SBrT 2021 1570724197

Improving Physical Layer Secret Key Generation in Fast Fading Environments Using Prediction

Pedro Ivo da Cruz, Alexandre Miccheleti Lucena, Ricardo Suyama and Murilo Bellezoni Loiola

***Abstract*— Physical layer security (PLS) is an alternative to traditional information security mechanisms. Physical layer secret key generation (PSKG) is a PLS technique that uses channel randomness to generate keys for encryption algorithms. Although several works investigated PSKG in slow fading channels, fast fading scenarios remain a challenge. We propose two predictors to deal with such scenarios, one based on the Recursive Least Squares (RLS) and another based on the Dual Extended Kalman Filter (DEKF). The aim is to mitigate the channel variation effects during PSKG, reducing the errors between keys generated by legitimate users.**

***Keywords*— Physical layer security, wireless communications, signal processing**

I. Introduction

Wireless communications systems are well known to be susceptible to security issues, such as eavesdropping, due to their broadcast nature. Nowadays, the security of the transmission relies on cryptography techniques, such as the advanced encryption system [1] and the Rivest–Shamir–Adleman (RSA) algorithm [2]. These techniques, however, rely on the assumption that it is difficult for an eavesdropper to guess the key used to encrypt the message. However, with the advance of technology, this might not hold for much longer. Furthermore, the distribution and management of such keys become exponentially difficult as networks increase in the number of nodes, for instance [3].

Physical layer security (PLS) has been a promising technique to work as an alternative or jointly with traditional cryptography-based algorithms. PLS techniques take advantage of the random characteristics of the wireless channel, such as fading, to provide secure communication. Physical layer Secret Key Generation (PSKG) is a PLS technique that uses the channel state information (CSI) to generate encryption keys.

PSKG highly depends on the assumption that the channel reciprocity between legitimate users [4], [5]. This has been widely investigated for slow fading channels, where the main reason for channel non-reciprocity is noise. The work in [6], for instance, uses a low pass filter to remove high frequencies components that are predominantly noise. The authors in [7] employ the Discrete Cosine Transform, while the works in [8] and [9] investigate the Discrete Wavelet Transform and Principal Component Analysis (PCA), respectively, for noise removal.

However, in fast fading channels, the hypothesis that the non-reciprocity is caused only by noise is no longer valid due to the increase in the Doppler frequency, which can happen by increasing the relative velocity between the transmitter and receiver or increasing the carrier frequency. It is important to consider higher relative velocity due to the increasing use of wireless networks in vehicular communications [10]. Additionally, technologies, such as mmWave, use higher carrier frequencies that can go from 30 GHz up to 300 GHz [11]. The 802.11ad Wi-Fi standard, for instance, uses a frequency of 60 GHz.

Therefore, this work proposes a new approach to deal with fast fading environments. The methods presented in this paper explore prediction techniques, based on Recursive Least Squares and the Dual Extended Kalman Filter (DEKF), to increase the cross-correlation between the channel measurements of the legitimate users to be used to generate encryption keys. We evaluate the proposed techniques in fast fading environments, considering the final cross-correlation obtained and the encryption key disagreement ratio (KDR).

The authors are with the Engineering, Modeling and Applied Social Sciences Center, Federal University of ABC, Santo André, SP, Brazil, E-mails: pedro.cruz@ufabc.edu.br, alexandre.lucena@ufabc.edu.br, ricardo.suyama@ufabc.edu.br, murilo.loiola@ufabc.edu.br. This work was funded in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 - and the National Council for Scientific and Technological Development – CNPq.

II. System Model

A. Channel Model

In this work, the wide sense stationary uncorrelated scattering channel model is considered [12]. The channel autocorrelation function provides the correlation between channel gains separated by a Δt time interval, which, for a Jakes Doppler power spectrum, is given by

$$r(\Delta t) = J_0(2\pi f_d \Delta t), \tag{1}$$

where $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind and

$$f_d = \frac{2vf_c}{c} \tag{2}$$

is the Doppler spread, which describes how fast the channel changes over time, v is the relative velocity between transmitter and receiver, f_c is the carrier frequency and c is the speed of the electromagnetic wave.

As it can be seen from (1) and (2), the correlation between channel measurements depends on Δt, v and f_c.

B. OFDM and Channel Estimation

This work considers the orthogonal frequency division modulation (OFDM) modulation [13], in which the n-th sample

of the time domain OFDM symbol is given by

$$x(n) = \sum_{k=0}^{N_c-1} X_k e^{j2\pi kn/N_c}, \tag{3}$$

where X_k are the quadrature modulated symbols in the k-th subcarrier and N_c is the number of subcarriers. The sum on the right side of the equation corresponds to the inverse discrete Fourier transform (IDFT). The time-domain OFDM symbol is formed by adding the cyclic prefix by copying the last N_{cp} samples obtained from the IDFT and adding them to the beginning. The resulting OFDM symbol is then transmitted through the channel, undergoing fading. After removing the cyclic prefix and evaluating the Discrete Fourier Transform (DFT) of the received signal in the time domain, the received signal in the frequency domain is given by

$$Y_k = H_k X_k + V_k, \tag{4}$$

where H_k is the channel response associated with the k-th subcarrier,

$$V_k = \sum_{k=0}^{N_c-1} v(n)\, e^{-j2\pi kn/N_c} \tag{5}$$

is the noise experienced at the k-th subcarrier and $v(n)$ is the additive white Gaussian noise (AWGN) with $v(n) \sim \mathcal{CN}(0, \sigma_v^2)$, where σ_v^2 is the variance of $v(n)$.

This work assumes that pilot symbols are transmitted using the block-type structure [13], in which the OFDM symbol contains pilots in all subcarriers, called the OFDM pilot symbol. Thus, it becomes possible to estimate the channel gains H_k in all subcarriers by

$$\hat{H}_k = \frac{Y_k}{X_k}. \tag{6}$$

A quantization algorithm converts these channel response into a random stream of bits.

III. Channel Probing and Prediction

In the fast fading scenario, the channel estimated by both legitimate users may not be sufficiently correlated due to the channel non-reciprocity. In this work, the AWGN is not the single cause for the cross-correlation reduction, since the CSI changes significantly over time. In such a scenario, a technique that only removes the noise from the signal is not enough to sufficiently increase the cross-correlation.

Thus, this work proposes prediction techniques to increase the cross-correlation between the measurements at the legitimate users, here called Alice and Bob. However, one of the users must collect a series of measurements that the prediction algorithms will use.

To do this and also collect enough channel measurements to generate long enough encryption keys for both Alice and Bob, they employ the following mechanism:

1) Alice sends N OFDM pilot symbols to Bob, where the first symbol is transmitted at the time t_0, the second at time t_1 and, therefore, the N-th symbols are transmitted at time t_{N-1}.
2) Bob then sends one OFDM pilot symbol to Alice. Assuming that the delay between Bob receiving the last symbol transmitted by Alice and sending its own is small enough to be considered zero, this symbol is transmitted at time t_N.
3) Alice estimates the channels in each subcarrier at time t_N.
4) Bob estimates the channel gains in each subcarrier at times $t_0,\ t_1,\ \cdots,\ t_{N-1}$, obtaining a set of N channel estimates for each subcarrier.
5) Bob predicts the channel at the instant t_N using the sets obtained in the previous step.

Alice and Bob can repeat this procedure until they have enough measurements to generate an encryption key whose length complies with the chosen standard requirements.

As the time instants t_m in which the channel were estimated are discrete, the channel estimates will be represented as $\hat{H}_k(m)$, where $t_m = mT_{OFDM}$, with $m \in \mathbb{N}$, and T_{OFDM} is the time necessary to transmit one OFDM symbol.

We present the prediction algorithms studied in this work in the sequel.

A. RLS Prediction

The time behaviour of the channel in fast fading environments decreases the cross-correlation between the channel estimates of Alice and Bob. In this case, the prediction goal is to produce a channel gain using the set of estimates in Bob that is closer to the channel estimate obtained in Alice and, therefore, has a higher cross-correlation with the measurement in Alice.

Predicting the channel using the RLS algorithm can be made by defining a window of length L_p, which is going to be the length of the predictor and also the number of taps of the linear filter, and by defining the vector $\mathbf{H}_k(m) = \left[\hat{H}_k(m),\ \hat{H}_k(m-1),\ \cdots,\ \hat{H}_k(m-L_p+1)\right]^{\mathrm{T}}$ containing the L_p most recent estimated channel samples at the k-th subcarrier. As the predictor runs independently for each subcarrier, the subscript k will be suppressed in the following equations for simplicity. The prediction, then, can be carried out by

$$\tilde{H}(m+1) = \mathbf{w}^{\mathrm{T}}\mathbf{H}(m), \tag{7}$$

where $\mathbf{w} = \left[w_0,\ w_1,\ \cdots,\ w_{L_p-1}\right]^{\mathrm{T}}$ are the predictor taps.

The goal of the exponentially weighted RLS algorithm is to find the best taps $\mathbf{w}$ that solves [14]

$$\begin{aligned} &\arg\min_{w} \lambda^N \mathbf{w}^{\mathrm{H}} \Pi \mathbf{w} \\ &+ \sum_{m=0}^{N-1} \lambda^{N-m-1} |\tilde{H}_k(m+1) - \mathbf{w}^{\mathrm{T}}\mathbf{H}(m)|^2 \end{aligned}, \tag{8}$$

where $(\cdot)^{\mathrm{H}}$ denotes the conjugate transpose, $(\cdot)^{\mathrm{T}}$ denotes the transpose, λ is a weighting variable called forgetting factor and Π is the regularization matrix.

The recursive algorithm to find the best solution $\mathbf{w}$ to (8) is summarized in Algorithm 1, where $(\cdot)^*$ denotes the conjugate of each element of a vector.

After the end of the recursions, the prediction of the channel at instant t_N in Bob is, thus, carried out by

$$\tilde{H}(N) = \mathbf{w}^{\mathrm{T}}\mathbf{H}(N-1). \tag{9}$$

Algorithm 1 RLS for prediction.

Initialization
$\mathbf{w} = \mathbf{0}_{L_p \times 1}$
$0 << \lambda \leq 1$
$\mathbf{P} = \Pi^{-1} = \mathbf{I}_{L_P}$
for $m \geq 0$ **do**
$\gamma = 1/(1 + \lambda^{-1}\mathbf{H}^{\mathrm{T}}(m)\mathbf{P}\mathbf{H}^{*}(m))$
$g = \lambda^{-1}\mathbf{P}\mathbf{H}^{*}(m)\gamma$
$\tilde{H}(m+1) = \mathbf{w}^{\mathrm{T}}\mathbf{H}(m)$
$e(m) = \hat{H}(m+1) - \tilde{H}(m+1)$
$\mathbf{w} = \mathbf{w} + ge(m)$
$\mathbf{P} = \lambda^{-1}\mathbf{P} - gg^{\mathrm{H}}/\gamma$
end for

B. Dual Extended Kalman Filter Prediction

The DEKF [15] allows the simultaneous estimation of states and unknown parameters of the model, by running two *Extended Kalman Filter* [16] concurrently, one for the state estimation (*state filter*) and another for parameter estimation (*parameter filter*). The idea is that the DEKF can also estimate a possible autoregressive (AR) model through the parameter estimation that describes the channel variation over time. To do this, a reformulation of the problem as a state-space representation is necessary. The state filter equations are given by

$$\xi(m) = f(\xi(m-1), \boldsymbol{\theta}) + \mathbf{z}(m-1) \tag{10}$$

$$\hat{H}(m) = \mathbf{C}\xi(m) + \boldsymbol{\varepsilon}(m). \tag{11}$$

For a AR(L_p) modeling of the prediction problem, $\xi(m)$ is the state vector composed of a window of past L_p channel coefficients $\xi(m) = \left[\hat{H}(m),\ \hat{H}(m-1),\ \cdots,\ \hat{H}(m-L_p+1)\right]^{\mathrm{T}}$. In addition, the term $f(\xi(m-1), \theta)$ in (10) can be replaced by $\mathbf{F}\xi(m-1)$ where the transition matrix $\mathbf{F}$ is an $L_p \times L_p$ matrix containing the unknown AR weights α_i that model the channel variation over time, given by

$$\mathbf{F} = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_{L_p-2} & \alpha_{L_p-1} & \alpha_{L_p} \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}. \tag{12}$$

The vector $\boldsymbol{\theta}$ is a parameter vector containing the unknown AR weights $\boldsymbol{\theta}(m) = \left[\alpha_1,\ \alpha_2,\ \cdots, \alpha_{L_p-1},\ \alpha_{L_p}\right]^{\mathrm{T}}$ to be estimated by the parameter filter. The observation matrix is defined as $\mathbf{C} = [1,\ 0,\ \cdots,\ 0]$, $\hat{H}(m)$ is the observed signal (estimated channel coefficient) and $\mathbf{z}(m)$ and $\boldsymbol{\varepsilon}(m)$ are the process and the observation noises, respectively, both with zero mean Gaussian distribution. The state-space equations of the parameter filter are then

$$\boldsymbol{\theta}(m) = \boldsymbol{\theta}(m-1) + \mathbf{q}(m-1), \tag{13}$$

$$\hat{H}(m) = g(\xi(m-1), \boldsymbol{\theta}(m)) + \mathbf{r}(m), \tag{14}$$

where $\mathbf{q}(m)$ and $\mathbf{r}(m)$ are the zero mean Gaussian process and observation noises, respectively. Using (10)-(14), the DEKF algorithm can be applied and is summarized in Algorithm 2 [17]. Variables with the $\hat{}$ symbol denote their respective estimate, while the apriori values are denoted by $\dot{()}^{-}$.

Algorithm 2 DEKF for prediction.

Initialization
$\hat{\xi}(-1) = 0$, $\hat{\boldsymbol{\theta}}(-1) \sim \mathcal{N}(0, 0.25)$, $\mathbf{P}_{\xi}(-1) = 1\mathrm{e}{-6}\mathbf{I}$, $\mathbf{Q}_{\xi} = 1\mathrm{e}{-3}\mathbf{I}$, $\mathbf{R}_{\xi} = 1\mathrm{e}{-2}\mathbf{I}$,
$\mathbf{P}_{\boldsymbol{\theta}}(-1) = 1\mathrm{e}{-6}\mathbf{I}$, $\mathbf{Q}_{\theta} = 1\mathrm{e}{-4}\mathbf{I}$, $\mathbf{R}_{\theta} = 1\mathrm{e}{-6}\mathbf{I}$
for $m \geq 0$ **do**
- State Filter
$\hat{\xi}^{-}(m) = \mathbf{F}\hat{\xi}(m-1)$
$\mathbf{P}_{\xi}^{-}(m) = \mathbf{F}\mathbf{P}_{\xi}(m-1)\mathbf{F}^{\mathrm{T}} + \mathbf{Q}_{\xi}(m-1)$
$\mathbf{K}_{\xi}(m) = \mathbf{P}_{\xi}^{-}(m)\mathbf{C}^{\mathrm{T}}\left[\mathbf{C}\mathbf{P}_{\xi}^{-}(m)\mathbf{C}^{\mathrm{T}} + \mathbf{R}_{\xi}(m)\right]^{-1}$
$\hat{\xi}(m) = \hat{\xi}^{-}(m) + \mathbf{K}_{\xi}(m)\left[\hat{H}(m) - \mathbf{C}\hat{\xi}^{-}(m)\right]$
$\mathbf{P}_{\xi}(m) = \left[\mathbf{I} - \mathbf{K}_{\xi}(m)\mathbf{C}\right]\mathbf{P}_{\xi}^{-}(m)$
- Parameter Filter
$\hat{\boldsymbol{\theta}}^{-}(m) = \hat{\boldsymbol{\theta}}(m-1)$
$\mathbf{P}_{\boldsymbol{\theta}}^{-}(m) = \mathbf{P}_{\boldsymbol{\theta}}(m-1) + \mathbf{Q}_{\boldsymbol{\theta}}(m-1)$
$\mathbf{K}_{\theta}(m) = \mathbf{P}_{\boldsymbol{\theta}}^{-}(m)\mathbf{G}^{\mathrm{T}}\left[\mathbf{G}\mathbf{P}_{\boldsymbol{\theta}}^{-}(m)\mathbf{G}^{\mathrm{T}} + \mathbf{R}_{\boldsymbol{\theta}}(m)\right]^{-1}$
$\hat{\boldsymbol{\theta}}(m) = \hat{\boldsymbol{\theta}}^{-}(m) + \mathbf{K}_{\theta}(m)\left[\hat{H}(m) - \mathbf{C}\hat{\xi}^{-}(m)\right]$
end for

The nonlinear function $g(\xi(m-1), \boldsymbol{\theta}(m))$ can be linearized to $\mathbf{G}\boldsymbol{\theta}(m)$ where $\mathbf{G}$ is obtained through the partial derivative of the state vector $\xi(m)$ with respect to the parameter vector $\boldsymbol{\theta}$ as in

$$\mathbf{G} = \left.\frac{\partial g(\xi(m-1), \boldsymbol{\theta}(m))}{\partial \boldsymbol{\theta}}\right|_{\hat{\boldsymbol{\theta}}^{-}(m)}. \tag{15}$$

which, for this particular case, can be replaced by $\mathbf{G} = \xi^{-\,\mathrm{T}}(m)$. The matrices $\mathbf{Q}_{\xi}$ and $\mathbf{R}_{\xi}$ are the covariance matrices of the State filter process noise $\mathbf{z}(m)$ and measurement noise $\boldsymbol{\varepsilon}(m)$ respectively, while $\mathbf{Q}_{\boldsymbol{\theta}}$ and $\mathbf{R}_{\boldsymbol{\theta}}$ are the covariance matrices of the parameter filter noises $\mathbf{q}(m)$ and $\mathbf{r}(m)$, respectively. Since, in this case, the interest is the predicted value, the prediction performance was given by the evaluation of the *a priori* values $\xi^{-}(m)$ obtained by the algorithm, where $\mathbf{F}$ is constantly updated through the parameter estimation process.

IV. Quantization

Once Alice and Bob collects enough channel estimates and predictions, respectively, these samples are organized according to the fine-grained CSI extraction for key generation [6], forming a vector,

$$X_a = [\Re\{\hat{H}(m_o)\}, \Im\{\hat{H}(m_o)\},\ \cdots,\ \Re\{\hat{H}(m_{T-1})\}, \Im\{\hat{H}(m_{T-1})\}], \tag{16}$$

$$X_b = [\Re\{\tilde{H}(m_o)\}, \Im\{\tilde{H}(m_o)\},\ \cdots,\ \Re\{\tilde{H}(m_{T-1})\}, \Im\{\tilde{H}(m_{T-1})\}], \tag{17}$$

in Alice and Bob, respectively. $\Re$ and $\Im$ denotes the real and imaginary part of their arguments, respectively. These vectors go through a quantization algorithm to generate a sequence of

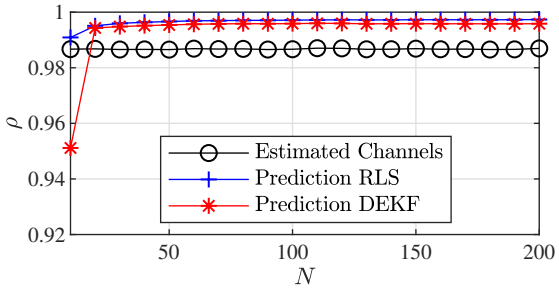

Fig. 1. Impact of N on the correlation coefficient. SNR $= 60$ dB and $f_d = 4.8$ kHz.

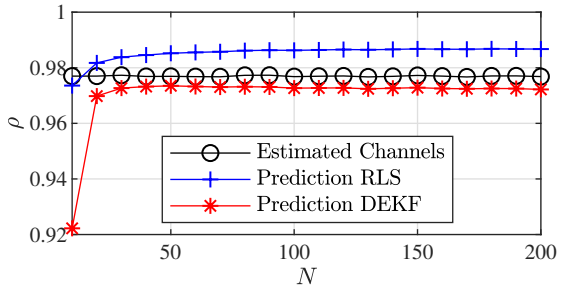

Fig. 2. Impact of N on the correlation coefficient. SNR $= 20$ dB and $f_d = 4.8$ kHz.

bits to be used as an encryption key. The algorithm used here is the mean based quantization [18], that generates the key at the user $u = \{a,\ b\}$, with a denoting Alice and b denoting Bob, and can be described by

$$K_u(i) = \begin{cases} 0, & X_u(i) \leq \mu \\ 1, & X_u(i) > \mu \end{cases}, \tag{18}$$

where μ is the mean of the input X_u.

V. Simulations and Results

We carried out simulations considering an OFDM system with 128 subcarriers with a cyclic prefix of 16 samples. The sample rate is 20 MHz, and the channel impulse response has five taps generated by the sum of sinusoids method [12]. The total number of trials in each simulation is 10^4.

The cross-correlation between X_a and X_b will be evaluated by the correlation coefficient, which can be computed, given that $X_a \sim \mathcal{N}(0, \sigma_a^2)$ and $X_b \sim \mathcal{N}(0, \sigma_b^2)$, as

$$\rho = \frac{\mathrm{E}\{X_a X_b^{\mathrm{H}}\}}{\sigma_a \sigma_b}. \tag{19}$$

The KDR provides the ratio of mismatched bits between the keys generated at Alice and Bob. It is defined as

$$KDR = \frac{\sum_i |K_a(i) - K_b(i)|}{N_k}, \tag{20}$$

where N_k is the length of the keys.

We include the cross-correlation when the measurements correspond to the estimated channels at Alice and Bob, i.e., $\hat{H}_k(N)$, estimated at Alice, and $\hat{H}_k(N-1)$ for comparison purposes. Figures 1 and 2 show the correlation obtained according to the number of OFDM symbols sent by Alice N, which produces the same number of channel estimates used to perform the prediction at Bob. The signal-to-noise ratio (SNR) is set to 60 dB and 20 dB, respectively. We chose these SNR values to evaluate a noisy and almost noiseless scenario, where only the maximum Doppler spread will affect the system performance. For this simulation, the carrier frequency was set to 120 GHz, and the relative velocity between Alice and Bob was 6 m/s, which results in a maximum Doppler spread of around 4.8 kHz. The predictors' length was set to $L_p = 5$. The RLS forgetting factor was set to $\lambda = 0.998$.

Figures 1 and 2 shows the correlation coefficient increases as N increases, achieving a plateau in $N = 60$ of around $\rho = 0.997$ for the RLS, and around $\rho = 0.996$ for the DEKF

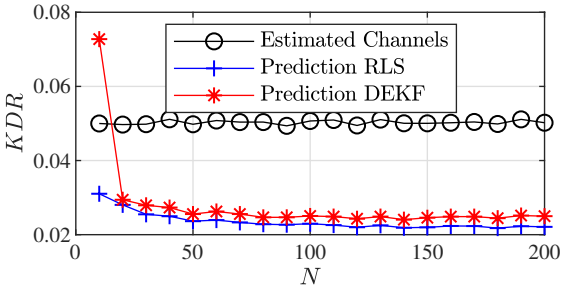

Fig. 3. Impact of N on the KDR. SNR $= 60$ dB and $f_d = 4.8$ kHz.

for a SNR regime of 60 dB. For 20 dB, the RLS achieves the plateau in $N = 80$ with $\rho = 0.986$, while the DEKF achieves its plateau in $N = 40$ with $\rho = 0.973$. The number of OFDM symbols N spans from 10 to 200 in both figures. According to Fig. 1 the RLS improves the cross-correlation for all N, compared to the cross-correlation between the direct channel estimates. The DEKF decreases the cross-correlation when $N = 10$ in 60 dB regime. For a 20 dB SNR, on the other hand, the RLS reduces its improvement because of the higher level of noise. The DEKF provided a cross-correlation below the one obtained by the direct estimated channels for all values of N.

The KDR produced for $SNR = 60$ dB is shown in Figure 3. It is possible to see that the RLS obtains the lowest KDR and, although the correlation achieves a plateau in $N \geq 20$, the KDR reaches a floor in $N \geq 120$. The DEKF reaches this floor around the same value of N but provides a slightly higher KDR. For $N = 10$, the KDR obtained using the DEKF is higher than the one obtained using the direct channel estimates, without prediction. The RLS, on the other hand, provides a lower KDR even for $N = 10$.

The following scenario evaluates the impact of the maximum Doppler spread on the predictors. In this evaluation, we set N to 50 and the SNR to 60 dB. Fig. 4 shows the correlation obtained when the maximum Doppler spread spans from 2.4 kHz to 24 kHz. Those values come, for instance, from a relative velocity of 6 m/s (equivalent to 21.6 km/h) in a carrier frequency of 60 GHz, and from a relative velocity of 30 m/s (equivalent to 108 km/h) in a carrier frequency of 120 GHz.

As can be seen, the general behaviour is that the correlation decreases as f_d increases. The reason behind that is that it is more difficult for the predictors to obtain a reasonable solution for $\mathbf{w}$, in the case of the RLS, and for $\mathbf{F}$ in the case of the DEKF,

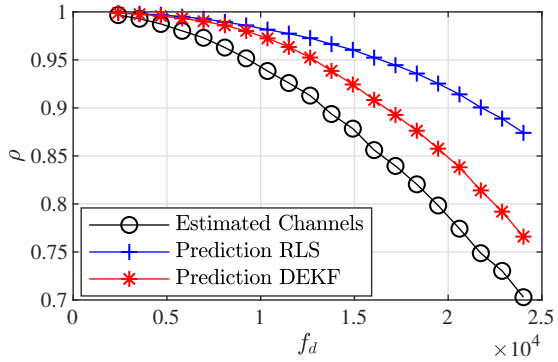

Fig. 4. Impact of maximum Doppler spread on the correlation coefficient. SNR $= 60$ dB and $N = 50$.

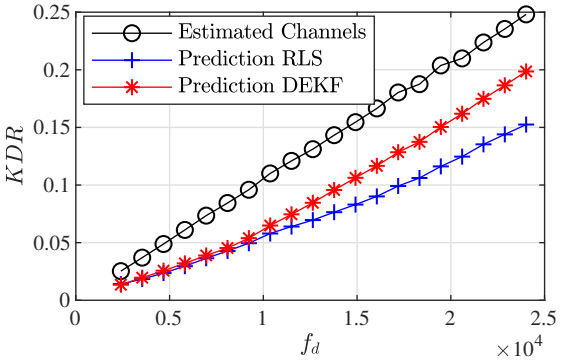

Fig. 5. Impact of maximum Doppler spread on the KDR. SNR $= 60$ dB and $N = 50$.

for higher values of f_d. A possible solution to that would be to increase N to provide more data to the algorithms. They could then converge to a better solution. Another possibility would be increasing L_p. The RLS, however, is still able to obtain a solution that provides highly correlated measurements. For the highest maximum Doppler spread evaluated, $f_d = 24$ kHz, the RLS attained approximately $\rho = 0.875$, while the DEKF obtained $\rho = 0.77$. For comparison purposes, the correlation between the directly estimated channels is slightly above $\rho = 0.7$, as shown in Fig. 4.

Fig. 5 shows the impact of this behaviour on the KDR. It is possible to see that the RLS obtained the best performance for higher f_d values. For $f_d = 24$ kHz, the KDR obtained was just above 0.15, while the DEKF attained a KDR around 0.2. An additional step of key reconciliation can then be employed to reduce the bit mismatch to zero. The work in [6] shows that a key reconciliation technique using Bose–Chaudhuri–Hocquenghem code with the parameters $\mathrm{BCH}(15,\ 3,\ 3)$ can correct up to 20% mismatch. For low f_d values, the gain of the RLS over the DEKF is not significant.

VI. Conclusions and Future Work

This paper proposes to use predictors to increase the correlation between channel measurements used for encryption key generation in the physical layer in scenarios with high mobility and, possibly, higher carrier frequencies. We designed two algorithms, one based on the RLS and another based on the DEKF.

The RLS has shown significantly better performance in extreme cases, where the maximum Doppler spread is high. For lower values, the RLS slightly outperforms the DEKF. The DEKF does not outperform the RLS in any scenario, which would make the RLS the best choice for this problem, since it has a lower computational cost.

Nonetheless, as this is a work in progress, further evaluations are going to be carried out. For instance, the impact of L_p on the predictor's performance, as well as the randomness of the generated encryption key. Furthermore, the authors intend to obtain the theoretical expressions for the correlation based on the error, mainly for the RLS algorithm that shows the best performance.

References

[1] *Advanced Encryption Standard*, Federal Information Processing Standards Publication 197 Std., 2001.
[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, Feb. 1978.
[3] J. M. Hamamreh, H. M. Furqan, and H. Arslan, "Classifications and applications of physical layer security techniques for confidentiality: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1773–1828, 2019.
[4] J. Zhang, R. Woods, T. Q. Duong, A. Marshall, and Yuan Ding, "Experimental study on channel reciprocity in wireless key generation," in *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, vol. 4, no. 99. IEEE, Jul. 2016, pp. 1–5.
[5] J. Zhang, B. He, T. Q. Duong, and R. Woods, "On the key generation from correlated wireless channels," *IEEE Communications Letters*, vol. 21, no. 4, pp. 961–964, Apr. 2017.
[6] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong, "Efficient key generation by exploiting randomness from channel responses of individual OFDM subcarriers," *IEEE Transactions on Communications*, vol. 64, no. 6, pp. 2578–2588, 2016.
[7] G. Margelis, X. Fafoutis, G. Oikonomou, R. Piechocki, T. Tryfonas, and P. Thomas, "Physical layer secret-key generation with discreet cosine transform for the Internet of Things," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, May 2017, pp. 1–6.
[8] F. Zhan and N. Yao, "On the using of discrete wavelet transform for physical layer key generation," *Ad Hoc Networks*, vol. 64, pp. 22–31, Sep. 2017.
[9] G. Li, A. Hu, J. Zhang, L. Peng, C. Sun, and D. Cao, "High-agreement uncorrelated secret key generation based on principal component analysis preprocessing," *IEEE Transactions on Communications*, vol. 66, no. 7, pp. 3022–3034, Jul. 2018.
[10] B. M. ElHalawany, A. A. A. El-Banna, and K. Wu, "Physical-layer security and privacy for vehicle-to-everything," *IEEE Communications Magazine*, vol. 57, no. 10, pp. 84–90, Oct. 2019.
[11] G. Li, C. Sun, J. Zhang, E. Jorswieck, B. Xiao, and A. Hu, "Physical layer key generation in 5G and beyond wireless communications: Challenges and opportunities," *Entropy*, vol. 21, no. 5, p. 497, May 2019.
[12] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, Eds., *Simulation of Communication Systems: Modeling, Methodology and Techniques*, 2nd ed. Norwell, MA, USA: Kluwer Academic Publishers, 2000.
[13] Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, *MIMO-OFDM Wireless Communications with MATLAB*. Wiley Publishing, 2010.
[14] A. Sayed, *Fundamentals of Adaptive Filtering*, ser. Wiley - IEEE. Wiley, 2003.
[15] E. A. Wan and A. T. Nelson, "Dual kalman filtering methods for nonlinear prediction, smoothing and estimation," in *Advances in neural information processing systems*, 1997, pp. 793–799.
[16] S. Haykin, *Adaptive filter theory*. Prentice-Hall, Inc., 1996.
[17] ——, *Kalman filtering and neural networks*. John Wiley & Sons, 2004, vol. 47.
[18] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy," in *Proceedings of the 14th ACM international conference on Mobile computing and networking - MobiCom '08*. New York, New York, USA: ACM Press, 2008, p. 128.