On the Security of Key Extraction from Measuring Physical Quantities*

Matt Edman¹, Aggelos Kiayias², Qiang Tang² and Bülent Yener¹

¹ Rensselaer Polytechnic Institute
{edmanm2,yener}@cs.rpi.edu
² University of Connecticut & University of Athens
{aggelos,qiang}@cse.uconn.edu

Abstract. Key extraction via measuring a physical quantity is a class of information theoretic key exchange protocols that rely on the physical characteristics of the communication channel to enable the computation of a shared key by two (or more) parties that share no prior secret information. The key is supposed to be information theoretically hidden to an eavesdropper. Despite the recent surge of research activity in the area, concrete claims about the security of the protocols typically rely on channel abstractions that are not fully experimentally substantiated. In this work, we propose a novel methodology for the experimental security analysis of these protocols. The crux of our methodology is a falsifiable channel abstraction that is accompanied by an efficient experimental approximation algorithm of the conditional min-entropy available to the two parties given the view of the eavesdropper. We focus on the signal strength between two wirelessly communicating transceivers as the measured quantity and we use an experimental setup to compute the conditional min-entropy of the channel given the view of the attacker which we find to be linearly increasing. Armed with this understanding of the channel, we showcase the methodology by providing a general protocol for key extraction in this setting that is shown to be secure for a concrete parameter selection. In this way we provide a first comprehensively analyzed wireless key extraction protocol that is demonstrably secure against passive adversaries. Our methodology uses hidden Markov models as the channel model and a dynamic programming approach to approximate conditional min-entropy but other possible instantiations of the methodology can be motivated by our work.

1 Introduction

Key extraction between two parties, Alice and Bob, by measuring a physical quantity is based on two basic premises. First, due to the physical properties of the quantity measured, Alice and Bob obtain highly correlated measurements. Second, the measurements obtained by the eavesdropper are weakly correlated. This hypothesized gap of the correlation of measurements between Alice and

* This work is supported by NSF grant 0831304
Bob and the correlation of the eavesdropper’s measurements opens the door for a key exchange mechanism that, would have to perform an information reconciliation step and privacy amplification step to enable the calculation of a secret key that is almost independent of the adversary’s view.

In the literature, one can distinguish two classes of such protocols. The first class is practice-oriented protocols in which the security of the key is based (at best) on statistical tests. The second class is theoretical protocols in which broad channel abstractions are made and then these are formally shown to imply a correlation gap between the adversary and the two parties, thus showing information theoretic security. Nevertheless, despite that security is proven formally, for a real-world implementation, a leap of faith is still required to accept the fact that the channel abstraction is indeed capturing the real-world setting of the adversary. The only exception is quantum key-exchange, where there are results from quantum mechanics such as the no-cloning theorem that do provide the underlying formal basis justifying the connection between the channel abstraction and the real-world. The lack of strong general arguments in key extraction from measuring physical quantities means that any deviation of the real-world adversary setting from the assumed channel abstraction will lead to a security breakdown and thus to a dubious security status for the key extraction protocol.

In this work we address this fundamental problem of bridging the gap between experimental analysis and theoretical security arguments, by providing a novel formal security methodology that enables us to argue experimentally about the security of such protocols. In a nutshell, we put forth a general framework that (1) enable us to mathematically argue the security of key extraction from measuring physical quantities, (2) the security analysis can be tied to a specific real-world setting through falsifiable assumptions that are verifiable through experiments.

1.1 Our Results and Related Work

Information-theoretic treatment of secure key exchange was initiated by Wyner [30], and later formalized by Maurer [23]. These works dealt with the feasibility of key exchange assuming a non-zero “equivocation-rate”, a measure expressing the uncertainty of the eavesdropper about the parties’ communication. Beyond generic schemes, for specific physical quantities, there has already been a series of theoretical studies, in quantum key exchange [6], and wireless key exchange focusing on the “secrecy capacity” between wirelessly communicating transceivers [5, 8, 11, 17, 20, 26]. These works, being information theoretic in nature, contain no experimental justification of the channel model they utilize: for the quantum ones, the underlying physics justify the model but the wireless ones have no real justification. Furthermore, most rely on the assumption that the signal measurements are independent across time and thus any existing non-zero equivocation rate that the eavesdropper may have can be magnified for a suitable number of transmissions in a straightforward manner.

Considering works that were more experimental in nature, there were several practical algorithms suggested for wireless key extraction [3, 4, 16, 19, 21, 22, 25, 20, 31, 32]. A common characteristic of these works is that significant attention
is given to demonstrate correctness (i.e., that Alice and Bob calculate the same key) and much less analysis (as we detail below) is performed to demonstrate security (i.e., that the eavesdropper cannot extract any information about the key). More specifically, in [22, 23, 32] no proper security analysis is provided whatsoever, while [21, 29] considers performing randomness tests but these are insufficient as security guarantees (such “random” sequences are not necessarily unpredictable). In [19] a type of attacks based on blind deconvolution is considered and it is argued experimentally that such attacks are unlikely to apply; however, this does not preclude other types of attacks. Works such as [31] provided informal security arguments based on calculating the correlation between the eavesdropper’s measurements and the parties’ measurements. Unfortunately, correlation measures used such as the Pearson correlation coefficient are insufficient for arguing about security. Indeed, even if no linear correlation exists, this does not preclude the existence of much more complex non-linear correlation functions that can be taken advantage of in an attack and the existence of such functions cannot be precluded by these arguments.

In fact, the lack of properly rigorous security claims can lead to (partial) key recovery attacks as demonstrated in [13, 15] that examined known protocols in a specific deployment. These works exemplified the fact that even though existing analysis has demonstrated successfully that the signal observed by the two parties has sufficient entropy it remains open whether the conditional entropy on the eavesdropper’s view is non-zero in a specific real-world deployment.

The authors of [14] also considered how to deploy a man-in-the-middle attack to recover part of the key generated from the wireless signals, and [11, 33] discussed the problem of extracting a key from wireless signal in an adaptive model where the adversary is active, she not only eavesdrops the communication but also tries to inject signals or to do anything she is able to influence the communication. We would like to remark that although we consider security in the passive model, our main objective is to derive a framework of estimating the conditional entropy contained in the physical quantities received by the communicating parties, this problem will still be fundamental when addressing the security in the adaptive model, and one may use our framework as a steppingstone to study this problem in the adaptive model; furthermore, we think it still meaningful to resolve this central issue in the passive model, as an example of bridging the gap between theoretical argument and experimental evaluation w.r.t. the security of extracting keys from physical quantities.

A number of techniques for reconciling the errors between the two communicating parties have been shown in the literature. For real valued measurements (as in the wireless key exchange setting), a popular approach employs a “thresholding” methodology [12, 21] where the signal is transformed to a bitstring using only the level of measurements where, for example, deep fades are observed. In such settings it can be shown that the errors are so few that reconciliation becomes easy. Unfortunately, such reconciliation requires interaction that may result in potentially zeroing the conditional entropy, a fact that experimentally
remains open. Other approaches suffer from similar problems, such as [16] that relies on the Cascade protocol or [3] that uses specialized antennas.

We conclude that information theoretic security in all these previous works relies on whether the channel abstraction fits the real-world adversary setting and whether any additional reconciliation interaction performed does not cut from the conditional entropy too much. While ad hoc lower bound assumptions on the conditional entropy of Alice and Bob’s measurements given the view of the adversary are sufficient to argue the security of the key, it is very hard to experimentally verify that such assumptions are indeed true. Indeed, in order to apply the textbook calculation of conditional entropy, one would have to carry experiments an immense number of times. Even worse, one can formally prove that an exponential number of samples are necessary for approximating conditional entropy in the general case (this can be derived from [28]).

In this work we address the above problems with the following contributions.

– First, we introduce a methodology to experimentally argue about the security of key exchange protocols that are based on measuring real-valued physical quantities in the passive model: The crux of our methodology is a falsifiable channel abstraction that is accompanied by an efficient experimental approximation algorithm of the conditional min-entropy of the two parties given the view of the eavesdropper. Specifically, we model measurements and adversarial observations via a hidden Markov model (HMM) and we utilize a dynamic programming approach to estimate the conditional min-entropy; we achieve that through a combination of the Viterbi algorithm [1] and the forward algorithm [18]. This algorithmic approach is beneficial as it allows with only a polynomial number of experiments in the size of the key to argue about the conditional min-entropy without oversimplifying the channel abstraction. Furthermore, the Markovian nature of the channel is falsifiable and is possible to verify experimentally. Given the conditional min-entropy bound, we eventually show security through the calculation of the min-entropy loss during the standard protocol steps of quantization, privacy amplification and information reconciliation. We showcase the result by providing a general protocol following these three steps that rely on “low-distortion” embeddings [2], secure sketches, [12] and randomness extractors [24]. We note that the HMM abstraction is only one out of many ways to instantiate our methodology for arguing about security and there could be others that may be discovered motivated by our work.

– Second, we apply the methodology we put forth above to the setting of wireless key exchange, where the signal strength is the target physical quantity that the two parties measure. We designed an experimental configuration that enabled us to measure the conditional min entropy as well as the correctness of channel abstraction our methodology utilizes. Our experimental configuration included a robotic mobile device equipped with various transceivers that performed measurements of signal strength over long periods of time. Using the data and our methodology we calculated the condi-
tional min-entropy and error rate over time, showing the feasibility of secure key exchange for the type of adversaries used in our experimental setup. Our experimental results enabled us to identify the concrete parameters needed for generating a secure key of a specific length and also predict how these parameters would need to change for generating longer keys.

2 Key Extraction from Measuring Physical Quantities

2.1 Definition

The problem of key extraction from physical measurements can be abstracted as a game between two players, Alice and Bob, who have access to their own separate measurement devices for a certain physical quantity. The physical quantity itself is generated through the actions of Alice, Bob and potentially also the adversary, Eve and for the purpose of this section will remain purposefully undetermined (we will detail a specific instantiation in section 3).

We will use $X_1, \ldots, X_n$ to denote the measurements obtained by Alice, and $X'_1, \ldots, X'_n$ the measurements obtained by Bob. Finally the adversary is also able to obtain measurements of the same physical quantity denoted by $Y_1, \ldots, Y_n$. The value $n$ is arbitrary and the algorithm for extracting the key should allow any choice for this value. The mechanism for determining the appropriate value of $n$, given a certain level of security that needs to be attained is, in fact, an important part of the protocol design problem. Beyond access to the measurements, we assume also that the parties, Alice and Bob have the ability to engage in “public discussion.”, i.e., they can utilize an authenticated channel. The implementation of this channel is separate and orthogonal to our objectives.

Given the above, a $(l, \epsilon_c, \epsilon_u)$ key generation system is a protocol between Alice and Bob running over an authenticated channel so that each party has access to their physical measurement devices and satisfies these properties:

- (Correctness) Alice and Bob both output the same $l$-bit key with probability $1 - \epsilon_c$.
- (Security) Conditional to the protocol view of any adversary the key calculated by the two parties has statistical distance from the uniform distribution over $\{0, 1\}^l$ at most $\epsilon_u$.

2.2 Learning Conditional Min-Entropy

The most important consideration in the process of describing how to extract the key from the physical measurements performed by Alice and Bob is determining the amount of uncertainty that exists in the measurements conditioned on the view of the adversary – if no sufficient uncertainty exists then they cannot be expected to complete the key extraction securely. We first recall the standard definitions of metrics for uncertainty: min-entropy and conditional min-entropy.
Definition 1. The min-entropy $H_\infty(A)$ of a random variable $A$ is defined as $H_\infty(A) = -\log(\max_a \Pr[A = a])$. The conditional min-entropy of $A$ on the event that another random variable $B$ equals a specific value $b$ is $H_\infty(A|B = b) = -\log(\max_a \Pr[A = a|B = b])$. Further, we define the (average-case) conditional min-entropy of $A$ given $B$ as $\tilde{H}_\infty(A|B) = -\log(E_{b\leftarrow B}[2^{-H_\infty(A|B=b)}])$.

Learning conditional min-entropy: Now we proceed to introduce the central problem for our security argument of any key-extraction system from measuring physical quantities. Fix a pair of jointly distributed random variables $X, Y$, the problem of learning/approximating conditional min-entropy is to return an estimation of $\tilde{H}_\infty(X|Y)$ from polynomially many measurements of $X, Y$. Unfortunately, in the general case, the problem is infeasible to solve. It’s shown in [28], that at least $O(\frac{N}{\log N})$ many samples are needed for estimating the entropy where $N$ is the support size. It is easy to see that conditional entropy is even more difficult to learn, since one can reduce the problem of learning the entropy to learning the conditional entropy over a uniformly distributed variable. It follows that in the general case the conditional min-entropy is not learnable efficiently. It is therefore important to identify classes of distributions for which the conditional min-entropy can be learned. Then, when a certain type of physical measurement can be justified experimentally to follow the distribution, our security argument will provide strong evidence regarding the security of key extraction from such measurements. We identify a wide class of distributions in the next section.

2.3 A Conditional Min-Entropy Learner for Markovian Processes

A simple observation is that the conditional min-entropy can be calculated directly (via textbook formula) in time polynomial in the support of $X$ and $Y$. This obviously does not help since the support sets are growing exponentially in $n$. Recall that Alice’s measurements are denoted by $X = X_1, \ldots, X_n$, while adversary Eve’s observations are $Y = Y_1, \ldots, Y_n$, both measurements are composed of a sequence of correlated variables with a relatively small support. A way to simplify the problem is to assume independence among these variables - however still this can be unrealistic. Here we propose a more realistic assumption which would still allow us to learn the conditional entropy efficiently. Our main idea is to consider Alice’s measurements $X_1, \ldots, X_n$ as hidden states of a Markov process, while taking Eve’s observations $Y_1, \ldots, Y_n$ as the visible output. The following are the basic elements of an HMM (see [27] for a survey):

- $k$, number of hidden states
- $S$, set of states, $S = \{s_1, \ldots, s_k\}$
- $m$, number of observed symbols
- $O$, set of observed symbols, $O = \{o_1, \ldots, o_m\}$
- $X = X_1, \ldots, X_n$, sequence of hidden states
- $Y = Y_1, \ldots, Y_n$, sequence of observed symbols
- $A$, the state transition probability matrix, $a_{ij} = \Pr[X_{t+1} = s_j | X_t = s_i]
Given the above we now turn to the conditional min-entropy calculation. From Definition 1 we get:

\[
\tilde{H}_\infty(X|Y) = -\log(E_{y\sim Y}[2^{-H_\infty(X|Y=y)}]) \\
= -\log(E_{y\sim Y}[\max_x Pr(X=x|Y=y)]) \\
= -\log(\sum_{y\sim Y} [\max_x Pr(X=x, Y=y)]) \\
\tag{1}
\]

We first calculate the maximum joint probability for a given sequence of observed symbols \(y_1, \ldots, y_n\): \(\max_{x_1, \ldots, x_n} Pr[X=x_1, \ldots, x_n, Y=y_1, \ldots, y_n]\).

This joint probability can be calculated using the Viterbi algorithm which is a dynamic programming algorithm we briefly describe below.

Consider the objective function \(\delta_t(s_i) = \max_{x_1, \ldots, x_{t-1}} Pr(X_1=x_1, \ldots, X_t=s_i, Y_1=y_1, \ldots, Y_t=y_t)\), our goal is then to calculate \(\max_s \delta_n(s)\), with the following recursive relation: \(\delta_{t+1}(s_j) = \max_{s_i} [\delta_t(s_i)a_{ij}]b_j(y_{t+1})\), which is satisfied by any HMM. The Viterbi algorithm consists of three key steps:

1. Initialization: \(\delta_1(s_i) = \pi_i \cdot b_1(y_1)\)
2. Recursion: \(\delta_{t+1}(s_j) = \max_{s_i} [\delta_t(s_i)a_{ij}]b_j(y_{t+1})\)
3. Termination: \(P^* = \max_s \delta_n(s)\)

It follows that we can calculate the maximum joint probability for a given observation sequence. Still, to get an estimation for the average case min-entropy by applying formula \(1\), we need to sum up all such joint probability for all possible observation sequences. It is very likely that there are exponentially many such sequences. For example, suppose for any state \(s_i\), there are only 3 possible observations, but they are equally possible. In this case, there would be \(3^n\) possible observation sequences with length \(n\).

To circumvent the above, we also study the property of the distribution of conditional entropy for fixed observed sequences (i.e., when conditioned on Eve’s measurements).

Note that the Viterbi algorithm gives only the maximum joint probability, in order to estimate the conditional min-entropy given each observed sequence, we need to further get the probability of appearance for each observed sequence. The forward algorithm \([13]\) can be used to calculate this quantity in an HMM. For a given observed sequence \(y_1, \ldots, y_n\), we define the objective function as: \(\alpha_t(s_i) = Pr[Y_1=y_1, \ldots, Y_t=y_t, X_t=s_i]\), then the probability of the sequence \(Pr[Y_1=y_1, \ldots, Y_n=y_n] = \sum_t \alpha_n(s_i)\). The forward algorithm proceeds as:

- Initialization: \(\alpha_1(s_i) = \pi_i \cdot b_i(y_1)\)
- Induction: \(\alpha_{t+1}(s_j) = [\sum_i \alpha_t(s_i) \cdot a_{ij}]b_j(y_{t+1})\)
- Termination: \(P = \sum_i \alpha_n(s_i)\)

If we assume that the conditional min-entropies on frequently appearing observations are stably distributed, then, the average case min-entropy can be
approximated by the average of conditional min-entropies on fixed observations. In other words, $E_{y \leftarrow Y}[2^{-H_\infty(X|Y=y)}]$ will be almost the same as the value $2^{-H_\infty(X|Y=y)}$ for each $y$, and thus we can approximate $H_\infty(X|Y)$ as the average of $H_\infty(X|Y=y)$ from all the measurements $y$ we observe.

Now, suppose we perform a series of $M$ experiments, each one with $n$ measurements. For the $j$-th experiment, we record the measurements of Eve, and we use them together with the HMM parameters to calculate the maximum of conditional probability as $P^*_j/P_j$, where $P^*_j$ is the maximum joint probability given an observed sequence, returned by the Viterbi algorithm. The conditional min-entropy for this observed sequence is $-\log(P^*_j/P_j)$. In this way, we will use the average of these entropies to approximate the average case min-entropy.

To summarize, with the following channel abstractions, one can learn the conditional min-entropy from only a reasonable number of experiments.

1. First, the measurements are produced from a memoryless Markov process, that is, $\forall i, j$, we have: $\Pr[X_i = x_1|X_{i-1} = x_0] = \Pr[X_j = x_1|X_{j-1} = x_0]$.
2. Second, the observation probability does not change over time: specifically, for all different times $i, j$, $\Pr[Y_i = y|X_i = x] = \Pr[Y_j = y|X_j = x]$.
3. Third, observations are independent, i.e., $\forall n$, $\Pr[Y_1 = y_1, \ldots, Y_n = y_n|X_1 = x_1, \ldots, X_n = x_n] = \prod_{i=1}^n \Pr[Y_i = y_i|X_i = x_i]$.
4. Finally, we assume that the random variable $H_\infty(X|Y=y)$ has small standard deviation (hence its expectation can be calculated via a small number of experiments). We call this the stable conditional entropy assumption.

The advantage of the above approach is that when applied to concrete application scenarios, the properties can be experimentally justified. Then, if the physical measurements do satisfy these properties, we can apply our method to approximate the conditional min-entropy thus laying a basis for secure extraction following the protocol and analysis we present in the next section.

### 2.4 A General Protocol

Next, we will present a general protocol for our key generation process from measuring physical quantities. It has three basic steps. First, $A$ and $B$ produce a sequence of events and their corresponding measurements and then convert their measurements to bitstrings denoted by $\rho_A$ and $\rho_B$, respectively. Subsequently they perform information reconciliation and privacy amplification. Our protocol design enable us to bound the entropy loss that takes place during each step of the execution of the algorithm and ensures the proper extraction of the key.

**Bit Quantization** Each party has at its disposal a series of measurements $X_1, \ldots, X_n$. We put forth a quantization approach that utilizes a low distortion embedding $\tau$ from $\ell_1$-distance metric space into Hamming distance metric space. The $\ell_1$-distance (or Manhattan distance) between two vectors $X, Y$ in an $n$ dimensional real vector space is defined as the sum of the absolute difference between corresponding coordinates, i.e., $\ell_1(X, Y) = \sum_n |X_i - Y_i|$, where $X = (X_1, \ldots, X_n), Y = (Y_1, \ldots, Y_n)$. 
An embedding is a mapping between two metric spaces. It is said to have distortion $c$ if and only if the distance is preserved with up to a multiplicative factor of $c$. Given a low distortion embedding $\tau$, we not only quantize the physical measurements, but also prepare two bitstrings to have a small Hamming distance which is critical for the reconciliation algorithm. Last, it is not hard to construct a low distortion embedding, one can verify the following proposition that a simple unary encoding is a good low distortion embedding.

**Proposition 1.** Suppose for any $i$, $X_i$ is a random variable over a finite set $B \subseteq \mathbb{Z}$ and $m = \max\{|b| \mid b \in B\}$. Consider the mapping $\tau : B \rightarrow \{0, 1\}^m$ defined by applying a unary encoding to the input, i.e., $\tau(x)$ is a bitstring of length $m + 1$ where the first bit represents $\text{sign}(x)$, followed by $m - |x|$ consecutive 0s and $|x|$ consecutive 1’s. Then, $\tau$ is an embedding with distortion at most 1.

**Reconciliation using Secure Sketches** After collecting measurements and performing the bit quantization step, Alice and Bob possess two $N$-bit strings $\rho_A, \rho_B$, respectively. For now, we assume that there exists a $d \in \mathbb{N}$ such that with overwhelming probability the Hamming distance satisfies $H(\rho_A, \rho_B) \leq d$. The value of $d$ in specific protocol can be determined experimentally for a given physical quantity, (e.g, we do that in the next section, see Figure 4 and Table 1).

We next describe our information reconciliation protocol that can be proven correct under the above definitions.

First, recall the definition of a secure sketch by Dodis et al [12]. An $(n, m, m', d)$-secure sketch is a pair of randomized procedures, “sketch” (SS), and “recover” (REC), such that the sketching procedure SS on input $n$-bit string $w$, returns a bit string $s \in \{0, 1\}^*$, and the recovery procedure REC takes an $n$-bit string $w'$ and a bit string $s \in \{0, 1\}^*$, and outputs an $n$-bit string.

SS and REC satisfy correctness and security. In particular, correctness means that if $H(w, w') \leq d$, then $\text{REC}(w', \text{SS}(w)) = w$. Security on the other hand, means that for any distribution $W$ with min-entropy $m$ over $\{0, 1\}^n$, the value of $W$ can be recovered by an adversary who observes $s$ with probability no greater than $2^{-m'}$, i.e., $H_\infty(W|\text{SS}(W)) \geq m'$.

Achieving information reconciliation between two parties holding data $w, w'$ respectively using sketches is simple. The first party transmits $s = \text{SS}(w)$ and the second computes $\text{REC}(w', s)$. Various secure sketches can be designed depending on the metric space. We next recall two secure sketches over the Hamming metric. Let $C$ be an error correcting code over $\mathbb{F}_2^n$ which corrects up to $d$ errors and $|C| = 2^k$. Thus, there is an algorithm Dec that given any $v \in C$ and any $e \in \{0, 1\}^n$ with $Wt(e) \leq d$, it holds that $\text{Dec}(v \oplus e) = v$, where $Wt(\cdot)$ stands for the Hamming weight of $e$. Observe that under our assumption $H(\rho_A, \rho_B)$ is at most $d$ and thus the output of Bob is exactly $\rho_A$. It follows that Alice and Bob recover $\rho_A$ and thus information reconciliation is achieved.

We note that the communication overhead of the above protocol is unnecessarily high. An improvement can be achieved by using a more communication efficient secure sketch. For a $(n, k)$ linear code, the $n \times (n-k)$ parity check matrix $H$ has the property that for every codeword $c$, $c \cdot H = 0$. For any transmitted
message $y = x + e$ of length $n$, the syndrome is defined as $\text{syn}(y) = y \cdot H$. Assuming that the error-codeword $e$ can be easily computed using $\text{syn}(e)$ (doing what is known as syndrome-decoding) the length of the sketch is only $n - k$. Bob computes the syndrome for his own bits, $\text{syn}(\rho_B)$, and finally computes $\text{syn}(e) = \text{syn}(\rho_B) - \text{syn}(\rho_A)$, from which he obtains $e$, and thus $\rho_A = \rho_B - e$. It is not hard to see that the syndrome based construction is an $(n, m, m-n+k, d)$-secure sketch, from the chain rule described in definition 1.

**Privacy Amplification** After both parties have derived the same bitstrings, they must “purify” the strings to make the joint output suitably random as a cryptographic key. For this task, we employ a randomness extractor [24]. We first recall the definition of statistical distance which characterizes how similar two random variables are. Then the level of randomness of a random variable can be measured by the statistical distance of the variable from the uniform distribution. The statistical distance of two probability distributions $X_1, X_2$ with support $S$ is: \[ \max_{T \subseteq S} \left\{ \Pr[X_1 \in T] - \Pr[X_2 \in T] \right\}, \] denoted by $\text{SD}(X_1, X_2)$. It can be also defined as \[ \frac{1}{2} \sum_{s \in S} |\Pr[X_1[s] - \Pr[X_2[s]]|. \] Two distributions $X_1, X_2$ are said to be $\epsilon$-close if $\text{SD}(X_1, X_2) \leq \epsilon$. We now define randomness extractors.

**Definition 2.** A function $\text{Ext} : \{0,1\}^t \times \{0,1\}^r \rightarrow \{0,1\}^l$ is a $(t, s, l, \epsilon)$-extractor if for every random variable $X$ over $\{0,1\}^t$ having min-entropy at least $s$, it holds that $\text{Ext}(X, U_r)$ is $\epsilon$-close to $U_l$, where $U_r, U_l$ denote uniform distributions over $\{0,1\}^r$ and $\{0,1\}^l$ respectively. Further, we say $\text{Ext}$ is an $(t, s, l, \epsilon)$-strong extractor if $\text{SD}((\text{Ext}(X, U_r), U_r), (U_l, U_r)) \leq \epsilon$. Finally, $\text{Ext}$ is an average-case $(t, s, l, \epsilon)$-strong extractor if the above holds when $\tilde{H}_\infty (X|Y) \geq s$ for some random variable $Y$ that is known to the adversary.

The following lemma shows that there are efficient constructions of (average case) strong extractor and thus we may utilize a public random seed to obtain a close to uniform key.

**Lemma 1.** Universal hash functions [10] are average case $(t, s, l, \epsilon)$ strong extractors whenever $s \geq l + 2 \log\left(\frac{1}{\epsilon}\right) - 2$.

### 3 Application: Physical Layer Key Extraction with Experimental Security Analysis

We now provide a concrete key extraction protocol from wireless signals, and analyze our protocol using the methodology proposed above. Finally we instantiate our protocol with concrete parameters. The justifications for all the assumptions we make using real experimental data are in the appendix.

#### 3.1 Key Extraction Protocol from Wireless Signal Envelope

Two nodes Alice and Bob execute a ping-like protocol, they send each other messages through a wireless channel and measure the received wireless signal
| Alice | Bob |
|-------|-----|
| $X_1$ | $\xleftarrow{\text{ping}}$ | $\xrightarrow{\text{pong}}$ | $X'_1$ |
| $\vdots$ | $\vdots$ |
| $X_n$ | $\xleftarrow{\text{ping}}$ | $\xrightarrow{\text{pong}}$ | $X'_n$ |

$s \xleftarrow{\{0, 1\}^r}$

$\rho_A = \tau(X_1, \ldots, X_n)$  $\rho_B = \tau(X'_1, \ldots, X'_n)$

$u = \text{syn}(\rho_A)$  $\xrightarrow{u, s}$  find $e$, s.t. $\text{syn}(e) = \text{syn}(\rho_B) - u$

Return $\text{key} = \text{Ext}(\rho_A, s)$  Return $\text{key} = \text{Ext}(\rho_B - e, s)$

**Fig. 1.** Our physical-layer key extraction. $\text{syn}$ is the algorithm computing the syndrome w.r.t. a public error-correcting code, $\tau(\cdot)$ is a low-distortion embedding into Hamming metric space, $s$ is the short random seed for extractor $\text{Ext}$.

...strength. Then they apply the general protocol of the previous section to those measurements to extract a key which will later be used for their secure communication. Note that since the actually messages communicated do not affect the exchange, the measurements can be collected during a regular plaintext communication between Alice and Bob. This means that key-exchange in this setting can piggyback on top of a plaintext communication. Our concrete key extraction protocol from received wireless signal strength is presented in Figure 1.

### 3.2 Experimental Setup

The hardware basis for our experimental platform was the Crossbow MICAz sensor mote, which contains a Chipcon CC2420 IEEE 802.15.4-compliant RF transceiver. The RF transceiver in our experiments operated at a frequency of 2.48 GHz and with a data rate of 250 kbps. Transmit power was set to the maximum level of 0 dBm for each MICAz device in our test network.

The software portion of our experimental platform primarily consisted of a ping-like application that we implemented on top of the TinyOS operating system. Our implementation consists of three components: a base station (Alice), a mobile client (Bob) and an eavesdropper implementation (Eve). Bob sends a PING frame containing a 4-byte sequence number to Alice, who records the sequence number and the received frame’s RSSI value as computed by the CC2420 transceiver. Alice then sends a PONG frame to Bob containing the sequence number from Bob’s original PING request. Bob records the response frame’s RSSI value, waits for 1300ms, increments the sequence number and repeats the above process. If Bob does not receive a response within 500ms of transmitting his ping request, he will increment the current sequence number and transmit a new ping request. The 1300ms delay between PING frames sent by Bob is intended to
ensure that our assumption of independence between non-consecutive channel samples apart holds true, which we experimentally evaluate in Section A.

The eavesdropper implementation, Eve, simply listens passively for any frames transmitted by any other nearby nodes. Eve records the source and destination addresses, frame type (i.e., PING or PONG), sequence number and observed RSSI value for that frame. After the experiment is manually terminated, the recorded data is uploaded from each sensor mote to a laptop for further processing and analysis. We resolve lost frames during post-processing by computing the intersection between frame sequence numbers observed by Alice and those observed by Bob. In practice, Alice and Bob could use an acknowledgement-based protocol (e.g., TCP) to detect lost frames; however, we opted for a post-processing approach for simplicity of implementation.

![Figure 2](image)

**Fig. 2.** Floorplan for our indoor test environment and layout of sensor motes. Node 1 is the stationary base station (Alice) and Node 2 (Bob) is fixed to a robotic platform that moves randomly throughout the test environment. The remaining five nodes are passive eavesdropper nodes (Eve) that simply record all observed frames and their observed RSSI values.

We conducted two separate data collection experiments in a basic indoor residential environment. In each experiment, we placed the stationary base station (Alice) in the middle of the environment with five eavesdropper nodes placed around the base station as shown in Figure 2. The mobile node (Bob) was mounted to a purpose-built robotic platform that was programmed to move throughout the environment in a random pattern for the duration of the experiment at a rate of approximately 14 cm/sec while avoiding obstacles. The robot would move forward for a period of time selected uniformly at random from between 5 and 15 seconds. At the end of the randomly selected travel time, the robot would turn at an angle selected uniformly at random from between -75 to 75 degrees indicating a left or right turn, respectively. If the robot detected an obstacle via its front-mounted bump sensors or ultrasonic sensor before the time expired, the robot would similarly choose a new direction and travel time at random. The travel time and turn angles were selected primarily based on the small size of the test environment.
Each sensor mote can store up to 25,000 channel measurements and so, with a 1300ms delay between PING frames, each experiment represents approximately nine hours of continuous movement over a total distance traveled of around 4 kilometers per experiment. The data for each node from both experiments were concatenated together resulting in slightly less than 50,000 samples for each node after accounting for lost or dropped frames. Note that the number of samples we collected is for the purpose of our security analysis; an actual key can be computed much faster. See Section 3.4 for more details on the number of samples required for key extraction.

3.3 Security Analysis of our Physical Layer Key Extraction Protocol

Each experimental dataset was downsampled by six samples in order to simulate a greater idle time between PING frames and ensure that a channel sample $X_i$ is correlated only with $X_{i-1}, X_{i+1}$, the reasoning for which will be explained in Section A. After downsampling, we still have around 8100 samples in the experiment. We took 8000 samples and divided them into 100-sample slices, each slice representing a single key experiment.

**Learning the conditional min-entropy:** First, we estimate the average case min-entropy of Alice’s channel measurements given a single adversary’s view with the methodology proposed in Section 2.3 by incorporating the data from the key experiments. In our setup, there are 5 adversarial nodes and we considered the entropy calculation for each one separately. All our measurements were calculated for 32 distinct signal levels. Moreover, to make our adversarial model stronger, we simply ignored for both Alice and Bob all samples that were not also observed by Eve. By doing so, we ensure that we could still obtain security without relying on frames being missed by the eavesdropper.

For every experiment of 100 samples, we first compute the maximum conditional probability of the observed sequence by using the Viterbi algorithm and forward algorithm as explained in Section 2.3, then get the conditional min-entropy for the 80 key experiments, and use the average to approximate the average case min-entropy because of the assumption that the conditional min-entropies on specific observations are stably distributed (experimental justification will be shown in Figure 6).

Since $X_i$, the random variable representing Alice’s $i$-th sample, follows the same distribution for all $i$ (will be justified in Section A), we get the initial state distribution $\pi$ by counting throughout all channel samples. We prepare the state transition probability matrix $A$ by estimating entry $a_{ij}$ by $\frac{n_j}{n_i}$, where $n_j$ is the number of measurements equal to $s_j$ such that the next measurement equals $s_j$, and $n_i$ is the number of measurements equal to $s_i$. Similarly, we approximate entry $b_j(k)$ for the observation probability distribution by $\frac{m_j}{m_k}$ where $m_j$ is the number of Alice’s measurements equal to $s_j$, and $m_k$ is the number of Eve’s observations equal to $a_k$ while the corresponding measurements of Alice is $s_j$. This is reasonable since the observation probability does not change (the independent observation assumption).
With all these parameters, we next apply the Viterbi algorithm and forward algorithm to compute the maximum conditional probability max
\[ \text{Pr}[X_1 = x_1, \ldots, X_{100} = x_{100} | y_1, \ldots, y_{100}] \]
for every experiment, where \( y_1, \ldots, y_{100} \) are Eve’s measurements for a 100-sample slice. We then apply minus logarithm to those probabilities, and average them to get our final approximation for the average case min-entropy contained in 100 samples of Alice. Comparisons for the five nodes are shown in the sixth column of Table 1.

| Node | \( M \) | \( n \) | \( N \) | \( d \) | \( \bar{H} \) | \( R_e \) | \( R \) |
|------|-----|-----|-----|-----|------|------|------|
| 3    | 80  | 100 | 800 | 4.42| 88.40| 0.55%| 11.05%|
| 4    | 80  | 100 | 800 | 4.29| 99.82| 0.54%| 12.48%|
| 5    | 80  | 100 | 800 | 4.36| 98.89| 0.55%| 12.36%|
| 6    | 80  | 100 | 800 | 4.28| 93.07| 0.54%| 11.63%|
| 7    | 80  | 100 | 800 | 4.35| 94.08| 0.54%| 11.76%|

Table 1. Comparison of conditional min-entropy for the 5 eavesdropping nodes. \( M \) denotes the number of experiments, \( n \) is number of channel samples in each experiment, \( N = |\rho_A| \) is the length of bit string after bit quantization, \( d \) is the average number of word errors, \( \bar{H} \) is the average case min-entropy, \( R_e \) is the average error rate, which is \( d/|\rho_A| \), and \( R \) is the average entropy rate. The node IDs are as shown in Figure 2.

**Accounting for Entropy Loss:** With a reasonable amount of entropy contained in Alice’s channel measurements, conditioned on adversary’s view, we now proceed to calculate the entropy loss for each step, and subtract those losses to get the final result about security of the key.

First, for the bit quantization step, the way we defined the embedding \( \tau \), it is a one-to-one map so, obviously, there is no entropy loss during this step. Thus,

\[ \bar{H}_\infty(\rho_A | Y) = \bar{H}_\infty(X | Y) \]

Next we consider the entropy loss in the information reconciliation step. Suppose an \((n, k)\) binary code is used to construct the syndrome based secure sketch \( u \) described in the protocol shown in figure 1, a total of \( |u| = n - k \) bits are transmitted during the information reconciliation step. Thus,

\[ \bar{H}_\infty(\rho_A | Y, u) = \bar{H}_\infty(X | Y, u) \geq \bar{H}_\infty(X | Y) - |u|. \]

Finally, if there is enough entropy left, and we want to get a key with length \( t \) and achieves a security level of \( \epsilon_u \), i.e., the distribution of the key is \( \epsilon_u \) close to the uniform distribution over \( \{0, 1\}^t \), according to lemma 1 if \( \bar{H}_\infty(\rho_A | Y, u) \geq s \geq l + 2 \log \frac{1}{\epsilon_u} \), we can apply a \((t, s, l, \epsilon_u)\) strong average case extractor such as a universal hash to extract an almost uniform key with length \( l \).

It is not hard to derive a sufficient condition for extracting a secure key by summarizing the above conditions. Note that as long as the error rate is not too high, and there is a non-zero entropy left after the phase of information reconciliation, it is always possible to amplify the entropy by collecting more samples.
before the bit quantization step, thus leaving room for privacy amplification. We provide concrete working parameters for our experimental setup in section 3.4.

![Graph](image.png)

**Fig. 3.** Linear growth of conditional min-entropy for node 4, x-axis is the number of samples in the sequence, y-axis is the value of conditional min-entropy, the (blue) points are average of conditional min-entropy in 40 experiments, the (red) lines start from mean-standard deviation and extend to mean+standard deviation.

### 3.4 Instantiation of our Protocol

In this section, we will concretely instantiate our general protocol for physical layer key extraction.

We first study the property of growth of conditional entropy in a block of samples with different size. By taking 10 samples as a unit, we slice the 8000 samples into 40 pieces, each representing an individual experiment with 200 samples. We then calculate conditional min-entropy (of 10 samples, 20 samples, and so forth until 200 samples) using our methodology by averaging among these 40 experiments. In this way we get the plot of the average conditional min-entropy with standard deviation of the 40 values as shown in Figure 3. We use the method of least squares to get an approximation of the average case min-entropy as a linear function of the number of samples and obtain the following equation: \( g(x) = \frac{985}{1000}x + \frac{1467}{1000} \), which means there is \( g(x) \) bits average case min-entropy in every \( x \) samples.

Similarly, we study the distribution of word (8-bit block) errors. We first divide the bitstrings \( \rho_A, \rho_B \) in 8-bit blocks, and count the number of different blocks between \( \rho_A, \rho_B \). The result is presented in Figure 4. In this way we obtain an approximation of the average word error \( e(x) = \frac{43}{1000}x + \frac{48}{1000} \), which means there are \( e(x) \) word errors on average in \( x \)-sample experiment.
Our results experimentally show the linear growth of conditional min-entropy and that of errors. We can now provide an explicit instantiation of our protocol.

**Theorem 1.** Suppose \( n \) is number of samples in our protocol. Using a Reed-Solomon code over \( F_{2^8} \) for the sketch, when \( n \geq \max(12.54\lambda + 6.27\ell - 3.24, 2326\ell - 3) \), we can successfully implement a \((l, e^{-c}, 2^{-\lambda})\) key generation system.

**Proof.** Given the estimated conditional entropy rate as \( g(n) = \frac{985}{1000} n + \frac{1467}{1000} \), and error rate as \( e(n) = \frac{38}{1000} n + \frac{48}{1000} \), we choose a RS code over \( F_{2^8} \) which can correct up to \( \frac{6}{5} e(n) \) errors (slice the bitstring into proper sized blocks if needed). Thus, the entropy loss of applying the code would be \( 8 \cdot \frac{6}{5} e(n) \), the entropy loss of applying extractor is \( 2\lambda \), and similar to the analysis at the end of section 3.3, we immediately get: if \( 159.4n + 549.4 > 1000l + 2000\lambda \), we will have a \( l \)-bits key with \( 2^{-\lambda} \) distance to \( U_l \). For correctness, from the Chernoff bound, the probability of having \( \frac{6}{5} e(n) \) errors is \( e^{-\frac{1}{1000}} \), we can easily derive another condition.

To put the above into perspective, if we set \( \epsilon_c = e^{-1}, \epsilon_u = 2^{-80} \), we can do the following: first we make around 1800 measurements, then use a 8-bit unary encoding. (Since we take 5-bit precision, and in our experiments almost all (99.95%) RSSI fall in [-8,0]. Also, we ignore the bit for sign, since they are all negative). Finally, we employ a (255,229)-RS code for secure sketch and apply a 2-universal hash function \([10]\) as randomness extractor and harvest a 128-bit key. Note that the success rate of the exchange is at least \( 1 - \frac{1}{e} \) and is detectable by the two parties hence in case of failure the extraction is repeated.

**4 Conclusions and Future work**

In this paper we presented a framework for a rigorous experimental security analysis that guarantees the agreed key extracted from physical quantities has reasonable level of security, and we show an application of the framework in the setting of extracting key from wireless signals and instantiate concrete parameters from our experimental data. Our novel methodology relies on hidden
Markov models and the estimation of conditional min-entropy through a dynamic programming approach that relies on the Viterbi and forward algorithms.

We believe that our work lays for the first time a comprehensive methodology for arguing experimentally the security of physical-layer key extraction against passive eavesdropper nodes and can be applied to a number of other similar protocols in a similar way as we demonstrated here. At a very high level, our methodology entails the following basic steps, (i) employ a falsifiable channel abstraction, equipped with an efficient conditional entropy approximation algorithm and conduct experiments to estimate the conditional min-entropy with the algorithm (ii) justify experimentally the channel abstraction, (iii) account for the entropy loss incurred due to quantization and information reconciliation.

While we believe the above are a significant step forward in the security analysis of physical-layer key extraction we recognize the still limited evidence for the security for these protocols. There exist still a multitude of open questions and our methodology can be extended in a number of directions: (i) increase the size of the HMM or even substitute the HMM with more general Bayesian networks to improve the accuracy of the estimation of conditional min-entropy and potentially decrease the time needed for secure key generation. (ii) consider the case of active adversaries and develop protocols with adversarial interference detection capability for which conditional min entropy can still be effectively estimated. (iii) consider observations from multiple eavesdropper nodes simultaneously.

References

1. A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, 13:260–269, 1967.
2. G. Anupam, K. Robert, and L. J. R. Bounded geometries, fractals, and low-distortion embeddings. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, FOCS ’03, pages 534–543, 2003.
3. T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka. Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels. *Antennas and Propagation, IEEE Transactions on*, 53(11):3776 – 3784, nov. 2005.
4. B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener. Robust key generation from signal envelopes in wireless networks. In *ACM CCS*, pages 401–410, 2007.
5. J. Barros and M. R. D. Rodrigues. On the secrecy capacity of wireless channels. In *Proceedings of the IEEE International Symposium on Information Theory*, 2006.
6. C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.
7. C. H. Bennett, G. Brassard, and J.-M. Robert. Privacy amplification by public discussion. *SIAM J. Comput.*, 17(2):210–229, 1988.
8. M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin. Wireless information-theoretic security. *IEEE Transactions on Information Theory*, 54(6):2515–2534, 2008.
9. G. Brassard and L. Salvail. Secret-key reconciliation by public discussion. In *EUROCRYPT*, pages 410–423, 1994.
1. L. Carter and M. N. Wegman. Universal classes of hash functions. *J. Comput. Syst. Sci.*, 18(2):143–154, 1979.

2. M. Clark. Robust wireless channel based secret key extraction. In *MilComm’12*, pages 1–6.

3. Y. Dodis, J. Katz, L. Reyzin, and A. Smith. Robust fuzzy extractors and authenticated key agreement from close secrets. In *CRYPTO*, pages 232–250, 2006.

4. N. Döttling, D. E. Lazich, J. Müller-Quade, and A. S. de Almeida. Vulnerabilities of wireless key exchange based on channel reciprocity. In *WISA*, pages 206–220, 2010.

5. S. Eberz, M. Strohmeier, M. Wilhelm, and I. Martinovic. A practical man-in-the-middle attack on signal-based key generation protocols. In *ESORICS*, pages 235–252, 2012.

6. M. Edman, A. Kiayias, and B. Yener. On passive inference attacks against physical-layer key extraction. In *Proceedings of the 2011 European Workshop on System Security (EUROSEC ’11)*, 2011.

7. S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. On the effectiveness of secret key extraction from wireless signal strength in real environments. In *MOBICOM*, pages 321–332, 2009.

8. A. Khisti, S. N. Diggavi, and G. W. Wornell. Secret-key generation using correlated sources and channels. *IEEE. Trans. on Inf. Forensics and Security*, 2011.

9. M. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73:360–363, 1967.

10. U. M. Maurer. Secret key agreement by public discussion from common information. *IEEE Trans on Information Theory*, 39(3):733–742, 1993.

11. N. Nisan and D. Zuckerman. Randomness is linear in space. *J. Comput. Syst. Sci.*, 52(1):43–52, 1996.

12. 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. Mob. Comput.*, 9(1):17–30, 2010.

13. S. Mathur, W. Trappe, N. B. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In *MOBICOM*, pages 128–139, 2008.

14. P. K. Gopala, L. Lai, and H. ElGamal. On the secrecy capacity of fading channels. *IEEE Transactions on Information Theory*, 54:4687–4698, 2008.

15. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

16. G. Valiant and P. Valiant. Estimating the unseen: an n/\(\log(n)\)-sample estimator for entropy and support size, shown optimal via new clts. In *STOC*, pages 685–694, 2011.

17. Q. Wang, H. Su, K. Ren, and K. Kim. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In *INFOCOM*, pages 1422–1430, 2011.
A Justification of Channel Assumptions

We now argue that all channel assumptions stated in Section 2.3 hold true based on the analysis of the experimental data collected in the test environment described in Section 3.2.

Stationary Memoryless Markov Process Assumption: We split assumption 1 to three sub-assumptions. Let us begin with the assumption that every channel measurement $X_i$ depends only on the previous measurement $X_{i-1}$ for all $i$. As mentioned in our description of our test platform, we used an artificial delay of 1300ms between ping frames in order to induce independence between channel samples greater than one sample apart. To evaluate this assumption of independence we performed the following analysis. We selected some $i \in [1+m, n]$ uniformly at random, where $n$ is the number of samples in a node’s measurements, and $m = 500$. We then created a $(m+1)$-length row vector as $\{X_i, X_{i-1}, \ldots, X_{i-m}\}$, where $X_{i-m}$ is the $(i-m)$-th sample from a node’s channel measurements. This was repeated 10,000 times, concatenating each of the 10,000 row vectors together to create a $10000 \times (m+1)$ matrix $A$.

The Pearson’s linear correlation coefficients of the resulting column vectors in $A$ were then computed to find the correlation of $X_i$ with $X_{i-k}$ for each $k \in [1, 500]$. From the correlation analysis of our experimental data, we found that there exists a statistically significant correlation between samples $X_i$ and $X_{i-k}$ at a significance level of $\alpha = 0.05$ for values of $k$ up to 6. Since we want to ensure that $X_i$ and $X_{i-k}$ are independent for $k > 1$ for our channel assumptions to hold, we further downsample the collected data by a factor of six in order to induce this level of independence (i.e., ensuring at least 7800ms between samples). The remainder of our experimental analysis is performed on this downsampled data.

The correlation among signals under downsampled data is shown in Figure 5.

Next, we show that if we treat every $X_i$ as a different random variable, they each follow an identical distribution. For this evaluation, we employ the two-sample Kolmogorov-Smirnov (K-S) test which accepts as input two input vectors $x_1$ and $x_2$ and evaluates the null hypothesis that $x_1$ and $x_2$ are from the same distribution against the alternative hypothesis that $x_1$ and $x_2$ are from different distributions. We randomly partitioned each node’s set of channel measurements $X$ into two non-overlapping vectors $x_1$ and $x_2$, each of length $|x_1| = |x_2| = \lfloor n/2 \rfloor$.
where \( n = |X| \) is the total number of channel measurements. We then performed a two-sample K-S test on \( x_1 \) and \( x_2 \) again using a significance level of \( \alpha = 0.05 \). This process was repeated 10,000 times each for Alice’s and Bob’s measurements (i.e., a total of 20,000 trials). Of those 20,000 trials, we had to reject the null hypothesis that \( x_1 \) and \( x_2 \) are identically distributed in 0.54% of the trials. While we did reject the null hypothesis in a non-zero percentage of trials, the results still lend strong support to the assumption that each node’s channel samples \( X_i \) are identically distributed throughout each experiment.

Finally, we consider the assumption that the Markov process transition probability is stationary. That is, we want to show that \( \Pr[X_i = x_1 | X_{i-1} = x_0] = \Pr[X_j = x_1 | X_{j-1} = x_0] \), holds for any \( i \) and \( j \) with \( i \neq j \). We randomly selected some \( i \) and \( j \) in \([1, N - 1]\) uniformly at random and append the channel sample \( X_{i+1} \) to a vector \( x_1 \) and \( X_{j+1} \) to a vector \( x_2 \). This was repeated 10,000 times to create two 10,000-element vectors. We then applied the two-sample K-S test to \( x_1 \) and \( x_2 \) to determine whether the sample distributions are identical. We found that in every test case the resulting distributions are identically distributed at a confidence level of \( \alpha = 0.05 \).

**Stationary Observation Probability Assumption:** Next, we consider our assumption that the channel sample observation probability is stationary. Specifically, we want to show that \( \Pr[Y_i = y | X_i = x] = \Pr[Y_j = y | X_j = x] \) holds true for any two sample indices \( i \) and \( j \) with \( i \neq j \), where \( x \) and \( y \) indicate concrete RSSI values observed by Alice and Eve, respectively. We can easily find the correlation between \( Y_i \) and \( X_i \) for \( k \in [-500, 500] \) for each Eve node using the same approach based on computing the linear Pearson’s correlation coefficient that we described above for evaluating the first assumption. The results were averaged across all five eavesdropper nodes. We found that an eavesdropper measurement \( Y_i \) was statistically independent from some sample \( X_j \) collected by Alice for \( i \neq j \) at a significance level of \( \alpha = 0.05 \).

We use a technique similar to those above to evaluate whether \( \Pr[Y_i | X_i] = \Pr[Y_j | X_j] \) for any two sample indices \( i \) and \( j \) with \( i \neq j \). We first partition the set of sample indices \( I = \{1, 2, \ldots, N\} \) into two equal-sized random subsets \( I_1' \) and \( I_2' \) such that \( |I_1'| = |I_2'| = \lfloor n/2 \rfloor \). Eve’s channel measurement set \( Y \) was similarly partitioned into two subsets \( Y_1' \) and \( Y_2' \), such that \( Y_1' \) contains the
measurements in $Y$ corresponding to the channel sample indices in $I'_1$ and $Y'_2$ contains the measurements in $Y$ corresponding to the channel sample indices in $I'_2$. Alice’s channel measurements were partitioned into two subsets $X'_1$ and $X'_2$ in the same manner. We then compute the joint distributions between $X'_1$ and $Y'_1$, as well as between $X'_2$ and $Y'_2$. For each RSSI value $x \in [-24, 0]$ (the range of possible RSSI values in our experimental platform), we use a two-sample K-S test to determine whether the two distributions $\Pr[Y'_1|X'_1 = x]$ and $\Pr[Y'_2|X'_2 = x]$ are identical. This process was repeated a total of 10,000 times for each of the five eavesdropper sets $Y$ (i.e., a total of 50,000 trials). Out of the 50,000 trials, we accepted the null hypothesis that the two distributions were identical 100% of the time with a significance level of $\alpha = 0.05$ and so we conclude that the assumption $\Pr[Y_i|X_i] = \Pr[Y_j|X_j]$ holds true when $i \neq j$.

**Independent Observation Assumption:** This states that the dependency that potentially exists between the adversary’s measurements comes strictly from the existing dependency of the corresponding measurements of Alice and not any other source. Note that from assumption 2, each $Y_i$ follows a distribution that depends only on the value of $X_i$ and this dependency does not change over time. This suggests that there exists a probabilistic function $F$ over $X_i$ which can simulate the observations of Eve, i.e., $Y_i = F(X_i)$. In such case we prove the independent observation assumption as follows:

$$
\Pr[Y_1 = y_1, \ldots, Y_n = y_n|x_1, \ldots, x_n] \\
= \Pr[F(X_1) = y_1, \ldots, F(X_n) = y_n|x_1, \ldots, x_n] \\
= \Pr[F(x_1) = y_1, \ldots, F(x_n) = y_n] \\
= \prod_{i=1}^{n} \Pr[F(x_i) = y_i] = \prod_{i=1}^{n} \Pr[F(X_i) = y_i|X_i = x_i] \\
= \prod_{i=1}^{n} \Pr[Y_i = y_i|X_i = x_i]
$$

**Stable Conditional Min-Entropy Assumption:** The intuition for assumption 4 is that even though at some location, some observations may have better advantage to predict the original measurement, the sequence would be long enough, and thus when averaging across the whole sequence, the advantage will diminish. To experimentally justify this assumption, we calculate the min-entropy of the sequence received by node1 conditioned on node4’s observation in each experiment (represented by each 100-sample block). We observe that it is stably distributed as shown in figure 6.
Fig. 6. Distribution of conditional min-entropy in node4. x-axis is the index of experiment, y-axis is the value of conditional min-entropy, 300 is the maximum entropy contained in the 100 samples