Adaptive Wireless Channel Probing for Shared Key Generation based on PID Controller

Yunchuan Wei, Kai Zeng, *Member, IEEE,* and Prasant Mohapatra, *Fellow, IEEE*

Abstract—Generating a shared key between two parties from the wireless channel is an increasingly interesting topic. The process of obtaining information from the wireless channel is called channel probing. Previous key generation schemes probe the channel at a preset and constant rate without any consideration of channel variation or probing efficiency. In order to satisfy the users' requirements for KGR and to use the wireless channel efficiently, we propose an adaptive channel probing scheme based on the Proportional-Integral-Derivative (PID) controller, which is used to tune the probing rate. Moreover, we use the Lempel-Ziv complexity to estimate the entropy rate of channel statistics (Received Signal Strength, RSS), which is considered as an indicator of probing efficiency. The experimental results show that the controller can dynamically tune the probing rate and, meanwhile, to achieve a user desired KGR. It stabilizes the KGR at the desired value with error of less than 1 bit/s. Besides, channel probing process is efficient under different user velocities, motion types and sites.

Index Terms—wireless channel probing, shared key generation, information theory, PID controller.

1 Introduction

Generating a shared key between two parties without pre-shared secret over a public channel is a challenging problem in symmetric key cryptography systems. Diffie-Hellman key exchange protocol is widely used for this purpose. However, it works under the assumption of the hardness of the discrete logarithm problem, which has been proven breakable in polynomial time using quantum computers [1]. Although realistic quantum computers may not become reality in years, it is desirable to search for other key agreement mechanisms which do not depend on the assumption of computational hardness. Furthermore, in practical implementations, Diffie-Hellman key exchange protocol may not produce a truly random key due to the use of pseudorandom generators.

With the spur of wireless communications, there is an increasing interest in generating a shared key between two parties from the wireless channel [2], [3], [4], [5], [6]. Two wireless entities exploit the reciprocity, randomness, and location-specific properties of a wireless fading channel, and obtain highly correlated channel states and produce shared key. A third party, who is more than half a wavelength away from the legitimate users, could eavesdrop but would be hard to generate the same key in rich scattering environments [7], [8]. Therefore, unlike the Diffie-Hellman key exchange protocol, generating key from the wireless channel is information-theoretically secure, i.e., no matter how much computing resources the attacker has, it is hard for the attacker to break the key even if it eavesdrops all the key generation messages exchanged between the two entities.

In recent implementations and experiments, the Received Signal Strength (RSS) is widely used as the parameter of the wireless channel to generate the shared key [2], [4], [5], [6]. The RSS can be easily obtained from current wireless device drivers, so it makes key generation using off-the-shelf devices feasible. In order to generate a shared key, two parties need to send channel probing frames to each other and measure the RSS. We call this process as *channel probing*. After this process, both parties quantize the measured RSS sequences into bit streams, and apply the information reconciliation method to make the bits agreed. Finally, they apply the privacy amplification method to discard the bit information revealed to an eavesdropper, and then generate a shared key.

As far as we know, all the existing key generation implementations probe the channel at a preset and constant rate without any consideration of channel variation or probing efficiency. On the one hand, if the channel does not change very frequently or drastically, even if a user can probe the channel at a high probing rate, it will get an RSS sequence with many consecutive duplicated values. These duplicated RSS values do not contribute new bit information to the final key, thus result in a low probing efficiency. On the other hand, it will take an intolerably long time to generate a key if the probing rate is too low.

The key generation rate (KGR) measures the number

- *Y. Wei is with the School of Automation, Beijing Institute of Technology, Beijing, China, 100081.*
E-mail:weiyunchuan1983@bit.edu.cn
- *K. Zeng is with the Department of Computer and Information Science, University of Michigan at Dearborn, MI, 48128.*
E-mail:kzeng@umich.edu
- *P. Mohapatra are with the Department of Computer Science, University of California at Davis, CA, 95616.*
E-mail:pmohapatra@ucdavis.edu

This work was partly presented at the IEEE Conference on Computer Communications (INFOCOM), Shanghai, China, April, 2011. This research was supported in part by the US National Science Foundation through the grant CNS-0709264, the US Army Research Office through the MURI grant W911NF-07-1-0318, the Chinese National Natural Science Foundation (61120106010, 60925011), and the China Scholarship Council (2009603052).

of shared secret bits generated per second between two parties. The KGR is largely determined by the channel variation and probing rate, and partially by the quantization, reconciliation and privacy amplification methods. S. Jana et. al [3] also verified that the bit extraction is affected by environmental dynamics and location characteristics. For example, bits extracted in dynamic environments showed a much higher secret bit rate than static environments. In this paper, a mathematical model of KGR is built, and the simulation results show a proportional relationship between probing rate and KGR.

The entropy rate is the time density of the average information in a stochastic process [9]. We consider the entropy rate as an indicator of probing efficiency. Channel probing process with low entropy rate obviously wastes network resource and computing power. We aim to strike a trade-off between KGR and probing efficiency.

In practice, since users always have requirements of how much time they can afford to generate a certain length of key, in order to satisfy users' requirements for KGR and to use the wireless channel efficiently, we introduce an adaptive channel probing scheme based on Lempel-Ziv complexity (LZ76) [10] and Proportional-Integral-Derivative (PID) controller. Our scheme uses LZ76 to estimate the entropy rate of the channel statistics (i.e., Received Signal Strength, RSS), and uses the PID controller to tune the probing rate. Since the classical definition of entropy rate is based on an asymptotic limit, it does not necessarily lead to an accurate estimator in the case of a finite-size time series [11]. However, LZ76 is a statistical estimator that is unbiased and converging fast enough to be accurate on a finite data sample. The PID controller is a generic feedback control loop mechanism widely used in industrial control systems. It is used to dynamically tune the probing rate in order to stabilize the output (KGR) under dynamic environment.

Our experimental results show that our adaptive channel probing scheme can adaptively tune its probing rate according to user mobility and environmental dynamics. Moreover, it can stabilize KGR by using the PID controller and satisfy the users' KGR requirements.

The contributions of our paper are:

- Mathematical model of KGR is built, and the proportional relationship between probing rate and KGR is verified through simulation.
- Desired KGR is satisfied by using a PID controller to adaptively tune the probing rate in different scenarios.
- Using LZ76 to estimate entropy rate of RSS sequence.

The rest of this paper is organized as follows. We discuss the related works in Section 2. Then we give the system model, adversary model and problem definition in Section 3. We derive the relationship between the probing rate and KGR from theoretic aspect in Section 4. Section 5 shows the workflow of our adaptive probing scheme. We present the experimental setup in section 6 and the results and evaluation in Section 7. We conclude this paper and discuss future work in Section 8.

2 Related Work

There has been an increasing interest in exploiting the randomness and reciprocity of the wireless channel to generate shared keys between two parties [2], [3], [4], [5], [6], [12], [13], [14], [15], [16], [17], [18]. Early research in this area focused on theoretical analysis [15], [16], [17], while most recent works were more interested in practical implementations of the key generation schemes using off-the-shelf wireless devices [2], [3], [4], [5]. Previous work assumed an authenticated channel while generating shared keys [12], [13], [14]. One recent work removed this assumption and proposed a shared key generation algorithm using level-crossings and quantization to extract secret bits from an unauthenticated wireless channel [4]. Another work proposed a method for key generation based on phase reciprocity of frequency selective fading channels [18].

To the best of our knowledge, there is no previous work discussing the trade-off among the KGR and channel resource consumption, nor adaptively tuning the channel probing rate according to the channel dynamics introduced by users' movement and/or the environment. In this paper, we address these problems and achieve adaptive channel probing in real scenarios using off-the-shelf devices.

3 System Model and Problem Definition

3.1 System Model

We consider two legitimate users, Alice and Bob, who want to generate a shared secret key at a target key generation rate (KGR) using the channel characteristics (RSS). Each of them has single antenna. Alice and Bob apply the following four steps to generate a key: channel probing, quantization, information reconciliation, and privacy amplification [3].

Channel probing is used to collect channel characteristics by legitimate users, who have an advantage compared with an eavesdropper, Eve [19]. We use the received signal strength (RSS) as the channel characteristics. In this step, Alice and Bob exchange request/reply probing frames for a duration, say T_p seconds. One of them instantly echoes a reply when he/she receives the request from the other. We assume there is a fixed interval, τ, between any two consecutive request (or reply) probing frames in one probing duration. The channel probing rate f is thus $1/\tau$. At the end of channel probing process, we assume Alice and Bob make N pairs of channel measurements. At Alice and Bob sides, they get, respectively,

$$\begin{aligned}\hat{\mathbf{h}}_{ab} &= \{\hat{h}_{ab}[1], \hat{h}_{ab}[2], ..., \hat{h}_{ab}[N]\}^T,\\ \hat{\mathbf{h}}_{ba} &= \{\hat{h}_{ba}[1], \hat{h}_{ba}[2], ..., \hat{h}_{ba}[N]\}^T,\end{aligned} \tag{1}$$

where the superscript T denotes matrix transpose and $\hat{h}[i]$ $(1 \leq i \leq N)$ is the channel characteristic estimation of $h[i]$ at time instant i. In our paper, the subscript of $\hat{h}_{uv}$ and other relative variables means the channel characteristics are measured by user u when user v sends the probing packets.

Quantization is used to quantize the measured channel characteristics $\hat{\mathbf{h}}_{ab}$ and $\hat{\mathbf{h}}_{ba}$ into bits.

Information reconciliation is a form of error correction carried out between Alice and Bob in order to ensure the keys generated separately on both sides are identical [16]. During the reconciliation, a few bits information will be revealed to Eve.

Privacy amplification is a method for reducing (or effectively eliminating) Eve's partial information about the legitimate key and for minimizing the correlation between the bits in a bit stream. Eve's partial information comes from the eavesdropping on the communication between Alice and Bob during the key generation process.

3.2 Adversary Model

There is an adversary, Eve, who tries to compromise the shared key by eavesdropping on the communication between Alice and Bob. Eve has single antenna. We assume that Eve can listen to all the communication between Alice and Bob. Eve can also measure both the channels between herself and legitimate users at the same time when Alice and Bob measure the channel between themselves for key generation. We also assume that Eve knows the key generation algorithm and the values of the parameters used in the algorithm. Eve can be close (e.g., several wavelengths away) to either Alice or Bob, but she cannot be in the exact same geographical position of Alice or Bob.

Let $\hat{\mathbf{h}}_{ea}$ and $\hat{\mathbf{h}}_{eb}$ be the channel estimation vector at Eve's side when Alice and Bob send the probing frames, respectively.

We assume that Eve neither jams the legitimate communication channel nor modifies any messages between Alice and Bob. Essentially, Eve will not disrupt the key establishment between Alice and Bob. We also assume that Eve cannot cause a man-in-the-middle attack, i.e., our methodology does not authenticate Alice or Bob.

In other words, we assume a passive adversary model.

3.3 Problem Definition

Intuitively, if Alice and Bob probe the channel at a higher rate under a given channel condition, they will get more samples (larger N) in the probing duration, and a higher KGR is expected to be achieved. However, it is difficult to decide which probing rate they should use to achieve a desirable KGR. The KGR depends on many factors such as the conditional mutual information between Alice and Bob given Eve's observation, the quantization, information reconciliation and privacy amplification methods. It is very hard to decide a deterministic function mapping a channel probing rate to a KGR, due to channel dynamics and uncertainty in each key generation step. Therefore, an adaptive channel probing scheme is desirable. For example, to achieve the same KGR, we should probe faster in a more static channel while the probing rate can be lower in a more variable channel.

An inappropriate approach is to set the probing rate at the possible highest rate without considering the channel conditions or key generation methods. This approach may achieve the maximum KGR, but is not resource efficient. Probing at the highest possible rate may consume a lot of bandwidth and energy. It can also introduce severe contention and interference to other communications in the network. Furthermore, when probing at a higher rate, the correlation between consecutive channel measurements may be increased, which also decreases the probing efficiency.

In this paper, we aim to addressing the adaptive channel probing problem to meet a desirable KGR and, meanwhile, make the probing process efficient. We introduce a mechanism to adaptively tune the probing rate in different scenarios and channel variations. Our mechanism is general enough to handle different key generation methods as long as they use the bi-directional channel probing schemes defined in this paper.

4 Theoretical Analysis: Probing Rate vs KGR

We analyze the relationship between the probing rate and the upper bound of KGR. We show that when probing rate is higher, we can achieve higher KGR. This property serves as a theoretical foundation to design the adaptive channel probing scheme based on a feedback controller.

Assume a noisy channel, according to [15], [20], [21], the upper bound of KGR is equal to the conditional mutual information between $\hat{\mathbf{h}}_{ab}$ and $\hat{\mathbf{h}}_{ba}$ given Eve's observation:

$$K_{ab} = I(\hat{\mathbf{h}}_{ab}; \hat{\mathbf{h}}_{ba} | \hat{\mathbf{h}}_{ea}, \hat{\mathbf{h}}_{eb}). \tag{2}$$

Now we derive the equation of K_{ab} under Gaussian channel assumption and Jake's correlation model in a rich scattering environment and isotropic distribution of incident power [22], [8].

Assuming the channel estimation error at Alice and Bob are $\mathbf{n}_{ab}$ and $\mathbf{n}_{ba}$, respectively, we have

$$\begin{aligned} \hat{\mathbf{h}}_{ab} &= \mathbf{h}_{ab} + \mathbf{n}_{ab}, \\ \hat{\mathbf{h}}_{ba} &= \mathbf{h}_{ba} + \mathbf{n}_{ba}, \end{aligned} \tag{3}$$

where $\mathbf{h}_{ab} = \{h_{ab}[1], h_{ab}[2], ..., h_{ab}[N]\}^T$ and $\mathbf{h}_{ba} = \{h_{ba}[1], h_{ba}[2], ..., h_{ba}[N]\}^T$ are the underlying channel characteristics. For simplicity, we assume the reciprocity holds, that is $h_{ab}[i] = h_{ba}[i] = h[i]$. This assumption usually holds if the time difference between the bi-directional probing is much smaller than the channel coherence time. We assume $h[i]$ $(1 \leq i \leq N)$ follows

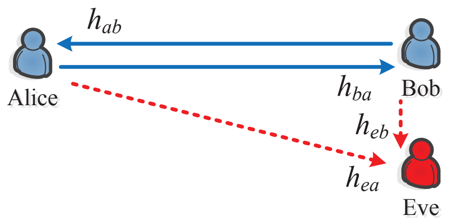


Fig. 1. An instance of channel characteristics among users

zero-mean Gaussian distribution with variance of σ_s^2. The measurement error/noise $\mathbf{n}_{ab}$ and $\mathbf{n}_{ba}$ follow independent and identically distributed (i.i.d.) zero-mean Gaussian distribution with variance of σ_n^2.

We assume the autocorrelation of the Alice-Bob channel follows the well-known Jake's model [22]. That is, the covariance between $h[i]$ and $h[j]$ is

$$R(h[i],h[j]) = \sigma_s^2 \cdot J_0(\frac{2\pi d_{ij}}{\lambda}), \tag{4}$$

where J_0 is the zero-order Bessel function, λ is the wavelength of the signal [1] , and d_{ij} is the distance between the measurements $h[i]$ and $h[j]$. Assuming Bob is stationary and Alice is moving towards or away from Bob in a straight line at a speed of v. $d_{ij} = v \cdot \tau|i-j|$. Assume $T_p = 1$, $\tau = \frac{1}{N-1}$.

At Eve side, we have

$$\begin{aligned} \hat{\mathbf{h}}_{ea} &= \mathbf{h}_{ea} + \mathbf{n}_{ea}, \\ \hat{\mathbf{h}}_{eb} &= \mathbf{h}_{eb} + \mathbf{n}_{eb}, \end{aligned} \tag{5}$$

where $\mathbf{h}_{ea}$ and $\mathbf{h}_{eb}$ are the underlying channel characteristics vector of $\hat{\mathbf{h}}_{ea}$ and $\hat{\mathbf{h}}_{eb}$, respectively. We assume $\mathbf{h}_{ea}$ and $\mathbf{h}_{eb}$ follow zero-mean Gaussian with variance of σ_e^2. $\mathbf{n}_{ea}$ and $\mathbf{n}_{eb}$ are measurement error/noise following i.i.d. zero-mean Gaussian with variance of σ_n^2.

Similar to the autocorrelation of Alice-Bob channel, the autocorrelation of Eve's measurement can also be modeled as Jakes' correlation model. That is

$$\begin{aligned} R(h_{ea}[i],h_{ea}[j]) &= \sigma_e^2 \cdot J_0(\tfrac{2\pi d_{ij}}{\lambda}), \\ R(h_{eb}[i],h_{eb}[j]) &= \sigma_e^2 \cdot J_0(\tfrac{2\pi d_{ij}}{\lambda}). \end{aligned} \tag{6}$$

Assuming Eve is close to Bob, shown in Fig. 1, the covariance between $h_{ea}[i]$ and $h[i]$ is modeled as

$$R(h_{ea}[i],h[i]) = \sigma_s\sigma_e \cdot J_0(\frac{2\pi d_0}{\lambda}). \tag{7}$$

Even though Eve is close to Bob (more than half a wavelength), according to our experimental results in Section 7, the cross correlation between $\hat{h}_{ea}$ and $\hat{h}_{ba}$ is small (say lower than 0.3). For simplicity, we assume that $h_{ea}[i]$ and $h[j]$ are uncorrelated when $i \neq j$.

Since Eve is close to Bob, she is far away from Alice given that Alice and Bob are well separated. Therefore, we can assume that $h_{eb}[i]$ is uncorrelated with $h[j]$ for $(1 \leq i,j \leq N)$. This assumption is validated by our experimental results in Section 7. Same assumption is also made in [8], [23].

According to the above assumption, we have

$$K_{ab} = I(\hat{\mathbf{h}}_{ab};\hat{\mathbf{h}}_{ba}|\hat{\mathbf{h}}_{ea}) = log2(\frac{|R_{AE}||R_{BE}|}{|R_E||R_{ABE}|}), \tag{8}$$

where $|\cdot|$ represents determinant of a matrix.

$$\begin{aligned} R_{AE} &= cov\{[\hat{\mathbf{h}}_{ab}^T, \hat{\mathbf{h}}_{ea}^T]^T, [\hat{\mathbf{h}}_{ab}^T, \hat{\mathbf{h}}_{ea}^T]\} \\ &= \begin{bmatrix} \hat{\mathbf{R}}_{aa} & \hat{\mathbf{R}}_{ae} \\ \hat{\mathbf{R}}_{ae} & \hat{\mathbf{R}}_{ee} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_{hh} + \sigma_n^2\mathbf{I} & \mathbf{R}_{he} \\ \mathbf{R}_{he} & \mathbf{R}_{ee} + \sigma_n^2\mathbf{I} \end{bmatrix} \end{aligned} \tag{9}$$

$$\begin{aligned} \mathbf{R}_{hh}(i,j) &= R(h[i],h[j]) \\ \mathbf{R}_{he}(i,j) &= R(h[i],h_{ea}[j]) \\ \mathbf{R}_{ee}(i,j) &= R(h_{ea}[i],h_{ea}[j]) \end{aligned} \tag{10}$$

Similarly,

$$\begin{aligned} R_{BE} &= cov\{[\hat{\mathbf{h}}_{ba}^T, \hat{\mathbf{h}}_{ea}^T]^T, [\hat{\mathbf{h}}_{ba}^T, \hat{\mathbf{h}}_{ea}^T]\} \\ &= \begin{bmatrix} \mathbf{R}_{hh} + \sigma_n^2\mathbf{I} & \mathbf{R}_{he} \\ \mathbf{R}_{he} & \mathbf{R}_{ee} + \sigma_n^2\mathbf{I} \end{bmatrix} \\ &= R_{AE} \end{aligned} \tag{11}$$

$$\begin{aligned} R_E &= cov\{\hat{\mathbf{h}}_{ea}, \hat{\mathbf{h}}_{ea}^T\} \\ &= \mathbf{R}_{ee} + \sigma_n^2\mathbf{I} \end{aligned} \tag{12}$$

$$\begin{aligned} R_{ABE} &= cov\{[\hat{\mathbf{h}}_{ab}^T, \hat{\mathbf{h}}_{ba}^T, \hat{\mathbf{h}}_{ea}^T]^T, [\hat{\mathbf{h}}_{ab}^T, \hat{\mathbf{h}}_{ba}^T, \hat{\mathbf{h}}_{ea}^T]\} \\ &= \begin{bmatrix} \mathbf{R}_{hh} + \sigma_n^2\mathbf{I} & \mathbf{R}_{hh} & \mathbf{R}_{he} \\ \mathbf{R}_{hh} & \mathbf{R}_{hh} + \sigma_n^2\mathbf{I} & \mathbf{R}_{he} \\ \mathbf{R}_{he} & \mathbf{R}_{he} & \mathbf{R}_{ee} + \sigma_n^2\mathbf{I} \end{bmatrix} \end{aligned} \tag{13}$$

Simulation. We set $\sigma_s = \sigma_e = 1$. The distance between Bob and Eve is 1 m. Fig. 2 and Fig. 3 show the numerical results of the upper bound of KGR at different probing rates with different noise variations σ_n and moving speeds v. It shows that the KGR is an increasing function of the probing rate. When moving faster, which induces a more dynamic channel, we can lower the probing rate and achieve the same KGR as those when moving slower. Therefore, in order to achieve a desired KGR, we need adaptively tune the probing rate under different channel conditions.

Discussion. We need to clarify that the results shown in Fig. 2 and Fig. 3 are under the assumption on the relative geographical location among Alice, Bob, and Eve shown in Fig. 1. When Eve is close to Alice, according to the reciprocity nature, we have similar results and conclusions. When Eve is far away from both Alice and Bob, Eq. 2 degenerates to $K_{ab} = I(\hat{\mathbf{h}}_{ab};\hat{\mathbf{h}}_{ba})$. Under this situation, KGR is also an increasing function of the probing rate, but larger than that in Fig. 2 and Fig. 3.

We should note that the KGR shown in Fig. 2 and Fig. 3 is the upper bound and could be loose. In practice, due to the non-perfect reciprocity and different efficiency of quantization, reconciliation, and privacy amplification

[1] We use standard 802.11n wireless card with average frequency 2.4 Ghz. Then, the wavelength λ is about 0.125 meter.

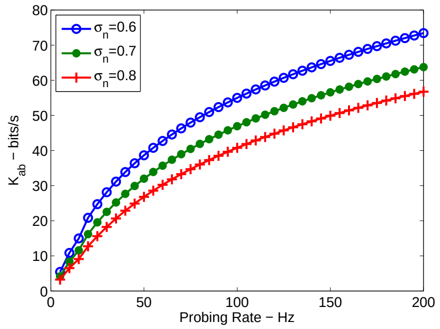


Fig. 2. Probing rate vs KGR under different σ_n

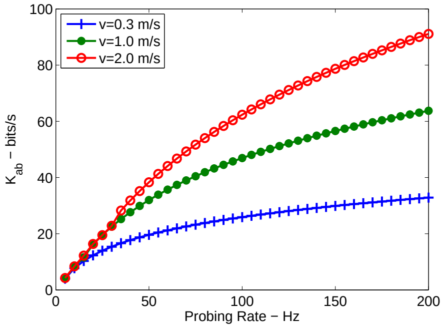


Fig. 3. Probing rate vs KGR under different velocities ($\sigma_n = 0.7$)

methods, the KGR would be lower. If the KGR as a function of probing rate could be determined, it would be trivial to determine a probing rate to achieve a desired KGR. However, it is very difficult to draw a close form deterministic relationship between the KGR and probing rate in practice due to the channel dynamics, unknown channel statistics, and uncertainty in each key generation step.

Therefore, in the next section, we introduce an adaptive channel probing scheme using the PID controller, which adaptively tunes the probing rate to achieve a desired KGR without knowing the exact mapping between the probing rate and KGR. We should also note that the proportional relationship between the probing rate and the upper bound of KGR serves as a guideline of designing our adaptive channel probing scheme. Although the relationship in practice might not be exactly the same as those shown in Fig. 2 and Fig. 3, the proportionality is likely to be held in practice as demonstrated by experiments in Section 7.

5 Adaptive Wireless Channel Probing for Shared Key Generation based on PID Controller

Based on theoretical foundation in previous sections, we propose an adaptive wireless channel probing scheme and its workflow is shown in Fig. 4. First, Alice initializes parameters, such as the probing rate f. Then, Alice and Bob both start to monitor the channel. In our system, WIRESHARK [24] is used to pull out the RSS information of probing packets. Two parties probe the channel by continually exchanging ICMP PING and REPLY packets for a fixed duration. The RSS values from the probing packets are recorded. Next, Alice and Bob estimate the entropy rate by LZ76, and the entropy rate is considered as an indicator of probing efficiency. After performing the quantization, reconciliation and privacy amplification, legitimate users obtain the KGR value. Last, the PID controller compares the KGR obtained in the current loop with user's desired KGR, denoted as κ, then decides a new probing rate for the next loop.

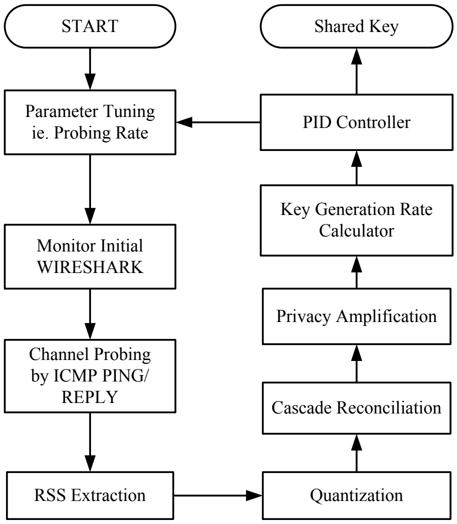


Fig. 4. Workflow of adaptive channel probing scheme

The process of sending and receiving packet pair, like ICMP PING and REPLY [25], is called a *probing process*. Recall that, the number of probing processes in one second is called *probing rate* f, measured by Hz. A series of probing processes at the same probing rate is called a *probing loop*. Duration of a probing loop is called *probing duration* T_p.

We introduce two main components of this probing scheme in the following subsections.

5.1 Lempel-Ziv Complexity

Entropy rate is considered as an indicator of probing efficiency in our work. Intuitively, high probing rate that aims to achieve a large KGR would result in low entropy rate, and vice versa; and this will be verified in Section 7. In order to estimate the entropy rate, we introduce an estimation method, namely Lempel-Ziv complexity (LZ76) [10].

Let X be a random variable or random vector, taking values in an arbitrary finite set A, its *alphabet*, and with distribution probability $p(x)$ = $\Pr\{X = x\}$ for $x \in A$. The *entropy* of X [9] is defined as,

$$H(X) = H(p) = -\sum_{x\in A} p(x)\log p(x). \tag{14}$$

The *entropy rate*, or per-symbol entropy, of X is

$$\lim_{n\to\infty} \frac{1}{n} H(X_1, X_2, \cdots, X_n), \tag{15}$$

whenever the limit exists, where $H(X_1, X_2, \cdots, X_n)$ is the entropy of the jointly distributed random variables $(X_1, X_2, \cdots, X_n)$.

We want to emphasize that the entropy rate is the property of a random process and difficult to evaluate [26]. In fact, knowledge of the probability distribution involved in its calculation requires an extensive sampling that usually cannot be performed [27]. In contrast, the complexity as originally formulated by Lempel and Ziv [10] is the property of individual sequences that can be used to estimate the entropy rate.

For a bitstring $X_1^N = [x_1, \cdots, x_N]$ of length N with $x_i \in \{0, 1\}$, a process that partitions X_N into non-overlapped substrings is called *parsing*. The first bit is always considered as the first substring, i.e., $B_1 = X_1^1 = x_1$. Assume we have

$$B_1 B_2, \dots, B_k = X_1^{N_k}, \tag{16}$$

in which $B_1 B_2, \dots, B_k$ is k adjacent and consecutive substrings, and $B_2 = X_2^{N_2}, \dots, B_k = X_{N_{k-1}+1}^{N_k}$, and $N_{k-1} + 1 \leq N_k < N (N_0 = 0, N_1 = 1)$. The definition

$$B_{k+1} = X_{N_k+1}^{N_{k+1}} \quad (N_k + 1 \leq N_{k+1} \leq N) \tag{17}$$

is the shortest substring that never appears in string $X_1^{N_{k+1}-1}$. Thus, we partition the string X_1^N into as fewer substrings as possible, and denote as

$$X_1^N = B_1 B_2, \dots, B_q, \tag{18}$$

where q is the amount of substrings and B_q is the only one substring that can probably appear in the whole string.

In order to make readers understand the parsing easier, here gives an instance. Assume a string $X_1^{19} =$ 0101101000110111 0010. First, we have $B_1 = x_1 = 0$. The bit $X_2^2 = 1$ does not appear in X_1^1, so $B_2 = X_2^2 = 1$. Next, both $X_3^3 = 0$ and $X_3^4 = 01$ appear in X_1^3, but $X_3^5 = 011$ does not appear in X_1^4, so we have $B_3 = X_3^5 = 011$, etc. Last, we obtain the partitioned string like

$$X_1^{19} = 0|1|011|0100|011011|1001|0, \tag{19}$$

where we can count the amount of substring $q = 7$.

Any further properties and formal expression can be found in reference [10].

In general, we define LZ76 value as

$$C_{LZ}(X_N) = \frac{q(\log_d q + 1)}{N}, \tag{20}$$

where d is the diversity of samples in X or range of x, and

$$0 \leq C_{LZ}(X_N) \leq \log_2 d. \tag{21}$$

For a random sequence X_N from an ergodic and stationary source [9], [28], entropy rate tends to

$$\lim_{N\to\infty} C_{LZ}(X_N). \tag{22}$$

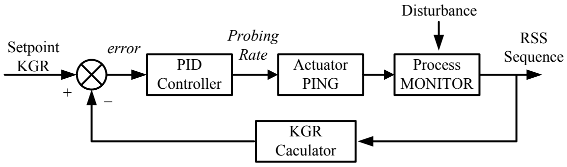


Fig. 5. Framework of the PID control system

In our paper, the RSS sequence is considered to be an ergodic and stationary source during a short probing duration, say $T_p = 1$ second, if the user velocity is not extremely high.

In order to keep the LZ76 calculator stable, i.e., LZ76 calculator results in small deviation, we should also limit the lower bound of f, denoted as f_{min}. Although there is no upper bound of f theoretically, due to the limitation of hardware, the upper bound of f is set as 300 Hz. Thus, we have

$$f_{min} < f \leq 300Hz. \tag{23}$$

5.2 PID Controller

In this subsection, we introduce feedback control to limit the error between the actual KGR and the desired KGR κ by adaptively tuning the probing rate f under different channel conditions.

Fig. 5 shows the framework of the control system. In the ith loop, we set probing rate f_i as input. At the end of this loop, we get the KGR K_i as output, feed it back, and compare it with κ. The PID controller then calculates a new probing rate f_{i+1} for the next loop. The controller model is

$$\begin{aligned} f_{i+1} = & f_i + G_P(K_i - \kappa) + G_I \textstyle\sum_{j=i-\alpha}^{i} (K_j - \kappa) \\ & + G_D(K_i - K_{i-1}), \end{aligned} \tag{24}$$

where $i = 1, 2, \cdots$, and α is the order of integral gain. G_P, G_I and G_D are the proportional gain, integral gain and derivative gain, respectively.

6 Experimental Setup

The adaptive probing scheme runs on a platform that is composed of three Gateway LT25 laptops (called Alice, Bob and Eve, respectively) with Atheros AR 5B95 802.11a/g/n wireless card. The operation systems are all Fedora Linux with the kernel version 2.6.34.8-68.fc13.i686. The RSS information of the probing packet is pulled out by WIRESHARK [24].

6.1 Experimental Scenarios

Outdoor and Indoor. The outdoor experiments were conducted at a place in Beijing Institute of Technology, Beijing, P.R.China. It is an open straight road with several cars parked along the side. The indoor experiments were conducted on the 5th floor of an academic building.

Static and Mobile (Line and Random). We consider it a static scenario if all users are stationary and no people or cars are moving on the road. We call it a mobile scenario if any user is moving. The motion type includes line and random movements.

The transmission power of all laptops are all set at 20 dBm. The velocity of the user is measured by a hand-held GPS. The weather of the day when the experiments were conducted was sunny. The outdoor temperature was 28 ^oC and relative humidity was 62%, while indoor temperature and relative humidity were 22 ^oC and 58%, respectively.

6.2 Performance Indices of the Controller

Denote K_i as the KGR at the ith loop, $i = 1, 2, \cdots, M$, and M is the amount of probing loops. The list of performance indices studied is as follows:

- KGR - mean: $\sum_{i=1}^{M} K_i/M$.
- KGR - std: standard deviation of KGR.
- KGR - error: $e(i) = |\sum_{i=1}^{M} K_i/M - \kappa|$.
- KGR oscillation frequency (KGR Osc. Freq.): M_{osc}/M, M_{osc} is the times that KGR values cross the setpoint κ.
- KGR overshooting (KGR Oversht.): the amount that KGR values exceed its desired value κ.
- KGR settling time: the time for KGR values to reach the setpoint at the first time, taking the loop number as settling time.
- ITAE (Integral of Time and Absolute Error): $(\sum_{i=1}^{M} e(i)M)/1000$.
- Efficiency: entropy rate estimated by LZ76.

The KGR oscillation frequency, overshoot and settling time jointly determine the ITAE. The smaller the ITAE, the better the controller works.

6.3 Parameters: LZ76 Calculator

According to Eq. 20, Lempel-Ziv complexity of a finite sequence is determined by q, d, N. In a loop, q is calculated by a Python script. N is the length of the RSS sequence, which relates to the probing rate f and probing duration T_p. d is a fixed number and is related to the diversity of the RSS values. As our wireless card provides RSS from -90 to -20 dBm, we consider the total range, $d = 70$, as the diversity.

From Eq. 22, if N is not large enough, LZ76 calculator cannot work well, i.e., results in large deviation. In order to make LZ76 calculator *stable*, i.e., making the deviation of entropy rate small, we should carefully consider T_p and f_{min}.

We conducted a series of outdoor-mobile-line experiments. In these experiments, the probing rate f was set as 5, 6, 8, 11, 22, 100 and 300 Hz, respectively. We calculated the standard deviation of the entropy rate by LZ76 calculator at different probing rate and probing duration, shown in Fig. 6. When the probing duration is 1 second, the standard deviations at different probing rate are all less than 0.2 bits/s per sample, which can be considered as small. Therefore, probing duration at 1 second (i.e., $T_p = 1s$) and a probing rate of no less than 5Hz (i.e., $f_{min} = 5$) can make LZ76 calculator stable.

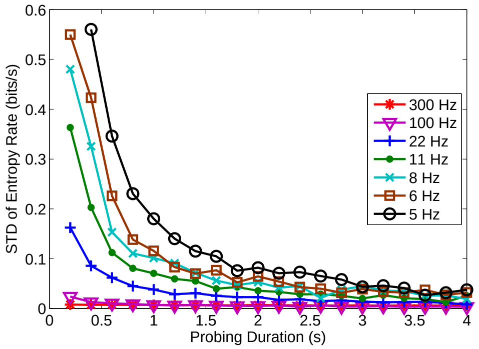


Fig. 6. Standard deviation of LZ76 calculator

6.4 Parameters: PID Controller

The Ziegler-Nichols tuning method is a heuristic method of tuning PID controller [29]. The setpoint of the controller (desired KGR) is 75 bits/s. The order of the integral gain $\alpha = 2$. By a series of tests, we get the proportional gain, integral gain and derivative gain as $G_P = 0.41, G_I = 0.23, G_D = 0.16$.

6.5 Cascade Reconciliation and Privacy Amplification

When using Cascade reconciliation [16], Alice permutes the bit stream randomly and divides it into small blocks, then sends permutation and parity information of each block to Bob. Bob permutes his bit stream in the same way, divides it into small blocks, calculates and checks if the parity information of those blocks are same or not. For each block whose parity does not match, Bob performs a binary search to find if a small number of bits in the block can be changed to make the block match the parity information. These steps are iterated multiple times until the probability of key agreement becomes higher than a desired threshold.

As information reconciliation is a probabilistic technique, it might fail occasionally. In those cases, the bit streams would be discarded and the key extraction process would be restarted by measuring RSS values again. However, low failure probability can be achieved by suitably choosing the amount of passes and the block size in each pass. In this paper, we set the amount of passes as 7 and the block size as 8, which can achieve acceptable low failure probability.

Privacy amplification solves the problems caused by correlation in bits and the revealed information to Eve. This is achieved by using universal hash function to map a long bit stream to a short one. Merkle-Damgard hash function is a collision-resistant one-way compression function used in our experiments [30]. It breaks input bit stream into small blocks of fixed size (5 bits). Bits in each block is hashed to 4 bits, so the average compression ratio is 0.8. According to the statistical analysis in Cascade reconciliation, the compression ratio is proper to amplify privacy by discarding those bit information revealed to Eve in our experiments. Note that our adaptive channel probing scheme will be applicable when a different compression ratio is used.

7 Experimental Results and Performance Evaluation

The relationships among the probing rate, entropy rate and KGR are analyzed by two groups of experiments. Then, we demonstrate the necessity of using a controller to satisfy the KGR requirement under dynamic environments. Last, we show experimental results under different situations and provide performance evaluation.

We should clarify that the KGR in the experiments is achieved by particular quantization, information reconciliation and privacy amplification method. Different methods may result in different KGRs. But it does not mean that the KGR is totally determined by those methods. Actually, the KGR is largely determined by the environmental dynamics, movements of users and spatial complexity.

The security of the shared key largely depends on the geographical location of adversary user and environmental dynamics, and partially depends on the privacy amplification method. Some privacy amplification methods may achieve very strict security but result in low KGR because they remove too many "safe" bits, which are neither leaked to Eve nor have any correlation with other bits. Therefore, an optimal privacy amplification method is the one that only remove those "unsafe" bits but keep "safe" bits. We should acknowledge that the privacy amplification method we used in this paper is not optimal, but its security and efficiency are acceptable.

7.1 Autocorrelation and Cross Correlation

In order to verify the mathematical models and the theoretic analysis in Section 4, we show the autocorrelation and cross correlation of the legitimate users and the adversary user in Fig. 7 and Fig. 8, respectively. Eve stayed away from Bob about 1 meter and much far away from Alice. Alice moved randomly and the other two were both stationary. The probing rate $f = 100$ Hz.

In Fig. 7, both the autocorrelations of Alice and Bob verify Eq. 4, and the correlation is larger than 0.8 when the lag interval is smaller than 5. There are two lines showing the autocorrelation of Eve (Eve-Alice channel and Eve-Bob channel) and they verify Eq. 6.

In Fig. 8, the cross correlation of Alice-Bob channel with zero lag verifies the reciprocity assumption, i.e., $h_{ab}[i] = h_{ba}[i] = h[i]$. The cross correlation of Eve-Alice channel verifies Eq. 7 and its value is very small, then it is reasonable to assume that $h_{ea}[i]$ and $h[j]$ are uncorrelated when $i \neq j$. Obviously, the Eve-Bob channel has very low correlation, then $h_{eb}[i]$ and $h[j]$ are considered as uncorrelated.

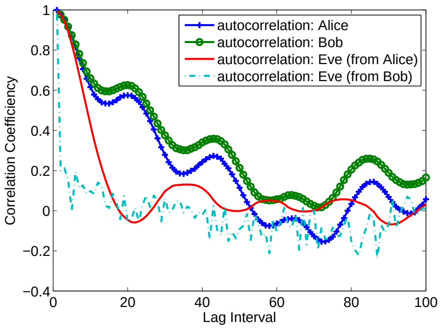


Fig. 7. Autocorrelations of Alice, Bob and Eve

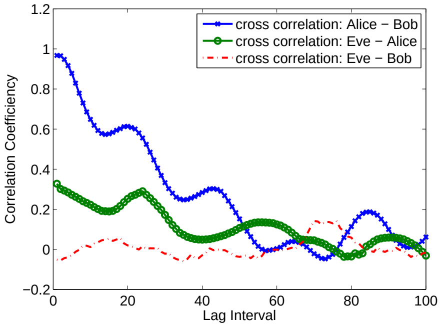


Fig. 8. Cross correlations among Alice, Bob and Eve

TABLE 1
Mutual information among users (mobile and static). Probing rate $f = 100$ Hz.

Mutual Information	Mobile (bits/s)	Static (bits/s)
$I_{ab} = I(\hat{\mathbf{h}}_{ab}; \hat{\mathbf{h}}_{ba})$	72.34	58.96
$I_{ae} = I(\hat{\mathbf{h}}_{ab}; \hat{\mathbf{h}}_{ea})$	8.65	3.17
$I_{be} = I(\hat{\mathbf{h}}_{ba}; \hat{\mathbf{h}}_{eb})$	0.07	0.02

7.2 Advantage of Legitimate Channel over Eavesdropping Channel

Here gives an experiment to demonstrate the advantage of legitimate channel over eavesdropping channel. The geographical positions of Alice, Bob and Eve in the static scenario are shown in Fig. 1. The distance between Bob and Eve was about 1 meter. In the mobile scenario, Alice moved at the speed of about 1 m/s and the others kept stationary. Table 1 shows Alice-Bob channel has significant larger mutual information than Alice-Eve and Bob-Eve channels, i.e., $I_{ab} >> I_{ae}$ and $I_{ab} >> I_{be}$. Please note the similar conclusion is also valid at other probing rates.

7.3 Probing Rate vs Entropy Rate

We focus on the relationship between the probing rate and the entropy rate in the static and mobile scenarios.

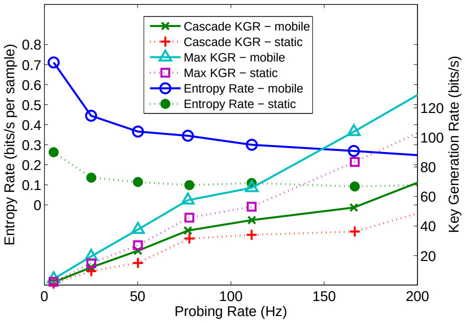


Fig. 9. Probing rate vs entropy rate and KGR (mobile and static)

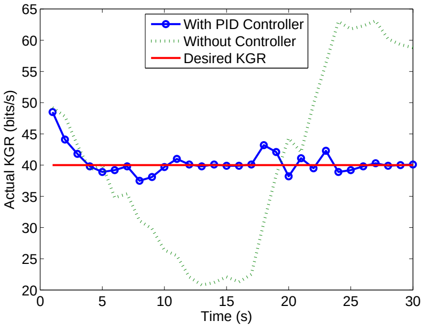


Fig. 10. Using PID Controller or not

Results are shown in Fig. 9 and it verifies that the entropy rate decreases when the probing rate increases. Higher probing rate induces higher correlation between consecutive channel measurements, so each measurement averagely contributes less information, thus leads to lower entropy rate, which indicates a lower probing efficiency.

Furthermore, the entropy rate at any probing rate in the mobile scenario is larger than the one in the static scenario. The entropy rate of the static scenario only can rise to 0.27 at the probing rate of 5 Hz, while in mobile scenario it reaches 0.71. Generally, the entropy rate in the mobile scenario is about two times of that in the static scenario.

7.4 Probing Rate vs Key Generation Rate

The relationship between the probing rate and KGR is derived from theoretic aspect in Section 4. Here give the empirical results in Fig. 9. As the method of calculating KGR in our experiments is based on the Cascade reconciliation, unless under specific clarification, the KGR refers to Cascade KGR. The maximum KGR is different from Cascade KGR. It directly compares binary streams of Alice and Bob by exchanging the secret information, and discards the bits in the same positions if they are not matched. Please note that the maximum KGR is not as same as the upper bound of KGR in theoretic analysis, mentioned in Section 4. It is only used as a baseline to demonstrate how many bits are agreed before reconciliation.

Fig. 9 show that both the Cascade KGR and Max KGR increase with probing rate. The line of the Max KGR is approximately proportional with probing rate. During 75 and 170 Hz, the line of the Cascade KGR increases relatively slow. Furthermore, the Cascade KGR and Max KGR in the mobile scenario are both larger than that in the static scenario.

7.5 Using PID Controller or Not

We conducted an indoor mobile experiment to demonstrate why we need a feedback controller. Assume user's KGR requirement is 40 bits/s. Eve was close to Bob and they were both stationary. Alice moved randomly, starting with slow moving and then being stationary, ending with fast moving. There were several people walking around randomly.

When we fix the probing rate at 100 Hz without using controller, the dash line in Fig. 10 shows that the KGR decreases or increases dramatically. Obviously, this cannot satisfy user's requirement and it is not efficient. The solid line, by contrast, shows the effectiveness when using the PID controller, that can stabilize the KGR around 40 bits/s by adaptively tuning the probing rate according to user's movement and environmental dynamics.

Note that the PID controller is just one type of feedback controllers. It is also worthwhile to consider other effective controllers.

7.6 Variable Motion

In order to validate the adaptive probing scheme and evaluate the performance of the PID controller, we conducted experiments under different situations.

Three experiments at different velocities were carried out, in which the moving user was walking, jogging and running, respectively. The desired KGR κ was 75 bits/s.

When the user moves faster, the KGR error becomes larger, from 0.15 to 0.59 bits/s, while the overshoot decreases, from 9.82 to 7.01 bits/s, shown in Table 2. Furthermore, the KGR reaches the setpoint more quickly. ITAE is smaller in the case of running, which indicates that the controller works better, i.e., it produces smaller overshoot and fewer oscillations. Fast moving provides more randomness and makes probing process more efficient.

Different motion types such as moving in line or randomly were also tested. From Table 2, we can see that random moving provides more randomness and results in more efficient probing process. However, it produces larger KGR error, reaching 0.42 bits/s. ITAE in the random scenario is smaller than that in the line scenario.

In summary, our adaptive probing scheme is adaptive to motion variations.

TABLE 2
Different Situation

Situation	**Velocities**			**Motion Types**		**Sites**	
Values	**0.4 m/s**	**1.0 m/s**	**1.5 m/s**	**Line**	**Random**	**Outdoor**	**Indoor**
KGR - error	0.15	0.38	0.59	0.15	0.42	0.15	0.33
KGR Osc. Freq.	0.29	0.34	0.30	0.29	0.43	0.29	0.26
KGR Oversht. - mean	9.82	7.01	6.17	9.82	7.29	9.82	6.10
Settling Time (loop)	4	3	3	4	3	4	4
ITAE	72.5	61.3	55.2	72.5	58.6	72.5	49.7
Probing Rate - mean	122.9	106.9	96.1	122.9	110.9	122.9	100.1
Probing Rate - std	7.1	5.8	5.3	7.1	8.4	7.1	9.2
Efficiency (LZ76)	0.318	0.320	0.337	0.318	0.345	0.318	0.362

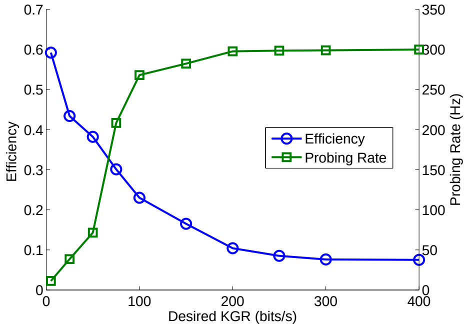


Fig. 11. Probing rate and efficiency under different desired KGR

7.7 Different Sites

Not only the movement produces more randomness, spatial complexity of environment also increases randomness because of more reflects and multi-paths. Two experiments were conducted in the outdoor and indoor scenarios. The mobile velocities were nearly about 0.3 m/s and motion types were both random.

Results are listed in Table 2. The indoor scenario has larger KGR error with 0.33 bits/s, as compared with 0.15 bits/s in outdoors. The indoor scenario results in smaller oscillation frequency and overshoot. As ITAE is much smaller, the controller works better indoors. It is more efficient to probe the channel indoors than outdoors.

7.8 Different Desired KGRs

The desired KGR κ was set at 75 bits/s in previous experiments. We conducted other indoor mobile experiments and set the desired KGR at 25, 50, 75, 100, 150, 200, 300 and 400 bits/s, respectively. The velocity of mobile user was about 0.3 m/s. Fig. 11 shows the results. When the desired KGR increases, the probing rate increases while the efficiency decreases. If we want to generate a key fast, we have to tolerate low efficiency, and vice versa. It also implies that if the users want to use the channel efficiently, they should not set their desired KGR too high.

TABLE 3
Different desired KGR

Desired KGR	Actual KGR	Probing Rate (Hz)
25	25.2	46.2
50	50.9	102.5
75	76.1	169.2
100	98.03	207.7
150	152.4	252.3
200	204.8	297.7
300	218.5	298.5
400	219.1	298.9

In Fig. 11, when the desired KGR is larger than 200 bits/s, the probing rate reaches the upper bound at 300 Hz and the efficiency becomes nearly unchanged. From Table 3, we can see that the actual KGR is around 200 bits/s and does not exceed 220 bits/s when the desired KGR is over 200 bits/s. That is, when the user requires a KGR over the limit of the system, our adaptive channel probing scheme just makes a best effort to satisfy the KGR. It is reasonable to believe that a higher KGR can be achieved when the probing rate is higher than 300 Hz if other specific platforms are used. In this paper, in order to probe the channel efficiently and obtain an acceptable KGR output, we suggest not set the desired KGR over 100 bits/s.

8 Conclusion and Discussion

In order to satisfy users' requirements for key generation rate under dynamic channel conditions and to use the wireless channel in an efficient way, we introduce an adaptive channel probing scheme based on the PID controller.

We present the proportional relationship between the probing rate and upper bound of KGR under eavesdropping attack. We use Lempel-Ziv complexity to estimate the entropy rate of channel statistics (Received Signal Strength, RSS), which is considered as an indicator of probing efficiency.

A series of experiments were conducted to evaluate the performance at different velocities, motion types, sites, and different desired KGRs. Experimental results show that our channel probing scheme can adaptively tune the probing rate according to users' movements and/or environmental dynamics. It not only satisfies user's KGR requirement, but also makes the probing process efficient.

However, the overshoot of KGR seems a little large, which may be due to the following reasons. First, the probing rate in the current loop is determined by the KGR in the last loop and the channel condition is not predictable. Therefore, it is impossible to stabilize the KGR exactly at the setpoint. Second, the parameters of the PID controller may not be optimal. Third, the control object is nonlinear but the controller is linear.

In order to solve the control problem mentioned above and improve the performance of the scheme, we can use the adaptive controller to cope with the fact that the parameters of the system being controlled are slowly time-varying or uncertain, and this approach is considered as our future work.

Acknowledgment

The authors would like to thank Prof. Lihua Dou, School of Automation, Beijing Institute of Technology, for discussions.

References

[1] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.

[2] Y. Wei, K. Zeng, and P. Mohapatra, "Adaptive wireless channel probing for shared key generation," in *Proceedings IEEE INFOCOM*, April 2011.

[3] 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 '09: Proceedings of the 15th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2009, pp. 321–332.

[4] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: extracting a secret key from an unauthenticated wireless channel," in *MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2008, pp. 128–139.

[5] K. Zeng, D. Wu, C. A., and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *Proceedings IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[6] N. Patwari, J. Croft, S. Jana, and S. Kasera, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *IEEE Transactions on Mobile Computing*, vol. 9, no. 1, pp. 17–30, Jan. 2010.

[7] T. S. Rappaport, *Wireless communications: principles and practice*. New Jersey,NJ,USA: Prentice Hall, 2001.

[8] C. Chen and M. Jensen, "Secret key establishment using temporally and spatially correlated wireless channel coefficients," *IEEE Transactions on Mobile Computing*, vol. 10, no. 2, pp. 205–215, Feb. 2011.

[9] J. Thomas, *Elements of information theory*. John Wiley and Sons, 1991.

[10] A. Lempel and J. Ziv, "On the complexity of finite sequences," *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 75–81, Jan. 1976.

[11] J.-L. Blanc, N. Schmidt, L. Bonnier, L. Pezard, and A. Lesne, "Quantifying neural correlations using lempel-ziv complexity," in *NEUROCOMP2008*, Marseille, France, 2008.

[12] T. Aono, K. Higuchi, M. Taromaru, T. Ohira, and H. Sasaoka, "Wireless secret key generation exploiting the reactance-domain scalar response of multipath fading channels : RSSI interleaving scheme," *The European Conference on Wireless Technology*, pp. 173–176, Oct. 2005.

[13] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener, "Robust key generation from signal envelopes in wireless networks," in *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*. ACM, 2007, pp. 401–410.

[14] R. Wilson, D. Tse, and R. Scholtz, "Channel identification: Secret sharing using reciprocity in ultrawideband channels," *IEEE International Conference on Ultra-Wideband*, pp. 270–275, Sept. 2007.

[15] U. M. Maurer, "Secret key agreement by public discussion from common information," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, May 1993.

[16] G. Brassard and L. Salvail, "Secret-key reconciliation by public discussion," in *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*. Springer-Verlag New York Inc., 1994, pp. 410–423.

[17] C. Cachin and U. M. Maurer, "Linking information reconciliation and privacy amplification," *Journal of Cryptology*, vol. 10, pp. 97–110, 1997.

[18] H. Koorapaty, A. Hassan, and S. Chennakeshu, "Secure information transmission for mobile radio," *IEEE Communications Letters*, vol. 4, no. 2, pp. 52–55, Feb. 2000.

[19] C. H. Bennett, G. Brassard, C. Crkpeau, and U. M. Maurer, "Generalized privacy amplification," *IEEE Transactions on Information Theory*, vol. 41, no. 6, pp. 1915–1923, Nov. 1995.

[20] A. Khisti, S. N. Diggavi, and G. W. Wornell, "Secret-key generation using correlated sources and channels," *IEEE Transactions on Infomation Forensics and Security, Special Issue on Physical Layer Security*, Sept. 2011.

[21] A. Khisti and S. N. Diggavi, "A remark on secret-key generation over correlated fading channels," in *Proc. GC'11 Workshop on Physical-Layer Security*, Dec. 2011.

[22] W. C. Jakes, *Microwave Mobile Communications*. Wiley-IEEE Press, 1994.

[23] J. Wallace and R. Sharma, "Automatic secret keys from reciprocal mimo wireless channels: Measurement and analysis," *IEEE Transactions on Wireless Communications*, vol. 5, no. 3, pp. 381–392, Sept. 2010.

[24] G. H. Gerald Combs, Gilbert Ramirez and R. Sharpe, 2006. [Online]. Available: http://en.wikipedia.org/wiki/Wireshark.

[25] J. Postel, "Request for comments," RFC:792, Sept. 1981. [Online]. Available: http://tools.ietf.org/html/rfc792.

[26] S. P. Strong, R. Koberle, de Ruyter van Steveninck, and W. Bialek, "Entropy and information in neural spike trains," *Phys. Rev. Lett.*, vol. 80, 1998.

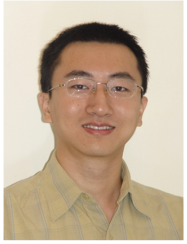
[27] J. M. Amigo, J. Szczepanski, ElekWajnryb, and M. V. Sanchez-Vives, "Estimating the entropy rate of spike trains via lempel-ziv complexity," *Neural Computation*, vol. 16, pp. 717–736, 2004.

[28] R. Badii and A. Politi, *Complexity: Hierarchical structures and scaling in physics*. Cambridge,UK: Cambridge University Press, 1997.

[29] J. B. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *ASME Transactions*, vol. 64, pp. 759–768, 1942.

[30] R. Merkle, "Secrecy, authentication, and public key systems," Ph.D., Stanford University, 1979.

Yunchuan Wei is currently a Ph.D. candidate in the School of Automation at the Beijing Institute of Technology, Beijing, China, and his supervisor is Prof. Lihua Dou. He was a visiting Ph.D. candidate under the supervision of Prasant Mohapatra in the Department of Computer Science at the University of California, Davis, supported by the China Scholar Council (No.2009603052) from 2009 to 2010. He received B.E. degree from the Beijing Institute of Technology in 2006.



Kai Zeng received his Ph.D. degree in Electrical and Computer Engineering at Worcester Polytechnic Institute (WPI) in 2008. He obtained MS in Communication and Information Systems and BS in Communication Engineering both from Huazhong University of Science and Technology, China in 2004 and 2001, respectively. He was a postdoctoral scholar in the Department of Computer Science at University of California, Davis (UCD) from 2008 to 2011. He joined the Department of Computer and Information Science at University of Michigan - Dearborn as an assistant professor in 2011. He is a recipient of the U.S. National Science Foundation Faculty Early Career Development (CAREER) award in 2012. He won Excellence in Postdoctoral Research Award at UCD in 2011 and Sigma Xi Outstanding Ph.D. Dissertation Award at WPI in 2008. His current research interests are in wireless network security, physical layer security, cognitive radio networks, energy efficiency, and cyber-physical systems.

Prasant Mohapatra is currently a Professor in the Department of Computer Science at the University of California, Davis. He has also held various positions at the Iowa State University, Michigan State University, Intel Corporation, Panasonic Technologies, Institute of Infocomm Research, Singapore, and the National ICT, Australia. Dr. Mohapatra received his Ph.D. in Computer Engineering from the Pennsylvania State University in 1993. He was/is on the editorial boards of the IEEE Transactions on computers, ACM/Springer WINET, and Ad hoc Networks Journal. He has served on numerous technical program committees for international conferences, and served on several panels. He was the Program Vice-Chair of INFOCOM 2004, and the Program Co-Chair of the First IEEE International Conference on Sensor and Ad Hoc Communications and Networks, (SECON-2004). Dr. Mohapatra's research interests are in the areas of wireless networks, sensor networks, Internet protocols and QoS.