High Entropy Secrecy Generation from Wireless CIR

Hao Li, Chen Shen, Yanxiang Zhao, Gokhan Sahin, Hyeong-Ah Choi, and Yogendra Shah

Abstract: The physical characteristics of wireless channels, in particular the channel impulse response (CIR), have become an attractive source of shared secrecy between two communicating parties for many wireless applications. However, ensuring high-entropy secret bits from the CIR may not be always feasible, including in IoT and/or indoor environments, where the coherence time may not be easy to obtain, or mobility is limited. Many techniques have been developed to resolve the above problems, including privacy amplification for higher entropy output, and pre-processing for data distillation. In this paper, we develop a framework for extracting entropy from sequentially arriving channel impulse response observation samples, and propose algorithms for both online and offline versions of the problem. We fully characterize the optimal solutions to these problems through a series of analytic results, and establish the optimality of our proposed online and offline algorithms, namely $A_{\text{EMSON}}$ and $A_{\text{EMSOFF}}$ to solve the problem. We also verify the performance of the algorithms through comparisons with the existing related techniques, and evaluate their capability to ensure high-entropy under different channel models using MATLAB simulations.

Index Terms: Entropy extraction, physical layer security, secrecy generation, wireless networks.

I. INTRODUCTION

WIRELESS mobile communication systems are characterized by multipath fading, where a signal can travel from a transmitter to a receiver over multiple reflective paths, resulting in fluctuations in the amplitude, phase, and angle of arrival of the received signal due to reflection, diffraction, and scattering [1], [2]. These effects, under commonly used assumptions, lead to the modeling of a wireless channel as a linear time variant (LTV) system relating the output (the received signal) to the input (the transmitted signal). Thus, the time-varying channel impulse response (CIR), or alternatively, the (time varying) channel frequency response could be used to characterize the overall physical link between the transmitter and the receiver. Accordingly, there is a vast amount of literature on estimation and modeling of wireless CIRs.

The channel impulse response and the channel state information have attractive properties that may be utilized in secrecy generation between two communicating parties [3]–[6]. A key property for shared secrecy is data randomness often characterized by the Shannon Entropy. However, in many IoT applications, the channel environment is nearly static or communicating parties may move slowly, which may result in significantly low entropy in the CIR data measured during a short period of time [7], [8]. It may also be possible for the CIR data observed for a longer period of time or combined from multiple monitoring periods to still have very low entropy. The entropy of the data obtained from CIR measurements also depends on the sampling interval used with respect to the Doppler frequency, or the channel coherence time. Thus, it becomes difficult to extract high entropy from CIR data in situations where the Doppler frequency or the channel coherence time is unknown or difficult to estimate [9].

Our main problem of interest is to extract high entropy data bits from random sources such as the aforementioned CIR data. We address both online and offline versions of this problem. The online version is described as: Given a set of data points arriving sequentially one at a time, our goal is to make an online decision whether to keep or discard a data point upon its arrival so that we always keep a data subset with a pre-determined size that has the maximum achievable entropy, or a minimum predetermined entropy. The offline version is described as to find a subset of given input data points that maximizes the subset size while its entropy satisfies a given threshold value and select the set with the highest entropy if there are multiple such subsets. We note that there are many sources of random data but the modeling of the data source and the ability to extract the randomness remains a challenge due to the underlying complexity of establishing an entropy model. Therefore, there is an opportunity to utilize a technique for adaptively extracting randomness from a potential source of data given a desired entropy. The framework developed in this paper may be useful in this broader context.

In Section II, we give the problem background and the current related works. We then present the wireless system model considered in formulating our problems followed by the notations used throughout this paper in Section III. In Section IV, we present a series of mathematical results that characterize the structure of the entropy maximizing subsets and that support the key ideas in developing our algorithms. Section V presents our algorithms with their optimality and complexity analysis. Performance analysis of our algorithms using the MATLAB toolbox is then presented in Section VI, followed by a comparison with the entropy obtained using the SHA-2 privacy amplification method. In Section VII, we provide a conclusion to the paper.

Manuscript received April 28, 2018; approved for publication by Zhu Han, Division III Editor, January 30, 2019.

H. Li, C. Shen, and H.-A. Choi are with the Department of Computer Science, The George Washington University, email: {haoli, shchen, hchoi}@gwu.edu.

Y. Zhao is with the Department of Mathematics, The George Washington University, email: yxzhaow@gwu.edu.

G. Sahin is with the Department of Electrical & Computer Engineering, Miami University, USA, email: sahing@miamioh.edu.

Y. Shah is an independent consultant with expertise in security and wireless systems, USA, email: yshah@ieee.org.

G. Sahin is the corresponding author.

Digital Object Identifier: 10.1109/JCN.2019.000019

© 2019 KICS

Creative Commons Attribution-NonCommercial (CC BY-NC).
This is an Open Access article distributed under the terms of Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided that the original work is properly cited.
The complete mathematical proofs of the theorems in this paper are given in the Appendix.

II. BACKGROUND AND RELATED WORK

We consider the physical radio channel between two communicating nodes as a source of entropy. In this paper we illustrate how a random bit stream may be extracted from the physical layer radio channel and to be able to measure the entropy of the resulting bit stream. Numerous mechanisms that make use of the key physical layer characteristics have been proposed for enhancing security in wireless networks, including 802.11 [10], [11]. Extraction of shared physical layer secrecy inherently involves interactions among a number of processing blocks such as quantization, information reconciliation, and privacy amplification [8]. If high entropy bits can be obtained after the quantization stages through sub-sampling or correlation-reduction techniques such as [5] or the methods proposed in this paper, the need for entropy extraction may be alleviated.

The conventional methods for obtaining high entropy mainly use the universal secure hash function, for which the output has been shown to be random enough when the input length is at least twice as big as that of the output [12]. Most work in privacy amplification of physical secrecy generation applies this method [9]. Some work [8] adds an extra layer of data distillation before applying the universal hash function. Our main contribution is to achieve high entropy output by source data selection instead of hashing. Our selection algorithm can approach the theoretical optimality for a given output size by strict mathematical proof. We compare the performance of our proposed algorithm with one of the popular secure hash functions, SHA-256 [12] which generates 256-bits hash code for an input of any length, in different scenarios under the specific settings. The simulation result shows that our algorithm outperforms SHA-256 when the data source has high entropy, or, when the channel is changing rapidly. There is another advantage of selection over hashing for higher entropy, which is the requirement of identical data pairs for the former. To guarantee that the hashing will work correctly, the data pair after information reconciliation have to be identical between the two sides. In contrast, data with imperfect matching can still use our selection algorithms for entropy enhancement, which may be helpful in authentication with imperfect shared keys [13]. Multiple works have pointed out the limitation of entropy gain by channel measurement probes when the channel is relatively constant [3], [9], [14]. The work in [15] provides an active method to increase the channel variation through sophisticated antenna manipulations, which may expand the application range of our entropy selection algorithm to cases where the channel does not change much. In [16], [17], secret key extraction from received signal strength is considered in vehicular environments, addressing the challenges of obtaining matching measurements despite extremely low coherence time. Techniques including data smoothing, online parameter learning, and randomness extraction are combined to address the challenges of this specific scenario. Our sample selection algorithm also differs from this work in that ours does not require any training period. Besides the received signal strength used in our simulation, some other work [18] proposed using channel phase information for better security and higher entropy of source data, since the signal strength is more related to the distance, which can be predictable to adversary, and the phase data is more likely to be evenly distributed. We note that our proposed entropy extraction algorithm is universally applicable in these cases. Due to the vast availability of the signal strength information and for simplification of the simulations, we only analyze the performance of the algorithm based on signal strength in simulations.

III. PROBLEM FORMULATION

In this section, we briefly describe the wireless system model we have in mind, as well as the process we follow to generate the bit streams from physical CIR data. We consider a time-division-duplexed (TDD) wireless system where the two sides of the channels take CIR measurements and convert them into bits following quantization, and possibly other processes. If multi-level quantization is employed, each quantized measurement can be considered as an input from an alphabet, as defined in section III.A, with an alphabet size equal to the number of quantization levels. Alternatively, codewords can be constructed from a stream of CIR-based bits using various approaches. Without loss of generality, in our simulations, the bits obtained from CIR measurements are converted into codewords of a fixed size (chosen as 8 bits in our simulations) by concatenating adjacent bits, and these serve as the input sequence \((x_1, x_2, \cdots)\) to our entropy maximizing subset selection problem. Thus, consecutive sets of 8-bit groups are mapped to a codeword in the alphabet. The codewords generated in the experiments are then counted to create a histogram Fig. 2 that is illustrative of the subsequent Theorems presented in Section III.

It is worth noting that our online and offline high entropy subset selection framework is general enough to work with any source of shared bit stream, even though we specifically refer to CIR-based secrecy generation. Our framework is also independent of the specific quantization schemes used to convert CIR measurements to bits, including those based on the level-crossing or censoring approaches [7], [14], [19] that have been proposed in physical layer security related work.

In our simulations, we consider a multi-path wireless channel, where the sum of all path gains in a CIR measurement is converted into a single bit depending on whether it is above or below the mean for the sum. These bits are serialized over time, forming a bit sequence. We then convert the bits in this sequence into words by concatenating a fixed number of adjacent bits (groups of 8 bits in our simulations).

A. Notations

Let \(A = \{a_1, a_2, \cdots, a_k\}\), called an alphabet, be a set of \(k\) distinct positive numbers called symbols. An input \(X = \{x_1, x_2, \cdots, x_n\}\) is defined as a set of \(n\) positive real numbers such that for \(1 \leq i \leq n\), each \(x_i \in A\). (In the discussion of an online problem formulation and its algorithm, input \(x_t\) is arriving one at a time for \(t = 1, 2, \cdots\)). For any subset \(X' \subseteq X\), define \(c_i(X'), 1 \leq i \leq k\), to denote the number of times \(a_i\) appears in \(X'\), and \(p_i(X'), 1 \leq i \leq k\), to denote the frequency of \(a_i\) appearing in \(X'\), i.e., \(p_i(X') = c_i(X')/|X'|\). When \(X' = X\),
Given a set $X$, the entropy of $X$ is defined as

$$H(X) = -\sum_{i=1}^{k} p_i(X) \cdot \log p_i(X).$$

We then observe the following result.

**Lemma 1:** Let $X_e = \{a_1, a_2, \ldots, a_k\}$ (i.e., each $a_i$ from the alphabet of size $k$ appears exactly once in $X_e$). Then, $H(X_e) = \log k$.

**Proof:** Note that $H(X_e) = -\sum_{i=1}^{k} \frac{c_i}{k} \log \frac{c_i}{k}$, which is $\log k$. It is noted that when each element $a_i$ $(1 \leq i \leq k)$ appears exactly $p$ times in $X_e$ for any integer $p \geq 1$, the same result still holds, i.e., $H(X_e) = \log k$.

**B. Problem Formulation**

The entropy maximizing subset problem is formulated in two versions: online and offline versions.

**B.1 Entropy Maximizing Subset: Online**

The **Entropy Maximizing Subset**: **Online** (EMS\textsc{On}) problem, an online formulation, is described as: **Given a set of input symbols arriving sequentially one at a time, our goal is to make an online decision whether to keep or discard a data point upon its arrival so that we always have a data set with a predetermined size that has the maximum entropy possible.** This problem is formally defined as follows.

Given an input $X = \{x_1, x_2, \ldots\}$ with each $x_t$ sequentially arriving one at a time where $x_t \in A = \{a_1, \ldots, a_k\}$ and also given an arbitrary integer $\beta \geq 1$, we want to find a subset $X(\ell) \subseteq \{x_1, \ldots, x_\ell\}$, for each $\ell$, by making an online decision when $x_t$ arrives for each $t \geq \beta$ such that $|X(t)| = \beta$ and the entropy of $X(t)$ is maximum among all possible subsets of $\{x_1, \ldots, x_\ell\}$ with size $\beta$.

**B.2 Entropy Maximizing Subset: Offline**

We are given an input $X = \{x_1, x_2, \ldots, x_n\}$ where each $x_i \in A = \{a_1, \ldots, a_k\}$ and an arbitrary number $\alpha$, $0 < \alpha \leq 1$. We define the **Entropy Maximizing Subset**: **Offline** (EMS\textsc{Off}) problem, an offline formulation, to find

$$\arg\max_{X_0 \in D} H(X_0),$$

where $D$ is the set of all maxima to the problem, such that

$$D := \arg\max_{X' \subseteq X: H(X') \geq \alpha \log k} |X'|.$$

In other words, we want to find a largest subset $X_0 \subseteq X$ such that $H(X_0) \geq \alpha \cdot H_{\text{max}}(X)$ and if there are multiple such subsets with the same size, $X_0$ must then be the one with the largest entropy.

**IV. CHARACTERIZATION OF ENTROPY MAXIMIZING SUBSET**

In this section, we present some mathematical results characterizing the input $X$ or its subset in terms of its entropy. These results will be used in the design of our algorithms for both online and offline formulations, namely the **EMS\textsc{On}** and **EMS\textsc{Off}** problems. We note that even though both offline and online algorithms appear to be intuitive, the optimality proof of each algorithm requires rigorous analysis. One of the main contributions in this paper is to justify the optimality through a series of necessary theorems as presented. We begin with the following lemma which will be used in Theorem 1, a key result in proving the optimality of our online algorithm A_\textsc{EMS\textsc{On}} discussed later. Theorem 1 will be also used to characterize the shape of any optimal solution to the **EMS\textsc{Off}** problem discussed in Theorem 2. All the proofs of the theorems are given in the Appendix part. We note that our entropy maximizing set analyses are conducted in a general sense. It is plausible that general mathematical theories that would directly lead to the same results may exist, although no such theories are known by the authors.

**Lemma 2:** Given any two numbers $x$ and $y$ such that $y \leq x$ and $x, y \geq 1$,

$$\frac{(x-1)^{y-1}(y+1)^{x+1}}{x^y y^x} < 1.$$

**Proof:** Consider the function

$$f(s) = \frac{(s-1)^{s-1}}{s^s},$$

which is easy to verify to be a decreasing function when $s \geq 1$. Therefore, for the given $x, y$, one has $y + 1 \leq x - 1 < x$, and

$$f(x) \leq f(y + 1),$$

which is the desired inequality.

**Theorem 1:** Let $X$ be an input with $c_1 \geq c_2 \geq \cdots \geq c_k$. (Consider $X$ as a staircase shape as depicted in Fig. 1(a).) Define $X' = X \cup \{a_p\}$, in which $a_p$ and $a_q$ are two symbols (see Fig. 1(b)) where $a_p$ (a green block) is located at the corner of the top row (where $a_p \in X$) and $a_q$ (a red block) is located on top of a middle row (where $a_q \in X$). We then claim that the entropy of $X'$ is larger than the entropy of $X$.

**Remarks:** Note that symbol $a_p$ occurs more than symbol $a_q$ in $X$. Therefore, this theorem essentially states that the entropy of any given set is increased by replacing a symbol with a higher frequency, with a symbol that has a lower frequency. The complete proof is given in Appendix VII.

We next proceed to discuss the shape of any optimal solution to the **EMS\textsc{Off}** problem.

**Theorem 2:** Any optimal solution $X_0 \subseteq X$ with total $r$ rows to the **EMS\textsc{Off}** problem is of the form depicted in Fig. 2 in which all symbols of $X$ in row 1 through $r - 1$ (for some $r, 1 < r \leq c_1$) must be included in $X_0$ while all or some of symbols in row $r$ may be included in $X_0$. We next present three additional results, Theorems 3-5 to conclude a key result, Theorem 6, that will be used to prove the correctness and then we analyze the complexity of our algorithm A_\textsc{EMS\textsc{Off}} discussed later.
Theorem 3: Let $X$ be an input set and $X_1 = X \cup \{a_1\}$. (See Fig. 3.) We then claim that $H(X_1) < H(X)$.

Theorem 4: Let $X$ be an input set and $X_1$ be obtained from $X$ by adding an entire row above the top row. (See Fig. 4.) We then claim that $H(X_1) < H(X)$.

Remarks: It should be noted that when Condition (12) is relaxed to as

$$c_1 \geq c_2 \geq \cdots \geq c_k,$$

the proof of Theorem 4 can be easily modified to show $H(X_1) \leq H(X)$. We next discuss Theorem 5 that shows how the entropy changes when adding more symbols. More precisely, consider $X$ represented as a staircase shape as discussed earlier and let $X_{r-1} \subseteq X$ be the set of all symbols upward from row 1 (i.e., the bottom row) through row $r-1$ of $X$. Now consider symbols on row $r$, and let $X_{r-1} + j$ be a set defined by adding $j$ symbols on row $r$ to $X_{r-1}$. (We can assume such $j$ symbols are left-most without loss of any generality in the context of the theorem.) Fig. 5 shows $X_{r-1} + j$ where the white areas only shows $X_{r-1}$.
Theorem 5: For some \( j_0 (1 \leq j_0 \leq l_r) \), \( H(X_{r-1} + j) < H(X_{r-1} + j - 1) \) when \( j \leq j_0 \) and \( H(X_{r-1} + j) > H(X_{r-1} + j - 1) \) when \( j > j_0 \), in which \( l_{r-1} \) denotes the length of row \( r - 1 \).

Remarks: This theorem states that \( H(X_{r-1} + j) \) decreases as \( j \) increases to a certain point \( j_0 \) (i.e., a point with minimum entropy) and then increases as \( j \) continues to increase until \( j = l_{r-1} \). As a simple case, consider a case that the frequency of every symbol in \( X_{r-1} \) is same, i.e., the entropy of \( X_{r-1} \) is \( \log k \), the maximum possible. It is easy to see that \( H(X_{r-1} + 1) < H(X_{r-1}) \) and also \( H(X_{r-1} + k - 1) < H(X_{r-1}) \); hence, as \( j \) increases from 1 to \( k - 1, H(X_{r-1} + j) \) should decrease and then increase. Fig. 6 illustrates this case.

Fig. 6. \( H(X_{r-1} + j) \) is first decreasing to a minimal point then increasing until all \( l_{r-1} \) symbols are added.

Fig. 6 depicts such behavior of the entropy stated in Theorem 5. It should be noted that \( H(X_{r-1} + 1) \) may be smaller or equal to \( H(X_{r-1} + l_{r-1}) \).

From Theorems 3-5, the following result is now established:

**Theorem 6:** Given an input \( X \) laid as a staircase shape, \( H(X_t) \leq H(X_{r-1}) \) for \( 1 < r \leq c_1 \).

V. ALGORITHMS

In this section, we present two optimal algorithms: \( A\_EMSON \) for EMSON and \( A\_EMSOFF \) for EMSOFF. Each algorithm is developed based on the theorems presented in the previous section.

We assume that the input \( X \) is always configured as a staircase shape as discussed in the previous section. (See Fig. 1(a) for example.)

A. \( A\_EMSON \)

We will outline the algorithm in a recursive manner. Initially, we keep the first \( \beta \) data points, i.e, \( X(\beta) = \{x_1, \cdots, x_\beta\} \). Now, assume \( X(t) \) has been computed for \( t \geq \beta \) and a new data point \( x_{t+1} \) has just arrived. We consider the following two cases.

**Case 1:** \( x_{t+1} \) is one of the symbols that appear most in \( X(t) \). If we let \( x_{t+1} \in X(t+1) \), we need to remove one element, and let \( a_q \) be the element removed from \( X(t) \). If the frequency of \( a_q \) is same as that of \( x_{t+1} \), two sets \( X(t) \) and \( X(t+1) \) essentially have the same entropy; hence, we can assume the frequency of \( a_q \) is smaller than that of \( x_{t+1} \). However, \( H(X(t)) > H(X(t+1)) \) by Theorem 1 (see its remarks). Therefore, in this case, \( x_{t+1} \) must be discarded and let \( X(t+1) = X(t) \).

**Case 2:** \( x_{t+1} \) is not one of the symbols that appear most in \( X(t) \). Let \( a_p \) be a symbol that appears most in \( X(t) \). We set \( X(t+1) = X(t) \setminus \{a_p\} \cup \{x_{t+1}\} \). Then, \( H(X(t+1)) > H(X(t)) \) by Theorem 1. Therefore, in this case, \( x_{t+1} \) is included in \( X(t+1) \) and a symbol that appears most in \( X(t) \) is removed.

By repeating the above process as \( t \) increases, it is clear that
we can always maintain a set $X(t)$ of size $\beta$ with a highest possible entropy. Hence, our algorithm is optimal.

A complete description of $A_{EMS\text{OFF}}$ is given in Algorithm 1 $A_{EMS\text{OFF}}$.

To analyze the time complexity of $A_{EMS\text{SON}}$, we note that a frequency table of $X(t)$, i.e., a table for each element in $A$ with its frequency in $X(t)$, can be maintained for each $t$ and updated in $O(1)$ time. When a new data $x_t$ arrives, the frequency of $x_t$ in $X(t - 1)$ is checked and a decision is made followed by an update all in $O(1)$ time. Therefore, $A_{EMS\text{SON}}$ runs in $O(1)$ time at each time step.

B. $A_{EMS\text{OFF}}$

Let $X$ be an input sorted according to the frequency of each symbol $a_i$, $1 \leq i \leq k$, in $X$ and assume it is configured as a stair-case shape as in Fig. 1(a). Note that the maximum possible entropy of a subset $X_0$ of $X$ is $\log k$ by Lemma 1. Let $\alpha$ be a real number also defined as an input to the $EMS\text{OFF}$ problem, and define $h_1 = \alpha \log k$.

Our algorithm works in two steps. The first step is based on Theorem 6 which states that the entropy of $X_t$ decreases or remains the same as $i$ increases in which $X_i$ is defined to be a subset of $X$ such that all elements are included from the bottom of the shape of $X$ (i.e., row 1) through row $i$ of $X$.

So, in Step 1, we compute the largest $r$ such that $H(X_r) < \alpha \log k$ and $H(X_{r-1}) \geq \alpha \log k$. Such $r$ can be computed linearly or using a binary search.

In Step 2, we add as many symbols in row $r$ as possible to $X_{r-1}$. So a goal is to find a largest $j_1$ such that $H(X_{r-1} + j_1) \geq h_1$. Note that by Theorem 5, the entropy of $X_{r-1} + j$ decreases as $j$ increases from $j = 0$ to a certain point $j_0$ and then increases as $j$ continues to increase. Hence, if $h_1 > \max\{H(X_{r-1} + 1), H(X_{r-1} + l_r)\}$, no symbol from row $r$ can be added, i.e., $j_1 = 0$. So now, we only have the following case to be considered:

\[ h_1 \leq \max\{H(X_{r-1} + 1), H(X_{r-1} + l_r)\}. \]  (3)

Suppose $H(X_{r-1} + 1) \leq H(X_{r-1} + l_r)$. Then, by (3), $H(X_{r-1} + l_r) \geq h_1$, a contradiction to the definition of $r$.

Hence, we have

\[ H(X_{r-1} + 1) > H(X_{r-1} + l_r). \]  (4)

From (3) and (4), we have

\[ h_1 \leq H(X_{r-1} + 1). \]  (5)

Since $H(X_{r-1} + l_r)$ cannot be greater than or equal to $h_1$ by the definition of $r$, we finally have

\[ H(X_{r-1} + l_r) < h_1 \leq H(X_{r-1} + l_r + 1). \]  (6)

Fig. 7 depicts conditions in (3)–(5) in which $j_1$ must be less than $j_0$ and such $j_1$ can be computed using a binary search.

From the above discussions, it is clear that we only need to find $r$ and $j_1$ to find an optimal solution. A complete description of $A_{EMS\text{OFF}}$ is given in Algorithm 2 $A_{EMS\text{OFF}}$.

To analyze the time complexity of $A_{EMS\text{OFF}}$, we first do some preprocessing of input data set $X$ sorted according to the frequency of each symbol appearing in $X$, which can be done in $O(n \log n)$ time where $|X| = n$. Also, we compute $H(X_i)$ for each $i$ in advance, which can be done in $O(n \log n)$ time. Once the preprocessing is done in $O(n \log n)$ time, we only need $O(\log n)$ time to compute $r$ and also $O(\log l_r) = O(\log k)$ time to compute $j_1$.

C. Discussions on $A_{EMS\text{SON}}$ and $A_{EMS\text{OFF}}$

The result of our online algorithm $A_{EMS\text{SON}}$ when terminated after making a decision for $x_n$ for any $n \geq \beta$ is essentially a subset $X_0 \subseteq \{x_1, \ldots, x_n\}$ of size $\beta$ with maximum possible entropy. Computing such a subset $X_0$ can also

---

Algorithm 1: $A_{EMS\text{OFF}}$

\[
\text{(}t \text{ refers the current time, and } x_t \text{ is available at time } t, \text{)} \\
\text{Let } X(0) = \emptyset \text{ and } t = 1. \\
\text{while } (t \leq \beta) \\
\quad \text{Let } X(t) = X(t - 1) \cup \{x_t\}; \\
\quad t = t + 1; \\
\text{endwhile} \\
\text{while (not finished) } \\
\quad \text{Let } a_p \text{ be a symbol that appears most in } X(t - 1). \\
\quad \text{if the frequency of } x_t \text{ is not less than the frequency } \\
\quad \text{of } a_p \text{, then } X(t) = X(t - 1) \\
\quad \quad \text{else } X(t) = X(t - 1) \cup \{x_t\} - \{a_p\}. \\
\quad t = t + 1; \\
\text{endwhile} \\
\text{Return } (X(t - 1)).
\]

Algorithm 2: $A_{EMS\text{OFF}}$

\[
\text{[Let } h_1 = \alpha \log k. \text{ Assume } H(X) < h_1. \text{ Let } l_r \text{ denote the number of symbols in row } i \text{ of } X.] \\
\text{Compute } r \text{ such that } H(X_{r-1}) \geq h_1 \text{ and } H(X_r) < h_1. \\
\text{Compute } j_1, 1 \leq j_1 \leq l_r, \text{ such that } \\
\quad H(X_{r-1} + j_1) \geq h_1 \text{ and } H(X_{r-1} + j_1 + 1) < h_1. \\
\text{Return } (X_{r-1} + j_1). \\
\]

Fig. 7. $h_1$ and $j_1$
be done offline using a slight modification of our offline algorithm $A_{EMS\text{OFF}}$ such that we add the entire elements of $X$ from row 1 to row $r - 1$ (where adding all elements in row $r$ exceed $\beta$) and then add elements in row $r$ from left to right until the size of $X_0$ becomes $\beta$.

It should be noted that such an offline computation can be done in $\log n$ time after all necessary preprocessing while the online approach takes only $O(1)$ time to make an online decision for each element upon its arrival; hence taking $O(n)$ time. Depending on application scenarios, either online or offline algorithm may be more desirable.

VI. PERFORMANCE ANALYSIS

A. CIR Data Generation using MATLAB Toolbox

We use the ITU channel models in the MATLAB Toolbox to generate CIR data to test our online and offline algorithms. The simulation follows the steps below.

1. **Channel creation.** Create the channel using “ITUr3GPAx” default configuration with 4 multipaths.

2. **Obtain CIR magnitude.** Use the path gains to obtain the magnitude of the CIR discretized in 4 different paths, and add them to obtain a single CIR measurement. Repeat the process over time to generate a sequence of CIR samples.

3. **Quantization.** After the sampling, compute the mean of the CIR samples, and quantize each value as 1 if it is larger than the mean value, and as 0 otherwise.

4. **Word generation.** Use the bit stream obtained from previous step and group them into words by concatenating adjacent bits. The word length can be chosen arbitrarily. In our experiments, we set the word length to 8 bits, corresponding to an alphabet size of 256.

5. **Apply the algorithms.** We implement two algorithms: our online and offline algorithms ($A\_EMSON$, and $A\_EMS\text{OFF}$). Our experiment also analyzes the entropy of raw data before any processing.

B. Impact of Sampling Rate on Entropy over Doppler shifts with and without Entropy Maximization Algorithms

A Motivating Experiment: We first illustrate the impact of the sampling rate on the entropy attained, and in particular, the deteriorating effect on the entropy as a result of sampling the channel at rates that do not match the Doppler shift. The experiments shown in Fig. 8 are based on the default ITU3GPA channel setting except that the Max Doppler shift is altered. The length of the CIR data input (the size of the input vector, which is 640 after conversion into words in this case) stays the same for each sampling rate, and the first 640 symbols are used for each. We note that this implies different CIR data capture times for different sampling rates. The entropy is upper-bounded to 8 bits by Lemma 1, since the alphabet size is 256. We observe that the entropy is around 7.5 bits for low sampling rates, but starts to drop when the sampling rate reaches about 3 times the Doppler rate. It then continues to decrease following a logarithmic-like shape. Accordingly, in situations where the Doppler shift may not be known or measured, which can happen in some IoT environments, the proposed entropy maximization algorithms can prove to be useful. While the higher entropy observed in Fig. 8 for lower sampling rates can be in part attributed to the advantage of observing the channel longer, the sampling rate also plays a distinct role, as will be discussed based on the results of Figs. 9 and 10, where the observation time is kept the same for all sampling rates.

Entropy versus Data Capture Time over Sampling Rates without Applying the Algorithm: The experiments shown below are based on ITUr3GPA channel using the default setting, except that the Max Doppler shift is altered.

1) Doppler shift = 20 Hz: The set of experiments shown in Fig. 9 are based on a max Doppler shift of 20 Hz. The $X$ axis here refers to the observation time elapsed of the experiment. Thus the longer the data capture time in Fig. 9 (and Fig. 10), the larger the data set size (without applying the algorithm). For the same data capture time and different sampling rates, a larger sampling rate will result in a larger set size. Therefore, in these experiments, the entropy differences among different sampling rates cannot be attributed to different data capture times. The entropy shown are of the set $X$ without applying the algorithm. The sampling rate is shown in the top right in Hz. We observe that the entropy initially increases with a longer observation time, but saturates after a certain time. This saturation entropy level depends on the sampling rate, and can be substantially lower than the theoretical upper bound for a given alphabet, especially when the sampling rate is much higher than the Doppler frequency of 20 Hz. This confirms the intuition that oversampling the CIR would result in a less diverse set of symbols.

2) Doppler shift = 100 Hz: For the experiments shown in Fig. 10, we set the Max Doppler shift to 100Hz, and show the entropy of $X$ without applying the algorithm. The sampling rates are shown in the top right in Hz. Our observations are in line with the case for the Doppler shift of 20 Hz: the entropy reaches a saturation level over time, with substantial reduction in achieved entropy when the sampling rate exceeds the Doppler shift. The entropy is reduced roughly by half when the sampling rate is an order of magnitude higher than the Doppler shift.

In summary, the entropy of $X$ tends to increase with increasing data capture time. However, it reaches a certain upper limit
depending on the sampling rate. This upper limit seems to drop as the sampling rate increases. This is because oversampling the CIR generates a less diverse set of symbols, and thereby reduces the entropy. Thus, there is a need for entropy extraction algorithms similar to those proposed in this paper.

Efficiency of the algorithm: We next illustrate that our algorithms, even when applied to symbols obtained from suboptimal CIR sampling, can achieve the same performance as sampling at the Doppler rate without our algorithm. The experiment shown in Fig. 11 depicts the time our algorithm needs to achieve the same subset size and the same entropy level as sampling at the Doppler rate without our algorithm. The experiment shown in Fig. 11 depicts the time our algorithm needs to achieve the same subset size and the same entropy level as sampling at the Doppler rate of 200 Hz without our algorithm. When the CIR is sampled at 200 Hz, which is also the Doppler rate for this experiment, the entropy achieved without applying the algorithm is 7.597 bits, and the maximum entropy subset size consists of 640 words. It takes 25.6 s to reach this entropy level.

We set our A_EMSON algorithm to reach an entropy level of 7.597 and a subset size of 640 words, and show how long it takes to reach these targets for different sampling rates in Fig. 11. We see that our algorithm is efficient in that we can reach these targets regardless of the sampling rate used. This shows the robustness of our algorithm in achieving a desired entropy level even when the sampling rate does not match the Doppler shift.

We also observe the following about the speed at which our algorithm achieves the target rates. Without the algorithm, as shown in the graphs shown in the Fig. 8, the entropy starts to drop when the sampling rate reaches 3 times the Doppler rate. With our algorithm, as shown in the Fig. 11, the time to reach the target entropy still keeps dropping until 800 Hz, which is 4 times the Doppler rate.

C. Experimental Results on the Algorithms

C.1 Online Algorithm Results

Following Steps 1–5 in Section VI.A, we experimented with the online algorithm discussed in Section V.A. We set $\beta = 640$, and $n = 2,475$ words (or 19,800 bits) arriving during the overall simulation time of 99 seconds. We set the channel Maximum Doppler shift to 100 Hz, and the sampling rate to 200 Hz. Fig. 12 shows the change of achieved entropy over time from 26 s to 99 s. The $x$-axis shows the data capture time, which can also be represented by the number of input words ranging from 650 to 2,475 symbols. We observe that A_EMSON rapidly enhances the entropy as soon as more symbols than the required $\beta = 640$ words become available.

C.2 Off-line Algorithm Results

In this experiment, we use the same experimental setting as in the online experiment and a slight variation of our off-line optimal algorithm discussed in Section V.B. Instead of maximizing the subset size given a threshold for the target entropy, we try to compute a subset of a given fixed size that maximizes the entropy. Fig. 13 shows the change of entropies over the size of the goal subset.

It is important to observe that there exist local minimal points in Fig. 13 as the subset size is increased. These results confirm the statement in Theorem 5 in which as we add more symbols on the top row (see Fig. 7), the entropy first decreases, and then increases resulting in a local minimum.

In our next experiment, we have the same experimental set-
Let et al. HIGH ENTROPY SECRECY GENERATION FROM WIRELESS CIR

Fig. 12. Entropy achieved versus data capture time (or input size) using A_EMSON with $\beta = 640$.

Fig. 13. Entropy achieved versus subset size threshold with modified A_EMSON.

Fig. 14. Subset size versus entropy threshold for various sampling rates with A_EMSON.

Fig. 15. Maximum entropy and maximum entropy subset size vs data capture time with A_EMSON.

Settings as in the previous two experiments except for applying different sampling rates and using our original offline algorithm discussed in Section V.B, in which we are given a threshold for the goal entropy and compute a subset of the maximum size. The input set consists of 2500 words generated from the CIR data. Results are shown in Fig. 14. The $x$-axis shows the threshold value for the goal entropy, and the $y$-axis shows the achieved subset size (in words), i.e., the size of the largest subset meeting the desired entropy. We observe a clear tradeoff between the desired entropy level and the size of the subset reaching that value.

We next use the same experimental setting as in the previous three experiments with our offline algorithm to look into the trade-off between the CIR input size (expressed as data capture time) and the maximum entropy subset size together with the maximum entropy level we could achieve. The results are shown in Fig. 15 with the $x$-axis as the data capture time and the $y$-axis as the maximum entropy subset size, i.e., the size of the subset which achieves the maximum entropy for the different CIR input sizes. The $z$-axis shows the maximum entropy that can be achieved. In the early phase of CIR collection, our algorithm achieves a reasonably high entropy level and subset size, even though the sample size is constrained. When we have more CIR data observed, it quickly improves in both achieved entropy and the subset size.

C.3 Comparison with SHA-256

To compare the performance of our algorithm with SHA-256, we designed several simulation scenarios from 3GPP-TS36.104, see Table 1. The data size is 40,000 bits at most and the alphabet size is 8 bits for entropy calculation. For these comparisons, after step 3 we first separate the bit stream into chunks of 512 bits, discarding any extra bits. We then run each chunk through SHA-256, resulting in chunks of 256 bits. We then link the chunks of 256 bits in the same order that we separated them to form the bitstream after SHA-256 privacy amplification. Fig. 16 shows the simulation result. Our algorithm performs better in most cases except the extended pedestrian scenario with sample rate of 20 Hz, which can be interpreted that the channel is relatively constant compared to the 20 samples/s. In other cases our proposed algorithm works better, where the channel has more variety or the sampling gap is long enough. It can also be derived that the performance of SHA-256 has little impact from the channel and sampling rate since its output is almost random. The alphabet size of 8 bits is set in our simulations. While this
size has more performance impact on our algorithm than SHA-256, we also designed simulations with alphabet size of 2, 4, and 6 bits under the same configuration. The results show that the corresponding entropy of SHA-256 output increases with data input and it is more close to theoretical optimal for alphabet size of 2 and 4 bits, while the entropy of ours are almost theoretical optimal even from very small data input size.

VII. CONCLUSIONS

Wireless channel characteristics have emerged as a promising source of secrecy due to the reciprocity and spatial decorrelation properties of the wireless radio channel. These properties allow the two communicating nodes to use near-simultaneous channel measurements taken independently at each side to generate (near-)identical bits after some processing such as quantization and information reconciliation. However, in many wireless settings and application scenarios, obtaining and sustaining high-entropy, high-rate secret bits remains a challenge due to factors such as low mobility or low storage (IoT environments). Thus, it is crucial to generate the highest number of secret bits while reaching the maximum (or a target) entropy-level possible from the arriving sequence of channel measurements.

In this paper, we consider both online and offline versions of the entropy maximizing subset selection problem, and fully characterize the optimal solutions through a series of theorems and algorithms. Our results are substantial. For the online version, we consider a set of sequentially arriving samples over time, and show how we can optimally decide whether to keep or discard each arriving sample in an online fashion in such a way that we always maintain a fixed-sized subset that has the optimal entropy over all possible subsets of the observed samples. For the offline version, we consider the problem of finding the highest-entropy subset among all maximum-sized subsets of the observed data subject to a minimum entropy threshold. We presented several theorems to reinforce the structure of the maximum-entropy (and maximum-size) solution to this problem. The algorithms were evaluated through MATLAB simulations with realistic wireless multipath channel models. The results indicate the resiliency of our algorithms in generating high entropy data sets over a range of input data sizes. In addition, the algorithms also sustain high-entropy secret bit generation even when the CIR sampling interval does not match the Doppler frequency, or the channel coherence time. This is important for many IoT and indoor environments where these channel parameters (Doppler, coherence time) may not be easy to obtain.

In this paper, our research focused on only Shannon’s entropy and algorithms were developed accordingly. Many issues including repeated patterns still need to be addressed in order to obtain truly random bit sequences. We plan to investigate additional randomness metrics incorporated in our algorithms that can pass existing benchmark testing including the NIST tool [12].

APPENDIX A

PROOF OF THE THEOREM 1

Proof: We first note that \(|X| = |X^{′}|\), and let \(M = |X|\). We also note that \(c_i(X^{′}) = c_i(X)\) for \(1 \leq i \leq k\) if \(i \neq p\) and \(i \neq q\), \(c_p(X^{′}) = c_p(X) - 1\), and \(c_q(X^{′}) = c_q(X) + 1\).

We next proceed to show \(H(X) - H(X^{′}) < 0\). By the definition of the entropy, we have

\[
H(X) - H(X^{′}) = -\sum_{i=1}^{k} \frac{c_i(X)}{M} \log \frac{c_i(X)}{M} + \sum_{i=1}^{k} \frac{c_i(X^{′})}{M} \log \frac{c_i(X^{′})}{M}
\]

\[
= -\frac{c_p(X)}{M} \log \frac{c_p(X)}{M} - \frac{c_q(X)}{M} \log \frac{c_q(X)}{M} + \frac{c_p(X^{′})}{M} \log \frac{c_p(X^{′})}{M} + \frac{c_q(X^{′})}{M} \log \frac{c_q(X^{′})}{M}.
\]

By setting \(c_p(X) = x\) and \(c_q(X) = y\), implying \(c_p(X^{′}) = x - 1\) and \(c_q(X^{′}) = y + 1\), we have

\[
H(X) - H(X^{′}) = -\frac{x}{M} \log \frac{x}{M} - \frac{y}{M} \log \frac{y}{M} + \frac{x - 1}{M} \log \frac{x - 1}{M}
\]

\[
+ \frac{y + 1}{M} \log \frac{y + 1}{M} = \frac{x}{M} \log M - \log x + \frac{y}{M} \log M - \log y
\]

\[
- \frac{x - 1}{M} \log M - \log(x - 1)
\]

\[
- \frac{y + 1}{M} \log M - \log(y + 1)
\]

\[
= \log M \left(\frac{x - 1}{M} + \frac{y + 1}{M}\right)
\]

\[
+ \frac{1}{M} \left(\frac{x - 1}{M} \log x - y \log y + (x - 1) \log(x - 1) + (y + 1) \log(y + 1)\right)
\]

\[
= \frac{1}{M} \log \frac{(x - 1)^{x-1}}{(y + 1)^{y+1}}.
\]

Since \(\frac{(x - 1)^{x-1}}{(y + 1)^{y+1}} < 1\) by Lemma 2, we establish

\(H(X) - H(X^{′}) < 0\),

completing the proof of Theorem 1.

APPENDIX B

PROOF OF THE THEOREM 2

Proof: Let \(X_1 \subseteq X\) be an optimal solution with its shape different from \(X_0\). We then note that there must be a symbol, say \(a_p\), which occurs more in \(X_1\) than in \(X_0\). Then, there must be a symbol, say \(a_q\), which occurs more in \(X_0\) than in \(X_1\) since \(|X_1| = |X_0|\). (Note that symbols \(a_p\) and \(a_q\) may occur multiple

| Channel type         | Doppler shift in Hz |
|----------------------|---------------------|
| Extended pedestrian A (EPA) | 5                   |
| Extended vehicular A (EVA)   | 70                  |
| Extended typical Urban (ETU) | 300                 |
times in $X_1$ as well as in $X_{01}$.) We then define $X_2 = X_1 \cup \{a_g\} \setminus \{a_p\}$. Clearly, $|X_2| = |X_1|$, but by Theorem 1 (see its remarks), $H(X_2) > H(X_1)$, a contradiction to the assumption that $X_1$ is an optimal solution. This concludes that $X_1$ cannot be an optimal solution, which completes the proof.

Indeed, we have (Let $S = \sum_{i=2}^{k} p_i$)

$$
\frac{dH}{d\epsilon} = -\ln(p_1 + \epsilon) + 1
$$

$$
- \sum_{i=2}^{k} \left[ -\frac{p_i}{S} \ln \left( \frac{p_i}{S} \right) - \frac{p_i}{\sum_{i=2}^{k} p_i} \right]
$$

$$
= - \ln(p_1 + \epsilon) + \sum_{i=2}^{k} \left[ \frac{p_i}{S} \ln \left( \frac{p_i}{p_i + \epsilon} \right) \right]
$$

$$
\leq - \ln(p_1 + \epsilon) + \ln \left[ \sum_{i=2}^{k} \frac{p_i}{S} \left( \frac{p_i}{p_i + \epsilon} \right) \right].
$$

Then the only inequality we need to prove is

$$
\sum_{i=2}^{k} \frac{p_i}{S} \left( \frac{p_i}{p_i + \epsilon} \right) \leq p_1 + \epsilon.
$$

Note that

$$
\sum_{i=2}^{k} \frac{p_i}{S} \leq p_1 \iff \sum_{i=2}^{k} \frac{p_i^2}{1 - p_i} \leq p_1
$$

If we can show the second inequality above, we are done with the proof. It turns out that:

$$
\sum_{i=2}^{k} \frac{p_i^2}{S} \leq p_1 \iff \sum_{i=2}^{k} \frac{p_i^2}{1 - p_i} \leq p_1
$$

$$
\iff p_2^2 + \cdots + p_k^2 \leq p_1 - p_1^2
$$

$$
\iff (p_1 - 1/2)^2 + p_2^2 + \cdots + p_k^2 \leq 1/4.
$$

Fig. 16. Output entropy comparison between SHA-256 and our proposed algorithm under different scenarios.

APPENDIX C

PROOF OF THE THEOREM 3

Proof: To show that adding one symbol on top of $X$ will decrease the entropy, we change it into a mathematical formulation as described next.

Assume $0 \leq p_2 \geq p_3 \geq \cdots \geq p_k$, and $\sum_{i=1}^{k} p_i = 1$. When the probability $p_1$ is increased by a small quantity $\epsilon$, namely, $\tilde{p}_1 = p_1 + \epsilon$, and other probabilities $p_2, p_3, \cdots, p_k$ change correspondingly but keep the ratio $p_2 : p_3 : \cdots : p_k$ unchanged, then we have:

$$
\tilde{p}_1 = p_1 + \epsilon
$$

$$
\tilde{p}_i = p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \epsilon, \quad i = 2, \cdots, k
$$

and the entropy $H$ in terms of $\epsilon$ reads:

$$
H(\epsilon) = -(p_1 + \epsilon) \ln(p_1 + \epsilon)
$$

$$
- \sum_{i=2}^{k} \left[ p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \right] \ln \left( p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \right).
$$

We only need to show that

$$
\frac{dH(\epsilon)}{d\epsilon} < 0 \quad \text{for } \epsilon > 0.
$$

Indeed, the probability $p_1$ is increased by a small quantity $\epsilon$, namely, $\tilde{p}_1 = p_1 + \epsilon$, and other probabilities $p_2, p_3, \cdots, p_k$ change correspondingly but keep the ratio $p_2 : p_3 : \cdots : p_k$ unchanged, then we have:

$$
\tilde{p}_1 = p_1 + \epsilon
$$

$$
\tilde{p}_i = p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \epsilon, \quad i = 2, \cdots, k
$$

and the entropy $H$ in terms of $\epsilon$ reads:

$$
H(\epsilon) = -(p_1 + \epsilon) \ln(p_1 + \epsilon)
$$

$$
- \sum_{i=2}^{k} \left[ p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \right] \ln \left( p_i - \frac{p_i}{\sum_{i=2}^{k} p_i} \right).
$$

We only need to show that

$$
\frac{dH(\epsilon)}{d\epsilon} < 0 \quad \text{for } \epsilon > 0.
$$

Indeed, we have (Let $S = \sum_{i=2}^{k} p_i$)

$$
\frac{dH}{d\epsilon} = -\ln(p_1 + \epsilon) + 1
$$

$$
- \sum_{i=2}^{k} \left[ -\frac{p_i}{S} \ln \left( \frac{p_i}{S} \right) - \frac{p_i}{\sum_{i=2}^{k} p_i} \right]
$$

$$
= - \ln(p_1 + \epsilon) + \sum_{i=2}^{k} \left[ \frac{p_i}{S} \ln \left( \frac{p_i}{p_i + \epsilon} \right) \right]
$$

$$
\leq - \ln(p_1 + \epsilon) + \ln \left[ \sum_{i=2}^{k} \frac{p_i}{S} \left( \frac{p_i}{p_i + \epsilon} \right) \right].
$$

Then the only inequality we need to prove is

$$
\sum_{i=2}^{k} \frac{p_i}{S} \left( \frac{p_i}{p_i + \epsilon} \right) \leq p_1 + \epsilon.
$$

Note that

$$
\sum_{i=2}^{k} \frac{p_i}{S} \leq p_1 \iff \sum_{i=2}^{k} \frac{p_i^2}{1 - p_i} \leq p_1
$$

If we can show the second inequality above, we are done with the proof. It turns out that:

$$
\sum_{i=2}^{k} \frac{p_i^2}{S} \leq p_1 \iff \sum_{i=2}^{k} \frac{p_i^2}{1 - p_i} \leq p_1
$$

$$
\iff p_2^2 + \cdots + p_k^2 \leq p_1 - p_1^2
$$

$$
\iff (p_1 - 1/2)^2 + p_2^2 + \cdots + p_k^2 \leq 1/4.
$$
The last inequality can be justified by a standard maximization problem with inequality constraints:

\[
\begin{align*}
\max \quad & (p_1 - 1/2)^2 + p_2^2 + \cdots + p_k^2 \\
\text{subject to} \quad & p_1 + \cdots + p_k = 1 \\
& p_i \geq p_{i+1}, \quad \text{for } i = 1, \cdots, k - 1.
\end{align*}
\]

Solving this problem using the Kuhn-Tucker conditions, we can easily find the maximum value is 1/4. Therefore:

\[
(p_1 - 1/2)^2 + p_2^2 + \cdots + p_k^2 \leq 1/4. \tag{11}
\]

Thus, \(H(X_1) < H(X)\). \(\Box\)

\section*{Appendix D
Proof of the theorem 4}

\textbf{Proof:} Let \(X_0 = \{x_1^{c_1}, \cdots, x_k^{c_k}\}\) be a set where \(x_i\) is counted \(c_i\) times for \(i = 1, \cdots, k\). Assume for some \(j\)

\[
c_1 = c_2 = \cdots = c_j > c_{j+1} \geq c_{j+2} \geq \cdots \geq c_k.
\]

Define

\[
X_\epsilon = \{x_1^{c_1+\epsilon}, x_2^{c_2+\epsilon}, x_{j+1}^{c_{j+1}+\epsilon}, \cdots, x_k^{c_k+\epsilon}\}, \quad C = \sum_{i=1}^{k} c_i,
\]

for \(\epsilon > 0\). Then the entropy \(H(X_\epsilon)\)

\[
H(X_\epsilon) = \sum_{i=1}^{j} \left\lfloor \frac{c_i + \epsilon}{C + j\epsilon} \ln \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) \right\rfloor
- \sum_{i=j+1}^{k} \left\lfloor \frac{c_i}{N + j\epsilon} \ln \left( \frac{c_i}{N + j\epsilon} \right) \right\rfloor \tag{14}
\]

satisfies

\[
\frac{dH(X_\epsilon)}{d\epsilon} \leq 0. \tag{15}
\]

From the definition of the entropy \(H(X_\epsilon)\), we have

\[
dH \frac{dH}{d\epsilon} = - \sum_{i=1}^{j} \left\lfloor \ln \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) + 1 \cdot \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) \right\rfloor
- \sum_{i=j+1}^{k} \left\lfloor \ln \left( \frac{c_i}{N + j\epsilon} \right) + 1 \cdot \left( \frac{c_i}{N + j\epsilon} \right) \right\rfloor
\]

\[
= - \sum_{i=1}^{j} \left\lfloor \ln \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) + 1 \cdot \frac{(C + j\epsilon) - (c_i + \epsilon)j}{(C + j\epsilon)^2} \right\rfloor
- \sum_{i=j+1}^{k} \left\lfloor \ln \left( \frac{c_i}{C + j\epsilon} \right) + 1 \cdot \frac{-c_i j}{(C + j\epsilon)^2} \right\rfloor
= - \sum_{i=1}^{j} \ln \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) \cdot \frac{1}{C + j\epsilon}
+ \sum_{i=j+1}^{k} \ln \left( \frac{c_i}{C + j\epsilon} \right) \cdot \frac{c_i}{C + j\epsilon}
\]

\[
\leq \left\lfloor \frac{-1}{j} \sum_{i=1}^{j} \ln \left( \frac{c_i + \epsilon}{C + j\epsilon} \right) + \ln \left( \sum_{i=1}^{j} \left( \frac{c_i + \epsilon}{C + j\epsilon} \right)^2 \right)
+ \sum_{i=j+1}^{k} \left( \frac{c_i}{C + j\epsilon} \right)^2\right\rfloor \cdot \frac{j}{C + j\epsilon}
\]

To show that \(dH/d\epsilon < 0\), we need

\[
\sum_{i=1}^{j} \left( \frac{c_i + \epsilon}{C + j\epsilon} \right)^2 + \sum_{i=j+1}^{k} \left( \frac{c_i}{C + j\epsilon} \right)^2 \leq \prod_{i=1}^{j} \left( \frac{c_i + \epsilon}{C + j\epsilon} \right)^{1/j}. \tag{16}
\]

Let

\[
p_i = \begin{cases} \frac{c_i + \epsilon}{C + j\epsilon}, & i = 1, \cdots, j; \\ \frac{c_i}{C + j\epsilon}, & i = j + 1, \cdots, k \end{cases}
\]

and notice that \(p_1 = \cdots = p_j\), the inequality (16) becomes

\[
p_1^2 + \cdots + p_k^2 \leq p_1 \iff (p_1 - 1/2)^2 + p_2^2 + \cdots + p_k^2 \leq 1/4. \tag{17}
\]

The last inequality can be justified by a standard maximization problem with inequality constraints:

\[
\max \quad (p_1 - 1/2)^2 + p_2^2 + \cdots + p_j^2
\]

\text{subject to} \quad \begin{align*}
p_1 + \cdots p_k &= 1 \\
p_i &\geq p_{i+1}, \quad \text{for } i = 1, \cdots, k - 1.
\end{align*}

Solving this problem using the Kuhn-Tucker conditions, we can easily find the maximum value is 1/4. Therefore the inequality (16) holds and we have

\[
dH \frac{de}{c} < 0.
\]

dH/de < 0 implies that H(X) is a decreasing function as e increases. In particular, we have H(X) ≤ H(X0). Namely, given a set

\[X_0 = \{x_1^0, \cdots, x_k^0\},\]

\[c_1 = c_2 = \cdots = c_j > c_{j+1} \geq c_{j+2} \geq \cdots \geq c_k,\]

then counting the first k elements one more time will lead to a decrement on the entropy. Thus H(X) < H(X).

**APPENDIX E**

**PROOF OF THE THEOREM 5**

*Proof:* We first translate the statements in the theorem into the following mathematical formulation.

Let \(X_0 = \{x_1^0, \cdots, x_k^0\}\) be a set where \(x_i\) is counted \(c_i\) times for \(i = 1, \cdots, k\). Rewrite it as in continuous form

\[X_0(n) = \left\{ \begin{array}{ll} c_1, & c \in [0, 1] \\ \vdots & \\ c_j, & c \in (j-1, j] \\ \vdots & \\ c_k, & c \in (k-1, k] \end{array} \right., \]

and assume for some \(j\)

\[c_1 = c_2 = \cdots = c_j > c_{j+1} \geq c_{j+2} \geq \cdots \geq c_k,\]

\[C = \sum_{i=1}^{k} c_i.\]

Define

\[X_\delta(c) = \left\{ \begin{array}{ll} c_1 + 1, & c \in [0, \delta] \\ c_1, & c \in (\delta, j] \\ c_{j+1}, & c \in [j, j+1] \\ \vdots & \\ c_k, & c \in (k-1, k] \end{array} \right., \]

for \(\delta \in (0, j]\). Then the entropy \(H(X_\delta)\)

\[H(X_\delta) = -\int_{\delta}^{c_j + 1} X_\delta dc \ln \left( \frac{X_\delta}{c_j + 1} \right) dc - \delta \frac{c_1 + 1}{C + \delta} \ln \left( \frac{c_1 + 1}{C + \delta} \right) - (j - \delta) \frac{c_1}{C + \delta} \times \ln \frac{c_1}{C + \delta} - \sum_{i=j+1}^{k} \left( \frac{c_i}{C + \delta} \ln \frac{c_i}{C + \delta} \right) \]

satisfies

\[
\frac{dH(X_\delta)}{dc} \bigg|_{c=0} < 0, \quad \frac{d^2H(X_\delta)}{dc^2} > 0. \tag{21}
\]

Taking the first derivative, we have

\[
\frac{dH(X_\delta)}{dc} = - \frac{c_1 + 1}{C + \delta} \ln \frac{c_1 + 1}{C + \delta} + \delta (c_1 + 1) \frac{1}{(C + \delta)^2} \left[ \ln \frac{c_1 + 1}{C + \delta} \right] - \frac{c_1}{C + \delta} \ln \frac{c_1}{C + \delta} + \frac{1}{(C + \delta)^2} \ln \frac{c_1}{C + \delta} + \frac{k}{(C + \delta)^2} \left[ 1 + \ln \frac{c_1}{C + \delta} \right]
\]

\[
= - \frac{(c_1 + 1)C}{(C + \delta)^2} \ln \frac{c_1 + 1}{C + \delta} + \frac{c_1 (C + j)}{(C + \delta)^2} \ln \frac{c_1}{C + \delta} + \frac{1}{C + \delta}
\]

\[
= \frac{a}{C + \delta} \ln \frac{c_1}{c_1 + 1} + a \ln \frac{c_1}{C + \delta} + \sum_{i=j+1}^{k} \frac{b_i}{C + \delta} \ln \frac{c_i}{c_i + 1} + \frac{1}{a + \sum_{i=1}^{k} b_i}, \tag{22}
\]

where

\[a = \frac{c_1 (C + j)}{C + \delta}, \quad b_i = \frac{c_i}{C + \delta}, \quad a + \sum_{i=1}^{k} b_i = \frac{(c_1 + 1) C}{C + \delta}, \]

and all sums \(\sum\) range from \(i = j + 1\) to \(k\). Note that

\[a + \sum_{i=1}^{k} b_i \ln \frac{c_i}{c_1 + 1} + \sum_{i=j+1}^{k} b_i \ln \frac{c_i}{c_1 + 1} + \frac{1}{a + \sum_{i=1}^{k} b_i}
\]

\[\leq \ln \left( \frac{a}{a + \sum_{i=1}^{k} b_i} \frac{a}{c_1 + 1} + \sum_{i=j+1}^{k} \frac{b_i}{a + \sum_{i=1}^{k} b_i} \frac{c_i}{c_1 + 1} + \frac{1}{a + \sum_{i=1}^{k} b_i} \right)
\]

\[= \ln \left( \frac{a}{(c_1 + 1)^2 C} + \frac{C + \delta}{(c_1 + 1) C} \right)
\]

\[\leq \ln \frac{c_1 (C + j) + c_1 \sum_{i=j+1}^{k} c_i}{(c_1 + 1)^2 C} + \frac{C + \delta}{(c_1 + 1) C}
\]

\[= \ln \frac{c_1 (C + j) + c_1 (C - j c_1)}{(c_1 + 1)^2 C} + \frac{C + \delta}{(c_1 + 1) C}
\]

\[= \ln \frac{c_1}{c_1 + 1} + \frac{C + \delta}{(c_1 + 1) C}.
\]

When \(\delta = 0\),

\[\ln \frac{c_1}{c_1 + 1} + \frac{C + \delta}{(c_1 + 1) C} = \ln \frac{c_1}{c_1 + 1} + \frac{1}{c_1 + 1} < 0
\]

by considering Taylor expansion. Therefore, we have

\[
\frac{dH(X_\delta)}{dc} \bigg|_{c=0} < 0.
\]
By taking second derivative, we have

\[
\frac{d^2 H(X_j)}{d\delta^2} = \frac{2(c_1 + 1)C}{(C + \delta)^3} \ln \frac{c_1 + 1}{C + \delta} + \frac{(c_1 + 1)C}{(C + \delta)^3} - \frac{2c_1(C + j)}{(C + \delta)^3} \ln \frac{c_1}{C + \delta} - \frac{c_1(C + j)}{(C + \delta)^3} - \sum_{i} \frac{2c_i}{(C + \delta)^3} \ln \frac{c_i}{C + \delta} - \sum_{i} \frac{c_i}{(C + \delta)^3} = \frac{2}{(C + \delta)^2} \left[ \frac{(c_1 + 1)C}{C + \delta} \ln \frac{c_1 + 1}{C + \delta} - \frac{c_1(C + j)}{C + \delta} \times \ln \frac{c_1}{C + \delta} - \sum_{i} \frac{c_i}{C + \delta} \ln \frac{c_i}{C + \delta} \right] = \frac{2}{(C + \delta)^2} \left[ a \ln \frac{c_1 + 1}{c_1} + \sum_{i} b_i \ln \frac{c_i}{c_i} \right] > 0
\]

as \(a, b_i > 0, i = j + 1, \ldots, k\) from (22) and \(c_1 > c_i, i = j + 1, \ldots, k\) from (18).

\[\square\]

REFERENCES

[1] K. Zeng, “Physical layer key generation in wireless networks: Challenges and opportunities,” IEEE Commun. Mag., vol. 53, no. 6, pp. 33–39, June 2015.

[2] Z. Li, W. Xu, R. Miller, and W. Trappe, “Securing wireless systems via lower layer enforcements,” in Proc. ACM WiSe, Sep. 2006, pp. 33–42.

[3] L. Xiao, L. J. Greenstein, N. B. Mandayam and W. Trappe, “Using the physical layer for wireless authentication in time-variant channels,” IEEE Trans. Wireless Commun., vol. 7, no. 7, pp. 2571–2579, July 2008.

[4] D. Kapetanovic, G. Zheng and F. Rusek, “Physical layer security for massive MIMO: An overview on passive eavesdropping and active attacks,” IEEE Commun. Mag., vol. 53, no. 6, pp. 21–27, June 2015.

[5] N. Patwari, J. Croft, S. Jana and S. K. Kasera, “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.

[6] B. Azimi-Sadjadi, A. Knayias, A. Mercado, and B. Yener. “Robust key generation from signal envelopes in wireless networks.” in Proc. ACM CCS, pp. 401–410, Oct. 2007.

[7] C. Ye, A. Reznik, G. Sternburg and Y. Shah, “On the secrecy capabilities of ITU channels,” in Proc. IEEE VTC, 2007, pp. 2030–2034.

[8] S. N. Premnath et al., “Secret key extraction from wireless signal strength in real environments,” IEEE Trans. Mobile Comput., vol. 12, no. 5, pp. 917–930, May 2013.

[9] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. 2008. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. in Proc. ACM MobiCom, Sep. 2008, pp. 128–139.

[10] C. Lee, Chen Shen, G. Sahin, K. Choi and H. Choi, “A novel and scalable communication-history-based Knapsack authentication framework for IEEE 802.11 networks,” in Proc. IEEE CNS, Oct. 2015, pp. 44–52.

[11] C. Lee, L. Zappaterra, K. Choi and H. Choi, “Securing smart home: Technologies, security challenges, and security requirements,” in Proc. IEEE CNS, Oct. 2014, pp. 67–72.

[12] L. E. Bassham, et al., “A statistical test suite for random and pseudorandom number generators for cryptographic applications,” Special Publication (NIST SP)-800-22 Rev 1a, 2010.

[13] C. Shen, H. Li, G. Sahin and H. Choi, “Low-complexity scalable authentication algorithm with imperfect shared keys for internet of things,” in Proc. IEEE ICC Workshop, May 2016, pp. 116–121.

[14] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe and N. B. Mandayam, “Information-theoretically secret key generation for fading wireless channels,” IEEE Trans. Inf. Forensics Security, vol. 5, no. 2, pp. 240–254, June 2010.

[15] T. Aono, K. Higuchi, T. Ohira, B. Komiyama and H. Sasaoka, “Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels,” IEEE Trans. Antennas Propag., vol. 53, no. 11, pp. 3776–3784, Nov. 2005.

[16] X. Zhu, F. Xu, E. Novak, C. C. Tan, Q. Li and G. Chen, “Using wireless link dynamics to extract a secret key in vehicular scenarios,” IEEE Trans. Mobile Comput., vol. 16, no. 7, pp. 2065–2078, July 2017.

[17] X. Zhu, F. Xu, E. Novak, C. C. Tan, Q. Li and G. Chen, “Extracting secret key from wireless link dynamics in vehicular environments,” in Proc. IEEE INFOCOM, Turin, 2013, pp. 2283–2291.

Yuxiang Zhao is currently an Assistant Professor of Mathematics at the George Washington University. He received his Ph.D. from the Department of Mathematics of the Pennsylvania State University under the supervision of Professor Quang Du.

Chen Shen received a B.S. degree in Software Engineering in 2011 from Huazhong University of Science and Technology, and his M.S. degree in Computer Science in 2013 from the George Washington University. After a short period of work at a startup company as an image processing engineer, he started the Ph.D. program in Computer Science at the George Washington University in 2014 under the supervision of Professor Hyeong-Ah Choi.

Hao Li earned his B.S. degree in Software Engineering in 2011 from Nankai University, and received his M.S. degree in Computer Science in 2013 from the George Washington University. He started the Ph.D. program in Computer Science at the George Washington University in 2014 under the supervision of Professor Hyeong-Ah Choi.
Gokhan Sahin is a Professor of Electrical and Computer Engineering at Miami University. He also served as a Nokia-Siemens Networks Invited Chair and Professor at Instituto de Telecomunicaes, Universidade de Aveiro, Portugal. He received his M.S. degree in 1997 and his Ph.D. in 2001, from the University of Washington, Seattle, both in electrical engineering. Dr. Sahin is a co-inventor of AT&T’s ROLEX restoration method, which made a substantial impact on AT&T’s backbone network and was later used in DARPA CORONET, a major national initiative aiming to design the next generation Internet. He currently serves as an Associate Editor of an IEEE Transactions on Network Science and Engineering special issue on network science for high-confidence cyber-physical systems.

Hyeong-Ah Choi is a Professor of Computer Science at the George Washington University. She obtained her Ph.D. in Computer Science from Northwestern University in 1986. Her research interests lie in network scheduling and optimization algorithms.

Yogendra Shah earned his B.Sc. and Ph.D. degrees in Electronic Engineering from London University, United Kingdom, in 1982 and 1985, respectively. He has worked in the wireless industry, developing products and services, incorporating security and wireless technologies, such as the early digital cordless phones to the the latest 5G wireless systems. He has research interests in developing security and advanced wireless communications technologies. He is a CISSP and a member of the IEEE.