Delay Analysis of Physical Layer Key Generation in Multi-user Dynamic Wireless Networks

Rong Jin*, Xianru Du*, Kai Zeng*, Laiyuan Xiao†, Jing Xu‡
*Department of Computer and Information Science, University of Michigan - Dearborn, MI 48128
†School of Software Engineering, Huazhong University of Science and Technology
‡Department of Electronic and Information Engineering, Huazhong University of Science and Technology
Email: *{ jinrong, duxianru, kzeng } @umich.edu †xiaol.y@263.net ‡xujing@mail.hust.edu.cn

Abstract—Secret key generation by extracting the shared randomness in wireless fading channel is a promising way to ensure wireless communication security. Previous works only consider key generation in static networks, but real-world key establishments are usually dynamic. In this work, for the first time we investigate the pairwise key generation in dynamic wireless networks with a center node (e.g., access point (AP)) and random arrival users. We establish the key generation model for this kind of networks. We propose a method based on discrete Markov chain to calculate the average time a user will spend on waiting and completing the key generation (average key generation delay, AKGD). Our method can tackle both serial and parallel key generation scheduling under various conditions. We conduct extensive simulations to show the effectiveness of our model and method. The analytical and simulation results match to each other.

I. INTRODUCTION

Establishing pairwise secret key between two communication parties is crucial to secure wireless communication. Physical layer key generation mechanisms that exploit reciprocal and location-specific properties of wireless fading channels, have been proposed [1, 2, 3]. Based on the reciprocity, the bidirectional channel states should be identical between two transceivers at a given instant of time. In a multipath or mobile environment, the channel states randomly fluctuate due to fading. Therefore, two legitimate parties can take advantage of this natural correlated random process to generate a shared key. Furthermore, the channel state observed at an eavesdropper is uncorrelated with the legitimate channel if the eavesdropper is more than half a wavelength away from legitimate parties. ⊙

The existing research on physical layer key generation mainly focuses on key generation rate (KGR) in static wireless networks. Most of the works discussed the KGR between two parties, including theoretical KGR analysis [1, 2, 4] and KGR optimization under practical conditions [5, 6, 7]. Some works discussed multiple-user case [8, 9], and theoretically derived KGR and secrecy capacities for multiple terminals in static wireless networks. To the best of our knowledge, physical layer key generation in dynamic wireless networks has not been well studied. However, it is a common case in practice.

In this paper, for the first time, we consider the key generation problem in dynamic wireless networks. Moreover, we focus on a more realistic metric: the users’ key generation delay. Since key generation is a pre-requisite for secure wireless communications, the delays that users suffer are their most concerned issue.

Delay analysis of physical layer key generation in dynamic networks is different from traditional delay analysis for wireless communication [10, 11] in the following two aspects. (1) Due to the specific mechanism of key generation (detailed in section II.C), total service rate (or KGR) depends on the number of users and channel probing scheduling; while in wireless communication, total service rates are the same, regardless of scheduling methods. For example, using time division multiplexing (TDM) scheme reduces individual user’s data transfer rate but will not affect total data transfer rate. (2) Key generation has an additional process: reconciliation, to ensure the keys generated on both sides are identical.

In this work, we consider the problem of pairwise key generation in dynamic wireless networks with a center node. We use the the scenario of access point (AP) and legitimate users shown in Fig. 1 to exemplify our model and method. The arrival of users is unknown in prior. We aim at calculating the average time a user will spend on waiting and completing the key generation (average key generation delay, AKGD). We propose the key generation model. Our model can tackle both serial and parallel key generation scheduling (section II). We notice that in key generation model, each key establishment period has predictable time. We make use of such characteristic to develop a method based on discrete Markov chain for calculating users’ AKGD (section III). We conduct simulations to show the effectiveness of our methods (section IV). Finally, we conclude this paper (section V).

Fig. 1. Key generations in a dynamic wireless network

This work is partially supported by the US National Science Foundation (NSF) CAREER award under Grant No. CNS-1149500, National Natural Science Foundation of China (61301127), and the Fundamental Research Funds for the Central Universities, HUST (2013QN142).
Our main contributions are:
- We formulate the physical layer key generation problem in dynamic wireless networks and analyse the delay.
- We propose the key generation model and develop a method to calculate AKGD.
- We conduct extensive simulation to verify and evaluate our model.

II. KEY GENERATION MODEL IN MULTI-USER DYNAMIC WIRELESS NETWORKS

A. System Setting

We consider a one-hop network with an AP generating pairwise keys for legitimate users (shown in Fig. 1). Users’ arrival is a stochastic process determined by a Poisson process with the rate of \( \lambda \). The key generation order follows the rule of first-come, first-serve. Users are numbered according to their arriving order \( u_1, u_2, \ldots \). Their key length requirements are \( L_1, L_2, \ldots \) taking values in the finite discrete set \( L \) containing all possible key length requirements. For simplicity, we assume users have similar channel variations, and mainly focus on the dynamic of users’ arrival.

We denote \( N \) as the number of users in the system, including waiting users and the users currently conducting key generations (we call them running users) [12]. Each user arrival increases \( N \) by 1; each key generation completion reduces \( N \) by 1.

We also assume a passive attacker trying to compromise the shared key by eavesdropping on the communication among AP and users.

B. Key Generation Process

The key generation process usually contains four steps: channel probing, quantization, information reconciliation, and privacy amplification [13, 1].

Channel probing is used to collect channel characteristics between AP and legitimate users shown in Fig. 2. The channel characteristics can be RSS (received signal strength), channel impulse response, or phase, etc. In this step, \( i \)th user and AP exchange bidirectional request/reply probing frames to probe the channel characteristics for a duration, say \( \delta \) seconds. Denote \( t_\lambda \) as the interval between any two consecutive request probing frames. The channel probing rate \( v \) is thus \( 1/t_\lambda \). At the end of channel probing process, we assume the \( i \)th user and AP make \( N_i \) pairs of channel measurements \( h_{u_i,a} = [h_{u_i,a}(1), h_{u_i,a}(2), \ldots, h_{u_i,a}(N_i)] \) and \( h_{a,u_i} \) within \( \tau \) seconds.

Quantization is used to quantize the measured channel characteristics into bits.

Information reconciliation is a discussion process used to correct measurement error between AP and users so as to make identical keys.

Privacy amplification is used to discard partial revealed information due to measurement correlation and eavesdropper.

C. Key Generation Scheduling

In the key generation, quantization and privacy amplification can be done locally at respective sides. While channel probing and information reconciliation need communication between the user and AP. For multiple users, scheduling is required to avoid interference. In our work, we consider the following two scheduling methods.

1) Serial Probing. AP probes the channel with users one by one according to users’ arriving order. However, this scheduling is not efficient. For serial probing, consecutive channel probing measurements have small time separation and thus highly correlated. The correlated part provides little extra information and should be largely discarded in the final key in privacy amplification [4, 5, 6, 3].

2) A more efficient optional key generation scheduling is Parallel Probing: multiple users share the channel using time-division method. An example with three users is shown in Fig. 2. AP probes the channel with multiple users in a round-robin way. The advantage of parallel probing is exemplified in Fig. 2: by inserting the channel probing between AP and the 2nd and 3rd users into the channel probing between AP and the 1st user. The time separation between consecutive channel probing measurements increases, thus correlation of the measurements reduces. From the point of view of the system, fewer correlated bits are discarded and higher total KGR can be achieved.

Our key generation scheduling model developed for parallel probing is shown in Fig. 3. We consider the following three conditions:

a) Light load: we define a default probing rate \( v_d \) as the maximum probing rate. AP parallel probes the channel with users with \( v_d \) when few users are in the system. Setting \( v_d \) is to avoid ultra low efficiency.

b) Medium load: AP parallel probes the channel with users until the number of parallel probing reaches its maximum \( N_p \).

c) Heavy load: when a large number of users come to the system \( N > N_p \), AP parallel probes the channel with first \( N_p \) users; remaining users wait at the service line.

After a period of probing process is finished, AP applies reconciliation for all of the channel measurements with the rate of \( v_{recon} \) s/bit. We denote the period from the beginning of channel probing to the end of reconciliation as a key establishment period.

Note that serial probing can be simply treated as a special case of parallel probing when \( N_p = 1 \). Our model tackles both serial probing and parallel probing cases.

D. User’s Key Generation Delay

For an individual user, his key generation delay \( D_{all} \) is the sum of waiting delay \( D_{wait} \) and service delay \( D_{service} \): \( D_{all} = D_{wait} + D_{service} \). According to the key generation process, \( D_{service} \) is divided into four parts: channel probing delay \( D_c \), quantization delay \( D_q \), information reconciliation
Reconciliation

III. Delay Analysis of Key Generation in Dynamic Wireless Networks

Now we introduce our methodology in detail. We notice that each key establishment period has predictable time determined by the number of running users. We make use of such characteristic and focus on the discrete time points at the end of each key establishment period; so that the key generations delay $D_a$, and privacy amplification delay $D_p$. $D_{\text{service}} = D_c + D_q + D_i + D_r$. $D_q$ and $D_p$ are very short compared with $D_c$ and $D_i$. Moreover, $D_q$ and $D_p$ are produced at respective local sides outside of our queueing system. For convenience, we define $D_{\text{service}} = D_c + D_i$

For a constant probing rate $v$, the channel probing delay $D_c$ can be written as

$$D_c = l/r(v) \quad (1)$$

where $l$ is the length of key generated in bit, and $r(v)$ is the KGR.

KGR can be calculated according to [6]. The maximum bits generated between AP and the i\textsuperscript{th} user can be represented by the conditional mutual information given an eavesdropper’s observation, say Eve:

$$K_{u,a} = I(\hat{h}_{u,a}, \hat{h}_{a}|\hat{h}_{e,a}) \quad (2)$$

where we assume the eavesdropper is near AP, $\hat{h}_{e,a}$ is the channel measurement of user to AP channel by Eve.

The other parameters affecting mutual information $K_{u,a}$ besides probing rate are: Doppler spread $f_m$ ($Hz$) to represent channel variation in time, signal variance $\sigma_s^2$, noise variance $\sigma_n^2$, eavesdropper channel variance $\sigma_e^2$ (channels are assumed zero mean Gaussian distribution), the distance between AP and the eavesdropper. The computational method for $K_{u,a}$ can be found in [6] (In that paper, it uses relative velocity to wavelength $v/\lambda$ to express $f_m$). Due to space limit, we do not repeat it here.

The upper-bound of KGR is $r_i^{\text{max}}(v) = K_{u,a}/\tau$. In practice, due to the non-perfect reciprocity and different efficiency of quantization, reconciliation, and privacy amplification, KGR is lower than the upper-bound: we use $\eta(v) < 1$ to denote the ratio between practical KGR and the upper-bound of KGR.

$$r_i = \eta r_i^{\text{max}} \quad (3)$$

For a given key generation protocol, $\eta(v)$ is determined.

Information reconciliation delay can be written as

$$D_i = l_{\text{raw}}v_{\text{recon}} \quad (4)$$

where $v_{\text{recon}}$ is the reconciliation rate; $l_{\text{raw}}$ is the amount of raw channel probing measurements within $t$ seconds. Under a constant probing rate $v$, $l_{\text{raw}} = vt$.

A. Establish One-step State Transition Matrix

To get $k$, we make the following phase decomposition:

Suppose we need $K$ key establishment periods to satisfy $i$\textsuperscript{th} user’s key length requirement $K = L_i/L_0$. We treat the shared key between $i$\textsuperscript{th} user and AP as $K$ phases. From each $t_n$ to $t_{n+1}$, the $i$\textsuperscript{th} user completes one phase (shown in Fig. 5). When all $K$ phases are completed, the key generation of the $i$\textsuperscript{th} user is completed. We use $X_n(1,c_1,c_2,...,C_n)$ to express the state at $t_n$: where $X_n$ is the number of users in the system and $c_1,c_2,...,c_X$ are their remaining phases arranged in user arrival order.

We denote $P_{X_n}(c_1,c_2,...,C_n)P_{X_n+1}(c_1,c_2,...,C_{n+1})$ as the transition probability from state $X_n(c_1,c_2,...,C_n)$ to state $X_{n+1}(c_1,c_2,...,C_{n+1})$ during $T_n$.

We write out one-step state transition matrix (Fig. 6).
(1) no user coming during $T_n$:
$$P(X_n = 0 | T_n)$$

(2) more than 1 user coming during $T_n$:
$$P(X_n = j | T_n) = \frac{1}{j!} \lambda^j e^{-\lambda}$$

where $j = X_n - 1 + k$ is the number of coming users. In the given example, $p_{3(122),2(11)} = P(Y_n = 1 | T_n(3)); p_{3(122),2(11)} = P(Y_n = 1 | T_n(3)(1/2)$; where underlined part denotes the initial phase of new coming users during $T_n$.

C. Stationary Distribution

The steady state $\pi$ is a row vector, whose entries $\pi_{X_n(c_1,c_2,..,c_n)}$ are non-negative and sum to 1, that satisfies the equation
$$\pi = \pi P$$

Thus $\pi$ is a normalized (meaning that the sum of its entries is 1) left eigenvector of $P$ associated with the eigenvalue 1.

Then we derive mean number of users between states $\phi X_n(c_1,c_2,..,c_n), X_{n+1}(c_1,c_2,..,c_{n+1})$.

D. Average Key Generation Delay

The steady state probabilities of continuous-time Markov process can be computed by multiplying the steady state probabilities of the discrete-time version of Markov chain by the mean times spent in the various states [14]. This leads to $\overline{P}_{X_n(c_1,c_2,..,c_n)} = L_0 T_n(c_1,c_2,..,c_n) T_n(N)$

where $T_n(0) = \Delta T$ $C$ is normalization constant making $\sum \overline{P}_{X_n(c_1,c_2,..,c_n)} = 1$;

The average number of users in the system is
$$L_s = \frac{1}{\lambda} \sum \overline{P}_{X_n(c_1,c_2,..,c_n)}$$

From Little’s law, we can get the AKGD
$$D_{all} = \frac{L_s}{\lambda}$$

IV. Simulations

In the following, we conduct numerical simulations to investigate our model and method. We set Doppler spread $f_m = 5$Hz, the variance of signal, noise and eavesdropper are $\sigma^2 = 0.2$. We assume an adversary is close to AP with the distance of 1m. We also assume bi-directional probing with $\theta = 2 \times 10^{-3}$; This makes the highest probing rate $v = 1/\delta = 500Hz$. The users have different key length requirements which takes the values of 64 and 128 bits with the same probability $P(L(u_k) = 64) = P(L(u_k) = 128) = 1/2$. We set $L_0 = 64$, reconciliation rate $v_{recon} = 64^{-1}s$bit, $\eta = 0.4$ and default probing rate $v_d = 250Hz$. 

Fig. 6. State transition matrix
with the analysis in section II.C. It conforms to be seen that parallel probing has smaller AKGD than serial probing in the condition of same users’ arrival rate. We set the maximum number of parallel probing N_p = 4. We record each user’s arrival time as well as key generation completion time, so as to calculate each user’s key generation delay. Fig. 7(left) shows the user’s key generation delay distribution with the interval of 1s. AKGD is \( \mathcal{D}_s = 13.42 s \). We also use our proposed method in section III to theoretically get AKGD \( \mathcal{D}_s^{\text{theo}} = 13.46 s \). It can be seen that simulation and theoretical AKGD are close.

We then observe the whole key generation procedure by counting the number of users in the system at every second (from the beginning of the first user arrival to the end of last user key generation completion). Fig. 7(right) shows the number of the observations falls into the case of 14 users in the system (k from 0 to 12). We also use equation (19) to theoretically derive the number of users distribution. Simulation and theoretical results are in good agreement.

B. Comparison between Serial and Parallel Probing

We then compare serial probing with parallel probing. We vary users’ arrival rate \( \lambda \) from \( 8^{-1} \) to \( 14^{-1} \) (that is from on average 8s a user to 14s a user). For parallel probing, we set the maximum number of parallel probing \( N_p = 4 \). Fig. 8 shows AKGD curves of serial and parallel probing. It can be seen that parallel probing has smaller AKGD than serial probing in the condition of same users’ arrival rate. It conforms with the analysis in section II.C.

Next, we investigate various parameters’ impact on AKGD and compare simulation and theoretical results.

C. Impact of Parallel Probing on Key Generation

We set \( \lambda = 6^{-1} \) to see the impact of the number of parallel probing on key generation. We vary the maximum number of users’ arrival rate \( \lambda \) from \( 5^{-1} \) to \( 20^{-1} \) (that is from on average 5s a user to 20s a user). Fig. 9(right) shows AKGD with respect to \( \lambda \). AKGD decreases from 15.86s to 8.84s as users’ arrival rate decreases from \( \lambda = 5^{-1} \) to \( \lambda = 20^{-1} \).

We compute the differences between simulation and theoretical results. We then fix \( N_p = 4 \) to see the impact of users’ arrival rate on key generation. We vary \( \lambda \) from \( 5^{-1} \) to \( 20^{-1} \) (that is from on average 5s a user to 20s a user). Fig. 9(right) shows AKGD and theoretical AKGD.

When \( N_p \) increases from 2 to 5, AKGD decreases from 16.14s to 13.44s (when \( N_p = 1 \) diverges because users’ arrival rate is larger than total KGR; converge region will be deduced later). When \( N_p \) increases from 5 to 9, AKGD stops decreasing. This is because efficiency is already high, increasing \( N_p \) will gain little KGR increment.

To get the difference between simulation and theoretical results, we compute mean absolute deviation \( \Delta \mathcal{D}_s = \sqrt{E(\mathcal{D}_s^{\text{sim}} - \mathcal{D}_s^{\text{theo}})^2} = 0.064 s \) and relative deviation \( e(\mathcal{D}_s) = \sqrt{E(\mathcal{D}_s^{\text{sim}} - \mathcal{D}_s^{\text{theo}})^2/\mathcal{D}_s^{\text{theo}}^2} = 0.44\% \).

D. Impact of Users’ Arrival Rate on Key Generation

We then fix \( N_p = 4 \) to see the impact of users’ arrival rate on key generation. We vary \( \lambda \) from \( 5^{-1} \) to \( 20^{-1} \) (that is from on average 5s a user to 20s a user). Fig. 9(right) shows AKGD and theoretical AKGD.

Note that when users’ arrival rate approaches to 0, AKGD approaches to its lower bound: whenever a new user comes, he always face an empty system and get an instant start. Its corresponding AKGD can be written as

\[
\mathcal{D}_s^{\text{min}} = \sum_{l \in L} P(L(u_k) = l)|l/r(v_d) + l/v_{\text{recon}}| \quad (21)
\]

In the example, the AKGD lower bound is \( \mathcal{D}_s^{\text{min}} = 7.54 s \). Fig. 10 shows the average number of users under different users’ arrival rates. When \( \lambda = 5^{-1} \), the probabilities of 0, 1, 2, 3, 4 and 5 users in the system are 5%, 19%, 17%, 18%, 16%, 17%. When users’ arrival rate decreases, the number of users in the system decreases. When \( \lambda = 20^{-1} \), the probabilities of 0, 1, 2 and 3 users in the system become 60%, 33%, 5%, 1%; the probability that more than 4 users is less than 1%.

Suppose the system can hold infinite users \( M = \infty \). Note that the system is diverge if users’ arrival rate is too fast and the number of users in the system cumulate to infinite. According to section II.C, the system has the highest total KGR when AP parallel probing the channel with \( N_p \) users. Users’ average arrival rate should be lower than the highest total KGR. Therefore, we can calculate converge region:

\[
\lambda^{-1} > \sum_{l \in L} P(L(u_k) = l)|l/r_{\text{all}} + l/v_{\text{recon}}| \quad (22)
\]
where the maximum overall KGR is $\rho_{\text{all}}^{\text{max}} = N_p K(v_{N_p})$, $v_{N_p} = 1/N_p \delta$. In the given example, convergence region is $\lambda < 3.86$.

**E. Impact of Doppler Spread on Key Generation**

Doppler spread $f_m$ is also an important parameter in key generation. Larger $f_m$ implies more drastic channel variation and leads to larger KGR. Indoor environment usually has larger Doppler spread than outdoor environment. We set $\lambda = 6^{-1}$, $N_p = 4$ to evaluate the impact of channel variation on key generation. We vary Doppler spread $f_m$ from 3 to 10 Hz.

Fig. 11(left) shows $f_m$ with AKGD. AKGD decreases as $f_m$ increases due to the increase of KGR. We compute the differences between simulation and theoretical results: $\Delta D_s = 0.1088s$, $e(D_s) = 0.86\%$

**F. Impact of SNR on Key Generation**

Finally, we investigate the impact of signal to noise ratio on key generation. We set $\lambda = 6^{-1}$, $N_p = 4$, $f_m = 5$ and vary SNR ($\sigma_n^2/\sigma_s^2$) from 0 dB to 30 dB.

Fig. 11(right) shows SNR with AKGD. AKGD decreases as SNR increases due to the improvement of channel quality. We compute the differences between simulation and theoretical results: $\Delta D_s = 0.029s$, $e(D_s) = 0.63\%$

**V. CONCLUSION**

In this paper, we considered the pairwise key generation in dynamic wireless networks with a center node (exemplified by the scenario of AP with random arrival users). We aim at deriving users’ average key generation delay.

We established the key generation model. We observed that in key generation model, each key establishment period has predictable time. We made use of such characteristic to treat the key generations as a discrete Markov process. We constructed state transition matrix. We deduced the probability of state transition and compute stationary distribution. We calculated mean number of users and timespan between states to get AKGD. Our method can deal with both serial and parallel probing scheduling.

We conducted simulations to evaluate the effectiveness of our model and method. The simulated AKGD and number of users distribution consist with our theoretical deduction. Simulation results show that parallel probing has shorter AKGD than serial probing due to its advantage on total KGR. We also compared simulation and theoretical AKGDs under various conditions. The results show that there are only slight differences between them.

Our model is useful in predicting AKGD under various conditions with different key generation scheduling. Extending our model to multi-hop wireless networks with heterogeneous channel conditions will be our future work.

**REFERENCES**


