f \cdot n \cdot h(p) \rceil$. The SCAN decoding was conducted described in [17]. The SCAN decoder is restricted to the maximum of 8 iterations.

As can be seen on the figure, the SCAN decoder can outperform the SC decoder in all QBER ranges. With maximum 8 iterations, the efficiency is always more than 11% lower. This gain of efficiency can effect final key rate of practical QKD.

VI. Conclusion

We have shown that SCAN decoding of polar codes can be used to reconcile two correlated discrete random variables. The results show that the SCAN decoders are a good alternative to the SC decoders. In terms of reconciliation efficiency, they show a similar behavior but a huge improvement for all QBER.

Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2018-2015-0-00385) supervised by the IITP(Institute for Information & communications Technology Promotion).

References

[1] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," in International Conference on Computers, Systems and Signal Processing, pp. 175–179, Dec 1984.

[2] C.H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin, " Experimental Quantum Cryptography", J. Cryptology, Vol. 5, No. 1, pp. 3-28, 1992.

[3] G. Brassard and L. Salvail, "Secret-Key Reconciliation by Public Discussion", Eurocrpyt'93, Workshop on the theory and application of cryptographic techniques on Advances in cryptology, pp. 410-23, 1994.

[4] W.T. Buttler, S.K. Lamoreaux, J.R. Torgerson, G.H. Nickel, C.H.Donahue, and C.G. Peterson, " Fast, efficient error reconciliation for quantum cryptography", Phys. Rev. A, 67, 052303, 2003.

[5] D. Elkouss, A. Leverrier, R. Alleaume, and J. Boutros, "Efficient reconciliation protocol for discrete-variable quantum key distribution", IEEE ISIT, pp. 145-149, 2009.

[6] D. Elkouss, J. Martinez-Mateo, and V. Martin, "Information Reconciliation for Qunatum Key Distribution", Quantum Inform. Comput., Vol. 11, No. 3, pp. 226-238, 2010.

[7] P. Jouguet and S. Kunz-Jacques, "High Performance Error Correction for Quantum Key Distribution using Polar Codes", Quantum Inform. Comput., Vol. 14, No. 3-4, pp. 329-338, 2013.

[8] G. Van Assche, "Quantum Cryptography and Secret-Key Distillation", Cambridge University Press., 2006.

[9] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources", IEEE Trans. Inf. Theory, Vol. 19, No. 4, pp. 471-480, 1973.

[10] E. Arikan, "Channel Polarization : A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels", IEEE Trans. Inf. Theory, Vol. 55, No. 7, pp. 3051-3073, 2009.

[11] C. Leroux, A. J. Raymond, G. Sarkis, I. Tal, A. Vardy, and W. J. Gross, "Hardware implementation of successive-cancellation decoders for polar codes", J. Signal Process. Syst., Vol. 69, No. 3, 305-315, 2002.

[12] H. Mahdavifar and A. Vardy, "Achieving the Secrecy Capacity of Wiretap Channels Using Polar Codes", IEEE Trans. Inf. Theory, pp. 642-6443, 2011.

[13] M.M. Wilde and S. Guha, "Polar codes for classical-quantum channels", IEEE Trans. Inf. Theory, Vol. 59, No. 2, pp. 1175-1187, 2013.

[14] S. Guha and M.M. Wilde, "Polar coding to achieve the Holevo capacity of a pure-loss optical channel", IEEE ISIT, pp. 546-550, 2012.

[15] P. Jouguet and S. Kunz-Jacques, "High Performance Error Correction for Quantum Key Distribution using Polar Codes", Quantum Inform. Comput., Vol. 14, No. 3-4, pp. 329-338, 2013.

[16] A. Nakassis and A. Mink, "Polar codes in a QKD environment", Quantum Information and Computation XII, 2014.

[17] U. U. Fayyaz and J. R. Barry, "Low-complexity soft-output decoding of polar codes," IEEE J. Selected Areas Commun., Vol. 32, No. 5, pp. 958–966, 2014

[18] E. Arikan, "Source Polarization", ISIT, 2010.

[19] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of Binary Source With Side Information at the Decdoer Using LDPC Codes", IEEE Commun. Lett. Vol. 6, No. 10, 2002