

# Phase Noise Impairment and Environment-Adaptable Fast (EAF) Optimization for Programming of Reconfigurable Radio Frequency (RF) Receivers

Minhee Jun\*, Rohit Negi\*, Shihui Yin<sup>†</sup>, Fa Wang<sup>†</sup>, Megha Sunny<sup>†</sup>, Tamal Mukherjee<sup>†</sup>, Xin Li<sup>†</sup>,  
Department of Electrical and Computer Engineering

Carnegie Mellon University, Pittsburgh, Pennsylvania 15213–3890

Email: \*mjun@ece.cmu.edu, \*negi@ece.cmu.edu,

<sup>†</sup>syin@andrew.cmu.edu, <sup>†</sup>fwang1@andrew.cmu.edu, <sup>†</sup>msunny@andrew.cmu.edu, <sup>†</sup>tamal@ece.cmu.edu, <sup>†</sup>xinli@cmu.edu,

**Abstract**—In order to support a multi-standard platform, a reconfigurable RF front-end needs an optimal configuration that adapts to a dynamic communication condition. To find an optimal configuration efficiently, we previously proposed the Environment-Adaptable Fast (EAF) optimization in terms of the RF impairments of gain, nonlinearity and noise figure. However, this preliminary study did not include the important impairment of phase noise. In this paper, we extend the EAF optimization algorithm to phase noise impairment in a reconfigurable RF front-end. In this study, we will propose a novel statistical estimation tool for obtaining phase noise spectrum information with the Interpolated FIR (IFIR) model and the least mean squares (LMS) adaptive algorithm. We formulate the calculation of the Signal-to-Interference-and-Noise Ratio (SINR) which hastens the optimization process. Phase noise is included in the SINR calculation. We demonstrate the efficient performance of the EAF optimization method even with phase noise impairment. This study shows that while finding an optimal configuration, the EAF optimization significantly reduces simulation time compared to the other four conventional optimization methods.

## I. INTRODUCTION

The concept of reconfigurable RF front-end (also called reconfigurable RF-FPGA) was recently proposed by the U.S. Defense Advanced Research Projects Agency (DARPA) in order to support a multi-standard platform of software-defined radio (SDR) [1]. SDR with a reconfigurable RF front-end reconfigures RF components (such as RF amplifiers) in addition to baseband (BB) components, unlike a fixed wide-band RF front-end that reconfigures only BB components (Figure 1). While a reconfigurable RF front-end can potentially outperform a fixed wide-band RF front-end in challenging RF environments, it requires an efficient optimization method for finding an optimal configuration for each communication standard. Previously, the Environment-Adaptable Fast (EAF) optimization method was proposed and demonstrated as an efficient optimization method [2]. The flow chart of this optimization method is given in Figure 2. First, the EAF optimization process finds the configuration of maximizing the calculated Signal-to-Interference-and-Noise Ratio (SINR). Second, the EAF optimization utilizes the calculated SINR of all configurations. When the calculated SINR of a configuration does not meet the SINR threshold, the configuration is discarded during the optimization process without simulation.

While only nonlinearity and thermal noise impairments were considered in the preliminary study of [2], this study

will extend its focus to **phase noise impairment**. Phase noise is one of the most important RF impairments with respect to baseband standard requirements as shown in Table I generated from a study published by Brandolini [3]. Phase noise is generated by oscillators. Although ideal oscillators are supposed to generate pure sinusoidal signals, real oscillators suffer from time-varying phase fluctuation of sinusoids.

In order to deal with the phase noise problem, previous studies have attempted to compensate signals impaired due to phase noise using estimation methods and signal processing. For example, Zou’s study in [4] used training signals for estimating signal samples impaired by phase noise and compensated the same impairment in data signals. This previous approach was typical due to widespread use of fixed RF front-end architecture.

The previous approaches for phase noise in fixed RF front-ends must be revised for reconfigurable RF front-ends. We will replace a local oscillator with another that has lower phase noise (assuming that phase noise is a dominant impairment). However, reconfiguring oscillators is not as simple as just choosing the oscillator of the lowest phase noise spectrum. There are factors other than phase noise to be considered, such as the power consumption of oscillators. Generally, the lower the phase noise spectrum, the higher the power consumption [5]. Thus, in order to reconfigure oscillators effectively, it is essential to investigate phase noise characteristics for the reconfigurable RF front-end.

In our study of phase noise for reconfigurable RF front-end, the contributions are as follows: (1) a statistical model and estimation method of phase noise spectrum using the Interpolated FIR (IFIR) model and the modified least mean squares (LMS) adaptive algorithm, (2) newly-included phase noise impairment for the SINR calculation, and (3) verification of the EAF optimization performance after including phase noise impairment.

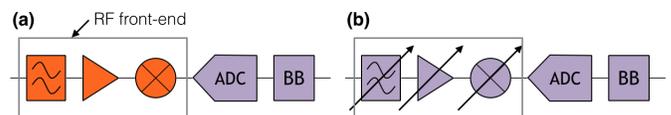


Fig. 1. Architectures of software-defined radio with (a) a fixed wide-band RF front-end and (b) a reconfigurable RF front-end.

TABLE I. BASEBAND STANDARD REQUIREMENT IN [3]. (PHASE NOISE IS SPECIFIED AT 1MHz FREQUENCY OFFSET.)

Standard	E-GSM (2G)	UMTS (3G)	802.11a/g	802.11b
f (GHz)	0.88-0.915	1.85-1.91	5.15-5.725	2.4-2.4835
Noise Figure	9 dB	6 dB	7.5 dB	14.8 dB
Phase Noise	-141 dBc/Hz	-150 dBc/Hz	-102 dBc/Hz	-101 dBc/Hz
IIP3	-18 dBm	-18 dBm	-16 dBm	-16 dBm
IIP2	+49 dBm	+46 dBm	-	-

The remainder of this paper is organized as follows: In Section II, we introduce phase noise impairment and its effect. In Section III, we model and estimate the phase noise spectrum. In Section IV, we explain the SINR calculation based on RF impairments including phase noise. We show numerical results in Section V when the updated SINR calculation is applied for the EAF optimization in [2]. We conclude the paper in Section VI.

## II. PHASE NOISE IMPAIRMENTS

In this section, we show how strong interference can interact with phase noise. We investigate the phase noise effect with and without interference using the IEEE 802.11a and the IEEE 802.11b transceivers as an example. Simulations are performed using Matlab Simulink with the SimRF block set. In Table I, we note that the IEEE 802.11a and IEEE 802.11b standards require -102 dBc/Hz and -101 dBc/Hz phase noise, respectively, at 1 MHz frequency offset.

First, we assumed that there is no interference and only the signal of interest is presented. The signal of interest was -65 dBm for the IEEE 802.11a and -76 dBm for the IEEE 802.11b as given in Figure 3. The signal-only case was plotted as the green curves in Figure 4 and 5. Because of other impairments such as nonlinearity, the simulated SINR curve converges to a constant in the low phase noise range. Otherwise, SINR decreases by 3 dB as phase noise increases by 3 dB in calculated SINR, which is derived from Eq. (12) in Section IV, and also in simulated SINR.

Second, we assumed that interference exists in addition to the signal of interest. In our scenario, we had the interferer of -30 dBm at 50 MHz frequency offset, the worst possible case

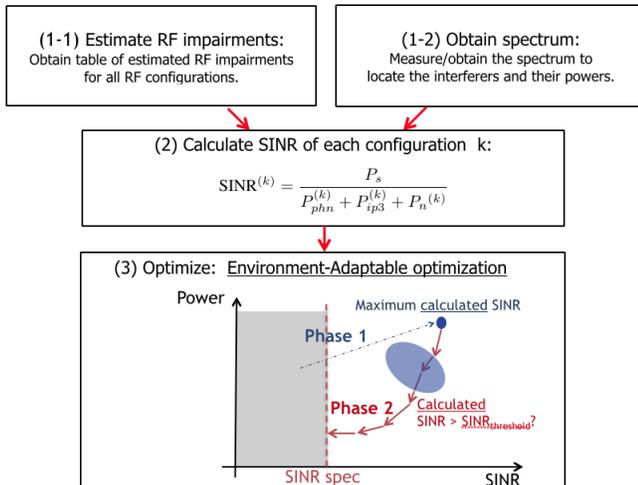


Fig. 2. Flow Chart of the EAF Optimization Algorithm.

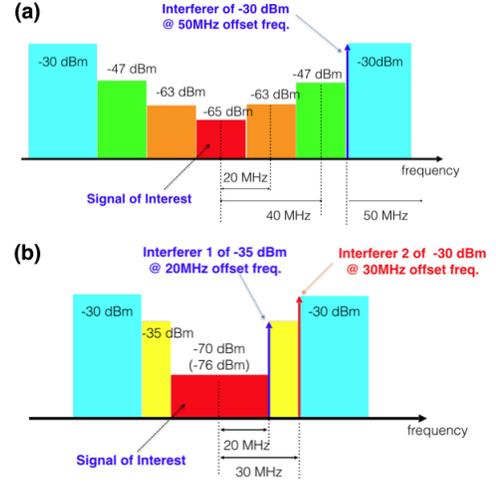


Fig. 3. Blocking Mask: (a) the IEEE 802.11a (b) the IEEE 802.11b [3].

specified by the blocking mask in the IEEE 802.11a as shown in Figure 3 (a). For the IEEE 802.11b, we considered the two possible scenarios specified by the standard: the interferer of -35 dBm at 20 MHz frequency offset and the interferer of -30 dBm at 30 MHz frequency offset.

For IEEE 802.11a, Figure 4 shows that the required phase noise is -108 dBc/Hz at 1MHz frequency offset, which is more stringent than -102 dBc/Hz from Brandolini's study [3] as

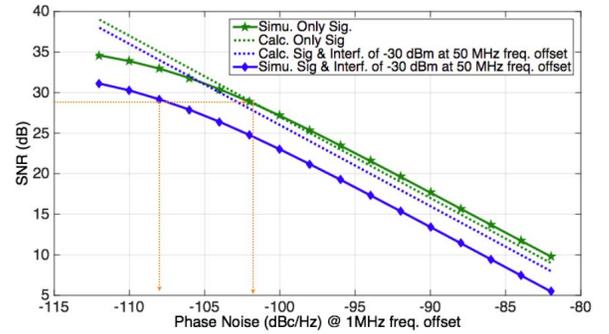


Fig. 4. SINR (dB) vs. Phase Noise (dBc/Hz at 1MHz frequency offset) for the IEEE 802.11a with the SINR requirement of 29 dB.

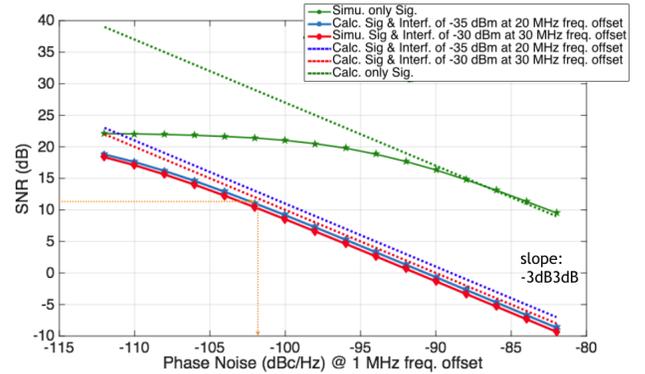


Fig. 5. SINR (dB) vs. Phase Noise (dBc/Hz at 1MHz frequency offset) for the IEEE 802.11b with the SINR requirement of 14.5 dB.

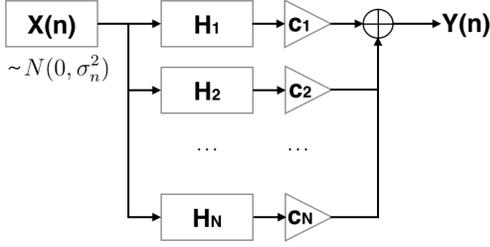


Fig. 6. The IFIR model for phase noise spectrum estimation: the basis filters  $H_1, H_2, \dots, H_N$  are fixed, and the gains  $c_1, c_2, \dots, c_N$  are adjustable.  $X(n)$  is AWGN of mean of 0 and variance of  $\sigma_n^2$ .

specified in Table I. This phenomenon is explained as follows: the impairing signal caused by phase noise and the signal of interest (in Eq. 12) and the impairing signal by phase noise and interference (in Eq. 13) have almost the same value. The summed signal is about 3 dB higher than either one of the impairing signals. Then, the SINR of the sum of impairing signal is about 3 dB lower than that found in Brandolini's study. Thus, SINR was impacted by the signal of interest and also by interference (in Section IV). This shows the need to derive a more accurate formula for SINR when there is phase noise and interference.

For IEEE 802.11b, our simulation results in Figure 5 show that phase noise requirement is about -102 dBc/Hz, which is consistent with -101 dBc/Hz in Table I. As expected, the SINR with interference is much lower than the SINR without interference because interference effect dominates phase noise.

### III. PHASE NOISE ESTIMATION

In order to calculate SINR in the EAF optimization, we need the phase noise information. Phase noise spectrum is described in terms of a designed estimator based on a stochastic signal model of phase noise.

There have been studies for modeling a stochastic process with the phase noise spectrum and generating stochastic signals caused by phase noise. The three main categories of phase noise spectrum modeling are given as (1) Autoregressive Moving Average (ARMA) based models, (2) fractional integration models, (3) wavelength based models of  $1/f$  noise [6]. In this paper, the fractional integration model - the most common approximation - was applied to model the phase noise process for reconfigurable RF front-ends.

The fractional integration model is a superposition of filters with one pole for converging to the phase noise spectrum. Instead of choosing the pole positions carefully, the Interpolated FIR (IFIR) approach fixes the pole positions for basis filters and carefully chooses gains of the filters [7]. The IFIR model in Figure 6 has the fixed basis filters  $H_1, H_2, \dots, H_N$  and the adjustable gains  $c_1, c_2, \dots, c_N$ . The discrete time sequence ( $X_n$ ) is AWGN of mean of 0 and variance of  $\sigma_n^2$ .

#### A. Interpolated FIR (IFIR) Model

The IFIR model efficiently builds lower order basis filters [8]. There are  $N$  (zero phase) basis filters, with frequency response  $H_1, H_2, \dots, H_N$ . We define the transition bandwidth of  $H_k$ ,  $(\omega_p^{(k)}, \omega_s^{(k)})$  such that  $\omega_s^{(k)} = m\omega_p^{(k)} = \omega_p^{(k+1)}$  where

$m$  is the rate constant that determines the passband of all basis filters. The highest frequency should be smaller than half of the sampling frequency,  $\omega_s^{(N)} < F_s/2$  where  $F_s$  is the sampling frequency. The transition bandwidth of  $H_k$  is given as follows,

$$(\omega_p^{(k)}, \omega_s^{(k)}) = m^{(k-1)} \cdot (\omega_p^{(1)}, \omega_s^{(1)}). \quad (1)$$

Then, the filters  $H_k$  are designed using Algorithm 1.

#### Algorithm 1 The Basis Filter Design Algorithm

- 1: **Generate a digital lowpass filter**  $G(e^{j\omega})$  **with transition bandwidth**  $(\omega_p^{(N)}, \omega_s^{(N)})$
- 2:  $H_N(e^{j\omega}) = G(e^{j\omega})$
- 3: **for**  $k = 1, 2, \dots, N - 1$  **do**
- 4:     **filter**  $G_k(e^{j\omega}) = G(e^{jm^{N-k}\omega})$
- 5:     **lowpass filter**  $I_k(e^{j\omega})$  **with stop frequency**  $\omega_s^{(N)}/m^{N-k}$
- 6:     **filter**  $H_k(e^{j\omega}) = G_k(e^{j\omega})I_k(e^{j\omega})$
- 7: **end for**
- 8: **Return**  $H_1(e^{j\omega}), H_2(e^{j\omega}), \dots, H_k(e^{j\omega}), \dots, H_N(e^{j\omega})$ .

The IFIR design algorithm efficiently reduces the order of designed basis filters. The Bode plots of the filters in Algorithm 1 are given in Figure 7. We first implemented the digital filter  $G_k(e^{j\omega})$  in line 1 using a Butterworth lowpass filter with a passband attenuation of 1 dB and a stopband attenuation of 20 dB. We set  $m = 2$  and  $N = 13$ . In line 4,  $G_k(e^{j\omega}) = G(e^{jm^{N-k}\omega})$  was obtained by upsampling the time domain signal  $g_k(t) = FT^{-1}(G_k(e^{j\omega}))$  with the upsampling factor  $m^{N-k}$  (the function  $FT^{-1}$  is the inverse fourier transform). Then, the lowpass filter  $I_k(e^{j\omega})$  eliminates the replica in the signal output of  $G_k(e^{j\omega})$ .

#### B. Statistical Signal Model and Estimation of Phase Noise

While the IFIR technique was previously implemented in a heuristic manner in [8], we establish a rigorous statistical signal model for utilizing the IFIR model. Then, we derive an estimator for the phase noise spectrum and apply the Least Mean Squares (LMS) adaptive algorithm to complete the estimator.

1) *Statistical Signal Model*: The signal subspace model is defined as follows [9],

$$Y = X \bullet H_C + N, \quad (2)$$

where  $Y, X, H_C, N$  are  $M \times 1$  vectors.  $Y$  is a vector of a received signal in the frequency domain,  $X$  is a vector of a

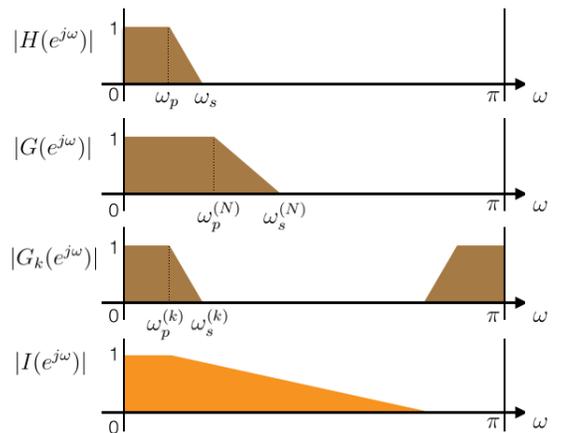


Fig. 7. Filters corresponding with the Algorithm 1 design.

AWGN in the frequency domain, and  $N$  is a vector of AWGN, measurement noise.  $H_C(f_k) = c_1 H_1(f_k) + c_2 H_2(f_k) + \dots + c_N H_N(f_k)$  where the vector  $H_i(f_k)$  is the impulse response of the  $i$ -th basis filter at the frequency  $f_k$  and the variable  $c_i$  is a gain of the  $i$ -th filter in the multirate filter bank. The  $\bullet$  operator is the element-wise product.

Then, the  $k$ -th element  $Y_k$  of the vector  $Y$  is,

$$Y_k = Y(f_k) = X(f_k) \cdot H_C(f_k) + N(f_k), \quad (3)$$

where  $Y(f_i)$  and  $Y(f_j)$  are independent if  $i \neq j$ , and  $Y(f_k) \sim N(0, \sigma_X^2 \cdot |H_C(f_k)|^2 + \sigma_n^2)$  for  $k = 1, 2, \dots, M$ .

**2) Problem Definition and Maximum Likelihood Estimator (MLE):** We need to obtain the gains  $c_1, c_2, \dots, c_N$  for the given model in Figure 6.

We define the vector  $C = [c_1, c_2, \dots, c_N]^T$ . Then, the log-likelihood function  $L(C, Y)$  of the vector  $Y$  is,

$$\begin{aligned} L(C, Y) &= \ln f_C(Y) \\ &= \ln \prod_{f_k} f_C(Y(f_k)) = \sum_{f_k} \ln f_C(Y(f_k)) \\ &= \frac{1}{2} \sum_{f_k} \left( -\ln |H_C(f_k)|^2 - \frac{|Y(f_k)|^2}{|H_C(f_k)|^2} \right) + \text{const.}, \end{aligned} \quad (4)$$

assuming  $\sigma_X^2 = 1$  and  $\sigma_n^2 = 0$ .

The maximum likelihood estimator (MLE)  $\hat{C}$  is obtained by maximizing the log-likelihood  $L(C, Y)$ .

$$\hat{C} = \max_C L(C, Y). \quad (5)$$

In order to prevent overfitting the vector  $C$ , we added the Lasso regularization  $\|C\|_1 = \sum_k |c_k|$  in (5).

$$\begin{aligned} \hat{C} &= \max_C L(C, Y) - \gamma \cdot \|C\|_1 \\ &= \max_C \frac{1}{2} \sum_{f_k} \left( -\ln |H_C(f_k)|^2 - \frac{|Y(f_k)|^2}{|H_C(f_k)|^2} \right) - \gamma \cdot \|C\|_1. \end{aligned} \quad (6)$$

Defining the cost function  $J(C) = -(L(C, Y) - \gamma \cdot \|C\|_1)$ , we can rewrite (6) in a way of minimizing  $J(C)$ ,

$$\begin{aligned} \hat{C} &= \min_C J(C) \\ &= \min_C \frac{1}{2} \sum_{f_k} \left( \ln |H_C(f_k)|^2 + \frac{|Y(f_k)|^2}{|H_C(f_k)|^2} \right) + \gamma \cdot \|C\|_1. \end{aligned} \quad (7)$$

**3) The Least Mean Squares (LMS) Adaptive Algorithm:**

Because of complications - dealing with the **complex** estimator  $\hat{C}$ , we applied the LMS adaptive algorithm.

The two main modifications from the primary LMS adaptive algorithm are initialization and termination: (1) The initialization condition in line 2 is modified from  $C^{(0)} = 0$  to  $C^{(0)} = (H^T H)^{-1} H^T Y_a$  where  $(Y_a)_k = |Y(f_k)|$ . This modification prevents the metric  $\Delta_i$  in line 13 from being  $\infty$  at the first iteration, and it also reduces the convergence time for the while loop. (2) The termination condition in line 8 is modified from estimator error convergence to spectrum error

convergence. Without the modification, the estimator error convergence condition suffers from local optimum convergence depending on the initial point  $C^{(0)}$ .

Applying the LMS adaptive algorithm, we need the metric  $\Delta_i$ , the derivative of  $J(C)$  in terms of  $c_i$ ,

$$\begin{aligned} \Delta_i &= \frac{dJ(C)}{dc_i} \\ &= \frac{1}{2} \sum_{f_k} \frac{H_i(f_k)}{H_C(f_k)} \left( 1 - \frac{|Y(f_k)|^2}{|H_C(f_k)|^2} \right) + \gamma \cdot \text{sign}(c_i). \end{aligned} \quad (8)$$

Therefore, we update the  $i$ -th element  $c_i$  of the vector  $C$  iteratively as follows,

$$c_i^{(k+1)} = c_i^{(k)} - \mu \cdot \Delta_i, \quad i = 1, 2, \dots, N, \quad (9)$$

for  $k \geq 1$  and the step size  $\mu$ .

The pseudocode of the LMS adaptive algorithm is given in Algorithm 2.

**Algorithm 2** The LMS Adaptive Algorithm for obtaining  $C$ .

```

1:  $\epsilon = \epsilon_0, \mu = \mu_0, \gamma = \gamma_0$ 
2:  $C^{(0)} = (H^T H)^{-1} H^T Y_a$  where  $H = [H_1, H_2, \dots, H_N]$ 
3:  $c_i^{(0)} = (C^{(0)})_i$  for  $i = 1, 2, \dots, N$ .
4:  $k = 0$ 
5: while  $k \leq k_{max}$  do
6:    $H_C = c_1^{(k)} H_1 + c_2^{(k)} H_2 + \dots + c_i^{(k)} H_i + \dots + c_N^{(k)} H_N$ 
7:   if  $k > 0$  then
8:      $e = \text{var}(|Y - H_C|)$ 
9:     if  $e < \epsilon$  then
10:       break
11:   end if
12:   end if
13:    $\Delta_i = \frac{1}{2} \sum_{f_k} \frac{H_i(f_k)}{H_C(f_k)} \left( 1 - \frac{|Y(f_k)|^2}{|H_C(f_k)|^2} \right) + \gamma \cdot \text{sign}(c_i)$ 
14:   for  $i = 1, 2, \dots, N$  do
15:      $c_i^{(k+1)} = c_i^{(k)} - \mu \cdot \Delta_i$ 
16:   end for
17:    $C^{(k+1)} = [c_1^{(k+1)}, c_2^{(k+1)}, \dots, c_N^{(k+1)}]$ 
18:    $k \leftarrow k + 1$ 
19: end while
20: return  $\hat{C} = C^{(k)}$ 

```

The Root-Mean-Square Error (RMSE) of the estimated phase noise using Algorithm 2 is plotted against the true phase noise (dBc/Hz) at 1 MHz frequency offset in Figure 8. The oscillator in our simulation has five configurations:  $-102, -104, -108, -118, -123$  dBc/Hz at 1 MHz offset frequency. The NMSE shows that the LMS adaptive algorithm provides estimates of phase noise within an acceptable range.

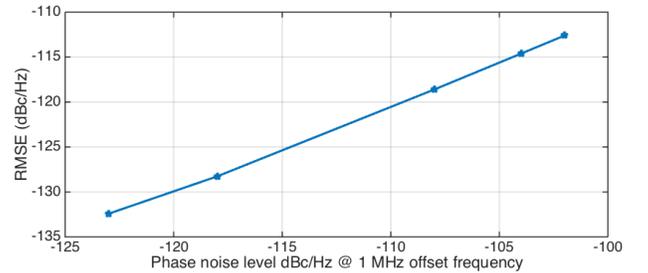


Fig. 8. The RMSE of the estimated phase noise (dBc/Hz at 1MHz frequency offset) vs. true phase noise (dBc/Hz at 1MHz frequency offset).

#### IV. SINR CALCULATION

In order to hasten the EAF optimization process, we need the calculated SINR. The SINR of the  $k$ -th configuration is calculated in terms of phase noise, nonlinearity and noise figure.

$$\text{SINR}^{(k)} = \frac{P_S}{P_{phn}^{(k)} + P_{ip3}^{(k)} + P_n^{(k)}}, \quad (10)$$

where  $P_S$  is the signal power.  $P_{phn}^{(k)}$  is the impairing signal power by phase noise impairment,  $P_{ip3}^{(k)}$  is the impairing signal power by nonlinearity impairment, and  $P_n^{(k)}$  is the impairing signal power by thermal noise impairment.

##### A. Phase Noise Impairment

The impairing signal power  $P_{phn}^{(k)}$  is given as,

$$P_{phn}^{(k)} = P_{phn,S}^{(k)} + P_{phn,I}^{(k)}, \quad (11)$$

where  $P_{phn,S}^{(k)}$  is due to the signal of interest, and  $P_{phn,I}^{(k)}$  is due to interferers.

1) *Phase Noise and the Signal of Interest:* The first term  $P_{phn,S}^{(k)}$  is calculated by the following equation,

$$P_{phn,S}^{(k)}(dBm) = P_S(dBm) + \int_{-\text{inf}}^{\text{inf}} L^{(k)}(f)df \Big|_{(dB)}, \quad (12)$$

where  $L^{(k)}(f)$  is the phase noise spectral density of the  $k$ -th configuration at the offset frequency  $f$ .

2) *Phase Noise and Interference:* The impairing signal power  $P_{phn,I}^{(k)}$  is,

$$P_{phn,I}^{(k)} = \sum_j P_{phn,I_j}^{(k)}, \quad (13)$$

where the interferer  $I_j$  has the power  $P_{I_j}$ . When  $I_j$  is located at the offset frequency  $f_j$ ,  $P_{phn,I_j}^{(k)}$  is,

$$P_{phn,I_j}^{(k)}(dBm) = P_{I_j}(dBm) + L^{(k)}(f_j)(dBc/Hz) + BW(dB), \quad (14)$$

where  $BW$  is the bandwidth of the signal of interest.

##### B. Nonlinearity Impairment

We will calculate  $P_{ip3}^{(k)}$  as follows,

$$P_{ip3}^{(k)} = P_{ip3,S}^{(k)} + P_{ip3,S,I}^{(k)} + P_{ip3,I}^{(k)}, \quad (15)$$

where  $P_{ip3,S}^{(k)}$  is the impairing signal power due to the signal of interest, and  $P_{ip3,I}^{(k)}$  is the impairing signal power due to interference.

1) *Nonlinearity and the Signal of Interest:* If there is no interference, the impairing signal power  $P_{ip3,S}^{(k)}$  is given as,

$$P_{ip3,S}^{(k)}(dBm) = 3 \cdot P_S(dBm) - 2 \cdot \text{IIP3}^{(k)}(dBm), \quad (16)$$

where  $\text{IIP3}^{(k)}$  is the Input third-order Intercept Point (IIP3) of the  $k$ -th configuration.

2) *Nonlinearity and the Signal of Interest and Interference:* If there is any interference, the signal of interest and the interferer are intermodulated as follows,

$$y^p(t) = \alpha_1(x^p(t) + A\cos(2\pi(f_c + f_0)t)) + \alpha_3(x^p(t) + A\cos(2\pi(f_c + f_0)t))^3. \quad (17)$$

Then, Eq. (17) is expanded to,

$$y^p(t) = \alpha_1 x^p(t) + \alpha_1 A \cos(2\pi(f_c + f_0)t) + \alpha_3 x^p(t)^3 + 3\alpha_3 x^p(t)^2 A \cos(2\pi(f_c + f_0)t) + 3\alpha_3 x^p(t) A^2 \cos^2(2\pi(f_c + f_0)t) + \alpha_3 A^3 \cos^3(2\pi(f_c + f_0)t), \quad (18)$$

where  $x^p(t) = x^I(t)\cos(2\pi f_c t) + x^Q(t)\sin(2\pi f_c t)$ .

In Eq. (18), the second, the fourth and the sixth terms are filtered out because these terms are out of band. The first term is the signal of interest, and the third term was already considered in Eq. (16). The fifth term is rewritten as,

$$\frac{3}{2} \alpha_3 x^p(t) A^2 (1 + \cos(4\pi(f_c + f_0)t)). \quad (19)$$

While the second term at  $2(f_c + f_0)$  Hz is filtered out, the first term at  $f_c$  still interferes with the signal of interest at  $f_c$ .

Then, the first term in Eq. (19) is the intermodulated signal by the signal of interest and interference that is given as,

$$P_{ip3,S,I}^{(k)} = \sum_j P_{ip3,S,I_j}^{(k)}, \quad (20)$$

$$P_{ip3,S,I_j}^{(k)}(dBm) = P_S(dBm) + 2 \cdot P_{I_j}(dBm) - 2 \cdot \text{IIP3}^{(k)}(dBm).$$

3) *Nonlinearity and Interference:* Finally, the signal power resulting from the intermodulation of the signal of interest and the interference is given as,

$$P_{ip3,I}^{(k)} = \sum_j P_{ip3,I_j}^{(k)}, \quad (21)$$

$$P_{ip3,I_j}^{(k)}(dBm) = 2 \cdot P_{I_j,1}(dBm) + P_{I_j,2}(dBm) - 2 \cdot \text{IIP3}^{(k)}(dBm),$$

where two interferers of the power  $P_{I_j,1}$  and  $P_{I_j,2}$  respectively are dintermodulated.

Last,  $P_n^{(k)}$  calculation is explained in [2].

#### V. SIMULATION RESULTS OF OPTIMIZATION

The performance of the EAF optimization method was demonstrated even when phase noise is included additionally.

##### A. Simulation Setup

This RF receiver is designed for the IEEE 802.11g WLAN standard where the carrier frequency is 2.4 GHz and the channel bandwidth is 20 MHz. While the received signal has a power of -65 dBm, a pulse blocker has a power of -35 dBm at 2.42 GHz. In this example, reconfigurable RF front-end is reconfigured by switching three amplifiers and one oscillator. The banks of three amplifiers have six possible configurations each. One oscillator has five possible configurations. Thus, there is a total of 1080 different candidate configurations for the reconfigurable front-end. The simulation is implemented in MATLAB Simulink.

TABLE II. PROGRAMMING RESULTS FOR THE RECONFIGURABLE RF FRONT-END

Algorithm	SNR (dB)	Power (mW)	No. Simulation Steps
(1) Exhaustive	30.5	71.7	1080
(2) Local Relaxation	29.4	96.8	12
(3) Simulated Annealing	30.5	71.7	293
(4) Two-phase [10]	31.2	99.5	26
(5) EAF optimization [2]	31.2	99.5	5

### B. Programming Results

In Table II, the performance of (5) the EAF optimization algorithm is compared to the performance of the other four conventional optimization methods: (1) Exhaustive search, (2) Local Relaxation search, (3) Simulated Annealing search and (4) Two-phase relaxation search (these four conventional algorithms were explained in our previous paper [2]). We set the SINR threshold to 24 dB for the SINR specification of 29 dB for the EAF optimization algorithm.

(1) The exhaustive optimization finds a global optimal configuration, but it is time-consuming with 1080 simulation steps. (2) The local relaxation search finds a local optimal configuration. This method is more practical compared to the exhaustive search as it takes only 12 simulation steps. (3) The simulated annealing optimization takes 293 simulation steps - it is better than the exhaustive search regarding speed, but still takes many more steps than the local relaxation search. The simulated annealing optimization finds the global optimum. (4) The two-phase relaxation search finds a local optimal configuration with slightly higher SINR than the local relaxation search. This method has more than twice the amount of simulation steps than the local relaxation search. It takes longer because the two-phase optimization detours the search space by finding a configuration of maximizing the calculated SINR first and then moving to the second-phase of optimization.

(5) Finally, the EAF optimization method is the fastest of the five methods. This method takes five simulation steps, and the SINR and power of the optimal configuration is that of the two-phase optimization. Therefore, we can verify that, even including phase noise impairment, the EAF optimization method significantly improves computational cost compared to the other four optimization methods, and it can still find an optimal configuration with acceptable performance.

## VI. CONCLUSION

In this paper, we studied phase noise impairment for a reconfigurable RF front-end. We proposed a novel statistical signal model for the Interpolated FIR (IFIR) model. Based on the statistical model, we designed estimators for the phase noise spectrum and applied the Least Mean Squares (LMS) adaptive algorithm. Also, SINR was calculated in terms of phase noise impairment and was updated in terms of nonlinearity impairment. Finally, we applied the calculated SINR to the Environment-Adaptable Fast (EAF) algorithm. We showed that the EAF algorithm significantly improves the efficiency of the optimization process of a reconfigurable RF front-end compared to four other conventional optimization methods. Therefore, the EAF algorithm, including phase noise impairment, is reliably designed for finding an optimal configuration of reconfigurable RF front-ends to support a multi-standard platform.

While our study demonstrated the efficiency of the EAF algorithm, the hypothetical example of 1080 configurations was only intended to initially test our exploration of the EAF algorithm. Future work should apply this method to more realistic large-scaled RF-FPGAs.

## ACKNOWLEDGMENTS

This study is sponsored by the DARPA RF-FPGA (Radio Frequency-Field Programmable Gate Arrays) program under Grant HR0011-12-1-0005. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

## REFERENCES

- [1] DARPA, "Radio Frequency-Field Programmable Gate Arrays (RF-FPGA)," November 2011. [Online]. Available: <https://www.fbo.gov/?tab=documents&tabmode=form&subtab=core&tabid=cd49cf41f5df4f860046cb3a9bdf8d9>
- [2] M. Jun, R. Negi, J. Tao, Y.-C. Wang, S. Yin, T. Mukherjee, X. Li, and L. Pileggi, "Environment-Adaptable Efficient Optimization for Programming of Reconfigurable Radio Frequency (RF) Receivers," in *Military Communications Conference (MILCOM), 2014 IEEE*. IEEE, 2014, pp. 1459–1465.
- [3] M. Brandolini, P. Rossi, D. Manstretta, and F. Svelto, "Toward multistandard mobile terminals-fully integrated receivers requirements and architectures," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 53, no. 3, pp. 1026–1038, 2005.
- [4] Q. Zou, A. Tarighat, and A. H. Sayed, "Compensation of phase noise in OFDM wireless systems," *Signal Processing, IEEE Transactions on*, vol. 55, no. 11, pp. 5407–5424, 2007.
- [5] S. SPIRIDON, F. SPIRIDON, D. Claudiu, and M. BODEA, "Phase noise and area- power consumption trade-off in the frequency synthesizers for software defined radio transceivers."
- [6] N. J. Kasdin, "Discrete simulation of colored noise and stochastic processes and  $1/f^\alpha$  power law noise generation," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 802–827, 1995.
- [7] P. P. Vaidyanathan, "Multirate digital filters, filter banks, polyphase networks, and applications: A tutorial," *Proceedings of the IEEE*, vol. 78, no. 1, pp. 56–93, 1990.
- [8] J. Park, K. Muhammad, and K. Roy, "Efficient modeling of  $1/f^\alpha$  noise using multirate process," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 25, no. 7, pp. 1247–1256, 2006.
- [9] L. L. Scharf, *Statistical signal processing*. Addison-Wesley Reading, MA, 1991, vol. 98.
- [10] J. Tao, Y.-C. Wang, M. Jun, X. Li, R. Negi, T. Mukherjee, and L. T. Pileggi, "Toward efficient programming of reconfigurable radio frequency (RF) receivers," in *Design Automation Conference (ASP-DAC), 2014 19th Asia and South Pacific*. IEEE, 2014, pp. 256–261.
- [11] G. Eynard, N. Lewis, D. Dallet, and B. Le Gal, "Efficient phase noise modeling of a PLL-based frequency synthesizer," in *Communications, Control and Signal Processing (ISCCSP), 2010 4th International Symposium on*. IEEE, 2010, pp. 1–4.
- [12] R. B. Staszewski, C. Fernando, and P. T. Balsara, "Event-driven simulation and modeling of phase noise of an RF oscillator," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 52, no. 4, pp. 723–733, 2005.