

High-Agreement Uncorrelated Secret Key Generation Based on Principal Component Analysis Preprocessing

Guyue Li, Aiqun Hu, Junqing Zhang, Linning Peng, Chen Sun, and Daming Cao

Abstract—Random and high-agreement secret key generation from noisy wideband channels is challenging due to the autocorrelation inside the channel samples and compromised cross correlation between channel measurements of two keying parties. This paper studies the signal preprocessing algorithms to establish high-agreement uncorrelated secret key in the presence of channel independent eavesdroppers. We first propose a general mathematical model for various preprocessing schemes, including principal component analysis (PCA), discrete cosine transform (DCT) and wavelet transform (WT). Among preprocessing schemes, PCA is proved to achieve the optimal secret key rate. Next, PCA with common eigenvector has been found to outperform PCA with private eigenvector in terms of an overall consideration of key agreement, information leakage and computational expense. Then, we propose a system level design of key generation, including quantization, information reconciliation and privacy amplification. Numerical results verify that the key generation enhanced by PCA with common eigenvector can achieve secret key with high key generation rate, low key error rate and good randomness.

Index Terms—Wireless communications, physical layer security, secret key generation, channel reciprocity, decorrelation, principal component analysis.

I. INTRODUCTION

The ongoing research and development of the fifth generation (5G) wireless communications has presented a very exciting vision by providing extremely low latency and high data rate, which will trigger many killer applications such as Internet of Things [1]. However, the security and privacy of the wireless communications is always the main challenge because the broadcast nature of wireless transmissions exposes the confidential data exchanged to any third party within the

Manuscript received July 17, 2017; revised November 15, 2017; accepted February 28, 2018. The associate editor coordinating the review of this paper and approving it for publication was Prof. Ragnar Thobaben. This paper was presented in part at the IEEE GLOBECOM Workshop on Trusted Communications with Physical Layer Security (TCPLS), Washington DC, USA, Dec. 2016. This work was supported in part by the National Natural Science Foundation of China (General Program: 61571110) and National Natural Science Youth Foundation of China under Grant 61602113. The work of L. Peng was supported in part by National Natural Science Youth Foundation of China under Grant 61601114, and Natural Science Youth Foundation of Jiangsu Province of China under Grant BK20160692. A. Hu is the corresponding author.

G. Li, A. Hu, L. Peng, C. Sun, and D. Cao are with School of Information Science and Engineering, Southeast University, Nanjing 210096, China. (email: guyuelee@seu.edu.cn; aqhu@seu.edu.cn; pengln@seu.edu.cn; sunchen@seu.edu.cn; and dmcao@seu.edu.cn)

J. Zhang is with the Department of Electrical Engineering and Electronics at the University of Liverpool, Liverpool, U.K. (email: junqing.zhang@liverpool.ac.uk)

communication range. This has attracted extensive research efforts with a focus on the classic cryptography [2].

Public key cryptography (PKC) is used to distribute the keys among users. This technology is based on the computational complexity of mathematical problems such as integer factorization and discrete logarithms, but it may be completely broken by the emerging quantum computers in future [3]. In addition, PKC requires a public key infrastructure which may not be available in many 5G applications such as device-to-device communications. Physical layer secret key generation has emerged as a strong candidate to complement PKC by exploiting common randomness from the wireless channels [4]. It exploits the unpredictable channel characteristics as the key and therefore is information theoretically secure [5]. The process can be completed by a pair of users and no help from other user is required. Finally, this technology is implementable in the commercial wireless systems, e.g., evidenced by prototypes with WiFi [6].

Key generation usually contains four main steps: channel sounding, quantization, information reconciliation and privacy amplification [4]. Two legitimate parties, namely Alice and Bob, successively and alternately send probe signals to each other for obtaining channel characteristics, such as received signal strength (RSS), channel state information (CSI), time delay, amplitude, phase and angle-of-arrive (AoA), etc. Both users will quantize their channel measurements into binary sequences once they collect enough data. Because key generation usually works in time-division duplex (TDD) mode, the channel measurements are subject to non-simultaneous sampling and noise. Therefore, there will probably be bit discrepancies, and information reconciliation is adopted to enhance bit agreement by applying error correction code [7]. Finally, privacy amplification such as universal hash functions increases randomness of the final key and wipes off the leaked information in the above procedures [8].

The randomness can be harvested from the temporal, frequency and spatial domains, by employing orthogonal frequency-division multiplexing (OFDM) techniques [9–12] or multiple antennas [13–15]. However, there may exist correlation between the adjacent measurements when the two probes are within the same coherence time and/or coherence bandwidth, which will introduce redundancy and may finally result in failure of key generation. Although privacy amplification methods are used to randomize the binary sequence, the randomness of sequences before privacy amplification is also important. A non-random sequence will be easily cracked by

dictionary attack.

The correlation within the collected data can be eliminated by introducing signal preprocessing procedure after channel sounding, such as principal component analysis (PCA) [15, 16], discrete cosine transform (DCT) [9, 17] and wavelet transform (WT) [18, 19]. PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. PCA is also named the discrete Karhunen-Loève transform (KLT) in signal processing domain. In [15, 16], channel measurements are decorrelated by transformation with their matrix of eigenvectors. There will be information leakage during the exchange of eigenvectors. The work in [20] proposes a channel trend information (CTI)-based algorithm without leaking any sensitive information. Alice and Bob first share the confidence constant, N , which indicates the number of agreeing ones or zeros required before a secret bit is generated. However, when N is small, this algorithm suffers from a low average success rate of key generation.

In [9], power spectrum is analyzed by the DCT and the high-frequency components are trimmed, so the remaining low-frequency components hold the power up to 90% in both time and frequency scale. Compressed time-variant frequency characteristics can then be obtained by inverse DCT. In the same way, DCT is employed to transform raw RSS samples and uncorrelated higher frequency components are discarded in [17]. In [18], channel measurements are mapped into WT domain and only the low frequency parts are used for key establishment to reduce key mismatch rate. Similarly, in [19], a compressor based on multi-level WT is applied to preprocess measurements so that some discrepancies between measurement sequences of different transceivers can be eliminated. The above preprocessing schemes are summarized and compared in [21]. Their results show that KLT is outperformed by DCT and WT, which is questionable. When analyzing the bit generation factor, the bit disagreement rate threshold in [21] is set too high (0.35). With such high bit disagreement, the key generation might be unsuccessful even with information reconciliation, which can significantly affect the efficiency of key generation. In addition, components selection and adaptive quantization methods are not considered in [21], but KLT features at concentrating the majority of information in a few principal components. A fair analysis is then still required to compare their performance. Regarding PCA, in [15, 16], Alice and Bob use the same covariance matrix for singular value decomposition to achieve key agreement, but the transmission of the eigenvector will leak information to eavesdropper which is not discussed.

The reciprocity of channel measurements is corrupted by non-simultaneous sampling, channel noise, and hardware imperfection [11, 22, 23], which is tackled mainly by low pass filtering [10] and interpolation [16]. On the other hand, PCA can also improve the cross-correlation between the channel measurements of Alice and Bob, because the noisy observation is removed by only keeping principal components [15]. However, a detailed and full analysis of how PCA affects channel cross-correlation is still missing from the existing work.

To bridge the above gaps, this paper carries out a comprehensive and theoretical study on the signal preprocessing algorithms to establish high-agreement uncorrelated secret key and uses an OFDM system as an example to evaluate the performance. Our contributions are as follows.

- Under the assumption that eavesdropper experiences independent fading channel, a general mathematical model of the signal preprocessing algorithms is built to find the optimal approach. PCA is proved in theory and later validated by Monte Carlo simulation to achieve a higher secret key rate than DCT and WT.
- Through comprehensive discussion and comparison in terms of key agreement, information leakage and computational expense, PCA with common eigenvector is proved to outperform PCA with private eigenvectors.
- A system level design of secret key generation is presented, including channel sounding, preprocessing, quantization, information reconciliation and privacy amplification. The system performance is evaluated in terms of secret key rate, key agreement and randomness.

In our previous work, we have investigated the optimal preprocessing approach in secret key generation [24]. We found that PCA achieves the highest secret key rate, and presented realization steps of PCA algorithm with common eigenvectors. In this paper, we considerably extend and complement this work by providing full discussion and proof of the optimality of PCA and comparing the performance of PCA with common and private eigenvectors from the perspective of key agreement, information leakage and computational expense.

Notation and Outline

Unless otherwise specified, we use the following notations throughout the manuscript: Upper (lower) bold-face letters denote matrices (column vectors); \mathbf{I} denotes the identity matrix, while $\mathbf{0}$ denotes zero matrix. Numeral subscripts of matrices and vectors, if needed, represent their sizes. Also, matrix superscripts $(\cdot)^H$, $(\cdot)^T$, $(\cdot)^*$ denote their conjugate-transpose, transpose, and conjugate, respectively. Superscripts $(\cdot)^c$ and $(\cdot)^p$ denote the value is calculated for PCA algorithm with common eigenvector and with private eigenvectors, respectively. We use $E\{\cdot\}$ to denote ensemble expectation and $|\cdot|$ to represent matrix determinant operations. We use \mathbf{R}_x and \mathbf{R}_{xy} to denote the covariance matrix of vector \mathbf{x} and the cross-covariance matrix of vectors \mathbf{x} and \mathbf{y} , respectively. The operation $\text{diag}\{x_1, x_2, \dots, x_N\}$ denotes a diagonal matrix with x_1, x_2, \dots, x_N along its main diagonal. The inequality $\mathbf{A} \succeq \mathbf{0}$ denotes a positive semi-definite Hermitian matrix \mathbf{A} , and $\mathbf{A} \succeq \mathbf{B}$ means that the matrix $\mathbf{A} - \mathbf{B}$ is a positive semi-definite Hermitian matrix.

The rest of the paper is organized as follows. Section II derives the general model for signal pre-processing and proves PCA can achieve the optimal secret key rate. Section III proposes and compares two realization of PCA schemes. Section IV designs other key generation procedures. Section V presents the simulation results and Section VI concludes the paper. All proofs are deferred to the Appendix.

II. SIGNAL PREPROCESSING MODEL AND OPTIMIZATION

We consider a general single-input single-output single-eavesdropper (SISOSE) model. All the users are equipped with a single antenna. Alice and Bob are two legitimate users who plan to extract key for secure communication. Eve, a passive eavesdropper, is located more than 10 wavelengths (1.25 m at the carrier frequency of 2.4 GHz) away from Alice and Bob, and there is no strong line-of-sight (LoS) between Eve and Alice or Bob. Therefore, Eve experiences independent fading from legitimate users. Besides, Eve knows the communication protocol and the information transmitted over the public channels between legitimate users. Key generation requires a temporally dynamic channel, and the channel variation can be introduced by the movement of users and/or surrounding objects [25].

This section presents a general model of preprocessing procedure including PCA, DCT, and WT. We then derive the secret key rate of the key generation with signal preprocessing.

A. Signal Preprocessing Model

During channel sounding, Alice and Bob alternately send probe signals to each other, and estimate the CSI. The k -th channel estimation vector with a length of N can be written as

$$\mathbf{h}_u^{(k)} = \mathbf{h}^{(k)} + \mathbf{n}_u^{(k)}, \quad (1)$$

where $u = \{a, b\}$ denotes Alice and Bob, respectively, $\mathbf{h}^{(k)}$ follows complex Gaussian distribution, and \mathbf{n}_u is independent and identically distributed (i.i.d.) zero-mean complex Gaussian noise with variance $E\{\mathbf{n}_u^{(k)}(\mathbf{n}_u^{(k)})^H\} = \sigma_n^2 \mathbf{I}_N$. After K channel samplings, Alice and Bob can construct matrix $\tilde{\mathbf{H}}_u$ as

$$\tilde{\mathbf{H}}_u = [\mathbf{h}_u^{(0)}, \mathbf{h}_u^{(1)}, \dots, \mathbf{h}_u^{(K-1)}], \quad (2)$$

where $\mathbf{h}_u^{(k)}$ and $\mathbf{h}_u^{(l)}$ are assumed to be i.i.d., $k, l \in [0, 1, \dots, K-1]$. Therefore, the superscript is omitted for simplicity from now on. Define the channel signal-to-noise ratio (SNR) as

$$SNR = \frac{E\{\mathbf{h}^H \mathbf{h}\}}{N\sigma_n^2}. \quad (3)$$

Linear signal processing transforms construct various sets of standard independent bases $\{\mathbf{u}^{(i)}\}$ and the transformation matrix can be given as a $N \times N$ unitary matrix

$$\mathbf{U} = [\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \dots, \mathbf{u}^{(N)}]. \quad (4)$$

DCT and WT are linear transforms which are independent of the data. The (i, j) -th element of the DCT matrix \mathbf{U}^{DCT} can be expressed as [26]

$$[\mathbf{U}^{\text{DCT}}]_{ij} = \frac{1}{\sqrt{N}} \begin{cases} 1, & i = 1 \\ \sqrt{2} \cos\left(\frac{(i-1)(2j-1)\pi}{2N}\right), & i > 1. \end{cases} \quad (5)$$

Haar transform (HT) is the simplest form of the WT. This transform cross-multiplies a function against the Haar wavelet

with various shifts and stretches. The (i, j) -th element of the HT matrix \mathbf{U}^{HT} is given by [27] for $i = 1$,

$$[\mathbf{U}^{\text{HT}}]_{ij} = \frac{1}{\sqrt{N}}, \quad j = 1, 2, \dots, N \quad (6)$$

and for any $i > 1$ and $i = 2^p + q - 1$, where 2^p is the largest power of 2 contained in i and $q - 1$ is the remainder,

$$[\mathbf{U}^{\text{HT}}]_{ij} = \frac{1}{\sqrt{N}} \begin{cases} 2^{p/2}, & (q-1)/2^p \leq j < (q-0.5)/2^p \\ -2^{p/2}, & (q-0.5)/2^p \leq j < q/2^p \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

On the other hand, PCA is a data dependent transform. The covariance matrix of the ideal channel \mathbf{h} can be decomposed as

$$\mathbf{R}_h = E\{\mathbf{h}\mathbf{h}^H\} = \mathbf{U}_h \mathbf{\Lambda}_h \mathbf{U}_h^H, \quad (8)$$

where $\mathbf{U}_h = [\mathbf{u}_h^{(1)}, \mathbf{u}_h^{(2)}, \dots, \mathbf{u}_h^{(N)}]$ is the eigenmatrix and $\mathbf{\Lambda}_h$ is a diagonal matrix, whose diagonal entries are the sorted eigenvalues, denoted by λ_i ($\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$). \mathbf{U}_h is the transformation matrix for PCA.

After transform domain mapping, matrix $\tilde{\mathbf{H}}_u$ is transformed to matrix $\mathbf{Y}_u = [\mathbf{y}_u^{(0)}, \mathbf{y}_u^{(1)}, \dots, \mathbf{y}_u^{(K-1)}]$ by expanding the CSI estimates with their projections on these bases

$$\mathbf{Y}_u = \mathbf{U}^H \tilde{\mathbf{H}}_u. \quad (9)$$

This process can be mathematically modeled as multiplying CSI estimates with specific unitary matrix \mathbf{U} .

However, not all column vectors in \mathbf{Y}_u can be employed for secret key generation purpose. For example, in WT, some high frequency components are discarded and in PCA, only a small part of principal components are used for key generation. Thus, signal reconstruction reorders the column vectors and select parts of them to form a $M \times K$ matrix $\tilde{\mathbf{Y}}_u$. On one hand, a large number of components are discarded since there are redundancies which do not contribute to secret key rate. On the other hand, some components which contain little information are not used in practice either. Although trimming them causes certain loss of information, it is too costly to use them as they are severely corrupted by noise. Combined with signal reconstruction, the mathematical model is modified to multiply CSI estimates $\tilde{\mathbf{H}}_u$ with a tall unitary matrix

$$\mathbf{V} = [\mathbf{v}^{(1)}, \mathbf{v}^{(2)}, \dots, \mathbf{v}^{(M)}], \quad (10)$$

which is a $N \times M$ matrix and satisfies $\mathbf{V}^H \mathbf{V} = \mathbf{I}_M$, $M \leq N$. After signal reconstruction, the output $M \times K$ matrix $\tilde{\mathbf{Y}}_u = [\tilde{\mathbf{y}}_u^{(0)}, \tilde{\mathbf{y}}_u^{(1)}, \dots, \tilde{\mathbf{y}}_u^{(K-1)}]$ is obtained by

$$\tilde{\mathbf{Y}}_u = \mathbf{V}^H \tilde{\mathbf{H}}_u, \quad (11)$$

which is the output signal of preprocessing procedure.

B. Secret Key Rate Optimization

The secret key rate of Alice's and Bob's mapped signals after transform domain mapping is computed by

$$\begin{aligned} R &= \frac{1}{N} I(\mathbf{y}_a; \mathbf{y}_b | \mathbf{h}_{ae}, \mathbf{h}_{be}, \mathbf{U}) \\ &= \frac{1}{N} (I(\mathbf{y}_a; \mathbf{y}_b) - I(\mathbf{y}_a; \mathbf{h}_{ae}, \mathbf{h}_{be}, \mathbf{U})), \end{aligned} \quad (12)$$

where \mathbf{h}_{ae} and \mathbf{h}_{be} is the channel from Alice to Eve and the channel from Bob to Eve, respectively. $I(\mathbf{y}_a; \mathbf{h}_{ae}, \mathbf{h}_{be}, \mathbf{U})$ is the mutual information between \mathbf{y}_a and $\mathbf{h}_{ae}, \mathbf{h}_{be}, \mathbf{U}$, which represents the information leaked to eavesdroppers. The leakage can be attributed to two reasons, i.e. Eve's passive eavesdropping on her channel observation \mathbf{h}_{ae} and \mathbf{h}_{be} , and transmission of unitary matrix \mathbf{U} over public channel. As shown in existing work [25, 28], when Eve is located more than 10 wavelengths away from Alice and Bob in an environment not with strong LoS, she experiences independent fading. Therefore, we do not take into account the information leakage due to Eve's passive eavesdropping on the channel observations. For data independent transform, e.g. DCT and WT, \mathbf{U} is independent of \mathbf{y}_a . As it will be analyzed in detail in Section III-C, the information leakage of transmitting eigenvector is very small, thus, we treat the amount of information leakage in PCA algorithm as zero. In summary, $I(\mathbf{y}_a; \mathbf{h}_{ae}, \mathbf{h}_{be}, \mathbf{U})$ is considered as zero in this section. According to [29],

$$\begin{aligned} R &= \frac{1}{N} I(\mathbf{y}_a; \mathbf{y}_b) \\ &= \frac{1}{N} \log_2 \frac{|\mathbf{R}_{y_a}| |\mathbf{R}_{y_b}|}{|\mathbf{R}_{y_a}| |\mathbf{R}_{y_b} - \mathbf{R}_{y_a y_b} \mathbf{R}_{y_a}^{-1} \mathbf{R}_{y_b y_a}|} \\ &= \frac{1}{N} \log_2 \frac{|\boldsymbol{\Lambda}_h + \sigma_n^2 \mathbf{I}|}{|\boldsymbol{\Lambda}_h + \sigma_n^2 \mathbf{I} - \boldsymbol{\Lambda}_h (\boldsymbol{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1} \boldsymbol{\Lambda}_h|} \\ &= \frac{1}{N} \sum_{i=1}^N \log_2 \frac{1}{1 - \left(\frac{\lambda_i / \sigma_n^2}{1 + \lambda_i / \sigma_n^2} \right)^2}. \end{aligned} \quad (13)$$

The secret key rate R is independent of the unitary matrix \mathbf{U} , only relying on the eigenvalues of the channel covariance matrix and the noise variance. As unitary transforms are invertible, they do not affect the secret key rate. In practice, due to the time and frequency correlation of the observation, the rank of the covariance matrix \mathbf{R}_h is much smaller than N , and only a few eigenvalues dominate the secret key rate. To reduce the redundancy, we only exploit M dominant components for key generation and drop the other weak components.

As shown in (11), the reconstructed signals obtained by Alice and Bob are $\tilde{\mathbf{y}}_a = \mathbf{V}^H \mathbf{h}_a$, $\tilde{\mathbf{y}}_b = \mathbf{V}^H \mathbf{h}_b$, respectively. Thus, the secret key rate is calculated as

$$\begin{aligned} \tilde{R} &= \frac{1}{N} \log_2 \frac{|\mathbf{R}_{\tilde{y}_a}| |\mathbf{R}_{\tilde{y}_b}|}{|\mathbf{R}_{\tilde{y}_a}| |\mathbf{R}_{\tilde{y}_b} - \mathbf{R}_{\tilde{y}_a \tilde{y}_b} \mathbf{R}_{\tilde{y}_a}^{-1} \mathbf{R}_{\tilde{y}_b \tilde{y}_a}|} \\ &= \frac{1}{N} \log_2 \frac{|\mathbf{V}^H (\mathbf{R}_h + \sigma_n^2 \mathbf{I}) \mathbf{V}|}{|\mathbf{V}^H (\mathbf{R}_h + \sigma_n^2 \mathbf{I}) \mathbf{V} - \mathbf{V}^H \mathbf{R}_h (\mathbf{R}_h + \sigma_n^2 \mathbf{I})^{-1} \mathbf{R}_h \mathbf{V}|}. \end{aligned} \quad (14)$$

Our main objective is to design the optimal tall unitary matrix \mathbf{V} , maximizing the key rate, which can be expressed as

$$\mathbf{V}^* = \arg \max_{\mathbf{V}} \tilde{R} \quad (15a)$$

$$\text{s.t. } \mathbf{V}^H \mathbf{V} = \mathbf{I}. \quad (15b)$$

We have the following theorem for this optimization problem.

Theorem 1: The optimal transform matrix \mathbf{V}^* is given by the eigenvectors of the channel covariance matrix corresponding to the M maximum eigenvalues, i.e.,

$$\mathbf{V}^* = \left[\mathbf{u}_h^{(1)}, \mathbf{u}_h^{(2)}, \dots, \mathbf{u}_h^{(M)} \right]. \quad (16)$$

Proof: See Appendix A. \blacksquare

From Theorem 1, we can observe that the optimal \mathbf{V} is consisted of eigenvectors corresponding to the maximum M eigenvectors of the covariance matrix. This means that PCA achieves the optimal performance among all tall unitary matrix transforms from the perspective of secret key rate. Particularly, when the rank of \mathbf{R}_h is M , $\tilde{R} = R$.

III. PRINCIPAL COMPONENT ANALYSIS ALGORITHM

In this section, we analyze two specific realization algorithms of PCA and present an overall comparison results of them.

A. Algorithm Description

The channel covariance matrix at user u is calculated as

$$\mathbf{R}_{h_u} = E \{ \mathbf{h}_u \mathbf{h}_u^H \} = \mathbf{U}_h (\boldsymbol{\Lambda}_h + \sigma_n^2 \mathbf{I}_N) \mathbf{U}_h^H, \quad (17)$$

which indicates that although the received signals at Alice and Bob are different, their eigenvectors are identical. However, in practice, the estimate of covariance matrix is always inaccurate as true channel probability distribution is unknown and ensemble average is approximated by time or frequency average, which is given as

$$\mathbf{R}_{h_u} = \mathbf{U}_u \boldsymbol{\Lambda}_u \mathbf{U}_u^H \approx \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{h}_u^{(k)} \mathbf{h}_u^{(k)H}. \quad (18)$$

Note that \mathbf{U}_u and $\boldsymbol{\Lambda}_u$ are different from \mathbf{U}_h and $\boldsymbol{\Lambda}_h$ due to the noise. \mathbf{U}_h and $\boldsymbol{\Lambda}_h$ are the exact eigenmatrix and diagonal matrix with eigenvalues of the covariance matrix of ideal channel \mathbf{h} , while \mathbf{U}_u and $\boldsymbol{\Lambda}_u$ are the eigenmatrix and diagonal matrix with eigenvalues of the estimated covariance matrix of channel at Alice and Bob.

As a result, there is a deviation between the covariance matrices of Alice and Bob. The relationship between channel estimates of Alice and Bob can be modified as

$$\mathbf{h}_b = \mathbf{h}_a + \mathbf{n}_d, \quad (19)$$

where $\mathbf{n}_d \in \mathbb{C}^{N \times 1}$ is independent with \mathbf{h}_a , representing the observation deviation noise. As the deviation is caused by the ensemble average approximation, we assume \mathbf{n}_d is colored Gaussian noise with the covariance \mathbf{R}_{n_d} .

Even a small deviation will result in an enormous difference to their corresponding eigenvalues and eigenvectors. To solve this problem, Alice can send her eigenvector to Bob and both will use it for signal reconstruction [15, 16], which is named as PCA algorithm with common eigenvector. On the other hand, Alice and Bob can calculate their own eigenvectors and use them for signal reconstruction without any interaction, which is called PCA algorithm with private eigenvectors in this paper. Implementation block diagrams of these two methods are summarized in Fig. 1.

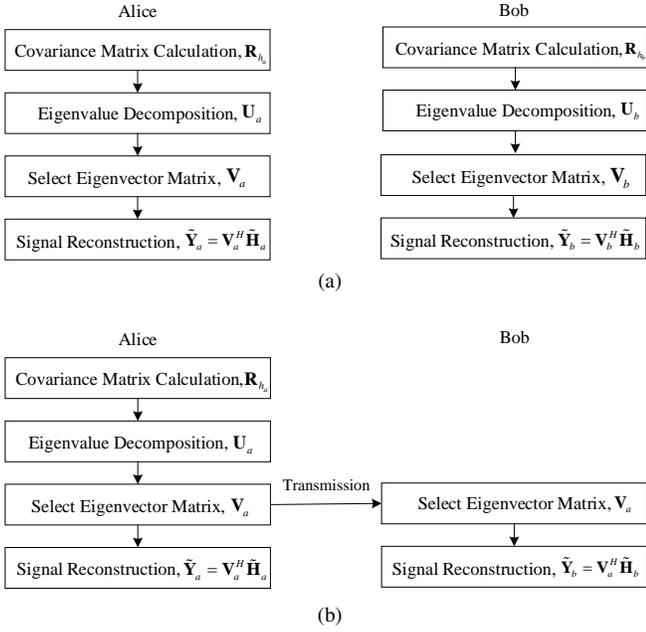


Fig. 1: (a) PCA with private eigenvector. (b) PCA with common eigenvector.

B. Key Agreement Analysis

Cross-correlation analysis can quantify the similarity between two signals, which is used here to evaluate the key agreement. The cross-correlation coefficient of user u 's and v 's i -th transformed components can be given as

$$\rho_i = \frac{|E\{y_{ui}y_{vi}\}|}{\sqrt{E\{y_{ui}y_{ui}\}E\{y_{vi}y_{vi}\}}}, i \in \{0, \dots, N\}. \quad (20)$$

When calculating ρ_i^c for PCA algorithm with common eigenvector, $y_{ui} = (\mathbf{u}_a^{(i)})^H \mathbf{h}_u$. On the other hand, when calculating ρ_i^p for PCA with private eigenvector, $y_{ui} = (\mathbf{u}_u^{(i)})^H \mathbf{h}_u$. For both cases, we calculate lower bounds of Alice's and Bob's i -th cross-correlation coefficients and compare them as shown by the following theorem.

Theorem 2: Alice's and Bob's correlation coefficient lower bounds of PCA with common and private eigenvectors are respectively given by

$$\rho_{i,lb}^c = \sqrt{\frac{\lambda_i}{\lambda_i + \delta}}, \quad (21)$$

$$\rho_{i,lb}^p = \sqrt{\frac{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2}{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2 + \sum_{j \neq i} \lambda_j |(\mathbf{u}_a^{(j)})^H \mathbf{u}_b^{(i)}|^2 + \delta}}, \quad (22)$$

where $\delta = \max_i (\mathbf{u}_a^{(i)})^H \mathbf{R}_{\mathbf{n}_d} \mathbf{u}_a^{(i)}$. The relationship between $\rho_{i,lb}^c$ and $\rho_{i,lb}^p$ is

$$\rho_{i,lb}^c \geq \rho_{i,lb}^p. \quad (23)$$

Proof: See Appendix B. ■

Remark 1: Consider a special case $\mathbf{R}_{\mathbf{n}_d} = \sigma_n^2 \mathbf{I}$, which means that the noise \mathbf{n}_d consists of i.i.d. zero-mean complex Gaussian white random variables with variance σ_n^2 . In this

case, the eigenvectors satisfy $\mathbf{u}_a^{(i)} = \mathbf{u}_b^{(i)}$, and Alice's and Bob's correlation coefficient $\rho_i^c = \rho_i^p = \sqrt{\lambda_i / (\lambda_i + \sigma_n^2)}$, which benefits from the increase of SNR and eigenvalue. When SNR is fixed, the components with higher variances can achieve better key agreement.

Remark 2: Theorem 2 reveals that for any i -th component, the key agreement lower bound of PCA algorithm with common eigenvector is higher than that with private eigenvectors. Moreover, when SNR is fixed, $\rho_{i,lb}^c$ benefits from the increase of eigenvalue. The above theorem proves that the transmission of eigenvector can indeed improve the key agreement between Alice and Bob.

Since eigenvalue decomposition is a high resource-consuming step, Bob's computational expense is much smaller in PCA algorithm with common eigenvector than that with private eigenvectors, which is beneficial and desirable in certain scenarios. For example, Alice is an access point with strong computational capacity while Bob is a low cost embedded device. Thus, it is essential to analyze whether transmitting \mathbf{V}_a to Bob will give Eve chance to guess the secret key.

C. Information Leakage Analysis

The transmission of eigenvector over an insecure public channel can cause information leakage, particularly when Eve can also obtain the eigenvalue. If Eve obtains enough information, she can perform a brute-force search to find the secret key. The eigenvector indicates the eigen-directions, and the eigenvalue indicates the energy on each direction. When Eve knows both eigenvector and eigenvalue, she can deduce the covariance matrix and vice versa. Therefore, we calculate the leakage ratio caused by transmitting covariance matrix, which is the upper bound of the leakage caused by transmitting eigenvector.

Covariance matrix provides the statistical information of channel vector \mathbf{h}_u which narrows down brute-force search scope of Eve. Since secret key is generated from instantaneous channel measurements which are unknown to Eve, the information leakage rate depends on the dimensions of statistical information compared with that of instantaneous information.

The information leakage caused by transmitting covariance matrix \mathbf{R}_{h_u} can be expressed as

$$\begin{aligned} I(\tilde{\mathbf{Y}}_u; \mathbf{R}_{h_u}) &= H(\tilde{\mathbf{Y}}_u) - H(\tilde{\mathbf{Y}}_u | \mathbf{R}_{h_u}) \\ &= H(\tilde{\mathbf{Y}}_u) - H(\tilde{\mathbf{Y}}_u | \mathbf{V}_u, \mathbf{R}_{h_u}). \end{aligned} \quad (24)$$

As $\tilde{\mathbf{Y}}_u = \mathbf{V}_u^H \tilde{\mathbf{H}}_u$ and $\mathbf{R}_{h_u} = \frac{1}{K} \tilde{\mathbf{H}}_u \tilde{\mathbf{H}}_u^H$, we can obtain $\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H = K \mathbf{V}_u^H \mathbf{R}_{h_u} \mathbf{V}_u$.

Let $\mathbf{U}_u = [\mathbf{V}_u \ \mathbf{A}]$ be a unitary matrix. We can construct \mathbf{Y}_u as

$$\mathbf{Y}_u = \begin{bmatrix} \tilde{\mathbf{Y}}_u \\ \mathbf{B} \end{bmatrix} = \mathbf{U}_u^H \tilde{\mathbf{H}}_u = \begin{bmatrix} \mathbf{V}_u^H \tilde{\mathbf{H}}_u \\ \mathbf{A}^H \tilde{\mathbf{H}}_u \end{bmatrix}. \quad (25)$$

As \mathbf{U}_u is the eigenmatrix of \mathbf{R}_{h_u} , we can have

$$\begin{aligned} H(\tilde{\mathbf{Y}}_u|\mathbf{V}_u, \mathbf{R}_{h_u}) &= H(\tilde{\mathbf{Y}}_u|\mathbf{U}_u, \mathbf{U}_u^H \mathbf{R}_{h_u} \mathbf{U}_u) \\ &= H(\tilde{\mathbf{Y}}_u|\mathbf{U}_u, \mathbf{Y}_u \mathbf{Y}_u^H) \\ &= H\left(\tilde{\mathbf{Y}}_u|\mathbf{U}_u, \begin{bmatrix} \tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H & \tilde{\mathbf{Y}}_u \mathbf{B}^H \\ \mathbf{B} \tilde{\mathbf{Y}}_u^H & \mathbf{B} \mathbf{B}^H \end{bmatrix}\right). \end{aligned} \quad (26)$$

When the data length is long enough, $\tilde{\mathbf{Y}}_u$ and \mathbf{B} becomes independent, and we have $\mathbf{B} \tilde{\mathbf{Y}}_u^H = \tilde{\mathbf{Y}}_u \mathbf{B}^H = \mathbf{0}$. Then,

$$H(\tilde{\mathbf{Y}}_u|\mathbf{V}_u, \mathbf{R}_{h_u}) = H(\tilde{\mathbf{Y}}_u|\mathbf{U}_u, \tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H, \mathbf{B} \mathbf{B}^H). \quad (27)$$

In addition, \mathbf{U}_u is the eigenmatrix of \mathbf{R}_{h_u} . Thus, \mathbf{U}_u depends on \mathbf{R}_{h_u} , independent of $\tilde{\mathbf{Y}}_u$, and we have

$$\begin{aligned} H(\tilde{\mathbf{Y}}_u|\mathbf{V}_u, \mathbf{R}_{h_u}) &= H(\tilde{\mathbf{Y}}_u|\mathbf{U}_u, \tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H, \mathbf{B} \mathbf{B}^H) \\ &= H(\tilde{\mathbf{Y}}_u|\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H). \end{aligned} \quad (28)$$

Thus,

$$I(\tilde{\mathbf{Y}}_u; \mathbf{R}_{h_u}) = H(\tilde{\mathbf{Y}}_u) - H(\tilde{\mathbf{Y}}_u|\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H). \quad (29)$$

As shown in (29), the information leakage is related to the equivocation of a $M \times K$ matrix $\tilde{\mathbf{Y}}_u$ on condition of a known $M \times M$ matrix $\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H$. In practice, K is usually much larger than M . As the elements of $\tilde{\mathbf{Y}}_u$ are independent, the matrix $\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H$ approximates to a diagonal matrix,

$$\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H \approx \tilde{\mathbf{\Lambda}}_u, \quad (30)$$

where $\tilde{\mathbf{\Lambda}}_u = \text{diag}\{\lambda_1 + \sigma_n^2, \lambda_2 + \sigma_n^2, \dots, \lambda_M + \sigma_n^2\}$. Therefore, matrix $\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H$ provides M constraint equations of the $M \times K$ elements in matrix $\tilde{\mathbf{Y}}_u$. At most M basic variables in $\tilde{\mathbf{Y}}_u$ can be solved in terms of $MK - M$ free variables. When these constraint equations are nonlinear or non-independent, the number of basic variables is less than M and the number of free variables is more than $MK - M$.

Assume that the elements of matrix $\tilde{\mathbf{Y}}_u$ is uniformly quantized in a discrete set with cardinality S . $H(\tilde{\mathbf{Y}}_u|\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H)$ can be approximated by

$$H(\tilde{\mathbf{Y}}_u|\tilde{\mathbf{Y}}_u \tilde{\mathbf{Y}}_u^H) \approx (MK - M) \log(S). \quad (31)$$

As $\tilde{\mathbf{Y}}_u$ has MK free variables, $H(\tilde{\mathbf{Y}}_u)$ can be expressed as

$$H(\tilde{\mathbf{Y}}_u) = MK \log(S). \quad (32)$$

Thus, the information leakage can be expressed as

$$I(\tilde{\mathbf{Y}}_u; \mathbf{R}_{h_u}) \approx M \log(S). \quad (33)$$

In this paper, the information leakage ratio η is defined as the ratio of the mutual information of $I(\tilde{\mathbf{Y}}_u; \mathbf{R}_{h_u})$ and the entropy of $\tilde{\mathbf{Y}}_u$ and can be given as

$$\eta = \frac{I(\tilde{\mathbf{Y}}_u; \mathbf{R}_{h_u})}{H(\tilde{\mathbf{Y}}_u)} \approx \frac{1}{K}, K > M, \quad (34)$$

which is a monotonic decreasing function. Therefore, a large K in calculating covariance matrix can reduce information leakage ratio. However, large K also means that secret key cannot be produced immediately. Therefore, K should be reasonably designed to make a tradeoff between real-time and

information leakage. For security purpose, we should at least wipe off $1/K$ bits in privacy amplification.

The covariance matrix provides information of both eigenvectors and eigenvalues. When Eve can only obtain eigenvectors instead of covariance matrix, the leakage ratio is much lower than η . Thus, when K is large, the information leakage caused by transmitting eigenvector is negligibly small.

In summary, sharing the eigenvectors between Alice and Bob can bring in higher key agreement, lower computational expense and negligibly small information leakage. Therefore, PCA algorithm with common eigenvector is superior to that with private eigenvector from an overall perspective.

IV. KEY GENERATION PROCEDURES

Besides channel sounding (with signal preprocessing), a key generation system also requires steps including quantization, information reconciliation, and privacy amplification, which are designed in this section.

After privacy amplification, Alice and Bob verify the agreement of their candidate keys in groups and each group has a length of L bits. Alice sends the Hash value of her candidate key to Bob. When their candidate keys match each other, a group of key with length L is established successfully. Otherwise, the group of candidate key fails.

A. Quantization

After preprocessing, different components have different SNRs, which can be expressed as

$$\gamma_i = \frac{(\mathbf{u}^{(i)})^H \mathbf{R}_h \mathbf{u}^{(i)}}{\sigma_n^2}. \quad (35)$$

For PCA preprocessing, γ_i can be written as

$$\gamma_i = \frac{\lambda_i^2}{\sigma_n^2}. \quad (36)$$

Note that γ_i in (36) represents the SNRs of different components, which is different from the channel SNR defined in (3). As the index of components increases, the SNR decreases. To make full use of high SNR of dominant components, we employ flexible quantization levels in the quantization algorithm.

Define key error rate (KER) as the number of failed groups divided by the number of total candidate key groups. In practice, to satisfy the high agreement, the KER need to be less than 10^{-3} . To meet the KER requirement, flexible quantization levels are set according to their SNRs. Employing the similar methodology in [15], we compute the SNR thresholds for various quantization levels \mathbf{Q} as shown in Table I. For each component with a SNR γ_i among these thresholds can set its quantization level \mathbf{Q}_i . Then, Alice and Bob use the cumulative distribution function (CDF) of their i -th components to quantize their range of values into \mathbf{Q}_i equally likely regions and generates $\log_2 \mathbf{Q}_i$ bits based on this quantization [16].

TABLE I: SNR THRESHOLDS TO ACHIEVE $KER = 10^{-3}$ FOR VARIOUS QUANTIZATION LEVELS WHEN $L = 128$

γ_i (dB)	$Q = 2^1$	$Q = 2^2$	$Q = 2^3$	$Q = 2^4$	$Q = 2^5$
	17.5	20	23	27	30

B. Information Reconciliation

To further reduce the KER, distributed source coding (DSC) [30] is used. Sartipi and Fekri proposed a scheme for DSC to achieve any arbitrary rate on the Slepian-Wolf rate region using low density parity check code (LDPC) [31]. However, the belief-propagation iterative decoding algorithm is of high computational complexity. To reduce the complexity, we propose a simple decoding algorithm based on bit-flipping.

For a group of quantized key sequences \mathbf{q}_a and \mathbf{q}_b with length L_R , the flow chart of the decoding algorithm is illustrated in Fig. 2. The algorithm includes the following main steps:

First, the flipping factor τ is initialized as

$$\tau = [\tau^{(0)}, \tau^{(1)}, \dots, \tau^{(L_R-1)}], \quad (37)$$

and

$$\tau^{(r)} = \begin{cases} 1 - \left| \frac{\tilde{y}_b^{(r)}}{\sigma_n^2} \right|, & \left| \frac{\tilde{y}_b^{(r)}}{\sigma_n^2} \right| \leq 1 \\ 0, & \left| \frac{\tilde{y}_b^{(r)}}{\sigma_n^2} \right| > 1, \end{cases} \quad (38)$$

where $\tilde{y}_b^{(r)}$ is the corresponding input signal of $\mathbf{q}_b^{(r)}$ before quantization.

Then, syndromes s_a and s_b are calculated respectively by

$$\mathbf{s}_a = \mathbf{q}_a \mathbf{G}^H, \quad \mathbf{s}_b = \mathbf{q}_b \mathbf{G}^H, \quad (39)$$

where \mathbf{G} is a $L_Q \times L_R$ check matrix.

A bit-flipping loop begins when $\mathbf{s}_b \neq \mathbf{s}_a$. The flipping factor $\tau^{(r)}$ is updated as

$$\tau^{(r)} = \begin{cases} \tau^{(r)} + 2\delta, & N^{(r)} = W^{(r)} \\ \tau^{(r)} + \delta, & N^{(r)} = W^{(r)} - 1, \end{cases} \quad (40)$$

where δ is constant, $N^{(r)}$ is the number of unequal equations for each $\mathbf{q}_b^{(r)}$, and $W^{(r)}$ is the r -th column weight of \mathbf{G} .

Then, each $\mathbf{q}_b^{(r)}$ with maximum $N^{(r)}$ is flipped if $\tau^{(r)} > \Delta$ and Δ is a constant threshold. When $\mathbf{q}_b^{(r)}$ is flipped, $\tau^{(r)}$ is updated to be 0.5. Otherwise, $\tau^{(r)} = \tau^{(r)} + \delta$. When none of $\mathbf{q}_b^{(r)}$ is flipped, $\mathbf{q}_b^{(r)}$ with the maximum $\tau^{(r)}$ is flipped and the corresponding $\tau^{(r)} = \tau^{(r)} - 2\delta$.

Finally, the information reconciliation process is completed when $\mathbf{s}_b = \mathbf{s}_a$ or after the set value of iterations.

Denoting $R_{ec} = 1 - L_Q/L_R$ as the LDPC rate, the KGR is defined as the number of candidate key bits generated per channel sounding and is given as

$$KGR = \sum_{i=1}^M \log_2(\mathbf{Q}_i) R_{ec} (1 - \eta). \quad (41)$$

For preprocessing algorithms without interaction, $\eta = 0$, and for PCA algorithm with common eigenvectors, η can be calculated according to (34).

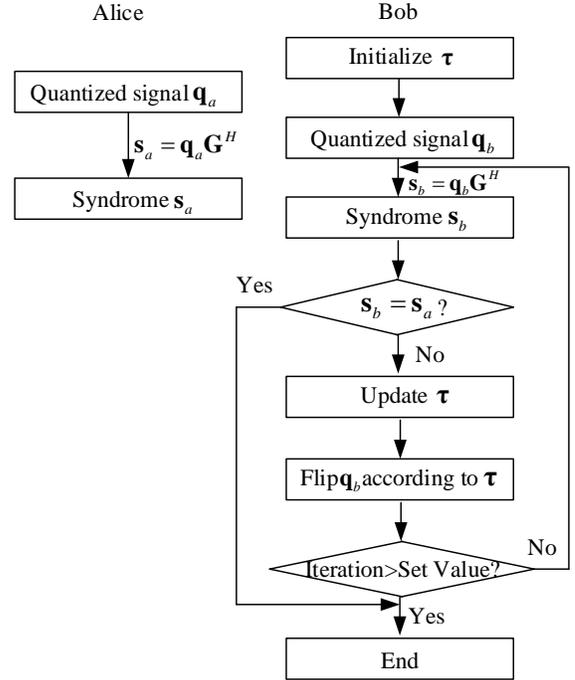


Fig. 2: LDPC-based information reconciliation.

C. Privacy Amplification

Privacy amplification allows legitimate users to distill a shorter but almost completely secret key from a common random variable about which Eve has partial information [32]. Cryptographic hash functions are applied to distill the gathered entropy to a final key. According to the leftover hash lemma, when an adversary only learns about t_{seq} bits of a n_{seq} bits sequence, we can produce a key of $L = n_{seq} - t_{seq}$ bits, over which the adversary has almost no knowledge [33].

We use Message-Digest Algorithm 5 (MD5) in this paper for privacy amplification. MD5 is a widely used hash function which maps data of arbitrary size to data of 128 bits, which is given as

$$g : \{0, 1\}^{n_{seq}} \rightarrow \{0, 1\}^{L=128}. \quad (42)$$

In order to perform the MD5 hash function g , Alice and Bob need to calculate the input sequence length n_{seq} . During the information reconciliation, the syndrome of LDPC transmitted over public channel leaks information to eavesdroppers. In addition, the information leakage ratio of the PCA algorithm is η as analyzed in Section III-C. The length of the secret key, L , can be given as

$$L = n_{seq} R_{ec} (1 - \eta). \quad (43)$$

Thus, in order to produce secret key with a length of L -bit, Alice and Bob should at least generate

$$n_{seq} = \lceil \frac{L}{R_{ec}(1 - \eta)} \rceil \quad (44)$$

bits common random sequence, where $\lceil \cdot \rceil$ represents ceiling operation. For example, when the rate of LDPC used in information reconciliation is $R_{ec} = 1/2$ and the number

TABLE II: SIMULATION PARAMETERS

Parameter	Value
Channel model	SCM
Scenario	Urban-macro
Path number N_p	6
Key length L	128
SNR	10 dB
Iterations of LDPC	200
Carrier frequency f_c	2 GHz
Bandwidth B_w	3.84 MHz
Carrier number F	256
Sampling interval	0.5 ms

of channel samplings $K = 500$, i.e., $\eta \approx 0.002$ for PCA algorithm with common eigenvector, n_{seq} is calculated to be 257.

The input sequence of the leftover hash lemma should be uniform random, otherwise, it will finally result in a weak key. For example, when there are long runs of 0s and 1s in the input, the output sequence of the MD5 function seems to be randomly distributed, but Eve can crack it with dictionary attack. The actual effective length of this key is far less than L , which will need further privacy amplification. Signal preprocessing, such as PCA, DCT and HT, etc., can condense the redundant signals, and therefore can prepare highly random bit sequence for privacy amplification. It is significant and efficient to compress these signals first, rather than processing redundant signals in quantization, information reconciliation and then wiping them out in privacy amplification. In the simulation, we apply the test suite recommended by the National Institute of Standards and Technology (NIST) [34] to verify the randomness of bit sequences before privacy amplification.

V. SIMULATION RESULTS

In this section, we evaluated the performance of the signal preprocessing through numerical simulations. We also compared with key generation without preprocessing, i.e., denoted as “Direct” in the figures. Haar wavelet is used in WT.

3GPP Spatial Channel Model (SCM) is a geometry-based channel model, where the channel parameters are based on statistical distributions extracted from channel measurements [35]. We built our simulation model based on a Matlab implementation of the SCM [36], with detailed parameters summarized in Table II. Alice and Bob are randomly distributed and the distance between them is uniformly distributed in [35, 200] m. We focus on the non line-of-sight (NLOS) scenario.

The decorrelation algorithms investigated in this paper work for any channel measurement with correlation, e.g., frequency correlation in OFDM systems and spatial correlation in MIMO systems. In this paper, we employed OFDM system as a case study. In particular, an OFDM model with 256 subcarriers is utilized. We considered two scenarios: (1) \mathbf{h}_u is constructed by the channel responses of all the 256 subcarriers in one OFDM symbol; (2) \mathbf{h}_u is constructed by combining subcarrier subsets from four continuous OFDM symbols, with each subset selecting one subcarrier out of every four adjacent subcarriers, i.e., the indexes of the selected subcarriers are

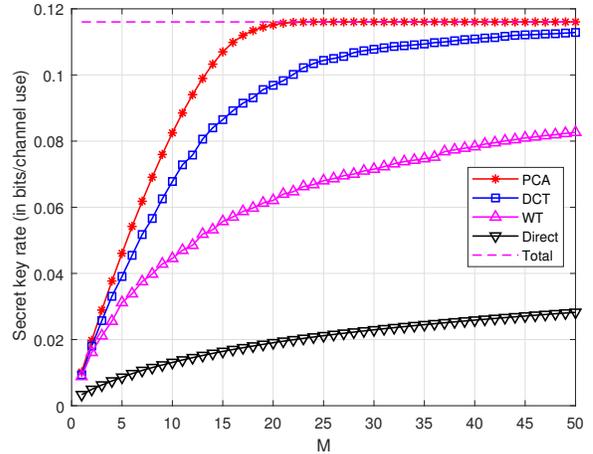


Fig. 3: Effects of the signal preprocessing on the secret key rate, Scenario (1).

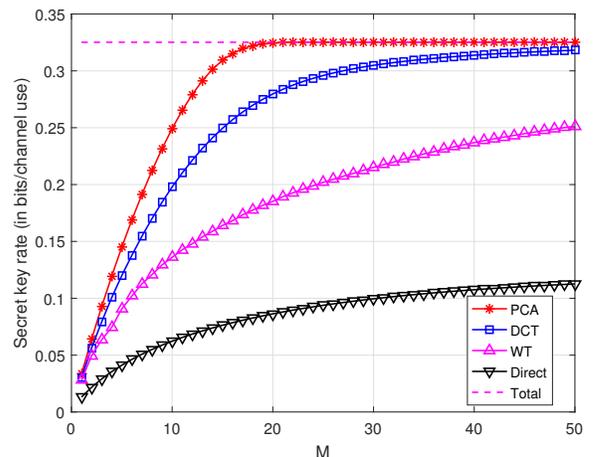


Fig. 4: Effects of the signal preprocessing on the secret key rate, Scenario (2).

$\{4, 8, \dots, 256\}$. The total selected subcarriers are also 256. We generate $K = 500$ independent channel vectors \mathbf{h}_u to calculate the channel covariance matrix.

Fig. 3 and Fig. 4 compare the effects of signal preprocessing on the secret key rate of scenario 1 and scenario 2. We keep the first M dominant components after the preprocessing and calculate the secret key rate. We also put the results of all the N components (“Total”) and M components but without signal preprocessing (“Direct”) as comparison. Theorem 1 has proven that PCA can achieve the optimal performance among all $N \times M$ tall unitary matrix transforms, which is validated by the numerical results here. With PCA, the secret key rate grows rapidly with the increase of M and approaches the total secret key rate even for a small M (e.g., when $M = 20$). This means that a few principal components contain the whole characteristics of the channel information.

We can also notice that Scenario (2) has higher secret key rates from Fig. 3 and Fig. 4. This is because in Scenario (1), we selected all the subcarriers from one OFDM symbol, within

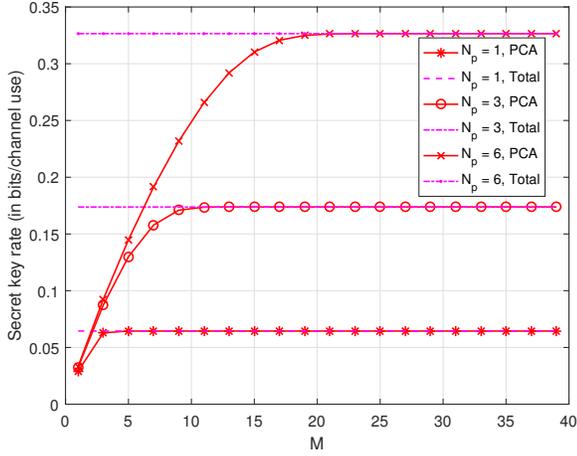


Fig. 5: Secret key rate in different multipath environments.

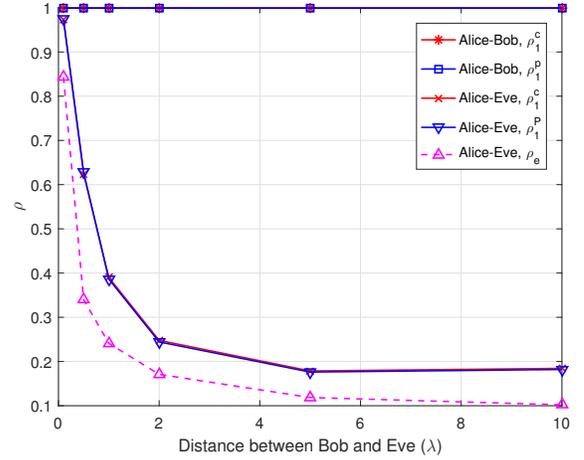


Fig. 7: Correlation coefficients vs. distance between Bob and Eve.

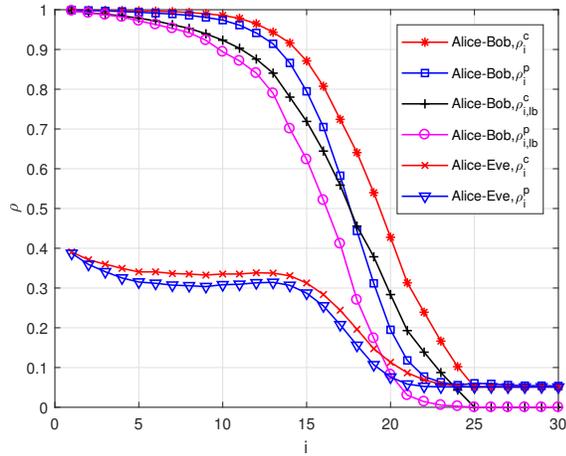


Fig. 6: Correlation coefficients of different PCA components.

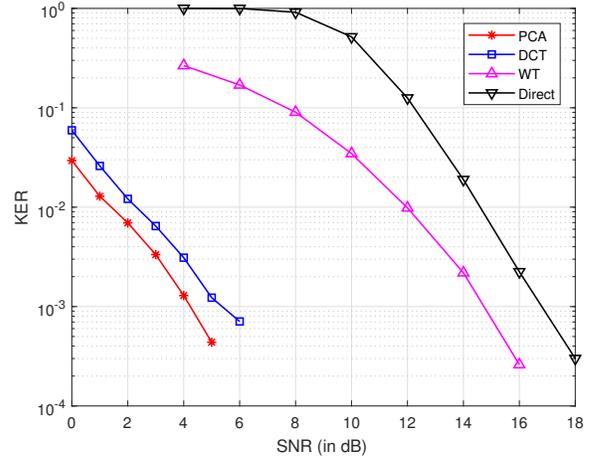


Fig. 8: KER performance comparison.

which there is correlation between any frequency falling into the coherence bandwidth. On the other hand, in Scenario (2), there will be less correlation between the channel responses of the subcarriers from the same symbol, and more randomness is introduced from the time domain by combining subcarriers from four OFDM symbols. Therefore, in the rest of this paper, we constructed \mathbf{h}_u as in Scenario (2).

We further study the secret key rate and dominant components in different multipath environments. As shown in Fig. 5, when the number of paths, N_p , increases, there is more independent information to generate secret key, which leads to the increase of the secret key rate.

Fig. 6 presents the comparison of the correlation coefficients of different components and their lower bounds of PCA with common eigenvector and private eigenvectors. When the principal component index i increases, the correlation coefficients and the lower bounds decrease, because the channel eigenvalues are sorted descending. From Fig. 6, it can be observed that the lower bound of common eigenvector is higher than that of private eigenvectors, which validates the Theorem 2.

We also investigate the eavesdropping channel (between Alice and Eve) with Eve located one wavelength away from Bob is considered as in [13]. As shown in Fig. 6, none of the principal components of the eavesdropping channel has a high correlation coefficient. In addition, we can observe that common eigenvector brings little performance gains to Eve.

Moreover, Fig. 7 presents the correlation coefficients of the first dominant component with different distances between Alice, Bob and Eve. As a comparison, we also illustrate ρ_e which denotes the correlation coefficient between original channel gains of Alice and Eve. When the eavesdropping distance is 10λ , ρ_e decreases to about 0.1, which is quite small. Moreover, it is observed that the coefficients of Eve with common eigenvector ρ_1^c are almost the same with those of private eigenvectors ρ_1^p , which means that the eigenvector transmission does not help eavesdropper.

Fig. 8 shows the KER performance of generated key with PCA, DCT, WT, and Direct. We select $M = 6$ dominant components, and for each component, we set quantization level $Q_i = 2$. Among these preprocessing algorithms, PCA achieves

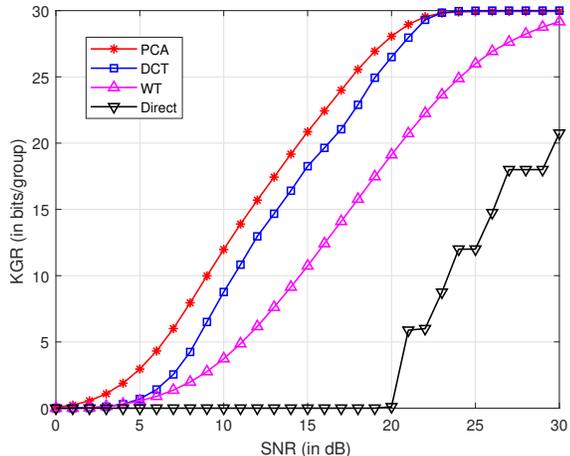


Fig. 9: KGR performance comparison.

the lowest KER approaching 10^{-3} at SNR of 4 dB. The KER of DCT preprocessing approaches that of PCA, with about 1 dB performance loss. Compared with the WT and direct preprocessing algorithms, the performance improved by PCA is larger than 10 dB.

Fig. 9 compares the KGR performance of different preprocessing schemes. We select $M = 6$ components, and the quantization level on each component is selected according to Table I, guaranteeing that the KER of each component is less than 10^{-3} . PCA scheme can reach the highest rate and have about 2 dB performance gains compared with DCT. When SNR is less than 20 dB, the received SNR on each component of Direct scheme cannot reach the lowest threshold, and thus, Direct scheme cannot generate secret key with 10^{-3} KER. In a certain range of SNR, each component of Direct has almost the same quantization level Q_i . Therefore, the KGR growth of Direct is discontinuous and the KGR curve has a stepwise form with the change of SNR. KGR of PCA, DCT, and WT all meets the key refresh requirements of the commercial systems. For example, WiFi recommends to update the key (with a length of 128-bit) once every hour [37].

NIST random test suite is a common tool to evaluate the randomness feature of binary sequences [34], which is also adopted in our work. The output results of each test is an indicator called p-value. A tested sequence passes a test when the p-value is greater than the threshold, usually chosen as 0.01. We perform 9 NIST statistical tests for 10,000 trials. Table III and Table IV show tests pass ratio and the averaged p-value of the 256 bits sequences before privacy amplification, respectively. First, the pass ratios of each specific test are given in Table III. Then, the all-pass ratio represents the number of bit sequences which pass all 9 tests divided by trials number. The maximum number of each row is highlighted in bold. It is observed that, PCA, DCT and HT achieve test pass ratios higher than 0.8, while the pass ratio of Direct is only 0.6.

VI. CONCLUSIONS

The paper has provided a theoretical study of the preprocessing technique in generating high-agreement uncorrelated

TABLE III: NIST STATISTICAL TEST PASS RATIO BEFORE PRIVACY AMPLIFICATION

	Direct	PCA	DCT	WT
Approximate Entropy	0.9250	0.9917	0.9812	0.9812
Runs	0.8562	0.9771	0.9917	0.9729
Ranking	0.9187	0.9146	0.9313	0.9021
Longest runs of ones	0.9521	0.9896	0.9854	0.9812
Frequency	0.8042	0.9896	0.9833	0.9792
FFT	0.9896	0.9979	1.0000	1.0000
Block frequency	0.9771	0.9896	0.9938	0.9938
Cumulative sums	0.9563	1.0000	0.9958	0.9979
Serial	0.8438	0.9833	0.9729	0.9563
All-pass ratio	0.6000	0.8562	0.8521	0.8208

TABLE IV: NIST STATISTICAL TEST P-VALUE BEFORE PRIVACY AMPLIFICATION

	Direct	PCA	DCT	WT
Approximate entropy	0.4080	0.5004	0.4925	0.4877
Runs	0.2510	0.4689	0.4817	0.4291
Ranking	0.4166	0.3753	0.3936	0.3884
Longest runs of ones	0.2919	0.3926	0.3863	0.3837
Frequency	0.2885	0.5048	0.4693	0.4612
FFT	0.4690	0.5852	0.6008	0.5962
Block frequency	0.4827	0.5062	0.4865	0.4920
Cumulative sums	0.5072	0.4996	0.5151	0.5104
Serial	0.2752	0.5049	0.4983	0.4538
	0.4165	0.5190	0.5061	0.4721

secret key from the time-variant OFDM channel. The general mathematical model of preprocessing approaches was first established. Secret key rate expression of preprocessed signals was deduced and it revealed that PCA can achieve the maximum rate among all preprocessing approaches including DCT and WT. Then, the discussion turned to the two specific realization algorithms, PCA with common eigenvector and PCA with private eigenvector. The comparison results proved that sharing the eigenvectors between Alice and Bob can bring in higher key agreement and lower computation expense with negligible information leakage. Finally, a system level design of key generation was proposed, including channel sounding, preprocessing, the flexible level quantization and information reconciliation using bit-flipping based LDPC codes and privacy amplification. Simulation results reconfirmed the optimality of PCA preprocessing and verified the superiority of PCA algorithm with common eigenvector. In the proposed PCA algorithm with common eigenvector, the computational expense of Bob is significantly reduced. Thus, this algorithm can be applied to the scenario where Alice is an access point with strong computational capacity while Bob is a low cost embedded device. Future work will focus on computation overhead reduction of Alice.

APPENDIX A PROOF OF THEOREM 1

Substituting $\mathbf{R}_h = \mathbf{U}_h \mathbf{\Lambda}_h \mathbf{U}_h^H$ into \tilde{R} in (14), we can have

$$\begin{aligned} \tilde{R} = & \log |\mathbf{V}^H \mathbf{U}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I}) \mathbf{U}_h^H \mathbf{V}| \\ & - \log |\mathbf{V}^H \mathbf{U}_h ((\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I}) - \mathbf{\Lambda}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1} \mathbf{\Lambda}_h) \mathbf{U}_h^H \mathbf{V}|. \end{aligned} \quad (45)$$

Define $\mathbf{U}_V = \mathbf{U}_h^H \mathbf{V}$, and thus, $\mathbf{U}_V^H \mathbf{U}_V = \mathbf{I}$. Then, \tilde{R} can be rewritten as

$$\begin{aligned} \tilde{R} &= \log |\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V| \\ &\quad - \log |\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V - \mathbf{U}_V^H (\mathbf{\Lambda}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1} \mathbf{\Lambda}_h) \mathbf{U}_V| \\ &= -\log |(\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V - \mathbf{U}_V^H (\mathbf{\Lambda}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1} \mathbf{\Lambda}_h) \\ &\quad \times \mathbf{U}_V) (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1}| \\ &= -\log \left| \mathbf{I} - (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1/2} \mathbf{\Lambda}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \right. \\ &\quad \left. \times \mathbf{U}_V^H \mathbf{\Lambda}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1/2} \right|. \end{aligned} \quad (46)$$

Let $\tilde{\mathbf{U}} = [\mathbf{U}_V \ \mathbf{U}_{V\perp}]$ be a unitary matrix, where $\mathbf{U}_{V\perp} \in \mathbb{C}^{N \times (N-M)}$ forms a unitary basis for the orthogonal complement of $\text{span}(\mathbf{U}_V)$, and then

$$\begin{aligned} (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1} &= \tilde{\mathbf{U}} (\sigma_n^2 \mathbf{I} + \tilde{\mathbf{U}}^H \mathbf{\Lambda}_h \tilde{\mathbf{U}})^{-1} \tilde{\mathbf{U}}^H \\ &= \tilde{\mathbf{U}} \left(\sigma_n^2 \mathbf{I} + \begin{bmatrix} \mathbf{U}_V^H \\ \mathbf{U}_{V\perp}^H \end{bmatrix} \mathbf{\Lambda}_h [\mathbf{U}_V \ \mathbf{U}_{V\perp}] \right)^{-1} \tilde{\mathbf{U}}^H \\ &= \tilde{\mathbf{U}} \begin{bmatrix} \mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V & \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_{V\perp} \\ \mathbf{U}_{V\perp}^H \mathbf{\Lambda}_h \mathbf{U}_V & \mathbf{U}_{V\perp}^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_{V\perp} \end{bmatrix}^{-1} \tilde{\mathbf{U}}^H. \end{aligned} \quad (47)$$

According to the block matrix inverse formula, we have

$$\begin{aligned} &\begin{bmatrix} \mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V & \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_{V\perp} \\ \mathbf{U}_{V\perp}^H \mathbf{\Lambda}_h \mathbf{U}_V & \mathbf{U}_{V\perp}^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_{V\perp} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} (\mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V)^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \mathbf{M}, \end{aligned} \quad (48)$$

where

$$\begin{aligned} \mathbf{M} &= \begin{bmatrix} -(\mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V)^{-1} \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_{V\perp} \\ \mathbf{I} \end{bmatrix} \mathbf{J}^{-1} \\ &\quad \times \begin{bmatrix} -\mathbf{U}_{V\perp}^H \mathbf{\Lambda}_h \mathbf{U}_V (\mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V)^{-1} & \mathbf{I} \end{bmatrix}, \end{aligned} \quad (49)$$

and

$$\begin{aligned} \mathbf{J} &= \mathbf{U}_{V\perp}^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_{V\perp} - \mathbf{U}_{V\perp}^H \mathbf{\Lambda}_h \mathbf{U}_V \\ &\quad \times (\mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V)^{-1} \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_{V\perp}. \end{aligned} \quad (50)$$

Since $(\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1}$ is positive definite, \mathbf{J} is a positive definite matrix, and then $\mathbf{M} \succeq \mathbf{0}$. Substituting (48) into (47), we have

$$\begin{aligned} (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1} &= \tilde{\mathbf{U}} \left(\begin{bmatrix} (\mathbf{U}_V^H (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h) \mathbf{U}_V)^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \mathbf{M} \right) \tilde{\mathbf{U}}^H \\ &= \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H + \tilde{\mathbf{U}} \mathbf{M} \tilde{\mathbf{U}}^H, \end{aligned} \quad (51)$$

which means $(\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1} \succeq \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H$. The equality holds up if $\mathbf{U}_V = \mathbf{E}$, where $\mathbf{E} = [\mathbf{I}_M \ \mathbf{0}_{(N-M) \times M}]^T$. Let $\tilde{\mathbf{\Lambda}}_h = (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1/2} \mathbf{\Lambda}_h$, and $\tilde{\mathbf{\Lambda}}_h^T = \mathbf{\Lambda}_h (\mathbf{\Lambda}_h + \sigma_n^2 \mathbf{I})^{-1/2}$. Then, we have

$$\tilde{\mathbf{\Lambda}}_h (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1} \tilde{\mathbf{\Lambda}}_h^T \succeq \tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T. \quad (52)$$

Let $\lambda_i(\mathbf{A})$ represent the i th sorted eigenvalue of \mathbf{A} ($\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \dots \geq \lambda_N(\mathbf{A})$), and then

$$\begin{aligned} 1 &\geq \lambda_i \left(\tilde{\mathbf{\Lambda}}_h (\sigma_n^2 \mathbf{I} + \mathbf{\Lambda}_h)^{-1} \tilde{\mathbf{\Lambda}}_h^T \right) \\ &\geq \lambda_i \left(\tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T \right). \end{aligned} \quad (53)$$

Due to

$$\text{Rank} \left(\tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T \right) = M, \quad (54)$$

where $\text{Rank}\{\cdot\}$ denotes the rank of the matrix. Thus, for $i = M+1, M+2, \dots, N$, we have [38]

$$\lambda_i \left(\tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T \right) = 0, \quad (55)$$

and

$$\begin{aligned} &\left| \mathbf{I} - \tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T \right| \\ &= \prod_{i=1}^M \left(1 - \lambda_i \left(\tilde{\mathbf{\Lambda}}_h \mathbf{U}_V (\sigma_n^2 \mathbf{I} + \mathbf{U}_V^H \mathbf{\Lambda}_h \mathbf{U}_V)^{-1} \mathbf{U}_V^H \tilde{\mathbf{\Lambda}}_h^T \right) \right) \\ &\geq \prod_{i=1}^M \left(1 - \lambda_i \left(\tilde{\mathbf{\Lambda}}_h \mathbf{E} (\sigma_n^2 \mathbf{I} + \mathbf{E} \mathbf{\Lambda}_h \mathbf{E})^{-1} \mathbf{E}^H \tilde{\mathbf{\Lambda}}_h^T \right) \right) \\ &= \left| \mathbf{I} - \tilde{\mathbf{\Lambda}}_h \mathbf{E} (\sigma_n^2 \mathbf{I} + \mathbf{E}^H \mathbf{\Lambda}_h \mathbf{E})^{-1} \mathbf{E}^H \tilde{\mathbf{\Lambda}}_h^T \right|. \end{aligned} \quad (56)$$

Thus, the maximal \tilde{R} is achieved when $\mathbf{U}_V = \mathbf{E}$. The optimal transform matrix $\mathbf{V} = [\mathbf{u}_h^{(1)}, \mathbf{u}_h^{(2)}, \dots, \mathbf{u}_h^{(M)}]$.

APPENDIX B PROOF OF THEOREM 2

For the common eigenvectors, we can calculate the covariance of the received signals as

$$E \{y_{ai} y_{ai}^*\} = (\mathbf{u}_a^{(i)})^H \mathbf{R}_{h_a} \mathbf{u}_a^{(i)} = \lambda_i, \quad (57a)$$

$$E \{y_{ai} y_{bi}^*\} = (\mathbf{u}_a^{(i)})^H E \{ \mathbf{h}_a (\mathbf{h}_a + \mathbf{n}_d)^H \} \mathbf{u}_a^{(i)} = \lambda_i, \quad (57b)$$

$$\begin{aligned} E \{y_{bi} y_{bi}^*\} &= (\mathbf{u}_a^{(i)})^H (\mathbf{R}_{h_a} + \mathbf{R}_{n_d}) \mathbf{u}_a^{(i)} \\ &= \lambda_i + (\mathbf{u}_a^{(i)})^H \mathbf{R}_{n_d} \mathbf{u}_a^{(i)}. \end{aligned} \quad (57c)$$

Let $\delta = \max_i (\mathbf{u}_a^{(i)})^H \mathbf{R}_{n_d} \mathbf{u}_a^{(i)}$, and the correlation coefficient is derived as

$$\rho_i^c = \sqrt{\frac{\lambda_i}{\lambda_i + (\mathbf{u}_a^{(i)})^H \mathbf{R}_{n_d} \mathbf{u}_a^{(i)}}} \geq \sqrt{\frac{\lambda_i}{\lambda_i + \delta}} \triangleq \rho_{i,lb}^c. \quad (58)$$

For the private eigenvectors, the covariance of the received signals can be expressed as

$$E \{y_{ai} y_{ai}^*\} = (\mathbf{u}_a^{(i)})^H \mathbf{R}_{h_a} \mathbf{u}_a^{(i)} = \lambda_i, \quad (59a)$$

$$\begin{aligned} E \{y_{ai} y_{bi}^*\} &= (\mathbf{u}_a^{(i)})^H E \{ \mathbf{h}_a (\mathbf{h}_a + \mathbf{n}_d)^H \} \mathbf{u}_b^{(i)} \\ &= (\mathbf{R}_{h_a} \mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)} = \lambda_i (\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}, \end{aligned} \quad (59b)$$

$$\begin{aligned} E \{y_{bi} y_{bi}^*\} &= (\mathbf{u}_b^{(i)})^H (\mathbf{R}_{h_a} + \mathbf{R}_{n_d}) \mathbf{u}_b^{(i)} \\ &= (\mathbf{u}_b^{(i)})^H \mathbf{R}_{h_a} \mathbf{u}_b^{(i)} + (\mathbf{u}_b^{(i)})^H \mathbf{R}_{n_d} \mathbf{u}_b^{(i)}. \end{aligned} \quad (59c)$$

Since $\mathbf{R}_{h_a} = \mathbf{U}_a \mathbf{\Lambda}_a \mathbf{U}_a^H = \sum_i \lambda_i \mathbf{u}_a^{(i)} (\mathbf{u}_a^{(i)})^H$, we can have

$$\begin{aligned} \rho_i^p &\geq \sqrt{\frac{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2}{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2 + \sum_{j \neq i} \lambda_j |(\mathbf{u}_a^{(j)})^H \mathbf{u}_b^{(i)}|^2 + \delta}} \\ &\triangleq \rho_{i,lb}^p. \end{aligned} \quad (60)$$

With the expressions (58) and (60), we can have

$$\frac{\rho_{i,lb}^c}{\rho_{i,lb}^p} = \sqrt{\frac{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2 + \delta + \sum_{j \neq i} \lambda_j |(\mathbf{u}_a^{(j)})^H \mathbf{u}_b^{(i)}|^2}{\lambda_i |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2 + \delta |(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2}}. \quad (61)$$

As $\sum_{j=1}^N |(\mathbf{u}_a^{(j)})^H \mathbf{u}_b^{(i)}|^2 = |\mathbf{u}_b^{(i)}|^2 = 1$, $|(\mathbf{u}_a^{(i)})^H \mathbf{u}_b^{(i)}|^2 \leq 1$. In addition, with $\sum_{j \neq i} \lambda_j |(\mathbf{u}_a^{(j)})^H \mathbf{u}_b^{(i)}|^2 \geq 0$, We can derive

$$\frac{\rho_{i,lb}^c}{\rho_{i,lb}^p} \geq 1. \quad (62)$$

REFERENCES

- [1] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5G wireless networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 1617–1655, Third Quarter 2016.
- [2] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, "A survey on wireless security: Technical challenges, recent advances, and future trends," *Proc. IEEE*, vol. 104, no. 9, pp. 1727–1765, Sep. 2016.
- [3] C. Cheng, R. Lu, A. Petzoldt, and T. Takagi, "Securing the Internet of Things in a quantum world," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 116–120, Feb. 2017.
- [4] J. Zhang, T. Q. Duong, A. Marshall, and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, Mar. 2016.
- [5] U. M. Maurer, "Secret key agreement by public discussion from common information," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 733–742, 1993.
- [6] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: extracting a secret key from an unauthenticated wireless channel," in *ACM International Conference on Mobile Computing and Networking*, San Francisco, California, USA, Sep. 2008, pp. 128–139.
- [7] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. on Computing*, vol. 38, no. 1, pp. 97–139, 2008.
- [8] U. Maurer and S. Wolf, "Secret-key agreement over unauthenticated public channels. ii. privacy amplification," *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 839–851, Apr. 2003.
- [9] S. Yasukawa, H. Iwai, and H. Sasaoka, "Adaptive key generation in secret key agreement scheme based on the channel characteristics in OFDM," in *Proc. Int. Symp. Information Theory and its Applications*, Auckland, New Zealand, Dec. 2008, pp. 1–6.
- [10] H. Liu, Y. Wang, J. Yang, and Y. Chen, "Fast and practical secret key extraction by exploiting channel response," in *Proc. 32nd IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Turin, Italy, Apr. 2013, pp. 3048–3056.
- [11] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong, "Efficient key generation by exploiting randomness from channel responses of individual OFDM subcarriers," *IEEE Trans. Commun.*, vol. 64, no. 6, pp. 2578–2588, Jun. 2016.
- [12] Y. Peng, P. Wang, W. Xiang, and Y. Li, "Secret key generation based on estimated channel state information for TDD-OFDM systems over fading channels," *IEEE Trans. Wireless Commun.*, 2017.
- [13] J. W. Wallace and R. K. Sharma, "Automatic secret keys from reciprocal MIMO wireless channels: Measurement and analysis," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 381–392, Sep. 2010.
- [14] K. Zeng, D. Wu, A. Chan, and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *Proc. 29th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, San Diego, California, USA, Mar. 2010, pp. 1–9.
- [15] C. Chen and M. A. Jensen, "Secret key establishment using temporally and spatially correlated wireless channel coefficients," *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 205–215, Feb. 2011.
- [16] N. Patwari, J. Croft, S. Jana, and S. K. Kasper, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *IEEE Trans. Mobile Comput.*, vol. 9, no. 1, pp. 17–30, Jan. 2010.
- [17] G. Margelis, X. Fafoutis, G. Oikonomou, R. Piechocki, T. Tryfonas, and P. Thomas, "Physical layer secret-key generation with discrete cosine transform for the Internet of Things," in *Proc. IEEE ICC*, Paris, France, May 2017.
- [18] Y. Wu, Y. Sun, L. Zhan, and Y. Ji, "Low mismatch key agreement based on wavelet-transform trend and fuzzy vault in body area network," *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1–16, 2013.
- [19] F. Zhan and N. Yao, "On the using of discrete wavelet transform for physical layer key generation," *Ad Hoc Networks*, vol. 64, pp. 22–32, Sep. 2017.
- [20] C. Sahin, B. Katz, and K. R. Dandekar, "Secure and robust symmetric key generation using physical layer techniques under various wireless environments," in *Proc. IEEE Radio and Wireless Symposium*, Austin, TX, USA, Jan. 2016, pp. 211–214.
- [21] S. Gopinath, R. Guillaume, P. Duplys, and A. Czulwik, "Reciprocity enhancement and decorrelation schemes for phy-based key generation," in *Proc. IEEE GLOBECOM Workshop Trusted Commun. with Physical Layer Security (TCPLS)*, Austin, Texas, USA, Dec. 2014, pp. 1367–1372.
- [22] J. Zhang, R. Woods, T. Q. Duong, A. Marshall, and Y. Ding, "Experimental study on channel reciprocity in wireless key generation," in *Proc. 17th IEEE Int. Workshop Signal Process. Advances in Wireless Commun. (SPAWC)*, Edinburgh, UK, Jul. 2016, pp. 1–5.
- [23] J. Zhang, B. He, T. Q. Duong, and R. Woods, "On the key generation from correlated wireless channels," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 961–964, Apr. 2017.
- [24] G. Li, A. Hu, L. Peng, and C. Sun, "The optimal preprocessing approach for secret key generation from OFDM channel measurements," in *Proc. IEEE GLOBECOM Workshop Trusted Commun. with Physical Layer Security (TCPLS)*, Washington DC, Dec. 2016, pp. 1–6.
- [25] J. Zhang, R. Woods, T. Q. Duong, A. Marshall, Y. Ding, Y. Huang, and Q. Xu, "Experimental study on key generation for physical layer security in wireless communications," *IEEE Access*, vol. 4, pp. 4464–4477, 2016.
- [26] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan 1974.
- [27] Y. Wu, Y. Sun, L. Zhan, and Y. Ji, "Low mismatch key agreement based on wavelet-transform trend and fuzzy vault in body area network," *International Journal of Distributed Sensor Networks*, vol. 2013, no. 2, pp. 1614–1617, 2013.
- [28] C. Zenger, H. Vogt, J. Zimmer, A. Sezgin, and C. Paar, "The passive eavesdropper affects my channel: Secret-key rates under real-world conditions," in *Proc. IEEE Globecom Workshops*, Washington DC, USA, Dec. 2016, pp. 1–6.
- [29] B. T. Quist and M. A. Jensen, "Maximization of the channel-based key establishment rate in MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 14, no. 10, pp. 5565–5573, Oct. 2015.
- [30] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Trans. Inf. Theory*, vol. 49, no. 3, pp. 626–643, 2003.
- [31] M. Sartipi and F. Fekri, "Distributed source coding in wireless sensor networks using LDPC coding: The entire Slepian-Wolf rate region," in *IEEE Wireless Communications and Networking Conference*, New Orleans, LA, USA, Mar. 2005, pp. 1939–1944.
- [32] C. Cachin and U. M. Maurer, "Linking information reconciliation and privacy amplification," *Journal of Cryptology*, vol. 10, no. 2, pp. 97–110, 1997.
- [33] R. Impagliazzo, L. A. Levin, and M. Luby, "Pseudo-random generation from one-way functions," in *Proc. 21st Annu. ACM Symp. Theory of Computing*, Seattle, Washington, USA, May 1989, pp. 12–24.
- [34] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," National Institute of Standards and Technology, Tech. Rep. Special Publication 800-22 Revision 1a, Apr. 2010.
- [35] 3rd Generation Partnership Project. TR 25.996 V10.0.0, "Spatial channel model for multiple input multiple output (MIMO) simulations," 2011.
- [36] J. Salo, G. D. Galdo, J. Salmi, P. Kyösti, M. Milojevic, D. Laselva, and C. Schneider, "MATLAB implementation of the 3GPP Spatial Channel Model (3GPP TR 25.996)," 2005. [Online]. Available: <http://www.tkk.fi/Units/Radio/scm/>
- [37] T. Moore, "802.1X and 802.11 key interactions," Microsoft, Tech. Rep. doc.: IEEE 802.11-01/610r02, Nov. 2001.
- [38] G. A. F. Seber, *A matrix handbook for statisticians*. Wiley Interscience, 2007.



Guyue Li received the B.S. degree in Information Science and Technology and the Ph.D. degree in Information Security from Southeast University, Nanjing, China, in 2011 and 2017, respectively.

She is currently a Lecturer at Southeast University. Her research interests include physical layer security, beamforming, artificial noise and blind source separation.



Daming Cao received the B. Eng. degree in information engineering from Southeast University, Nanjing, China, in 2013.

He is currently a Ph.D student in Southeast University. His research mainly focuses on information theory and security.



Aiqun Hu received the B.Sc.(Eng.), the M.Eng.Sc. and Ph.D. degrees from Southeast University in 1987, 1990, and 1993 respectively.

He was invited as a post-doc research fellow in The University of Hong Kong from 1997 to 1998, and TCT fellow in Nanyang Technological University in 2006. His research interests include data transmission and secure communication technology. He has published two books and over 100 technical papers in wireless communications field.



Junqing Zhang received the B.Eng and M.Eng degrees in Electrical Engineering from Tianjin University, China in 2009 and 2012, respectively, and the Ph.D degree in Electronics and Electrical Engineering from Queen's University Belfast, UK in 2016.

From 2016 to 2017, he was a Postdoctoral Research Fellow with Queen's University Belfast, UK. Since 2018, he has been a Tenure Track Fellow with University of Liverpool, UK. His research interests include wireless security, physical layer security and

key generation.



Linning Peng received his PhD degree from the Electronics and Telecommunications Institute of Rennes laboratory at the National Institute of Applied Sciences of Rennes, France, in 2014.

From 2014, he has been a Research Associate at the Southeast University, Nanjing, China. His research interests include the design and optimization of digital communication systems and physical layer security in wireless communications.



Chen Sun (S'13) received the B.E. degree in electrical engineering from Southeast University, Nanjing, China, in 2011.

Currently he is working towards the Ph.D. degree in the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China. From September 2015 to August 2016, he was a visiting student in the Department of Electrical and Computer Engineering, University of California, Davis, USA. His research interests include communications, and information theory, with emphasis on

massive MIMO communications.