# MODELLING JITTER IN WIRELESS CHANNEL CREATED BY PROCESSOR-MEMORY ACTIVITY

*Baki Berkay Yilmaz and Alenka Zajić*

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA, 30332 USA

*Milos Prvulovic*

School of Computer Science
Georgia Institute of Technology
Atlanta, GA, 30332 USA

## ABSTRACT

A wireless communication created by a computer software activity is described and modelled. The generation of this communication link is a consequence of electromagnetic (EM) emanations emitted during computer activity. This wireless channel in addition to channel errors due to noise, also experiences jitter created by the software activity "transmitter" which lacks precise synchronization. Also, the "transmitter" gets interrupted with other (system) activity, and the transmitted signal goes through a channel obstructed by metal, plastic, etc. To capture all these effects, we have modelled transmitted sequence as a pulse amplitude modulated (PAM) signal with random varying pulse position. From the model, we have derived the power spectral density and the bit error rate of the transmitted signal and presented performance analysis of such a channel.

***Index Terms***— Wireless security, electromagnetic information leakage, covert channel attacks

## 1. INTRODUCTION

This paper models and analyses wireless communication link created as a consequence of electromagnetic (EM) emanations (side-channels) emitted during the computer software activities. The generation of such a wireless communication channel can encourage motivated attackers to leak some valuable information from the systems. In that respect, side channel attacks generally require some degree of direct access to the targeted systems. Some examples of these attacks can be power analysis [1, 2, 3, 4, 5, 6, 7, 8, 9], temperature analysis [10, 11] or caches-based [12, 13, 14]. Fortunately, these attacks often face with risk of detection due to need of direct access. On the other hand, attacks based on EM emanations only require physical proximity. Since the transmitter code is innocuous-looking and the attacker does not require a direct access, many attacks can be performed with little risk of detection. An example of a wireless communication system exploiting the emanations during the system activity is demonstrated in [15], removing any doubts whether such a wireless communication exists.

In contrast to most communication systems, which are designed to avoid symbol loss, side-channels are not designed to transfer information at all and their transmission is often corrupted by erroneous transfer of bits. Like in traditional wireless communications, some errors in side-channel occur due to variation in the propagation environment. However, in addition to channel errors, the software activity "transmitter" lacks precise synchronization, causing jitter that reduces the signal's effective bandwidth *and* increases the noise level. Also, the "transmitter" gets interrupted with other (system) activity, and the transmitted signal goes through a channel obstructed by metal, plastic, etc. To capture all effects of the observed behaviour, we have modelled the transmitted sequence as a pulse amplitude modulated (PAM) signal with random varying pulse position. From the model, we have derived the power spectral density and the bit error rate of the transmitted signal with only substitution errors.

The rest of the paper is organized as follows: Section 2 introduces the creation of the wireless communication based on software activity, Section 3 models the overall communication system, and calculates the PSD of the jitter noise and the signal, and Section 4 provides BER performance for different program activities, experimental results and concluding remarks.

## 2. SOFTWARE-ACTIVITY-CREATED WIRELESS COMMUNICATION CHANNEL

In this section, the creation of the modulated signal is considered. We first need to generate a carrier. The method we use to produce a carrier is to create repetitive variations in software activity [15]. We choose T, the period (duration) of each repetition, two types of activity (A and B), and write a small software code (i.e. microbenchmark) that in each period does activity A in the first half and B in the second half of the period. The intuition behind this is that, if activity A and activity B result in non-identical EM fields around

This work has been supported, in part, by NSF grants 1563991 and 1318934, AFOSR grant FA9550-14-1-0223, and DARPA LADS contract FA8650-16-C-7620. The views and findings in this paper are those of the authors and do not necessarily reflect the views of NSF, AFOSR, and DARPA.

the processor or the system, repetition of this A-then-B pattern will create oscillations (with period T) in this EM field, i.e. it will result in a "carrier" RF signal at frequency $1/T$. The period T will be selected to correspond to a specific frequency, e.g. to produce a radio signal at 1 MHz, we should set $T = 1\mu s$. This carrier-generation approach is illustrated in Figure 1. Next, AM modulation is achieved by inserting

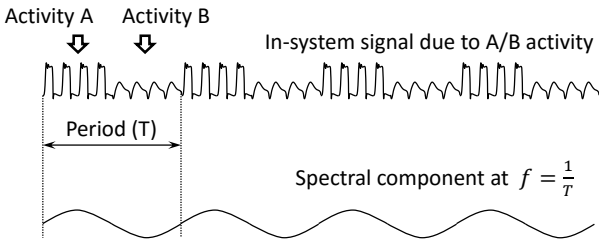



**Fig. 1**. Emanations at a specific radio frequency induced by half-periods of activities A and B.

intervals during which only activity B is performed in both half-periods which means any carrier signal produced by differences between A and B should be absent when only B is used, resulting in the simplest form of AM modulation (on-off keying). This approach is illustrated in Figure 2. Note that other modulations (e.g. frequency modulation or even some non-standard modulation) can just as easily be used to create a truly covert transmission that can still be received by a customized receiver/demodulator.

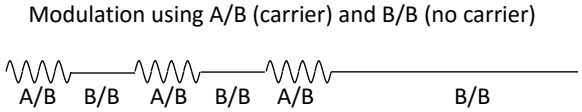



**Fig. 2**. Modulating the signal into the carrier.

This framework results in a simple form of on-off keying. Two experiments are conducted to verify that the received signal is the transmitted message [15, 16]. However, the pulses generated by on-off keying do not have equal timing due to varying timing of instruction executions. Therefore, precise synchronization could not be achieved and the system faces with jitter noise which must be considered during the analysis of the system.

## 3. TRANSMISSION MODEL FOR SOFTWARE-ACTIVITY-CREATED SIGNALS

In this section, we introduce the model for the transmitted signal and obtain the PSD of received signal including white and jitter noise.

In Section 2, the structure of the generated signal is shown to be "on-off" keying, therefore, we model the baseband signal as a PAM signal corrupted by jitter noise. In a synchronized system, the baseband PAM signal is given as [17]

$$\tilde{x_p}(t) = \sum_k x_k p(t - kT) \tag{1}$$

where $\mathbf{x}_k = (x_k, x_{k-1}, x_{k-2}, \ldots)$ is the sequence of data symbols that are chosen from a finite alphabet and p(t) is a shaping pulse. Unlike synchronized channels, the pulses created by software activities are not well synchronized and, therefore, utilizing (1) could not capture the structure of the overall scheme. To model the proposed framework, we need to incorporate (1) with jitter noise. In that respect, we insert a random pulse shifter, $\mathbf{T_k}$, whose pdf is supported between $[-T/2, T/2)$ and obtain the following PAM signal:

$$x_p(t) = \sum_k x_k p(t - kT - \mathbf{T_k}). \tag{2}$$

If we assume that $\delta(t)$ is chosen as the pulse shaping function, the PSD of PAM signal with random pulse position at the receiver side can be written as

$$S_y(f) = \frac{1}{T} S_x(f)\Phi(f) + \frac{R_x[0]}{T}(1 - \Phi(f)) \tag{3}$$

where $\Phi(f)$ is the Fourier transform of $\phi(\tau)$ and

$$\phi(\tau) = \int f_{\mathbf{T}}(\tau + t) f_{\mathbf{T}}(t) dt = f_{\mathbf{T}}(\tau) * f_{\mathbf{T}}(-\tau) \tag{4}$$

and $f_{\mathbf{T}}(\bullet)$ is the pdf of the random pulse position.

*Proof.* See Section 6. □

We can observe from (1) that the pdf of the jitter noise determines the PSD of the received signal. In Fig. 3.(a), we plot the experimental results for the timing distributions of software activities with and without memory activities. Although we constraint the support set of the distributions to be in a finite interval, the best fit for the experimental data appears to be a normal distribution. Fortunately, the total probability beyond our constraint is almost zero, hence, a normal distribution with mean $\mu$ and standard deviation $\sigma$ is considered for the pdf of random pulse shift. Keeping Fourier transform of Gaussian distribution, i.e. $e^{-j2\pi f\mu} e^{-2\pi^2\sigma^2 f^2}$, in mind, we have $\Phi(f) = e^{-4\pi^2\sigma^2 f^2}$ and finally

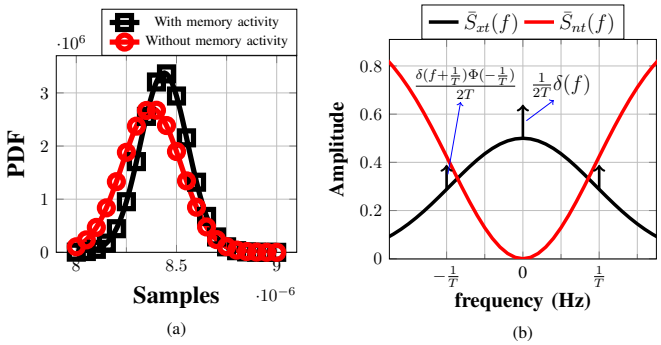
$$S_y(f) = \frac{1}{T} S_x(f) e^{-4\pi^2\sigma^2 f^2} + \frac{R_x[0]}{T}(1 - e^{-4\pi^2\sigma^2 f^2}). \tag{5}$$

The first and second summands of (5) represent the PSD of the signal and jitter noise denoted by $S_{xt}(f)$ and $S_{nt}(f)$, respectively.

For the received signal, multipath does not play a significant role in the generated communication described in Section 2 because it occurs at lower frequencies. Therefore, the received signal can be written as

$$r(t) = y(t) + n(t). \tag{6}$$

Using (3) and (6), the PSD of received signal can be written as $S_r(f) = S_{xt}(f) + S_{nt}(f) + N_0/2$ where $N_0/2$ is the additive




(a) (b)

**Fig. 3**. a) Symbol timing distributions with memory and without memory activity, b) Power spectral density of jittery noise and signal.

white noise power. Since we model the transmitted signal as a PAM signal, we first need to write the autocorrelation function of the input sequence as

$$R_x[m] = \mathcal{A}^2 \cdot \left(\frac{1}{2} - \frac{\Im\{m \neq 0\}}{4}\right) \tag{7}$$

where $\Im$ is the indicator function whose output is one if its argument is true and zero otherwise, and $\mathcal{A}$ is the amplitude of symbols when the symbols are "on". Therefore, the PSD of the input sequence can be written as

$$S_x(f) = \frac{R_x[0]}{2}\left(1 + \frac{1}{T}\sum_m \delta(f - m/T)\right). \tag{8}$$

Merging the results in (3) and (8), the PSD of the received signal, $S_y(f)$, including jitter noise can be written as

$$\frac{\mathcal{A}^2}{2T}\left(\underbrace{\left(1 + \sum_m \frac{\delta(f - m/T)}{T}\right)\frac{\Phi(f)}{2}}_{\bar{S}_{xt}(f)} + \underbrace{(1 - \Phi(f)}_{\bar{S}_{nt}(f)}\right) \tag{9}$$

where $\bar{S}_{xt(f)}$ and $\bar{S}_{nt}(f)$ represent the normalized versions of signal and jitter noise spectrum. In Fig. 3.(b), we plot $\bar{S}_{xt(f)}$ and $\bar{S}_{nt}(f)$ based on the pulse shifter distribution given in Fig. 3.(a) without memory activity. It is clear that the jitter noise beats the signal power for higher frequencies. Therefore, we convolve the received signal with low pass filter whose bandwidth is $1/2T$ since the signal period is $T$.

## 4. EXPERIMENTAL RESULTS

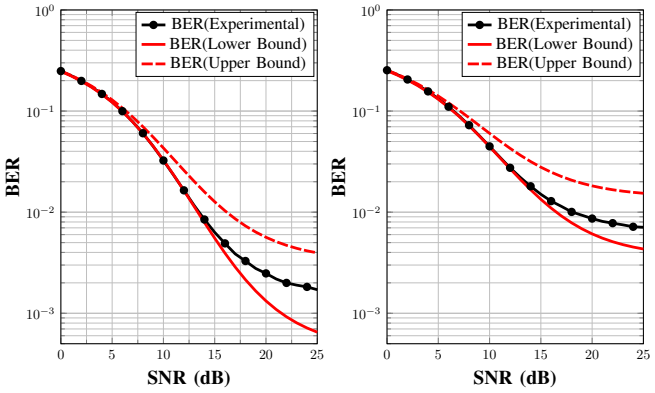
Having the PSD of the signal and the noise should be sufficient to calculate BER performance of the communication system. Such a measure will unveil how reliable a communication system is for a targeted Signal-to-Noise ratio (SNR). The probability of error for "on-off" keying is given as [17]

$$P_{PAM} = Q\left(\sqrt{\frac{P_s}{2P_n}}\right). \tag{10}$$

For the proposed scheme, the ratio between transmitted and the overall noise signals including both jitter and white noise is given as

$$\begin{aligned}\frac{P_s}{P_n} &= \frac{\int_{-1/2T}^{1/2T} S_{xt}(f)df}{\int_{-1/2T}^{1/2T} (S_{nt}(f) + N_0/2)df} \\ &= \frac{\frac{\mathcal{A}^2}{2} + \frac{\sqrt{\pi}}{4}\mathrm{erf}\left(\pi\sigma/T\right)/(\pi\sigma/T)}{\mathcal{A}^2 \cdot N_J + N_0}\end{aligned} \tag{11}$$

where $N_J = (1 - \frac{\sqrt{\pi}}{2}\mathrm{erf}\left(\pi\sigma/T\right)/(\pi\sigma/T))$ and SNR is defined as $\mathcal{A}^2/N_0$. Note that the equation given in (10) assumes the noise is white. However, the jitter noise does not behave like a white noise, and therefore, we distribute the overall power of jitter noise over the support range of the low pass filter in (11). Distribution of all available power over all frequency components equally means the decrease in the power margin of high frequency components. Therefore, a better BER performance for the actual system is obtained because the reconstruction errors occur due to high frequency components of the jitter noise. Therefore, inserting (11) into (10) will give a lower bound for the BER for the proposed system.




(a) (b)

**Fig. 4**. BER for the communication channels created a) without memory and b) with memory activity.

Since we have a lower bound for BER, having also an upper bound will provide the range for the actual BER. In that respect, we add an extra term to the denominator which is equal to total loss in signal power due to jitter effect. This added noise power is the half of the total jitter noise power and, therefore, SNR can be written as

$$\frac{P_s}{\hat{P}_n} = \frac{\frac{\mathcal{A}^2}{2} + \frac{\sqrt{\pi}}{4}\mathrm{erf}\left(\pi\sigma/T\right)/(\pi\sigma/T)}{\frac{3\mathcal{A}^2}{2} \cdot N_J + N_0}. \tag{12}$$

Fig. 4 plots BER for the experimental result, and lower and upper bounds. The bounds are very strict when the additive noise power is dominant and gets looser as SNR increases. However, when additive noise power is negligible by comparing jitter noise power, both the bounds and actual BER do not decrease further for a specific jitter power and the gap between lower and upper bounds stays same. Moreover, as the jitter noise power decreases, the bounds get stricter as expected.

## 5. CONCLUSIONS

A wireless communication created by a computer software activity is described and modelled. The generation of this communication link is a consequence of electromagnetic (EM) emanations emitted during computer activity. This wireless channel in addition to channel errors due to noise, also experiences jitter created by the software activity "transmitter" which lacks precise synchronization. Also, the "transmitter" gets interrupted with other (system) activity, and the transmitted signal goes through a channel obstructed by metal, plastic, etc. To capture all these effects, we have modelled transmitted sequence as a pulse amplitude modulated (PAM) signal with random varying pulse position. From the model, we have derived the power spectral density and the bit error rate of the transmitted signal and presented performance analysis of such a channel.

## 6. APPENDIX - DERIVATION OF EQUATION 3

In this section, we provide the derivation of (3). To simplify the derivation, we assume $p(t) = \delta(t)$ and define the transmitted signal as

$$y(t) = \sum_k x_k \delta(t - kT - \mathbf{T_k}). \tag{13}$$

It can be shown that the signal given in (13) is cyclostationary process assuming $\mathbf{T_k}$ and $x_k$ are i.i.d and stationary. Therefore, the autocorrelation of $y(t)$ can be written as

$$\begin{aligned} R_y(\tau) &= \lim_{K\to\infty} \frac{1}{KT} \int_{-\frac{KT}{2}}^{\frac{KT}{2}} y(t)y^*(t-\tau)dt \\ &= \frac{1}{T}\int_0^T \lim_{K\to\infty} \frac{1}{K} \sum_{k=-\frac{K}{2}}^{\frac{K}{2}} y(t+kT)y^*(t+kT-\tau)dt \\ &= \frac{1}{T}\int_0^T \mathbb{E}\left[y(t)y^*(t-\tau)\right] dt \end{aligned} \tag{14}$$

where the last equation follows that the process is periodic in time with a period $T$. If we define $R_y(t,\tau) = \mathbb{E}\left[y(t)y^*(t-\tau)\right]$, we can write $R_y(t,\tau)$ as:

$$\mathbb{E}\left[\sum_i \sum_j x_i x_j \delta(t - iT - \mathbf{T}_i)\delta(t - \tau - jT - \mathbf{T}_j)\right]. \tag{15}$$

Exploiting the assumption $x_k$ and $\mathbf{T}_k$ are independent and stationary, $\delta(t-t_0)f(t) = \delta(t-t_0)f(t_0)$, defining $m = j-i$ and denoting $R_x[m] = \mathbb{E}\left[x_i x_j\right]$, we can rewrite (15) as

$$\sum_m R_x[m]\tilde{y}(t) \tag{16}$$

where $\tilde{y}(t) = \sum_i \mathbb{E}\left[\delta(t - iT - \mathbf{T}_0)\delta(\mathbf{T}_i - \tau - mT - \mathbf{T}_m)\right]$. By combining (14) and (16), we have

$$R_y(\tau) = \frac{1}{T}\sum_m R_x[m] \int_0^T \tilde{y}(t)dt = \frac{1}{T}\sum_m R_x[m] r_y(\tau). \tag{17}$$

If we define $\lambda = t - iT$ and change the variable of integration, and assume the random pulse positions are distributed normally (although it violates the finite support set assumption), we have for $m \neq 0$

$$\begin{aligned} r_y(\tau) &= \sum_i \int_{-iT}^{-(i-1)T} \mathbb{E}\left[\delta(\lambda - \mathbf{T}_0)\delta(\mathbf{T}_0 - \tau - mT - \mathbf{T}_m)\right] d\lambda \\ &= \int_{-\infty}^{\infty} \mathbb{E}\left[\delta(\lambda - \mathbf{T}_0)\delta(\mathbf{T}_0 - \tau - mT - \mathbf{T}_m)\right] d\lambda \\ &= \mathbb{E}\left[\delta(\mathbf{T}_0 - \tau - mT - \mathbf{T}_m)\right] \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \big[\delta(-\tau - mT + t_0 - t_m) \times \\ &\qquad f_{\mathbf{T}_0}(t_0) f_{\mathbf{T}_m}(t_m) dt_0 dt_m\big] \\ &\overset{(a)}{=} \int_{-\infty}^{\infty} f_{\mathbf{T}_0}(\tau + mT + t_m) f_{\mathbf{T}_0}(t_m) dt_m \\ &= f_{\mathbf{T}_0}(-\tau + mT) * f_{\mathbf{T}_0}(\tau) \\ &= \delta(\tau - mT) * f_{\mathbf{T}_0}(-\tau) * f_{\mathbf{T}_0}(\tau) \\ &= \delta(\tau - mT) * \phi(\tau) \end{aligned} \tag{18}$$

where (a) follows that the random pulse position distributions are iid. When $m = 0$, $r_y(\tau)$ is equal to $\delta(\tau)$. Hence, $R_y(\tau)$ can be written as

$$\frac{1}{T}\left(R_x(0)\delta(\tau) + \sum_{m\neq 0} R_x(m)\Big(\delta(\tau - mT) * \phi(\tau)\Big)\right). \tag{19}$$

By adding and substracting $\frac{R_x(0)\phi(\tau)}{T}$, we can rewrite (19) as

$$\sum_m \frac{R_x(m)}{T}\delta(\tau - mT) * \phi(\tau) + \frac{R_x(0)}{T}\Big(\delta(\tau) - \phi(\tau)\Big). \tag{20}$$

To obtain the PSD of $y(t)$, we need to take the Fourier transform of $R_y(\tau)$. Therefore, if we take the transform of (20), we have the PSD of the signal as

$$S_y(f) = \frac{1}{T}S_x(f)\Phi(f) + \frac{R_x(0)}{T}(1 - \Phi(f)) \tag{21}$$

where $\Phi(f)$ is the Fourier transform of $\phi(\tau)$ which concludes the proof.

## 7. REFERENCES


[1] A G Bayrak, F Regazzoni, P Brisk, F.-X. Standaert, and P Ienne, "A first step towards automatic application of power analysis countermeasures," in *Proceedings of the 48th Design Automation Conference (DAC)*, 2011.

[2] Dan Boneh and David Brumley, "Remote Timing Attacks are Practical," in *Proceedings of the USENIX Security Symposium*, 2003.

[3] S Chari, C S Jutla, J R Rao, and P Rohatgi, "Towards sound countermeasures to counteract power-analysis attacks," in *Proceedings of CRYPTO'99, Springer, Lecture Notes in computer science*, 1999, pp. 398–412.

[4] B Coppens, I Verbauwhede, K De Bosschere, and B De Sutter, "Practical Mitigations for Timing-Based Side-Channel Attacks on Modern x86 Processors," in *Proceedings of the 30th IEEE Symposium on Security and Privacy*, 2009, pp. 45–60.

[5] L Goubin and J Patarin, "DES and Differential power analysis (the "duplication" method)," in *Proceedings of Cryptographic Hardware and Embedded Systems - CHES 1999*, 1999, pp. 158–172.

[6] P Kocher, J Jaffe, and B Jun, "Differential power analysis: leaking secrets," in *Proceedings of CRYPTO'99, Springer, Lecture notes in computer science*, 1999, pp. 388–397.

[7] T S Messerges, E A Dabbish, and R H Sloan, "Power analysis attacks of modular exponentiation in smart cards," in *Proceedings of Cryptographic Hardware and Embedded Systems - CHES 1999*, 1999, pp. 144–157.

[8] W Schindler, "A timing attack against RSA with Chinese remainder theorem," in *Proceedings of Cryptographic Hardware and Embedded Systems - CHES 2000*, 2000, pp. 109–124.

[9] Daniel Genkin, Itamar Pipman, and Eran Tromer, "Get your hands off my laptop: Physical side-channel key-extraction attacks on pcs," in *Cryptographic Hardware and Embedded Systems CHES 2014*, Lejla Batina and Matthew Robshaw, Eds., vol. 8731 of *Lecture Notes in Computer Science*, pp. 242–260. Springer Berlin Heidelberg, 2014.

[10] Michael Hutter and Jrn-Marc Schmidt, "The temperature side channel and heating fault attacks," in *Smart Card Research and Advanced Applications*, Aurlien Francillon and Pankaj Rohatgi, Eds., vol. 8419 of *Lecture Notes in Computer Science*, pp. 219–235. Springer International Publishing, 2014.

[11] J. Brouchier, T. Kean, C. Marsh, and D. Naccache, "Temperature attacks," *Security Privacy, IEEE*, vol. 7, no. 2, pp. 79–82, March 2009.

[12] E Bangerter, D Gullasch, and S Krenn, "Cache games - bringing access-based cache attacks on AES to practice," in *Proceedings of IEEE Symposium on Security and Privacy*, 2011.

[13] Yukiyasu Tsunoo, Etsuko Tsujihara, Kazuhiko Minematsu, and Hiroshi Miyauchi, "Cryptanalysis of block ciphers implemented on computers with cache," in *Proceedings of the International Symposium on Information Theory and its Applications*, 2002, pp. 803–806.

[14] Zhenghong Wang and Ruby B Lee, "New cache designs for thwarting software cache-based side channel attacks," in *ISCA '07: Proceedings of the 34th annual international symposium on Computer architecture*. 2007, pp. 494–505, ACM.

[15] A. Zajic and M. Prvulovic, "Experimental demonstration of electromagnetic information leakage from modern processor-memory systems," *Electromagnetic Compatibility, IEEE Transactions on*, vol. 56, no. 4, pp. 885–893, Aug 2014.

[16] Milos Prvulovic and Alenka Zajic, "Rf emanations from a laptop," 2012, `http://youtu.be/ldXHd3xJWw8`.

[17] J.G. Proakis, *Digital Communications*, McGraw-Hill Series in Electrical and Computer Engineering. Computer Engineering. McGraw-Hill, 2001.