Co-Learning Bayesian Model Fusion: Efficient Performance Modeling of Analog and Mixed-Signal Circuits Using Side Information

Fa Wang[1], Manzil Zaheer[1], Xin Li[1], Jean-Olivier Plouchart[2] and Alberto Valdes-Garcia[2]
[1]ECE Department, Carnegie Mellon University, Pittsburgh, PA 15213
[2]IBM T. J. Watson Research Center, Yorktown Heights, NY 10598
{fwang1, manzil, xinli}@ece.cmu.edu, {plouchar, avaldes}@us.ibm.com

ABSTRACT

Efficient performance modeling of today's analog and mixed-signal (AMS) circuits is an important yet challenging task. In this paper, we propose a novel performance modeling algorithm that is referred to as Co-Learning Bayesian Model Fusion (CL-BMF). The key idea of CL-BMF is to take advantage of the additional information collected from simulation and/or measurement to reduce the performance modeling cost. Different from the traditional performance modeling approaches which focus on the prior information of model coefficients (i.e. the coefficient side information) only, CL-BMF takes advantage of another new form of prior knowledge: the performance side information. In particular, CL-BMF combines the coefficient side information, the performance side information and a small number of training samples through Bayesian inference based on a graphical model. Two circuit examples designed in a commercial 32nm SOI CMOS process demonstrate that CL-BMF achieves up to 5× speed-up over other state-of-the-art performance modeling techniques without surrendering any accuracy.

1. INTRODUCTION

The aggressive scaling of integrated circuits (ICs) leads to large-scale process variations that cannot be easily mitigated. Process variations, including inter-die variations and local mismatches, significantly impact the parametric yield of analog and mixed signal (AMS) circuits and must be appropriately modeled, analyzed and optimized throughout the AMS design flow [1]-[2]. Historically, various performance modeling approaches have been proposed to address the variability issue [3]-[5]. A performance model approximates the circuit-level performance (e.g. phase noise) as an analytical function (e.g. polynomial) of device-level variations (e.g. ΔV_{TH}) and/or device-level operation point (e.g. DC bias current). Once the performance model is built, it can be applied to various important applications, such as yield estimation [6]-[7], corner extraction [8], design optimization [9]-[10], post-silicon tuning [11]-[15], etc.

Although the problem of performance modeling has been extensively studied in the literature, new technical challenges have recently emerged with the advance of nanoscale technology. Due to the higher complexity and faster speed of AMS circuits, the cost of both pre-silicon simulation and post-silicon measurement substantially increases [16]. For example, a single transistor-level simulation for the 1dB compression point of a radio-frequency (RF) mixer in a commercial 32nm SOI CMOS process can take one hour to finish due to its complicated netlist after parasitic extraction. The expensive circuit simulation and/or measurement lead to a prohibitively high cost for performance modeling.

To combat this challenge, several state-of-the-art performance modeling techniques have been proposed [17]-[23]. In [17]-[22], sparse regression (SR) is used to efficiently determine model coefficients. In [23], a Bayesian model fusion (BMF) framework is developed, which borrows the data generated from an early stage to facilitate efficient performance modeling at a late stage. However, to further reduce the modeling cost and, hence, speed up the design process, there is a strong need to rethink the fundamental strategy for performance modeling.

Most performance modeling approaches [17]-[23] attempt to take advantage of the *side information* to reduce the number of required samples and, hence, the modeling cost. Here we define the side information as the information that is *not* present in the original performance modeling problem, but is indirectly associated with it. Taking sparse regression [17]-[22] as an example, side information refers to the prior knowledge that the model coefficients are sparse (i.e., most coefficients are close to zero). For BMF [23]-[28], side information indicates that the early-stage and late-stage models are similar and, hence, their model coefficients are close. It is important to note that these conventional approaches only focus on the side information related to model coefficients, referred to as the *coefficient side information* (CSI).

In this paper, we further improve the efficacy of performance modeling by considering other side information. Toward this goal, we propose a novel technique that is referred to as *Co-Learning Bayesian Model Fusion* (CL-BMF). CL-BMF considers two different models that predict the same circuit performance. We assume that the complexity of these two models is different: a model with low/high complexity must be trained from a small/large number of samples. The key idea of CL-BMF is to pass the knowledge from the low-complexity model to the high-complexity model (i.e. *co-learning*) to reduce its training cost. In other words, once the low-complexity model is accurately built based on a small number of training samples, the high-complexity model can be trained by borrowing the information from the low-complexity model, instead of relying on the expensive training data only. In our work, the performance model of interest is treated as the high-complexity model, while the low-complexity model provides the *performance side information* (PSI) to reduce the training cost of the high-complexity model.

Mathematically, the proposed CL-BMF method is derived from the Bayesian inference that can be represented as a graphical model [29]-[30]. To build a performance model, CL-BMF combines the following information: (i) the PSI which enables co-learning, (ii) the CSI which provides prior knowledge, and (iii) a small number of training samples collected by pre-silicon simulation or post-silicon measurement. Once the Bayesian inference is constructed, the unknown model coefficients can be accurately determined by maximizing the posterior distribution.

The remainder of this paper is organized as follows. We briefly review the background of performance modeling in Section 2, and then describe the proposed CL-BMF approach in Section 3. Several implementation issues are discussed in Section 4. Two circuit examples are presented to demonstrate the efficacy of CL-BMF in Section 5. Finally, we conclude in Section 6.

978-1-4673-8388-2/15/$31.00 ©2015 IEEE

2. BACKGROUND

For an AMS circuit (e.g. a voltage controlled oscillator), the performance model is an analytical function (e.g. polynomial) of device-level variations (e.g. ΔV_{TH}) and/or device-level operation point (e.g. DC bias current):

$$y \approx f_1(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m \cdot b_m(\mathbf{x}), \quad (1)$$

where y denotes the *performance of interest* (PoI), $\mathbf{x}$ is a vector representing the device-level variations and/or device-level operation point, f_1 denotes the performance model of interest which establishes a mapping from $\mathbf{x}$ to y, $\{\alpha_m;\ m = 1, 2, ..., M\}$ contains the model coefficients, $\{b_m(\mathbf{x});\ m = 1, 2, ..., M\}$ contains the basis functions (e.g., linear and quadratic polynomials), and M denotes the total number of basis functions.

To determine the performance model in (1), the model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ must be solved. Toward this goal, the traditional least-squares fitting method first collects a number of sampling points of $\mathbf{x}$ and y, and then solves the model coefficients from the following optimization problem:

$$\min_{\boldsymbol{\alpha}} \quad \left\| \mathbf{y} - \mathbf{B} \cdot \boldsymbol{\alpha} \right\|_2^2, \quad (2)$$

where

$$\mathbf{B} = \begin{bmatrix} b_1(\mathbf{x}^{(1)}) & b_2(\mathbf{x}^{(1)}) & \cdots & b_M(\mathbf{x}^{(1)}) \\ b_1(\mathbf{x}^{(2)}) & b_2(\mathbf{x}^{(2)}) & \cdots & b_M(\mathbf{x}^{(2)}) \\ \vdots & \vdots & \vdots & \vdots \\ b_1(\mathbf{x}^{(K)}) & b_2(\mathbf{x}^{(K)}) & & b_M(\mathbf{x}^{(K)}) \end{bmatrix} \quad (3)$$

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_M \end{bmatrix}^T \quad (4)$$

$$\mathbf{y} = \begin{bmatrix} y^{(1)} & y^{(2)} & \cdots & y^{(K)} \end{bmatrix}^T. \quad (5)$$

In (3)-(5), $\mathbf{x}^{(k)}$ and $y^{(k)}$ are the values of $\mathbf{x}$ and y at the k-th sampling point respectively, K represents the total number of sampling points, and $\|\bullet\|_2$ stands for the L_2-norm of a vector. To avoid over-fitting, the number of sampling points (i.e., K) must be substantially greater than the number of unknown coefficients (i.e., M).

To reduce the number of required sampling points, several advanced performance modeling approaches [17]-[23] have recently been proposed. These approaches take advantage of the CSI (i.e. the extra prior knowledge related to the model coefficients). Taking SR as an example [22], the model coefficients are solved from the following convex optimization problem:

$$\min_{\boldsymbol{\alpha}} \quad \left\| \mathbf{y} - \mathbf{B} \cdot \boldsymbol{\alpha} \right\|_2^2 + \lambda_1 \left\| \boldsymbol{\alpha} \right\|_1 + \lambda_2 \left\| \boldsymbol{\alpha} \right\|_2^2, \quad (6)$$

where $\|\bullet\|_1$ stands for the L_1-norm of a vector. In (6), the CSI is encoded by the regularization terms $\|\boldsymbol{\alpha}\|_1$ and $\|\boldsymbol{\alpha}\|_2$, and λ_1 and λ_2 denote two parameters controlling the regularization. Specifically, the CSI here is the prior knowledge that the model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ should be sparse and do not take any extremely large values.

On the other hand, consider BMF [23] as another example. The late-stage model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ are efficiently solved by borrowing the early-stage model information. The performance modeling problem is then formulated as a maximum-a-posteriori (MAP) estimation:

$$\min_{\boldsymbol{\alpha}} \quad \left\| \mathbf{y} - \mathbf{B} \cdot \boldsymbol{\alpha} \right\|_2^2 + \lambda \cdot \boldsymbol{\alpha}^T \cdot \mathbf{D} \cdot \boldsymbol{\alpha}, \quad (7)$$

where λ denotes a parameter controlling the regularization term, and $\mathbf{D}$ stands for a diagonal matrix learned from the early-stage model. In (7), the CSI is encoded in the second term which indicates the similarity between the early-stage and late-stage models.

While the CSI has been successfully applied to many practical applications, it is possible to exploit other side information to further reduce the performance modeling cost. Motivated by this observation, we will propose a new CL-BMF technique that takes into account the PSI in this paper.

3. PROPOSED APPROACH

Unlike the conventional approaches that focus on CSI only, CL-BMF considers both CSI and PSI. In what follows, we will discuss the mathematical formulation of CL-BMF in detail.

3.1 Performance Side Information for Co-Learning

For a given performance modeling task, obtaining the PoI value y at each sampling point is often expensive (e.g., by running a transistor-level simulation). Therefore, to reduce the overall performance modeling cost, it is extremely important to reduce the number of required PoI samples for model fitting. Toward this goal, we propose a co-learning method to generate *pseudo samples* of PoI without running actual circuit simulation or performing physical silicon measurement. These pseudo samples are considered as the PSI in our work.

In particular, we consider a vector of performance metrics $\mathbf{z}$ that satisfies the following three criteria: (i) $\mathbf{z}$ is inexpensive to simulate or measure, (ii) $\mathbf{z}$ can be used to accurately predict the PoI y, and (iii) $\mathbf{z}$ is low-dimensional and, hence, the following model has low complexity and can be accurately learned from a small number of training samples:

$$y \approx f_2(\mathbf{z}) = \sum_{t=1}^{T} \beta_t \cdot c_t(\mathbf{z}), \quad (8)$$

where $\{\beta_t;\ t = 1, 2, ..., T\}$ contains the model coefficients, $\{c_t(\mathbf{z});\ t = 1, 2, ..., T\}$ contains the basis functions, and T denotes the total number of basis functions. To simplify our discussion, we further assume that the vector $\mathbf{x}$ in (1) and the vector $\mathbf{z}$ in (8) do not share any common element. Namely, the i-th element x_i of $\mathbf{x}$ is not identical to the j-th element z_j of $\mathbf{z}$ for any i and j.

The key idea of co-learning is to amalgamate the PSI when fitting $f_1(\mathbf{x})$. Since a much smaller training set is needed to fit $f_2(\mathbf{z})$ than $f_1(\mathbf{x})$, we first learn $f_2(\mathbf{z})$ based on a small number of training samples of y and $\mathbf{z}$. Next, $f_1(\mathbf{x})$ can be trained using the PSI generated from $f_2(\mathbf{z})$, instead of relying on the expensive simulation or measurement samples only.

To elaborate, we consider the co-learning procedure shown in Figure 1. First, a small number of *physical samples* $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}, y^{(r)});\ r = 1, 2, ..., R\}$ are collected by simulation or measurement and they are used to fit the low-complexity model $f_2(\mathbf{z})$ accurately. Next, a large number of samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)});\ r = R + 1, R + 2, ..., K\}$ are collected, and a set of pseudo samples $\{(\mathbf{x}^{(r)}, f_2(\mathbf{z}^{(r)}));\ r = R + 1, R + 2, ..., K\}$ are generated for the PoI, as shown by the dashed box in Figure 1. Note that since $\mathbf{z}$ is inexpensive to simulation or measure, these pseudo samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}, f_2(\mathbf{z}^{(r)}));\ r = R + 1, R + 2, ..., K\}$ can be obtained with low cost. Finally, the high-complexity model $f_1(\mathbf{x})$ is fitted by using all the samples, including both the physical samples $\{(\mathbf{x}^{(r)}, y^{(r)});\ r = 1, 2, ..., R\}$ and the pseudo samples $\{(\mathbf{x}^{(r)}, f_2(\mathbf{z}^{(r)}));\ r = R + 1, R + 2, ..., K\}$. By taking advantage of the extra pseudo samples, the aforementioned co-learning is expected to result in a more accurate model $f_1(\mathbf{x})$

than the traditional approach that uses the physical samples $\{(\mathbf{x}^{(r)}, y^{(r)}); r = 1, 2, ..., R\}$ only. Alternatively speaking, applying co-learning can reduce the number of required physical samples and, hence, the overall modeling cost for $f_1(\mathbf{x})$.

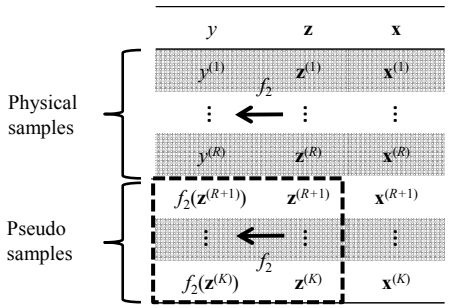

Figure 1. The co-learning procedure is illustrated. The low-complexity model $f_2(\mathbf{z})$ is first fitted using a small number of physical samples that are collected by simulation or measurement. Next, a set of pseudo samples are generated for the PoI, as shown by the dashed box. Finally, the high-complexity model $f_1(\mathbf{x})$ is fitted by using all the samples, including both the physical samples and the pseudo samples.

Note that the low-complexity model $f_2(\mathbf{z})$ may not be highly accurate in practice. Hence, directly using the pseudo samples $\{(\mathbf{x}^{(r)}, f_2(\mathbf{z}^{(r)})); r = R + 1, R + 2, ..., K\}$ to fit the high-complexity model $f_1(\mathbf{x})$ may result in large modeling error. It is crucial to develop a statistical framework that can appropriately incorporate the PSI into our regression modeling process. To this end, a Bayesian inference will be constructed and represented by graphical model. We will discuss the proposed Bayesian interface, including the likelihood models of physical and pseudo samples respectively, in the following sub-sections.

3.2 Likelihood Model of Physical Samples

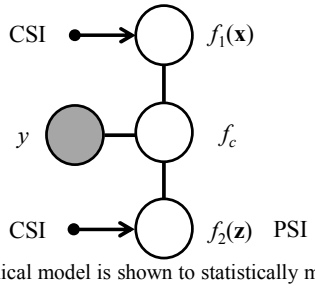

Figure 2. A graphical model is shown to statistically model the likelihood of physical samples.

As shown in Figure 2, a graphical model is constructed to statistically model the likelihood of physical samples. In the graphical model, each node represents a random quantity, and each directed/undirected edge represents a unidirectional/non-directional dependency. The small solid circle represents the prior information as our CSI. The filled node indicates that the node has been observed (i.e. the physical samples of y have been collected). In the co-learning process, we expect that the two models f_1 and f_2 are consistent with each other because they are predicting the same PoI. In this regard, we explicitly define a consensus function f_c to represent the true value of PoI predicted by f_1 and f_2.

According to the graphical model in Figure 2, the joint probability density function (PDF) of f_1, f_2, f_c and y can be represented as:

$$pdf(f_1, f_2, f_c, y) \propto \exp\left[-\frac{(f_1 - f_c)^2}{2\sigma_1^2}\right] \cdot \exp\left[-\frac{(f_2 - f_c)^2}{2\sigma_2^2}\right] \cdot \exp\left[-\frac{(y - f_c)^2}{2\sigma_c^2}\right], \tag{9}$$

where σ_1, σ_2 and σ_c are three parameters. By integrating over the consensus function f_c, we have:

$$pdf(f_1, f_2, y) \propto \exp\left[-\lambda_1 \cdot (y - f_1)^2\right] \cdot \exp\left[-\lambda_2 \cdot (y - f_2)^2\right] \cdot \exp\left[-\lambda_3 \cdot (f_1 - f_2)^2\right], \tag{10}$$

where λ_1, λ_2 and λ_3 are three constants depending on σ_1, σ_2 and σ_c.

Given a number of independent physical samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}, y^{(r)}); r = 1, 2, ..., R\}$ collected by pre-silicon simulation or post-silicon measurement, the joint PDF for all these physical samples can be represented as:

$$pdf(\mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P) \propto \exp\left[-\lambda_1 \cdot \left\|\mathbf{y}_P - \mathbf{f}_{P,1}\right\|_2^2\right] \cdot \exp\left[-\lambda_2 \cdot \left\|\mathbf{y}_P - \mathbf{f}_{P,2}\right\|_2^2\right] \cdot \exp\left[-\lambda_3 \cdot \left\|\mathbf{f}_{P,1} - \mathbf{f}_{P,2}\right\|_2^2\right], \tag{11}$$

where

$$\mathbf{y}_P = \left[y^{(1)} \quad y^{(2)} \quad \cdots \quad y^{(R)}\right]^T \tag{12}$$

$$\mathbf{f}_{P,1} = \left[f_1(\mathbf{x}^{(1)}) \quad f_1(\mathbf{x}^{(2)}) \quad \cdots \quad f_1(\mathbf{x}^{(R)})\right]^T \tag{13}$$

$$\mathbf{f}_{P,2} = \left[f_2(\mathbf{z}^{(1)}) \quad f_2(\mathbf{z}^{(2)}) \quad \cdots \quad f_2(\mathbf{z}^{(R)})\right]^T. \tag{14}$$

In (13)-(14), $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$ are defined by (1) and (8), respectively. Hence, $\mathbf{f}_{P,1}$ and $\mathbf{f}_{P,2}$ can be re-written as:

$$\mathbf{f}_{P,1} = \mathbf{B}_P \cdot \boldsymbol{\alpha} \tag{15}$$

$$\mathbf{f}_{P,2} = \mathbf{C}_P \cdot \boldsymbol{\beta}, \tag{16}$$

where

$$\boldsymbol{\beta} = [\beta_1 \quad \beta_2 \quad \cdots \quad \beta_T]^T \tag{17}$$

$$\mathbf{B}_P = \begin{bmatrix} b_1(\mathbf{x}^{(1)}) & b_2(\mathbf{x}^{(1)}) & \cdots & b_M(\mathbf{x}^{(1)}) \\ b_1(\mathbf{x}^{(2)}) & b_2(\mathbf{x}^{(2)}) & \cdots & b_M(\mathbf{x}^{(2)}) \\ \vdots & \vdots & \vdots & \vdots \\ b_1(\mathbf{x}^{(R)}) & b_2(\mathbf{x}^{(R)}) & & b_M(\mathbf{x}^{(R)}) \end{bmatrix} \tag{18}$$

$$\mathbf{C}_P = \begin{bmatrix} c_1(\mathbf{z}^{(1)}) & c_2(\mathbf{z}^{(1)}) & \cdots & c_T(\mathbf{z}^{(1)}) \\ c_1(\mathbf{z}^{(2)}) & c_2(\mathbf{z}^{(2)}) & \cdots & c_T(\mathbf{z}^{(2)}) \\ \vdots & \vdots & \vdots & \vdots \\ c_1(\mathbf{z}^{(R)}) & c_2(\mathbf{z}^{(R)}) & & c_T(\mathbf{z}^{(R)}) \end{bmatrix}. \tag{19}$$

Substituting (15)-(16) into the joint PDF in (11) yields the likelihood function:

$$pdf(\mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \exp\left[-\lambda_1 \cdot \left\|\mathbf{y}_P - \mathbf{B}_P \cdot \boldsymbol{\alpha}\right\|_2^2\right] \cdot \exp\left[-\lambda_2 \cdot \left\|\mathbf{y}_P - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2\right] \cdot \exp\left[-\lambda_3 \cdot \left\|\mathbf{B}_P \cdot \boldsymbol{\alpha} - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2\right]. \tag{20}$$

The likelihood function in (20) consists of three L_2-norm terms. The first term represents the difference between the physical samples of y and the approximated function $f_1(\mathbf{x})$, the second term represents the difference between the physical samples of y and the approximated function $f_2(\mathbf{z})$, and the third term represents the

difference between $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$. The aforementioned likelihood function will be further used to solve the Bayesian inference in Section 3.4.

3.3 Likelihood Model of Pseudo Samples

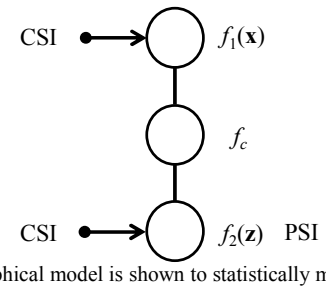

Figure 3. A graphical model is shown to statistically model the likelihood of pseudo samples.

As shown in Figure 3, a graphical model is constructed to statistically model the likelihood of pseudo samples. According to the graphical model in Figure 3, the joint PDF of f_1, f_2 and f_c can be represented as:

$$pdf(f_1, f_2, f_c) \propto \exp\left[-\frac{(f_1 - f_c)^2}{2\sigma_1^2}\right] \cdot \exp\left[-\frac{(f_2 - f_c)^2}{2\sigma_2^2}\right]. \quad (21)$$

By integrating over the consensus function f_c, we have:

$$pdf(f_1, f_2) \propto \exp\left[-\lambda_4 \cdot (f_1 - f_2)^2\right], \quad (22)$$

where λ_4 is a constant depending on σ_1 and σ_2. Given a number of independent pseudo samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}); r = R+1, R+2, ..., K\}$, the joint PDF for all these pseudo samples can be represented as:

$$pdf(\mathbf{f}_{S,1}, \mathbf{f}_{S,2}) \propto \exp\left[-\lambda_4 \cdot \left\|\mathbf{f}_{S,1} - \mathbf{f}_{S,2}\right\|_2^2\right], \quad (23)$$

where

$$\mathbf{f}_{S,1} = \left[f_1(\mathbf{x}^{(R+1)}) \quad f_1(\mathbf{x}^{(R+2)}) \quad \cdots \quad f_1(\mathbf{x}^{(K)})\right]^T \quad (24)$$

$$\mathbf{f}_{S,2} = \left[f_2(\mathbf{z}^{(R+1)}) \quad f_2(\mathbf{z}^{(R+2)}) \quad \cdots \quad f_2(\mathbf{z}^{(K)})\right]^T. \quad (25)$$

In (24)-(25), $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$ are defined by (1) and (8), respectively. Hence, $\mathbf{f}_{s,1}$ and $\mathbf{f}_{s,2}$ can be re-written as:

$$\mathbf{f}_{S,1} = \mathbf{B}_S \cdot \boldsymbol{\alpha} \quad (26)$$

$$\mathbf{f}_{S,2} = \mathbf{C}_S \cdot \boldsymbol{\beta}, \quad (27)$$

where

$$\mathbf{B}_S = \begin{bmatrix} b_1(\mathbf{x}^{(R+1)}) & b_2(\mathbf{x}^{(R+1)}) & \cdots & b_M(\mathbf{x}^{(R+1)}) \\ b_1(\mathbf{x}^{(R+2)}) & b_2(\mathbf{x}^{(R+2)}) & \cdots & b_M(\mathbf{x}^{(R+2)}) \\ \vdots & \vdots & \vdots & \vdots \\ b_1(\mathbf{x}^{(K)}) & b_2(\mathbf{x}^{(K)}) & & b_M(\mathbf{x}^{(K)}) \end{bmatrix} \quad (28)$$

$$\mathbf{C}_S = \begin{bmatrix} c_1(\mathbf{z}^{(R+1)}) & c_2(\mathbf{z}^{(R+1)}) & \cdots & c_T(\mathbf{z}^{(R+1)}) \\ c_1(\mathbf{z}^{(R+2)}) & c_2(\mathbf{z}^{(R+2)}) & \cdots & c_T(\mathbf{z}^{(R+2)}) \\ \vdots & \vdots & \vdots & \vdots \\ c_1(\mathbf{z}^{(K)}) & c_2(\mathbf{z}^{(K)}) & & c_T(\mathbf{z}^{(K)}) \end{bmatrix}. \quad (29)$$

Substituting (26)-(27) into the joint PDF in (23) yields the likelihood function:

$$pdf(\mathbf{f}_{S,1}, \mathbf{f}_{S,2} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \exp\left[-\lambda_4 \cdot \left\|\mathbf{B}_S \cdot \boldsymbol{\alpha} - \mathbf{C}_S \cdot \boldsymbol{\beta}\right\|_2^2\right]. \quad (30)$$

The likelihood function in (30) contains only one L_2-norm term, which represents the difference between $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$. It encodes our prior knowledge that the function values approximated by $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$ are "likely" to be similar. The likelihood function in (30) will be combined with the likelihood function in (20) to solve the Bayesian inference in Section 3.4.

3.4 Bayesian Inference for Co-Learning

Based on the likelihood function defined in Section 3.2 and 3.3, the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can be optimally determined by MAP estimation through Bayesian inference [30]. Given that the physical and pseudo samples are independently generated, the likelihood function in (20) and (30) can be combined:

$$\begin{aligned} pdf(\mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P, \mathbf{f}_{S,1}, \mathbf{f}_{S,2} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) &\propto \exp\left[-\lambda_1 \cdot \left\|\mathbf{y}_P - \mathbf{B}_P \cdot \boldsymbol{\alpha}\right\|_2^2\right] \\ \cdot \exp\left[-\lambda_2 \cdot \left\|\mathbf{y}_P - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2\right] &\cdot \exp\left[-\lambda_3 \cdot \left\|\mathbf{B}_P \cdot \boldsymbol{\alpha} - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2\right] \\ \cdot \exp\left[-\lambda_4 \cdot \left\|\mathbf{B}_S \cdot \boldsymbol{\alpha} - \mathbf{C}_S \cdot \boldsymbol{\beta}\right\|_2^2\right] & \end{aligned}. \quad (31)$$

Similar to [23], we statistically represent the CSI of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ as a PDF that is referred to as the prior distribution. Here, we define the prior distribution of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in a general form:

$$pdf(\boldsymbol{\alpha}, \boldsymbol{\beta}). \quad (32)$$

In (32), the prior knowledge is encoded in the PDF template (e.g., Gaussian distribution, Laplace distribution, etc.) and/or the PDF parameters (e.g. mean and standard deviation for a Gaussian distribution). More details about the prior definition will be discussed in Section 4.1.

Once the prior distribution $pdf(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is specified, we will combine $pdf(\boldsymbol{\alpha}, \boldsymbol{\beta})$ with the physical samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}, y^{(r)}); r = 1, 2, ..., R\}$ and the pseudo samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}); r = R+1, R+2, ..., K\}$ to solve the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ by MAP estimation [30]. The key idea of MAP estimation is to find the optimal values of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to maximize the posterior distribution $pdf(\boldsymbol{\alpha}, \boldsymbol{\beta} \mid \mathbf{f}_{p,1}, \mathbf{f}_{p,2}, \mathbf{y}_p, \mathbf{f}_{s,1}, \mathbf{f}_{s,2})$. Intuitively, the posterior distribution indicates the remaining uncertainty of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, after we observe all physical and pseudo samples. Hence, MAP attempts to find the optimal $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ that are most likely to occur.

Based on Bayes' theorem, the posterior distribution can be represented as:

$$\begin{aligned} pdf(\boldsymbol{\alpha}, \boldsymbol{\beta} \mid \mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P, \mathbf{f}_{S,1}, \mathbf{f}_{S,2}) \propto \\ pdf(\boldsymbol{\alpha}, \boldsymbol{\beta}) \cdot pdf(\mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P, \mathbf{f}_{S,1}, \mathbf{f}_{S,2} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) \end{aligned}. \quad (33)$$

Mathematically, the MAP solution can be found by solving the following optimization problem:

$$\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \quad pdf(\boldsymbol{\alpha}, \boldsymbol{\beta} \mid \mathbf{f}_{P,1}, \mathbf{f}_{P,2}, \mathbf{y}_P, \mathbf{f}_{S,1}, \mathbf{f}_{S,2}). \quad (34)$$

Combining (31)-(34) and taking the logarithm for the posterior distribution, we can convert (34) to the following equivalent optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \quad & -\log\left[pdf(\boldsymbol{\alpha}, \boldsymbol{\beta})\right] + \lambda_1 \cdot \left\|\mathbf{y}_P - \mathbf{B}_P \cdot \boldsymbol{\alpha}\right\|_2^2 \\ & + \lambda_2 \cdot \left\|\mathbf{y}_P - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2 + \lambda_3 \cdot \left\|\mathbf{B}_P \cdot \boldsymbol{\alpha} - \mathbf{C}_P \cdot \boldsymbol{\beta}\right\|_2^2 \cdot \\ & + \lambda_4 \cdot \left\|\mathbf{B}_S \cdot \boldsymbol{\alpha} - \mathbf{C}_S \cdot \boldsymbol{\beta}\right\|_2^2 \end{aligned} \quad (35)$$

In (35), the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are solved together. The first term in the cost function represents the prior knowledge which encodes the CSI. The second term penalizes the difference between the physical samples of y and the approximated function $f_1(\mathbf{x})$. The first two terms resemble the traditional regression approaches as described in (6) and (7). The third term penalizes the difference between the physical samples of y and the

approximated function $f_2(\mathbf{z})$, and the fourth and fifth terms penalize the inconsistency between $f_1(\mathbf{x})$ and $f_2(\mathbf{z})$ (i.e. the PSI) for physical and pseudo samples respectively.

While the basic idea of likelihood modeling and Bayesian inference for co-learning is illustrated in this section, several implementation issues must be carefully considered in order to make CL-BMF of practical utility. These implementation details will be further discussed in the next section.

4. IMPLEMENTATION DETAILS

In this section, we further discuss several important implementation issues for CL-BMF, including (i) prior definition, and (ii) cross-validation.

4.1 Prior Definition

The prior knowledge (i.e. the CSI) of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is statistically encoded by the prior distribution $pdf(\boldsymbol{\alpha}, \boldsymbol{\beta})$ in (32). In this sub-section, we consider two special cases: (i) the Laplace prior distribution, and (ii) the Gaussian prior distribution. These prior distributions have been extensively used in the literature [23]-[25], [30]-[32]. It should be noted that the proposed CL-BMF framework can be easily extended to accommodate other prior distributions [26]-[28], even though the details of these possible extensions are not included in this paper due to the page limit.

1) Laplace prior distribution: The Laplace prior distribution is defined as:

$$pdf(\boldsymbol{\alpha},\boldsymbol{\beta}) \propto \exp\left[-\kappa\cdot\left\|\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\end{bmatrix}\right\|_1\right] = \exp\left[-\kappa\cdot\|\boldsymbol{\alpha}\|_1\right]\cdot\exp\left[-\kappa\cdot\|\boldsymbol{\beta}\|_1\right], \quad (36)$$

where κ is a hyper-parameter and $\|\bullet\|_1$ represents the L_1-norm of a vector. The Laplace prior distribution attempts to promote sparsity for the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. In other words, by applying the Laplace prior distribution, the MAP solution of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is likely to be sparse. Combining (35) and (36), we obtain the following optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\beta}}\quad & \|\boldsymbol{\alpha}\|_1 + \|\boldsymbol{\beta}\|_1 + \frac{\lambda_1}{\kappa}\cdot\left\|\mathbf{y}_P - \mathbf{B}_P\cdot\boldsymbol{\alpha}\right\|_2^2 \\ & + \frac{\lambda_2}{\kappa}\cdot\left\|\mathbf{y}_P - \mathbf{C}_P\cdot\boldsymbol{\beta}\right\|_2^2 + \frac{\lambda_3}{\kappa}\cdot\left\|\mathbf{B}_P\cdot\boldsymbol{\alpha} - \mathbf{C}_P\cdot\boldsymbol{\beta}\right\|_2^2 . \\ & + \frac{\lambda_4}{\kappa}\cdot\left\|\mathbf{B}_S\cdot\boldsymbol{\alpha} - \mathbf{C}_S\cdot\boldsymbol{\beta}\right\|_2^2\end{aligned} \quad (37)$$

It is straightforward to verify that the optimization problem in (37) is convex and, hence, can be efficiently solved to find its global optimum.

2) Gaussian prior distribution: In the case where the CSI contains the magnitude information of the model coefficients from an early stage [23], a Gaussian prior distribution can be used. To this end, we consider the following models from the early-stage:

$$y \approx f_{E,1}(\mathbf{x}) = \sum_{m=1}^{M}\alpha_{E,m}\cdot b_m(\mathbf{x}) \quad (38)$$

$$y \approx f_{E,2}(\mathbf{z}) = \sum_{t=1}^{T}\beta_{E,t}\cdot c_t(\mathbf{z}), \quad (39)$$

where $f_{E,1}$ and $f_{E,2}$ denote the early-stage versions of f_1 and f_2, $\{\alpha_{E,m};\ m = 1, 2, ..., M\}$ contains the model coefficients of $f_{E,1}$, and $\{\beta_{E,t};\ t = 1, 2, ..., T\}$ contains the model coefficients of $f_{E,2}$. We expect that the model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ and $\{\beta_t;\ t = 1, 2, ..., T\}$ are close to $\{\alpha_{E,m};\ m = 1, 2, ..., M\}$ and $\{\beta_{E,t};\ t = 1, 2, ..., T\}$ respectively, since they are associated with the same PoI. Following this assumption, the joint prior distribution of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can be defined as a multivariate Gaussian distribution [23]:

$$pdf(\boldsymbol{\alpha},\boldsymbol{\beta}) \propto \exp\left[-\frac{1}{2}\cdot\left(\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\end{bmatrix}\right)^T\cdot\mathbf{D}\cdot\left(\begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\end{bmatrix}\right)\right], \quad (40)$$

where

$$\mathbf{D} = \begin{bmatrix}\mathbf{D}_{\boldsymbol{\alpha}} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}_{\boldsymbol{\beta}}\end{bmatrix} \quad (41)$$

$$\mathbf{D}_{\boldsymbol{\alpha}} = \mathrm{diag}\left(\alpha_{E,1}^{-2}, \alpha_{E,2}^{-2}, \cdots, \alpha_{E,M}^{-2}\right) \quad (42)$$

$$\mathbf{D}_{\boldsymbol{\beta}} = \mathrm{diag}\left(\beta_{E,1}^{-2}, \beta_{E,2}^{-2}, \cdots, \beta_{E,T}^{-2}\right). \quad (43)$$

In (40)-(43), diag(•) represents the operator to construct a diagonal matrix and $\mathbf{D}$ is the diagonal covariance matrix of the prior distribution encoding our CSI. Combining (35) and (40), we obtain the following optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\beta}}\quad & \boldsymbol{\alpha}^T\cdot\mathbf{D}_{\boldsymbol{\alpha}}\cdot\boldsymbol{\alpha} + \boldsymbol{\beta}^T\cdot\mathbf{D}_{\boldsymbol{\beta}}\cdot\boldsymbol{\beta} + \lambda_1\cdot\left\|\mathbf{y}_P - \mathbf{B}_P\cdot\boldsymbol{\alpha}\right\|_2^2 \\ & + \lambda_2\cdot\left\|\mathbf{y}_P - \mathbf{C}_P\cdot\boldsymbol{\beta}\right\|_2^2 + \lambda_3\cdot\left\|\mathbf{B}_P\cdot\boldsymbol{\alpha} - \mathbf{C}_P\cdot\boldsymbol{\beta}\right\|_2^2 . \\ & + \lambda_4\cdot\left\|\mathbf{B}_S\cdot\boldsymbol{\alpha} - \mathbf{C}_S\cdot\boldsymbol{\beta}\right\|_2^2\end{aligned} \quad (44)$$

Eq. (44) represents a convex quadratic programing problem. Its globally optimal solution can be analytically solved by using the first-order optimality condition [33].

4.2 Cross-Validation

To solve the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in (37) or (44), the parameters λ_1, λ_2, λ_3, λ_4 and κ must be carefully determined. First, we consider the optimization problem in (44) with four parameters λ_1, λ_2, λ_3 and λ_4. Note that these four parameters are not independent because they can be uniquely determined by σ_1, σ_2 and σ_c from (9) and (21). In this paper, we assume that the simulation or measurement noise is negligible for y. Namely, the consensus function f_c (i.e. the true value of PoI) is equal to y (i.e. the observed value of PoI). In this case, the parameter σ_c can be set to a value that is close to zero and only two other parameters σ_1 and σ_2 should be determined. Second, we consider the optimization problem in (37) with five parameters λ_1, λ_2, λ_3, λ_4 and κ. It is straightforward to note that the parameter κ can be absorbed into λ_1, λ_2, λ_3 and λ_4. Therefore, we only need to determine two parameters $\kappa\cdot\sigma_1$ and $\kappa\cdot\sigma_2$ eventually.

It is important to find the optimal values of the aforementioned two parameters so that the modeling error is minimized. Toward this goal, we use N-fold cross-validation [30] to estimate the modeling error for different parameter values. In particular, we partition the entire data set into N groups. Modeling error is estimated from N independent runs using the physical samples. In each run, one of the N groups is used to estimate the modeling error and all other groups are used to calculate the model coefficients. In addition, different groups should be selected for error estimation in different runs. As such, each run gives an error value e_n (n = 1, 2, ..., N) that is estimated from a unique group of data points. The final modeling error is computed as the average of $\{e_n;\ n = 1, 2, ..., N\}$, i.e., $e = (e_1 + e_2 + ... + e_N)/N$.

4.3 Summary

Algorithm 1 summarizes the major steps of the proposed CL-BMF method. It consists of two major components: (i) prior distribution definition, and (ii) MAP estimation. CL-BMF can automatically assign the appropriate weights to the PSI, the CSI and the physical samples by optimally tuning the parameters λ_1,

$\lambda_2, \lambda_3, \lambda_4$ and κ in (37) or (44) based on cross-validation. If the PSI or CSI is not accurate, cross-validation can automatically assign a small weight value to the inaccurate side information. As such, the performance model $f_1(\mathbf{x})$ will not be distorted due to the inaccurate side information.

Algorithm 1: Co-Learning Bayesian Model Fusion (CL-BMF)

1. According to the CSI, define the prior distribution for the model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ and $\{\beta_t;\ t = 1, 2, ..., T\}$ by (36) or (40).
2. Collect the physical samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)}, y^{(r)});\ r = 1, 2, ..., R\}$ and the pseudo samples $\{(\mathbf{x}^{(r)}, \mathbf{z}^{(r)});\ r = R + 1, R + 2, ..., K\}$.
3. Solve the model coefficients $\{\alpha_m;\ m = 1, 2, ..., M\}$ and $\{\beta_t;\ t = 1, 2, ..., T\}$ based on the optimization problem in (37) or (44) where the parameters λ_1, λ_2, λ_3, λ_4 and κ are determined by cross-validation.

To exploit the benefit of PSI, it is important to appropriately identify a vector of performance metrics $\mathbf{z}$ that satisfies the criteria listed in Section 3.1. In practice, these performance metrics can be selected by an analog/RF designer according to his design experience, as will be illustrated by our circuit examples in Section 5.

5. NUMERICAL EXAMPLES

In this section, we demonstrate the efficacy of CL-BMF by several circuit examples designed in a commercial 32nm SOI CMOS process. Our objective is to build performance models for either pre-silicon validation or post-silicon tuning of these circuits. For testing and comparison purposes, three different performance modeling techniques are implemented: (i) the traditional sparse regression method based on L_1-norm regularization (SR-L1) [32], (ii) Bayesian model fusion using Gaussian prior (BMF) [23], and (iii) CL-BMF. Here, the SR-L1 and BMF methods are chosen for comparison, since they are among the state-of-the-art techniques in the literature.

In our experiments, a two-level cross-validation is used. In the inner loop, the first-level cross-validation is used to determine the parameters $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ and κ. In the outer loop, the second-level cross-validation is applied to estimate the modeling error. All numerical experiments are run on a 2.53GHz Linux server with 64GB memory.

5.1 Ring Oscillator

Figure 4 shows the simplified circuit schematic of a ring oscillator (RO) designed in a commercial 32nm SOI CMOS process. In this example, there are 7693 independent random variables to model device-level process variations, including both inter-die variations and random mismatches. Our objective is to approximate the phase noise (PN) of the RO as a linear function of these 7693 random variables.

Table 1 summarizes the side information for the aforementioned performance modeling problem. The sparsity of the model coefficients is considered as the CSI, which is encoded by a Laplace prior distribution shown in (36). On the other hand, the center frequency (F_0) of the RO is strongly correlated with its PN according to our design experience. Simulating the F_0 by transient analysis has significantly lower computational cost than simulating the PN by periodic steady-state analysis. For these reasons, we choose the F_0 as our alternative performance metric to define the PSI. Statistically, the PSI is encoded by the Gaussian likelihood function in (31).

We collect a set of Monte Carlo samples for both PN and F_0. Next, we build the performance models for PN by two different approaches: (i) SR-L1 and (ii) CL-BMF. When CL-BMF is applied, 750 Monte Carlo samples of F_0 are used to build the likelihood function for PSI. Figure 5 shows the performance modeling error as a function of the number of Monte Carlo samples for PN.

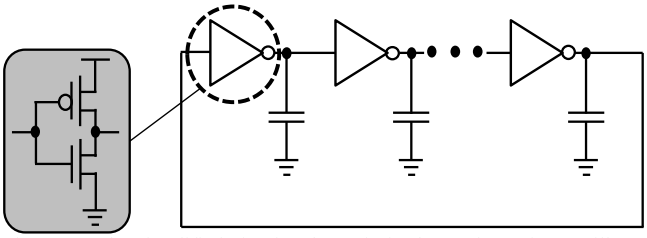

Figure 4. The simplified circuit schematic is shown for a ring oscillator (RO) designed in a commercial 32nm SOI CMOS process.

Table 1. Side information for performance modeling of RO

	Description	Distribution	SR-L1	CL-BMF
CSI	Sparsity	Laplace prior	Yes	Yes
PSI	F_0	Gaussian likelihood	No	Yes

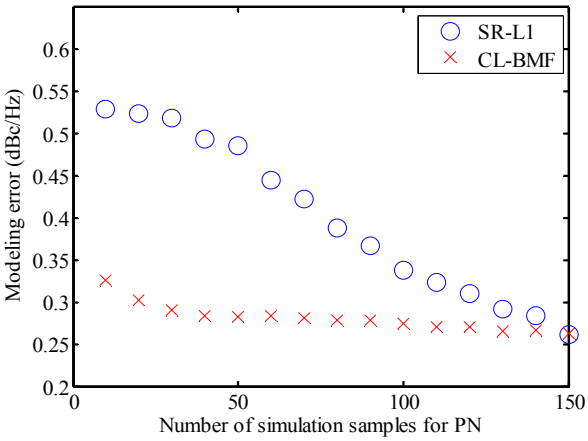

Figure 5. The performance modeling error of phase noise (PN) is shown as a function of the number of simulation samples for PN.

Table 2. Performance modeling error and cost for RO

	SR-L1	CL-BMF
Modeling error (dBc/Hz)	0.34	0.33
Number of simulation samples for PN	100	10
Simulation time for PN (Sec.)	5540	554
Number of simulation samples for F_0	0	750
Simulation time for F_0 (Sec.)	0	540
Simulation cost (Sec.)	5540	1094
Fitting cost (Sec.)	5.4	108.6
Overall modelling cost (Sec.)	5545.4	1202.6

Studying Figure 5 reveals several important observations. First, for both SR-L1 and CL-BMF, the modeling error decreases as the number of PN samples increases. Second, but more importantly, given the same number of PN samples, CL-BMF is able to achieve substantially higher accuracy than SR-L1, especially when few samples are available. In this example, CL-BMF offers superior modeling accuracy, because it takes into account both the CSI and the PSI, while SR-L1 considers the CSI only.

Table 2 further compares the modeling error and cost for SR-L1 and CL-BMF. The overall modeling cost consists of two major

parts: (i) the simulation cost (i.e., the cost of running transistor-level simulations to collect sampling points), and (ii) the fitting cost (i.e., the cost of solving all unknown model coefficients). As shown in Table 2, the overall modeling cost is dominated by the PN simulation in this example. CL-BMF needs less PN samples and, hence, has lower modeling cost than SR-L1. CL-BMF achieves 4.6× runtime speed-up over SR-L1 in this example.

5.2 Low-Noise Amplifier

In this sub-section, we consider a tunable 60GHz low-noise amplifier (LNA) designed in a commercial 32nm SOI CMOS process. The simplified circuit schematic of the LNA is shown in Figure 6. It consists of three stages and their bias currents are tunable. Our objective is to efficiently measure the circuit performance metrics by low-cost on-chip sensors and then adaptively tune the bias currents to maximize the performance and/or reduce the power of the LNA.

Toward this goal, indirect performance sensing has been proposed in the literature to estimate the performance metrics that cannot be easily measured by on-chip sensors [11]-[15]. In this example, we consider the noise figure (NF) of the LNA as our PoI. Instead of directly measuring the NF, the key idea of indirect performance sensing is to predict its value by using a set of other performance metrics that are referred to as the performances of measurements (i.e. PoMs). The choices of PoMs must satisfy the following two criteria. First, the PoMs should be highly correlated with the NF. Second, the PoMs can be easily measured by low-cost on-chip sensors.

To predict the NF of the LNA in Figure 6, we choose three performance metrics as the PoMs based on our design knowledge: (i) the bias current of the first stage (I_B), (ii) the drain voltage of the transistor T_4 (V_N), and (iii) the environmental temperature (*Temp*). Once the PoMs are chosen, we need to further build a performance model to approximate the NF as a polynomial function of the PoMs.

Table 3 summarizes the side information for the aforementioned performance modeling problem. At the pre-silicon stage, a performance model is fitted based on the simulation data generated by random sampling. The magnitude of the coefficients of this pre-silicon model is considered as our CSI, which is encoded by a Gaussian prior distribution shown in (40).

At the post-silicon stage, both the NF and the PoMs (i.e. I_B, V_N and *Temp*) are measured from a number of chips by wafer probe test. Here, we have to measure the NF by wafer probe, because it is a high-frequency performance metric at 60GHz and, hence, cannot be measured at the package level. In addition, each chip must be measured individually at such a high frequency, implying that the measurement cost is prohibitively high for this LNA example.

For testing and comparison purposes, we build the performance models for NF by two different approaches: (i) BMF, and (ii) CL-BMF. When CL-BMF is applied, we consider the S21 as our alternative performance metric to define the PSI, because it is strongly correlated with the NF according to our design experience and measuring the S21 is significantly less expensive than the NF for high-frequency wafer probe test. In our experiment, the S21 values measured from 13 chips are used to build the likelihood function for PSI.

Figure 7 and Table 4 compare the modeling error and cost for BMF and CL-BMF. In this example, the overall modeling consists of four major parts: (i) the measurement cost for NF, (ii) the measurement cost for S21, (iii) the measurement cost for probe alignment when testing each die, and (iv) the fitting cost for solving the unknown model coefficients. As shown in Table 4, the overall modeling cost is dominated by the measurement cost for NF. CL-BMF requires less measurement data for NF and, hence, has lower modeling cost than BMF. In this example, CL-BMF achieves 5× speed-up over BMF in terms of the overall modeling cost.

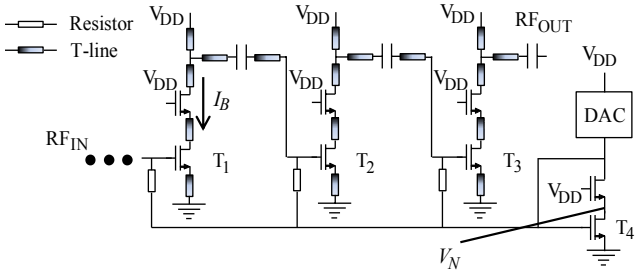

Figure 6. The simplified circuit schematic is shown for a three-stage 60GHz low noise amplifier (LNA) designed in a commercial 32nm SOI CMOS process.

Table 3. Side information for performance modeling of LNA

	Description	Distribution	BMF	CL-BMF
CSI	Magnitude	Gaussian prior	Yes	Yes
PSI	S21	Gaussian likelihood	No	Yes

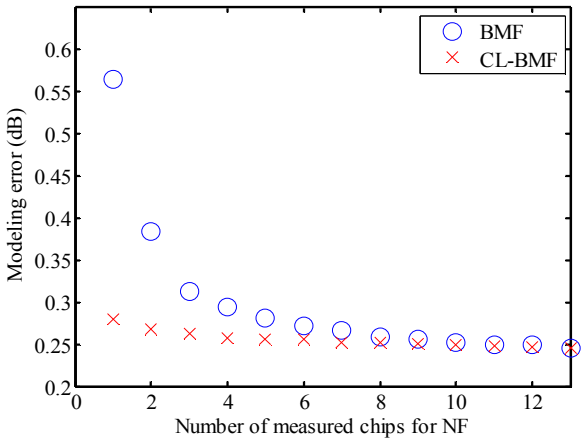

Figure 7. The performance modeling error of noise figure (NF) is shown as a function of the number of measured chips for NF.

Table 4. Performance modeling error and cost for LNA

	BMF	CL-BMF
Modeling error (dB)	0.28	0.28
Number of measured chips for NF	5	1
Measurement time for NF (Sec.)	250	50
Number of measured chips for S21	0	13
Measurement time for S21 (Sec.)	0	0.0039
Alignment time for wafer probe (Sec.)	0.63	1.63
Measurement cost (Sec.)	250.63	51.63
Fitting cost (Sec.)	0.011	0.241
Overall modeling cost (Sec.)	250.6	51.9

6. CONCLUSIONS

In this paper, a novel CL-BMF algorithm is proposed for efficient performance modeling of AMS circuits. CL-BMF optimally combines the following information: (i) the CSI, (ii) the PSI, and (iii) a small number of training samples. Bayesian inference is constructed and represented as a graphical model,

where the CSI and the PSI are encoded by the prior distribution and the likelihood function respectively. From the Bayesian inference, the unknown model coefficients can be accurately determined by maximizing the posterior distribution. As is demonstrated by our circuit examples designed in a commercial 32nm SOI CMOS process, the proposed CL-BMF method achieves up to 5× cost reduction over other state-of-the-art performance modeling techniques. In our future work, we will further study a number of implementation issues for CL-BMF (e.g., automatic construction of the PSI for a given circuit).

7. ACKNOWLEDGMENT

This work is sponsored in part by the DARPA HEALICS program under Air Force Research Laboratory (AFRL) contract FA8650-09-C-7924 and the National Science Foundation under contract CCF–1316363. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. The authors would like to thank Bodhisatwa Sadhu for valuable discussions and Daniel Friedman for management support.

8. REFERENCES

[1] Semiconductor Industry Associate, *International Technology Roadmap for Semiconductors*, 2011.

[2] X. Li *et al.*, *Statistical Performance Modeling and Optimization*, Now Publishers, 2007.

[3] A. Singhee *et al.*, "Beyond low-order statistical response surfaces: latent variable regression for efficient, highly nonlinear fitting," *DAC*, pp. 256-261, 2007.

[4] A. Mitev *et al.*, "Principle Hessian direction based parameter reduction for interconnect networks with process variation," *ICCAD*, pp. 632-637, 2007.

[5] T. McConaghy *et al.*, "Template-free symbolic performance modeling of analog circuits via canonical-form functions and genetic programming," *IEEE TCAD*, vol. 28, no. 8, pp. 1162-1175, Aug. 2009.

[6] X. Li *et al.*, "Asymptotic probability extraction for nonnormal performance distributions," *IEEE TCAD*, vol. 26, no. 1, pp. 16-37, Jan. 2007.

[7] F. Gong *et al.*, "Variability-aware parametric yield estimation for analog/mixed-signal circuits: concepts, algorithms, and challenges," *IEEE D&T*, vol. 31, no. 4, pp. 6-15, Jan. 2014.

[8] H. Zhang *et al.*, "Efficient design-specific worst-case corner extraction for integrated circuits," *DAC*, pp. 386-389, 2009.

[9] X. Li *et al.*, "Robust analog/RF circuit design with projection-based performance modeling," *IEEE TCAD*, vol. 26, no. 1, pp. 2-15, Jan. 2007.

[10] Y. Wang *et al.*, "Enabling efficient analog synthesis by coupling sparse regression and polynomial optimization," *DAC*, pp. 1-6, 2014.

[11] S. Sun *et al.*, "Indirect performance sensing for on-chip analog self-healing via Bayesian model fusion," *CICC*, 2013.

[12] S. Sun *et al.*, "Indirect performance sensing for on-chip self-healing of analog and RF circuits," *IEEE TCAS-I*, vol. 61, no. 8, pp. 2243-2252, Jul. 2014.

[13] M. Andraud *et al.*, "One-shot calibration of RF circuits based on non-intrusive sensors," *DAC*, pp. 1-6, 2014.

[14] J.-O. Plouchart *et al.*, "Adaptive circuit design methodology and test applied to millimeter-wave circuits," *IEEE D&T*, vol. 31, no. 6, pp. 8-18, Jal. 2014.

[15] J.-O. Plouchart *et al.*, "A 18mW, 3.3dB NF, 60GHz LNA in 32nm SOI CMOS technology with autonomic NF calibration," *RFIC*, pp. 319-322, 2015.

[16] V. Natarajan *et al.*, "Yield recovery of RF transceiver systems using iterative tuning-driven power-conscious performance optimization," *IEEE D&T*, vol. 32, no. 1, pp. 61-69, Jan. 2015.

[17] X. Li *et al.*, "Statistical regression for efficient high-dimensional modeling of analog and mixed-signal performance variations," *DAC*, pp. 38-43, 2008.

[18] X. Li, "Finding deterministic solution from underdetermined equation: large-scale performance modeling by least angle regression," *DAC*, pp. 364-369, 2009.

[19] W. Zhang *et al.*, "Toward efficient large-scale performance modeling of integrated circuits via multi-mode/multi-corner sparse regression," *DAC*, pp. 897-902, 2010.

[20] X. Li, "Finding deterministic solution from underdetermined equation: large-scale performance variability modeling of analog/RF circuits," *IEEE TCAD*, vol. 29, no. 11, pp. 1661-1668, Nov. 2010.

[21] X. Li *et al.*, "Large-scale statistical performance modeling of analog and mixed-signal circuits," *CICC*, 2012.

[22] T. McConaghy, "High-dimensional statistical modeling and analysis of custom integrated circuits," *CICC*, pp. 1-8, 2011.

[23] F. Wang *et al.*, "Bayesian model fusion: large-scale performance modeling of analog and mixed-signal circuits by reusing early-stage data," *DAC*, pp. 1-6, 2013.

[24] X. Li *et al.*, "Efficient parametric yield estimation of analog/mixed-signal circuits via Bayesian model fusion," *ICCAD*, pp. 627-634, 2012.

[25] X. Li *et al.*, "Bayesian model fusion: a statistical framework for efficient pre-silicon validation and post-silicon tuning of complex analog and mixed-signal circuits," *ICCAD*, pp. 795-802, 2013.

[26] C. Fang *et al.*, "BMF-BD: Bayesian model fusion on Bernoulli distribution for efficient yield estimation of integrated circuits," *DAC*, 2014.

[27] C. Fang *et al.*, "Efficient bit error rate estimation for high-speed link by Bayesian model fusion," *DATE*, pp. 1024-1029, 2015.

[28] Q. Huang *et al.*, "Efficient multivariate moment estimation via Bayesian model fusion for analog and mixed-signal circuits," *DAC*, 2015.

[29] S. Yu *et al.*, "Bayesian co-training," *NIPS*, pp. 1-8, 2008.

[30] C. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[31] L. Yu *et al.*, "Remembrance of transistors past: compact model parameter extraction using bayesian inference and incomplete new measurements," *DAC*, pp. 1-6, 2014.

[32] W. Zhang *et al.*, "Virtual probe: a statistical framework for low-cost silicon characterization of nanoscale integrated circuits," *IEEE TCAD*, vol. 30, no. 12, pp. 1814-1827, Dec. 2011.

[33] S. Boyd *et al.*, *Convex Optimization*, Cambridge University Press, 2004.