

# A Computational Platform for Visible Light Communications

Felipe Barboza da Silva and Wallace Alves Martins

**Abstract**—This paper proposes a computational platform for simulations of visible light communication (VLC) systems. Special emphasis is placed on the modeling of light-emitting diode and its nonlinear aspects, optical channel, and photodiode. In addition, the equalization performance of VLC systems is assessed using adaptive filters, such as Volterra decision-feedback equalizer. Bit error rate results indicate that adaptive nonlinear techniques outperform linear-in-parameter approaches when it comes to coping with VLC nonlinearities.

**Keywords**—Visible light communication, light-emitting diode, adaptive filters, Volterra series, decision-feedback equalizer.

## I. INTRODUCTION

Visible light communication (VLC) is a technique that employs light to transmit data. When compared to traditional radiofrequency (RF) communications, the key components that enable VLC to work are a light-emitting diode (LED), responsible for transforming electrical signals into light signals at the transmitter, and a photodiode, responsible for converting this optical signal into a corresponding current level at the receiver end. VLC provides an unregulated spectrum from 400 THz to 800 THz, while RF-based communications occupy a band from 3 kHz to 300 GHz, which means VLC overall frequency band is 10,000 higher than RF's. This is one of the reasons why VLC is considered to be an alternative solution that addresses the RF spectrum scarcity by relieving this crowded frequency band. Moreover, VLC has some other benefits as compared to RF, such as the aforementioned non-regulated spectrum, non interference with most electromagnetic waves used in applications, which allows its use in hospitals and airplanes, no health regulations to restrict the transmitted power, low deployment cost, just to mention a few.

Precursors of modern VLC systems date back to 1880s with the photophone [1], developed by Alexander Graham Bell. More recently, VLC technology flourished in the late 1990s, when LED traffic lights were employed to broadcast audio messages to drivers [2]. In the early 2000s, a Japanese research group connected VLC to power line communication (PLC), while using white LEDs for both illumination and transmitting data purposes [3], [4]. Since the 2000s, lots of works in VLC popped up, including [5], which is one of the first papers to present actual experimental results. An open-source platform for developing practical VLC solutions is detailed in [6]. However, to the best of our knowledge, no work focusing on developing an open-source simulation environment for this application has been presented.

This work proposes a practical computational platform for VLC, which can be used, for instance, to analyze the bit

error rate (BER) for an on-off keying (OOK) under a given bit rate and signal-to-noise ratio (SNR), or to compare the performance of two different modulation schemes, among other possible uses. The platform uses some models from the literature for the main components (Section II) of a VLC system, namely, LED, optical channel, and photodiode. First, a model for the LED frequency response is employed, following the proposal in [7]. Then, a model that relates the LED nonlinear  $I$ - $V$  curve is used [8], followed by a modeling of the inherent nonlinear effect of the electrical-to-optical conversion, which is based on a study conducted in [9]. The proposed simulator also considers an optical channel and noise model based on [10]. A simple scheme for the receiving photodiode is considered, treating it as a simple gain. It is worth mentioning this work employs the parameters of a specific commercial LED, such as maximum electrical power,  $I$ - $V$  curve, but the user can change them easily (Section III).

Another contribution of this paper is the evaluation of some adaptive equalization techniques in the VLC context using the proposed simulator. The performance of linear-in-parameter and Volterra-based nonlinear structures are assessed for both forward and decision-feedback equalizers (Section IV).

## II. VLC SYSTEM OVERVIEW

The work principle of a VLC system is quite simple. The data to be transmitted is mapped into an electrical signal, which is converted to light using an LED. This light signal propagates through the air (free space) until it reaches the receiver where it is transformed over again into an electrical signal using a receiver sensor. This process resembles what occurs in radiofrequency (RF) communications, except for the electrical conversion to light at the transmitter and the inverse process at the receiver. Nevertheless, in the RF framework the amplitude of the received electrical signal is usually a linear function of the electrical field, instead of nonlinear as in the VLC case [10]. VLC schemes employ intensity modulation (IM), where the data to be transmitted is modulated varying the luminous intensity of the emitted light [10]. At the receiver, the fluctuations in the light luminosity are converted into a proportional current level by the photodiode. This process is the so-called direct detection (DD) [10].

In order to clarify the entire process of sending and receiving data through a VLC system, consider the following mathematical modeling: let a baseband discrete-time signal  $s[n]$  represent a data signal mapped from a given modulation scheme (OOK, PAM, PPM, etc). Before passing through the LED,  $s[n]$  must be converted into either current or voltage, depending on the LED driving circuit. This paper considers a voltage-based driving circuit, so that the LED input signal is

$$V_{in}[n] = V_{DC} + \alpha s[n], \quad (1)$$

F. B. da Silva and W. A. Martins are with the Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil (emails: felipe.silva@smt.ufrj.br, wallace.martins@smt.ufrj.br). The authors are grateful for the financial support provided by CNPq, CAPES, and FAPERJ, Brazilian research councils.

where  $V_{DC} \in \mathbb{R}_+$  is a DC voltage bias added to ensure  $V_{in}[n]$  is nonnegative, and  $\alpha \in \mathbb{R}_+$  is a voltage constant, which is chosen to keep the LED input signal inside a predefined operational range. From (1), one can see that  $V_{in}[n]$  is a signal comprised of a DC part,  $V_{DC}$ , and an AC part,  $\alpha s[n]$ , where  $V_{max}$  will denote the maximum amplitude of  $V_{in}[n]$ . The so-called modulation index (MI) [11] quantifies how much the amplitude of the data signal  $V_{in}[n]$  varies around  $V_{DC}$ , i.e.

$$MI = \frac{\Delta V}{V_{DC}} = \frac{V_{max} - V_{DC}}{V_{DC}}. \quad (2)$$

If MI and  $V_{DC}$  values are predefined parameters, then by using (1) and (2) one can write an expression for  $\alpha$  as

$$\alpha = \frac{MI \cdot V_{max}}{(MI + 1) \max\{s[n]\}}. \quad (3)$$

Once the LED input signal model was presented, the computational process to convert  $V_{in}[n]$  into a corresponding current level is modeled by the  $I$ - $V$  LED curve, as will be better explained in Section III. The product of  $V_{in}[n]$  and its corresponding current is then translated into a certain level of luminous intensity signal,  $\mathcal{I}[n]$ , according to the LED luminous efficacy. After that,  $\mathcal{I}[n]$  is convolved with a linear time-invariant filter  $h[n]$ , which emulates the light signal propagation through the air (optical channel [10]), and then is converted to a corresponding current level  $r[n]$  corrupted by a zero-mean white Gaussian noise  $g[n]$  at the receiver [12]. The resulting received signal is described by

$$r[n] = R \{A \cdot (\mathcal{I} * h)[n]\} + g[n], \quad (4)$$

where  $*$  represents a linear convolution operation,  $A \in \mathbb{R}_+$  is the photodiode (PD) detection area, and  $R\{\cdot\}$  denotes the PD responsivity [A/W]. A simplified block diagram of a VLC system is shown in Fig. 1. Next section describes some computational models for LED, optical channel, and receiver that will be incorporated in the computational platform.

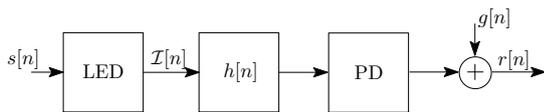


Fig. 1. Simplified VLC model.

### III. VLC SIMULATOR MODELING

In this section, some intrinsic effects induced by the VLC components, such as LED frequency response in the case of the transmitter, DC gain in the case of the channel, and wavelength responsivity of the photodiode, are modeled.

#### A. LED Model

The LED is usually the component that generates the most severe distortion effects on a VLC system. For instance, the LED very limited bandwidth, the nonlinear relation between the voltage applied to the LED and the corresponding output current, and the nonlinear conversion of instantaneous electrical power to instantaneous luminous intensity are issues that will be modeled in this paper. To do that, a low cost white LED, whose datasheet can be found in [13], is considered.

1) *Frequency-Response Model*: Low cost white LEDs feature low-pass frequency response, where the 3-dB bandwidth is usually around 2 MHz [14]. The frequency-response model adopted in this paper is the one proposed in [7], where the authors performed an experiment to measure the frequency response  $F(\omega)$  of a commercial white LED, approximated as:

$$F(\omega) = \begin{cases} e^{-|\omega|/\omega_1}, & \text{if } |\omega| < \omega_c \\ e^{-|\omega_c|/\omega_1} e^{|\omega_c|/\omega_2} e^{-|\omega|/\omega_2}, & \text{otherwise,} \end{cases} \quad (5)$$

where  $\omega_1 = 2\pi \cdot 3.26$ ,  $\omega_2 = 2\pi \cdot 10.86$ , and  $\omega_c = 2\pi$  Mrad/s. In order to employ this frequency response model in a digital environment,  $F(\omega)$  should be truncated and sampled at a rate  $\omega_s$ , generating the discrete-time model  $F_s(e^{j\omega})$ . This work assumes that  $s[n]$  denotes a pre-filtered data signal, as in [15].

It is worth mentioning that, even though the LED used in [7] is different from the one employed in this paper, the frequency response modeled by (5) will be used here; nonetheless, the proposed computational platform encapsulates this part so that the user can easily change it.

2) *I-V Curve Model*: After defining the LED input signal and its frequency response, it is necessary to model the behavior of the LED current as a function of the input voltage [8]:

$$I_{LED}[n] = \begin{cases} I_s (e^{V_{in}[n]/n_{LED} V_T} - 1), & \text{if } V_{in}[n] \geq 0 \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

where  $I_s \in \mathbb{R}_+$  represents the saturation current,  $n_{LED} \in \mathbb{R}_+$  is the LED ideality factor, and  $V_T$  denotes the thermal voltage. The LED current grows exponentially with voltage; however, there is a region where current varies approximately linearly with voltage, thus suggesting that the operational point ( $V_{DC}$ ) should be set within this region.

The resulting instantaneous electrical power is

$$P_{electrical}[n] = V_{in}[n] \cdot I[n], \quad (7)$$

which now must be converted into optical power. As will be shown, this conversion may be nonlinear, depending on the level of the instantaneous electrical power.

3) *Electrical-to-Optical Conversion Model*: In the linear region of the  $I$ - $V$  curve, the luminous intensity is proportional to the electrical power, i.e.,  $\mathcal{I}[n] = P_{electrical}[n] \cdot \eta$ , where  $\eta \in \mathbb{R}_+$  is an electrical-to-optical conversion factor. Nevertheless, there is a certain level of electrical power where the luminous intensity saturates, i.e., part of the electrical power is dissipated as heat. This effect can be modeled as [9]:

$$\mathcal{I}[n] = \frac{P_{electrical}[n] \cdot \eta}{\left(1 + \left(\frac{P_{electrical}[n] \cdot \eta}{\mathcal{I}_{max}[n]}\right)^{2k}\right)^{1/2k}}, \quad (8)$$

where  $k \in \mathbb{R}_+$  is a knee factor that adjusts the smoothness of the luminous intensity saturation, and  $\mathcal{I}_{max} \in \mathbb{R}_+$  is the maximum luminous intensity emitted by the LED. Fig. 2 illustrates the conversion using (8) for different values of  $k$ .

Note that the nonlinear effect imposed by the electrical-to-optical conversion is directly connected to MI in (2). Indeed, when MI increases, the maximum allowed value for the LED input signal also increases, leading to a higher degree of nonlinearity imposed by the  $I$ - $V$  curve, and possibly, causing saturation in the LED's electrical-to-optical conversion.

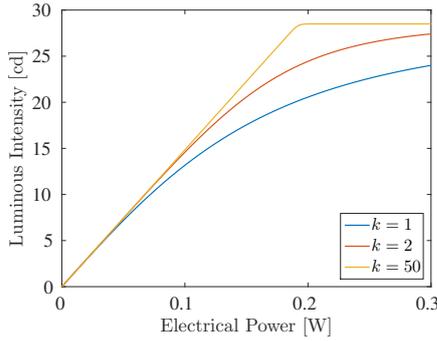


Fig. 2. Nonlinear behavior of luminous intensity for different  $k$ 's.

After modeling the main effects imposed by an LED when it is working as a VLC transmitter, the next step is to consider an optical channel model from the literature and incorporate it to the computational platform developed in this paper.

### B. Optical Channel Model

In [16] the authors proposed a model for wireless infrared channel that is also suitable for VLC systems:

$$h(t) = H(0) \frac{6a^6}{(t+a)^7} u(t), \quad (9)$$

where  $H(0)$  denotes the DC gain,  $a = 12\sqrt{11/13}D$ , with  $D$  denoting the root mean square (RMS) delay spread of the channel, and  $u(t)$  denotes the unit step function. In an optical channel,  $D$  is usually in the order of nanoseconds. Therefore, if a VLC system is working in a bit rate remarkably lower than a few Gbps, (9) can be approximated by a Dirac impulse, giving rise to channel without intersymbol interference (ISI). As this work considers bit rates in the order of Mbps,  $h[n] \approx \frac{2\pi}{\omega_s} H(0) \delta[n]$  in the discrete-time domain.<sup>1</sup>

The model employed in this work for  $H(0)$  considers that the angular distribution of radiant output power of the LED can be modeled with a generalized Lambert law [17]. In addition, it is assumed that the optical channel features a line-of-sight (LOS), i.e., the direct path between transmitter and receiver is not obstructed. Based on those assumptions, one has

$$H(0)_{\text{LOS}} \approx \begin{cases} \frac{A}{d^2} \frac{m+1}{2\pi} \cos^m(\phi) \cos(\theta), & \text{if } |\frac{\theta}{\text{FOV}}| \leq 1 \\ 0, & \text{otherwise,} \end{cases} \quad (10)$$

where FOV denotes the photodiode field of view, and  $m = -\ln 2 / \ln(\cos \Phi_{1/2})$ , with  $\Phi_{1/2}$  being the LED's half-power angle. As for  $\phi$  and  $\theta$ , consider Fig. 3, in which the vectors  $\mathbf{r}_T, \mathbf{r}_R \in \mathbb{R}^{3 \times 1}$  respectively denote the LED and photodiode positions, and  $\hat{\mathbf{n}}_T, \hat{\mathbf{n}}_R \in \mathbb{R}^{3 \times 1}$  represent the normal vectors to their surfaces. Then,  $\phi$  is the angle between  $\hat{\mathbf{n}}_T$  and  $(\mathbf{r}_R - \mathbf{r}_T)$ , while  $\theta$  is the angle between  $\hat{\mathbf{n}}_R$  and  $(\mathbf{r}_R - \mathbf{r}_T)$ .

It is important to highlight that, as in the RF case, the optical channel is also corrupted by an additive noise, which is modeled at the receiver as zero mean white Gaussian [12].

### C. Receiver Model

The receiver model employed in this paper is represented only by a gain  $R$ , which is the photodiode conversion factor.

<sup>1</sup> $\delta[n]$  denotes a discrete unit impulse.

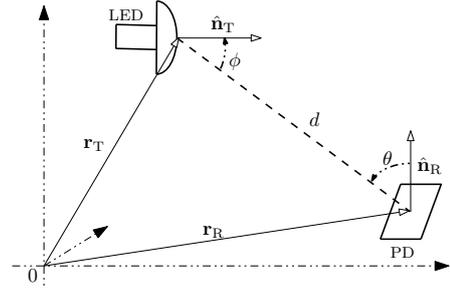


Fig. 3. Geometry of the LED and photodiode.

Indeed, it was considered that the photodiode exhibits a flat response over the entire visible spectrum. This was assumed for the sake of simplicity and does not affect the generality of this work. It is worth mentioning that the bandwidth of a photodiode is often much greater than the LED's.

## IV. TRANSCEIVER EQUALIZATION

This section describes some equalization techniques, focusing on adaptive filtering methods to equalize the received signal provided by the VLC simulator.

VLC systems suffer from inherent nonlinear effects, mostly imposed by the LED. Besides, LEDs also introduce some ISI due to their memory. Moreover, the VLC channel may be shadowed, i.e., the transmitted luminous intensity may be severely attenuated by an obstacle. Therefore, adaptive nonlinear techniques are more capable to cope with these issues and should yield better results than linear equalizers with fixed taps.

### A. Adaptive Linear-in-Parameter Forward Equalization

In an adaptive filtering context, the equalization technique consists of minimizing some function of the error between the training sequence and the equalizer outputs. The filter's input signal vector and weights are given as

$$\mathbf{x}[n] = [x[n] \quad x[n-1] \quad \cdots \quad x[n-N]]^T, \quad (11)$$

$$\mathbf{w}[n] = [w[n] \quad w[n-1] \quad \cdots \quad w[n-N]]^T, \quad (12)$$

where  $N+1$  is the adaptive filter length. It is important to emphasize that the samples of input signal vector  $\mathbf{x}[n]$  are zero-mean normalized versions of the signal  $r[n]$  (see Fig. 1). This pre-processing must be performed to adjust  $r[n]$  to the receiver dynamic range of operation.

Considering the error equation as  $e[n] = d[n] - \mathbf{w}[n]^T \mathbf{x}[n]$ , where  $d[n]$  represents the desired signal (training sequence), the adaptive filter coefficients may be updated according to the normalized least-mean-square (NLMS) algorithm [18]:

$$\mathbf{w}[n+1] = \mathbf{w}[n] + \frac{\mu}{\mathbf{x}[n]^T \mathbf{x}[n] + \delta} e[n] \mathbf{x}[n], \quad (13)$$

where  $\mu \in [0, 1]$  is the step-size parameter and  $\delta \in \mathbb{R}_+$  is a regularization factor.

### B. Adaptive Volterra Forward Equalization

The Volterra filter is a versatile type of nonlinear structure. Practical systems that perform nonlinear operations on the inputs to produce outputs can be modeled via Taylor series. The truncation of a Taylor series expansion can approximate

the nonlinear system behavior as a function of weighted combinations of products of the input samples [18], [19], describing a large class of nonlinear systems [19].

In the adaptive Volterra filtering scheme, the output should follow a Volterra series [18], generating the expanded vectors  $\bar{\mathbf{x}}[n]$  and  $\bar{\mathbf{w}}[n]$ , increasing the filter length to  $(N + 1) + (N + 1)^2$ . These vectors are represented here using the second-order Volterra series, as follows:

$$\bar{\mathbf{x}}[n] = \begin{bmatrix} x[n] \\ \vdots \\ x[n-N] \\ x^2[n] \\ x[n]x[n-1] \\ \vdots \\ x[n]x[n-N] \\ \vdots \\ x[n-N]x[n-N+1] \\ x^2[n-N] \end{bmatrix}, \quad \bar{\mathbf{w}}[n] = \begin{bmatrix} w_0[n] \\ \vdots \\ w_N[n] \\ w_{0,0}[n] \\ w_{0,1}[n] \\ \vdots \\ w_{0,N}[n] \\ \vdots \\ w_{N,N-1}[n] \\ w_{N,N}[n] \end{bmatrix}. \quad (14)$$

It is worth highlighting that the Volterra kernel has redundant terms, e.g.,  $x[n]x[n-1] = x[n-1]x[n]$ . These redundancies will be disregarded on both  $\bar{\mathbf{x}}[n]$  and  $\bar{\mathbf{w}}[n]$  vectors so as to reduce the filter length to  $(N + 1) + \left\lceil \frac{(N+1)^2 + (N+1)}{2} \right\rceil$ , thus decreasing the computational burden. In addition, the authors in [11] show that a second-order Volterra kernel is able to cope with the VLC nonlinearities, justifying why this paper considers only linear and quadratic terms of the Volterra series.

The process of training the adaptive Volterra filter is the same as in the linear-in-parameter case, except for some mild modifications. In (13), the vectors  $\mathbf{x}[n]$  and  $\mathbf{w}[n]$  should be replaced by  $\bar{\mathbf{x}}[n]$  and  $\bar{\mathbf{w}}[n]$ , respectively.

### C. Adaptive Decision-Feedback Equalization

The basic idea behind DFE is to use past symbols to improve the detection performance of the present symbol [20]. DFE consists in a feedforward filter (FF), a feedback filter (FB), and a decision device that introduces a nonlinearity in the equalization process, as illustrated in Fig. 4. The FF and FB filters can be linear-in-parameter or nonlinear filters (e.g. Volterra). The input signal vector and the weights for the FF and the FB filters are

$$\mathbf{x}[n] = [x[n] \quad x[n-1] \quad \cdots \quad x[n-L_{\text{FF}}]]^T, \quad (15)$$

$$\mathbf{w}_{\text{FF}}[n] = [w_{\text{FF}}[n] \quad \cdots \quad w_{\text{FF}}[n-L_{\text{FF}}]]^T, \quad (16)$$

$$\hat{\mathbf{y}}[n] = [\hat{y}[n] \quad \hat{y}[n-1] \quad \cdots \quad \hat{y}[n-L_{\text{FB}}]]^T, \quad (17)$$

$$\mathbf{w}_{\text{FB}}[n] = [w_{\text{FB}}[n] \quad \cdots \quad w_{\text{FB}}[n-L_{\text{FB}}]]^T, \quad (18)$$

where  $L_{\text{FF}}$  and  $L_{\text{FB}}$  represent the length of the feedforward and feedback filters, respectively. Then, the DFE output is  $y[n] = \mathbf{w}_{\text{DFE}}[n]^T \mathbf{x}_{\text{DFE}}[n]$ , where  $\mathbf{w}_{\text{DFE}}[n] = [\mathbf{w}_{\text{FF}}[n]^T \quad \mathbf{w}_{\text{FB}}[n]^T]^T$ , and  $\mathbf{x}_{\text{DFE}}[n] = [\mathbf{x}[n]^T \quad \hat{\mathbf{y}}[n]^T]^T$ .

## V. PERFORMANCE EVALUATION

This section assesses the performance of the techniques described in Subsection IV-A (linear-in-parameter forward equalization), Subsection IV-B (Volterra forward equalization), and Subsection IV-C (linear-in-parameter DFE and Volterra DFE) using the data obtained from the proposed computational

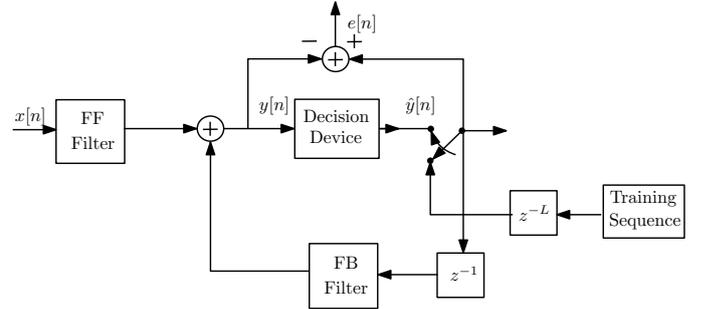


Fig. 4. Adaptive DFE.

platform. The methodology consists in: training the adaptive filters, describing the resulting mean squared error (MSE), and using filters obtained after convergence in the equalization process, presenting their respective bit error rates (BERs).

### A. Simulation Procedures

The simulations to train the adaptive filters used 1000 independent runs, step-size  $\mu = 0.8$ , an SNR of 30 dB, and a 4-PAM symbol constellation with a bandwidth of 1 MHz. The SNR computation disregards the DC component used to ensure the transmitted signals are nonnegative and to set the LED to its operational point. It was considered a distance between LED and photodiode of 10 cm, which were treated as perfectly aligned, leading to  $\phi = \theta = 0^\circ$ . The LED has half-power angle of  $\Phi_{1/2} = 15^\circ$ . The photodiode's responsivity, detection area, and FOV were set as  $R = 0.5$ ,  $A = 1 \text{ cm}^2$ , and  $\text{FOV} = 25^\circ$ , respectively. The knee-factor in (8) was  $k = 2$ , inducing a high degree of nonlinearity in the electrical-to-optical conversion. The DC bias in the simulations was 3.25 V, chosen so as to place the operational point around the middle of the linear part of the aforementioned LED  $I$ - $V$  curve.

The parameter  $N = 11$  led to a total length of 12 for the linear-in-parameter filters, and  $\frac{12^2 + 12}{2} + 12 = 90$  for Volterra filters. In the case of DFE and Volterra DFE, FF and FB filter lengths were  $L_{\text{FF}} = 12$  and  $L_{\text{FB}} = 12$ . Due to the Volterra kernel, the FF section length of DFE Volterra is also increased to 90. Those parameters were chosen following [11]. For BER simulations, 40000 symbols were transmitted and 1000 Monte Carlo runs were performed.

### B. Simulation Results

Fig. 5 shows the learning curve of the equalization techniques presented in this paper. One can note in Fig. 5b and Fig. 5d that the nonlinear equalizers achieved the smallest MSEs, especially the Volterra DFE. Nonetheless, the speed of convergence of Volterra-based filters is much slower than the linear-in-parameter filters due to the difference of lengths. In addition, as the modulation index increases, a higher degree of nonlinearity is imposed on the system, either by the larger signal excursion in  $I$ - $V$  curve, or by the LED electrical-to-optical conversion, thus increasing MSE.

Fig. 6 depicts BER for different electrical SNR levels. Once again, the DFE Volterra is the one that achieved the best results, as shown in Fig 6d, obtaining a BER of  $2 \times 10^{-5}$  for SNR = 30 dB and MI = 0.1. Volterra equalizer achieved results close to Volterra DFE's in Fig 6b.

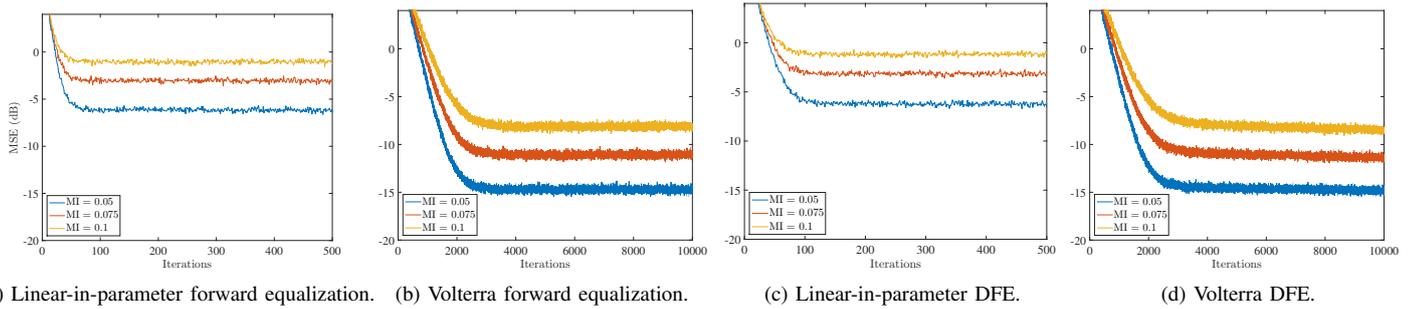


Fig. 5. MSE learning curves for different MI's.

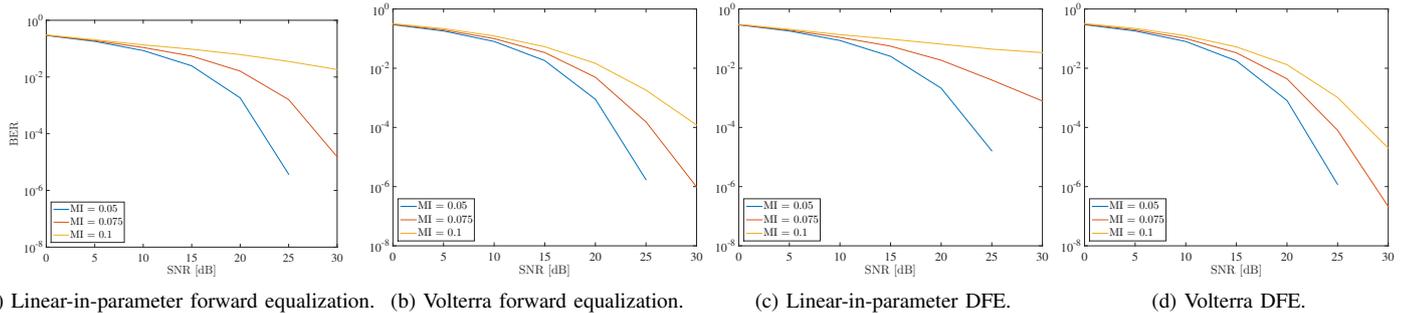


Fig. 6. BER vs. SNR using the adaptive filter after convergence for different MI's.

## VI. CONCLUSIONS

This paper proposed a practical computational platform for VLC. It described the models for LED, such as frequency-response and electrical-to-optical conversion, and some key aspects of the optical channel and photodiode models. Adaptive techniques were evaluated under the task of equalizing the data signal provided by the simulator, indicating that Volterra-based equalization schemes are better than linear-in-parameter techniques, particularly when the nonlinearity level imposed by the VLC system is high. As future works, the computational platform will be tailored to employ models for non-line-of-sight optical channels, as well as a larger number of LEDs and photodiodes, allowing the simulation of MIMO systems. The codes of the proposed computational platform can be found in [21].

## REFERENCES

- [1] A. G. Bell, W. G. Adams, Tyndall, and W. H. Preece, "Discussion on "the photophone and the conversion of radiant energy into sound";" *Telegraph Engineers, Journal of the Society of*, vol. 9, no. 34, pp. 375–383, Jan 1880.
- [2] G. Pang, T. Kwan, C.-H. Chan, and H. Liu, "Led traffic light as a communications device," in *IEEE/IEE/JSAI International Conference on Intelligent Transportation Systems*, Tokyo, Japan, Oct 1999, pp. 788–793.
- [3] Y. Tanaka, T. Komine, S. Haruyama, and M. Nakagawa, "Indoor visible communication utilizing plural white LEDs as lighting," in *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 2, San Diego, CA, USA, Sep 2001, pp. 81–85.
- [4] T. Komine and M. Nakagawa, "Integrated system of white LED visible-light communication and power-line communication," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 1, pp. 71–79, Feb 2003.
- [5] M. Z. Afgani, H. Haas, H. Elgala, and D. Knipp, "Visible light communication using OFDM," in *2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, Barcelona, Spain, Mar 2006, pp. 129–134.
- [6] Q. Wang, D. Giustiniano, and D. Puccinelli, "OpenVLC: software-defined open architecture for embedded visible light networks," in *1st ACM Workshop on Visible Light Communication Systems*, Maui, HI, USA, Sep 2014, pp. 15–20.
- [7] M. Biagi, T. Borogovac, and T. D. C. Little, "Adaptive receiver for indoor visible light communications," *Journal of Lightwave Technology*, vol. 31, no. 23, pp. 3676–3686, Dec 2013.
- [8] A. S. Sedra and K. C. Smith, *Microelectronic Circuits*, 1st ed., New York, NY, USA, 1998.
- [9] H. Elgala, R. Mesleh, and H. Haas, "An LED model for intensity-modulated optical communication systems," *IEEE Photonics Technology Letters*, vol. 22, no. 11, pp. 835–837, Jun 2010.
- [10] J. M. Kahn and J. R. Barry, "Wireless infrared communications," *Proceedings of the IEEE*, vol. 85, no. 2, pp. 265–298, Feb 1997.
- [11] G. Stepniak, J. Siuzdak, and P. Zwierko, "Compensation of a VLC phosphorescent white LED nonlinearity by means of Volterra DFE," *IEEE Photonics Technology Letters*, vol. 25, no. 16, pp. 1597–1600, Aug 2013.
- [12] E. A. Lee and D. G. Messerschmitt, *Digital Communication*. AZ Dordrecht, Netherlands: Springer, 2012.
- [13] [http://www1.futureelectronics.com/doc/EVERLIGHT%C2%A0/334-15\\_T1C1-4WYA.pdf](http://www1.futureelectronics.com/doc/EVERLIGHT%C2%A0/334-15_T1C1-4WYA.pdf).
- [14] J. Grubor, S. Randel, K. D. Langer, and J. W. Walewski, "Broadband information broadcasting using LED-based interior lighting," *Journal of Lightwave Technology*, vol. 26, no. 24, pp. 3883–3892, Dec 2008.
- [15] H. Qian, S. J. Yao, S. Z. Cai, and T. Zhou, "Adaptive postdistortion for nonlinear LEDs in visible light communications," *IEEE Photonics Journal*, vol. 6, no. 4, pp. 1–8, Aug 2014.
- [16] J. B. Carruthers and J. M. Kahn, "Modeling of nondirected wireless infrared channels," *IEEE Transactions on Communications*, vol. 45, no. 10, pp. 1260–1268, Oct 1997.
- [17] F. R. Gfeller and U. Bapst, "Wireless in-house data communication via diffuse infrared radiation," *Proceedings of the IEEE*, vol. 67, no. 11, pp. 1474–1486, Nov 1979.
- [18] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, 4th ed. New York, NY, USA: Springer, 2012.
- [19] M. Zeller and W. Kellermann, "Fast and robust adaptation of dft-domain volterra filters in diagonal coordinates using iterated coefficient updates," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1589–1604, Mar 2010.
- [20] P. P. Vaidyanathan, S. Phoong, and Y. Lin, *Signal Processing and Optimization for Transceiver Systems*. New York, NY, USA: Cambridge University Press, 2010.
- [21] F. B. da Silva, Apr 2017, <https://github.com/fbs2112/Visible-Light-Communication.git>.