Efficient Performance Modeling via Dual-Prior Bayesian Model Fusion for Analog and Mixed-Signal Circuits

Qicheng Huang[1], Chenlei Fang[1], Fan Yang[1,*], Xuan Zeng[1,*], Dian Zhou[1,2], and Xin Li[1,3]
[1]State Key Lab of ASIC & System, Microelectronics Department, Fudan University, Shanghai, P. R. China
[2]Electrical Engineering Department, University of Texas at Dallas, Richardson, TX, USA
[3]Electrical & Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA, USA

ABSTRACT

In this paper, we propose a novel Dual-Prior Bayesian Model Fusion (DP-BMF) algorithm for performance modeling. Different from the previous BMF methods which use only one source of prior knowledge, DP-BMF takes advantage of multiple sources of prior knowledge to fully exploit the available information and, hence, further reduce the modeling cost. Based on a graphical model, an efficient Bayesian inference is developed to fuse two different prior models and combine the prior information with a small number of training samples to achieve high modeling accuracy. Several circuit examples demonstrate that the proposed method can achieve up to 1.83× cost reduction over the traditional one-prior BMF method without surrendering any accuracy.

1. INTRODUCTION

The continuous scaling of integrated circuits (ICs) leads to severe process variations. These device–level process variations (e.g., ΔV_{th}, ΔT_{ox}, etc.) pose large-scale uncertainties in circuit performances and, hence, impact the parametric yield of analog and mixed-signal (AMS) circuits [1]. To model, analyze and optimize process variations at all levels of design hierarchy, various techniques have been developed for performance modeling during the past decades [2]-[4]. The objective is to describe the performance of interest (e.g., offset of an operational amplifier) by an analytical (e.g., linear, quadratic, etc.) function of device-level variations and/or environmental conditions. Once the performance models are created, they can be applied to various applications such as parametric yield prediction [5] and worst-case corner extraction [6].

Although many performance modeling techniques were developed, the evolution of AMS circuits, especially the increase of circuit size and complexity, has posed a number of new challenges in this area. On one hand, a large number of random variables have to be used to model the process variations associated with large-scale circuits. In consequence, a huge amount of simulation samples must be generated for high-dimensional modeling. On the other hand, the computational cost of circuit simulation increases significantly due to increasingly large circuit size, which makes circuit simulation extremely time-consuming. These recent trends have made performance modeling prohibitively expensive today [7].

To address this challenging issue of modeling cost, several advanced performance modeling techniques (e.g., sparse regression [8], elastic net regularization [9], etc.) have been proposed. In particular, a framework of Bayesian Model Fusion (BMF) was developed for efficient high-dimensional performance modeling [10]-[11]. BMF optimally combines the early-stage (e.g., schematic-level) information and a small number of late-stage (e.g., post-layout) samples via Bayesian inference. The late-stage model coefficients are then determined by maximizing the posterior distribution. An extended version of BMF, referred to as Co-Learning BMF (CL-BMF), was recently proposed to further reduce the modeling cost [12]. CL-BMF trains an extra low-complexity model to generate pseudo samples for fitting a high-complexity performance model. In this way, it greatly reduces the number of required physical samples and, hence, the overall modeling cost.

The aforementioned BMF approaches attempt to exploit only one source of prior knowledge (i.e. early-stage model coefficients). In practice, we can often obtain useful knowledge from multiple sources to facilitate late-stage performance modeling. For example, to model the performance metrics based on post-silicon measurements, we can take advantages of the models fitted by (i) the pre-silicon data collected from simulation and (ii) the post-silicon data measured from a previous tape-out. Consider another important application of modeling the aging behavior for analog circuits. To capture the aged performance metrics at the post-layout stage, we can borrow the prior knowledge from the models fitted by (i) the schematic-level simulation data for the aged performance metrics and (ii) the post-layout simulation data at $t = 0$. These various sources of prior knowledge are expected to provide the useful information that facilitates us to efficiently fit the performance model of interest. Therefore, a new BMF framework must be created to properly fuse multiple sources of prior knowledge for performance modeling.

Towards this goal, we propose a novel BMF technique referred to as Dual-Prior Bayesian Model Fusion (DP-BMF) that takes into account multiple sources of prior knowledge. DP-BMF is derived from the Bayesian inference that can be represented as a graphical model [13]-[14]. The performance model of interest is built by combining multiple prior models and a small number of training samples. As will be demonstrated by our experimental results in Section 5, the proposed method can achieve up to 1.83× cost reduction over the conventional BMF approach.

The reminder of this paper is organized as follows. We briefly review the background of BMF in Section 2 and then derive our DP-BMF method in Section 3. In Section 4, several implementation issues are further discussed. The efficacy of the proposed method is demonstrated by two circuit examples in Section 5. Finally, we conclude in Section 6.

2. BACKGROUND

The performance modeling of an AMS circuit aims to describe certain performances of interests with an analytical function of device-level variations and/or environmental conditions. For example, we can approximate the offset of an operational amplifier as a polynomial function of variables like ΔV_{th}, ΔL, and DC bias current.

[*] Corresponding authors: {yangfan, xzeng}@fudan.edu.cn.

DAC '16, June 05-09, 2016, Austin, TX, USA
© 2016 ACM. ISBN 978-1-4503-4236-0/16/06…$15.00
DOI: http://dx.doi.org/10.1145/2897937.2898014

Generally, the performance model of a given circuit can be described as:

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

where y denotes the performance metric to be estimated, **x** is a vector representing the variations and operation point, f is the performance function. The performance function is a linear combination of M basis functions (e.g., linear or quadratic polynomials) $\{g_m(\mathbf{x});\ m = 1, 2, \ldots, M\}$, and $\{\alpha_m;\ m = 1, 2, \ldots, M\}$ are the model coefficients.

The unknown coefficients in (1) are traditionally determined by solving the following least-squares regression problem:

$$\min_{\boldsymbol{\alpha}} \ \left\| \mathbf{y}_{\mathrm{L}} - \mathbf{G} \cdot \boldsymbol{\alpha} \right\|_2 , \tag{2}$$

where

$$\mathbf{G} = \begin{bmatrix} g_1(\mathbf{x}^{(1)}) & g_2(\mathbf{x}^{(1)}) & \cdots & g_M(\mathbf{x}^{(1)}) \\ g_1(\mathbf{x}^{(2)}) & g_2(\mathbf{x}^{(2)}) & \cdots & g_M(\mathbf{x}^{(1)}) \\ \vdots & \vdots & \vdots & \vdots \\ g_1(\mathbf{x}^{(K)}) & g_2(\mathbf{x}^{(K)}) & \cdots & g_M(\mathbf{x}^{(K)}) \end{bmatrix} \tag{3}$$

$$\boldsymbol{\alpha} = [\alpha_1 \ \alpha_2 \ \cdots \ \alpha_M]^T \tag{4}$$

$$\mathbf{y}_{\mathrm{L}} = [y^{(1)} \ y^{(2)} \ \cdots \ y^{(K)}]^T , \tag{5}$$

In (2)-(5), $\|\bullet\|_2$ stands for the L_2-norm of a vector, K represents the total number of sampling points, and $\mathbf{x}^{(k)}$ and $y^{(k)}$ are the k-th sampling values of **x** and y respectively. $\boldsymbol{\alpha}$ is a vector containing all the model coefficients and $\mathbf{y}_{\mathrm{L}}$ is a vector consisting of all the samples of y. When applied to high-dimensional performance modeling, the least-squares fitting method requires a huge number of sampling points and thus leads to extremely expensive modeling cost.

To address the complexity issue of least-squares fitting, sparse regression method [9] has recently been developed by exploiting the fact that most high-dimension model coefficients are close to zero. However, traditional sparse regression approach only fits the performance model based on the simulation data of a single stage. BMF can be applied to further combine the knowledge from early stage and thus reduce the performance modeling cost.

In the conventional BMF method, the prior knowledge of model coefficients obtained from early stage data (e.g., schematic-level simulation data) is encoded into a nonzero-mean Gaussian prior distribution. With only a few samples from late-stage data (e.g., post-layout simulation data), the estimated late-stage model coefficients $\boldsymbol{\alpha}_L$ can then be derived as:

$$\boldsymbol{\alpha}_{\mathrm{L}} = \left[\eta \cdot \mathbf{D} + \mathbf{G}^T \cdot \mathbf{G} \right]^{-1} \cdot \left[\eta \cdot \mathbf{D} \cdot \boldsymbol{\alpha}_{\mathrm{E}} + \mathbf{G}^T \cdot \mathbf{y}_{\mathrm{L}} \right] , \tag{6}$$

where

$$\boldsymbol{\alpha}_{\mathrm{E}} = [\alpha_{\mathrm{E},1} \ \alpha_{\mathrm{E},2} \ \cdots \ \alpha_{\mathrm{E},M}]^T \tag{7}$$

$$\mathbf{D} = diag\left(\alpha_{\mathrm{E},1}^{-2}, \alpha_{\mathrm{E},2}^{-2}, \cdots, \alpha_{\mathrm{E},M}^{-2} \right) . \tag{8}$$

$\boldsymbol{\alpha}_{\mathrm{E}}$ is the coefficient vector of a model fitted by the early-stage data. **G** and $\mathbf{y}_{\mathrm{L}}$ are defined by (3) and (5) respectively, and $diag(\bullet)$ represents the operator to construct a diagonal matrix. η is a hyper-parameter controlling the confidence in prior knowledge. In the case that η is sufficiently large, (6) can be reduced as:

$$\boldsymbol{\alpha}_{\mathrm{L}} \approx \boldsymbol{\alpha}_{\mathrm{E}} , \tag{9}$$

which reveals that the prior knowledge is very accurate; In another extreme case that η is very small, we have:

$$\boldsymbol{\alpha}_{\mathrm{L}} = \left(\mathbf{G}^T \cdot \mathbf{G} \right)^{-1} \cdot \mathbf{G}^T \cdot \mathbf{y}_{\mathrm{L}} , \tag{10}$$

which implies that the prior knowledge is inaccurate, so the least-squares regression is applied on the late-stage data alone for model fitting. The optimal value of η can be determined by the cross-validation technique [13].

While the aforementioned BMF method proves to achieve significant speedup over least-squares regression, it restricts the prior knowledge to single source. In practice, we often have multiple sources of prior knowledge. It is possible to exploit more correlated information from different aspects to further facilitate the late-stage statistical analysis. With this motivation, we propose a new BMF strategy to utilize two sources of prior knowledge.

3. PROPOSED APPROACH

In this section, we develop our proposed DP-BMF method to borrow prior knowledge from two sources of early-stage data for more efficient performance modeling.

3.1 Problem Formulation

Specifically, we denote the two groups of prior model coefficients as $\boldsymbol{\alpha}_{\mathrm{E},1}$ and $\boldsymbol{\alpha}_{\mathrm{E},2}$, where

$$\boldsymbol{\alpha}_{\mathrm{E},1} = [\alpha_{\mathrm{E},1,1} \ \alpha_{\mathrm{E},1,2} \ \cdots \ \alpha_{\mathrm{E},1,M}]^T , \tag{11}$$

$$\boldsymbol{\alpha}_{\mathrm{E},2} = [\alpha_{\mathrm{E},2,1} \ \alpha_{\mathrm{E},2,2} \ \cdots \ \alpha_{\mathrm{E},2,M}]^T . \tag{12}$$

The two groups of coefficients are obtained by fitting from two different sources of existing data respectively, using the same set of basis functions for late-stage performance modeling. Therefore, we are supposed to know the two groups of coefficients before fitting the late-stage model.

The fundamental problem of the proposed method then becomes: with the input information of (i) two groups of early-stage model coefficients $\{\alpha_{\mathrm{E},1,m};\ m = 1, 2, \ldots, M\}$ and $\{\alpha_{\mathrm{E},2,m};\ m = 1, 2, \ldots, M\}$ and (ii) a few late-stage samples of **x** and y, how to properly estimate the late-stage model coefficients $\boldsymbol{\alpha}$ by borrowing prior knowledge of the two sources. To this end, a Bayesian inference strategy will be constructed and represented by a graphical model.

3.2 Graphical Model and Likelihood Function

We consider three different performance models: two single-prior model $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and the late-stage model $f_c(\mathbf{x})$ we aim to fit:

$$y \approx f_1(\mathbf{x}) = \sum_{m=1}^{M} \alpha_{1,m} \cdot g_m(\mathbf{x}) \tag{13}$$

$$y \approx f_2(\mathbf{x}) = \sum_{m=1}^{M} \alpha_{2,m} \cdot g_m(\mathbf{x}) \tag{14}$$

$$y \approx f_{\mathrm{c}}(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m \cdot g_m(\mathbf{x}) , \tag{15}$$

where $\{\alpha_{1,m};\ m = 1, 2, \ldots, M\}$ and $\{\alpha_{2,m};\ m = 1, 2, \ldots, M\}$ represent the coefficients of two single-prior models, and $\{\alpha_m;\ m = 1, 2, \ldots, M\}$ denote the late-stage model coefficients to be estimated. We assume $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and $f_c(\mathbf{x})$ share the same set of basis functions. The meaning of single-prior models will be explained soon in the details of the graphical model.

As shown in figure 1, a graphical model is constructed to illustrate the main strategy of our method. Each node represents a random quantity, and each directed/undirected edge represents a unidirectional/non-directional dependency. The two small solid circles stand for the two sources of prior knowledge. The filled node indicates that the corresponding data (i.e., the physical samples of **y**) have been observed. We call $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ single-prior models because they aim to predict the late-stage model coefficients based on single source of prior knowledge (i.e. $\boldsymbol{\alpha}_{\mathrm{E},1}$ or $\boldsymbol{\alpha}_{\mathrm{E},2}$) respectively. Our target model $f_c(\mathbf{x})$ then works as a consensus function to balance the two single-prior models and combine the useful information they extract from their respective prior information. We expect that the three models $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and $f_c(\mathbf{x})$ are consistent, since they are supposed to predict the same performance metric, although in different ways.

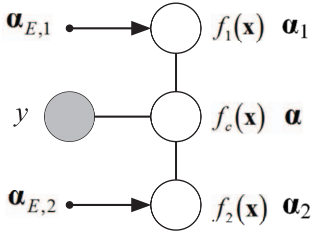
$\boldsymbol{\alpha}_{E,1}$
$f_1(\mathbf{x})$ $\boldsymbol{\alpha}_1$
y
$f_c(\mathbf{x})$ $\boldsymbol{\alpha}$
$\boldsymbol{\alpha}_{E,2}$
$f_2(\mathbf{x})$ $\boldsymbol{\alpha}_2$

Figure 1. A graphical model is shown to illustrate the Bayesian inference strategy of DP-BMF.

According to the graphical model in Figure 1, we can derive the joint probability density function (PDF) as:

$$pdf(f_1,f_2,f_c,y) \propto \exp\left[-\frac{(f_1-f_c)^2}{2\sigma_1^2}\right] \cdot\exp\left[-\frac{(f_2-f_c)^2}{2\sigma_2^2}\right]\cdot\exp\left[-\frac{(y-f_c)^2}{2\sigma_c^2}\right]. \tag{16}$$

Here we assume the distribution of the difference between $< f_1, f_c>$, $< f_2, f_c>$ and $<f_c, y>$ are all zero-mean Gaussian, with σ_1, σ_2 and σ_c to be the corresponding standard deviations. Given a number of independent late-stage samples $\{(\mathbf{x}^{(r)}, y^{(r)});\ r = 1, 2, ..., K\}$, the joint PDF for all the collected samples can be described as:

$$pdf(\mathbf{f}_1,\mathbf{f}_2,\mathbf{f}_c,\mathbf{y}_L) \propto \exp\left[-\frac{\|\mathbf{f}_1-\mathbf{f}_c\|_2^2}{2\sigma_1^2}\right] \cdot\exp\left[-\frac{\|\mathbf{f}_2-\mathbf{f}_c\|_2^2}{2\sigma_2^2}\right]\cdot\exp\left[-\frac{\|\mathbf{y}_L-\mathbf{f}_c\|_2^2}{2\sigma_c^2}\right], \tag{17}$$

where

$$\mathbf{f}_1=\left[f_1\left(\mathbf{x}^{(1)}\right)\ f_1\left(\mathbf{x}^{(2)}\right)\ \cdots\ f_1\left(\mathbf{x}^{(K)}\right)\right]^T, \tag{18}$$

$$\mathbf{f}_2=\left[f_2\left(\mathbf{x}^{(1)}\right)\ f_2\left(\mathbf{x}^{(2)}\right)\ \cdots\ f_2\left(\mathbf{x}^{(K)}\right)\right]^T, \tag{19}$$

$$\mathbf{f}_c=\left[f_c\left(\mathbf{x}^{(1)}\right)\ f_c\left(\mathbf{x}^{(2)}\right)\ \cdots\ f_c\left(\mathbf{x}^{(K)}\right)\right]^T, \tag{20}$$

and $\mathbf{y}_L$ is defined as (5).

According to (13)-(15), $\mathbf{f}_1$, $\mathbf{f}_2$ and $\mathbf{f}_c$ can be re-written as:

$$\mathbf{f}_1=\mathbf{G}\cdot\boldsymbol{\alpha}_1 \tag{21}$$

$$\mathbf{f}_2=\mathbf{G}\cdot\boldsymbol{\alpha}_2 \tag{22}$$

$$\mathbf{f}_c=\mathbf{G}\cdot\boldsymbol{\alpha}, \tag{23}$$

where $\mathbf{G}$ and $\boldsymbol{\alpha}$ are defined as (3) and (4) respectively, and

$$\boldsymbol{\alpha}_1=[\alpha_{1,1}\ \ \alpha_{1,2}\ \ \cdots\ \ \alpha_{1,M}]^T, \tag{24}$$

$$\boldsymbol{\alpha}_2=[\alpha_{2,1}\ \ \alpha_{2,2}\ \ \cdots\ \ \alpha_{2,M}]^T. \tag{25}$$

Then, by substituting (21)-(23) into the joint PDF (17), we can get the likelihood function:

$$pdf(\mathbf{y}_L|\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}) \propto \exp\left[-\frac{\|\mathbf{G}\cdot\boldsymbol{\alpha}_1-\mathbf{G}\cdot\boldsymbol{\alpha}\|_2^2}{2\sigma_1^2}\right] \cdot\exp\left[-\frac{\|\mathbf{G}\cdot\boldsymbol{\alpha}_2-\mathbf{G}\cdot\boldsymbol{\alpha}\|_2^2}{2\sigma_2^2}\right]\cdot\exp\left[-\frac{\|\mathbf{y}_L-\mathbf{G}\cdot\boldsymbol{\alpha}\|_2^2}{2\sigma_c^2}\right]. \tag{26}$$

The first two L2-norm terms represent the differences between the target model $\mathbf{f}_c$ and two single-prior models $\mathbf{f}_1$ and $\mathbf{f}_2$ respectively, and the third L2-norm term stands for the modeling error of $\mathbf{f}_c$, as compared to the observed samples of y.

3.3 Prior Distribution Definition

Since $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ aim to estimate the late-stage model coefficients based on prior knowledge $\boldsymbol{\alpha}_{E,1}$ and $\boldsymbol{\alpha}_{E,2}$ respectively, we expect that the model coefficients $\{\alpha_{1,m};\ m = 1, 2, \ldots, M\}$ are close to $\{\alpha_{E,1,m};\ m = 1, 2, \ldots, M\}$, and the coefficients $\{\alpha_2;\ m = 1, 2, \ldots, M\}$ are close to $\{\alpha_{E,2,m};\ m = 1, 2, \ldots, M\}$. Therefore, we construct two nonzero-mean Gaussian distributions for each of the two sets of coefficients:

$$pdf\left(\alpha_{1,m}\right)\sim Gauss\left(\alpha_{E,1,m},\ k_1\cdot\alpha_{E,1,m}^2\right)\quad (m=1,\ 2,\ \cdots,\ M), \tag{27}$$

$$pdf\left(\alpha_{2,m}\right)\sim Gauss\left(\alpha_{E,2,m},\ k_2\cdot\alpha_{E,2,m}^2\right)\quad (m=1,\ 2,\ \cdots,\ M). \tag{28}$$

k_1 and k_2 are two hyper-parameters that can be determined by cross-validation as will be discussed in detail in Section 4. In the prior distribution (27), we assume $pdf(\alpha_{1,m})$ is peaked at its mean value $\alpha_{1,m} = \alpha_{E,1,m}$, implying the early-stage coefficient $\alpha_{E,1,m}$ and the late-stage $\alpha_{1,m}$ are likely to be similar. Also, the standard deviation of $pdf(\alpha_{1,m})$ is assumed to be proportional to $|\alpha_{E,1,m}|$, which provides each late-stage coefficient $\alpha_{1,m}$ with a relatively equal opportunity to deviate from the corresponding early-stage coefficient $\alpha_{E,1,m}$. The prior distribution $pdf(\alpha_{2,m})$ in (28) can be interpreted in a similar way.

Then, we further assume that all late-stage model coefficients are statistically independent. In this way, we can encode the prior knowledge of $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$, and $\boldsymbol{\alpha}$ as the joint PDF function $pdf(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha})$:

$$pdf(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}) \propto \exp\left[-\frac{1}{2}\left(\begin{bmatrix}\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_{E,1}\\ \boldsymbol{\alpha}_2-\boldsymbol{\alpha}_{E,2}\end{bmatrix}^T\cdot\begin{bmatrix}\mathbf{D}_1 & 0\\ 0 & \mathbf{D}_2\end{bmatrix}\cdot\begin{bmatrix}\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_{E,1}\\ \boldsymbol{\alpha}_2-\boldsymbol{\alpha}_{E,2}\end{bmatrix}\right)\right]\cdot 1, \tag{29}$$

where

$$\mathbf{D}_1=k_1\cdot diag\left(\alpha_{E,1,1}^{-2},\alpha_{E,1,2}^{-2},\cdots,\alpha_{E,1,M}^{-2}\right), \tag{30}$$

$$\mathbf{D}_2=k_2\cdot diag\left(\alpha_{E,2,1}^{-2},\alpha_{E,2,2}^{-2},\cdots,\alpha_{E,2,M}^{-2}\right). \tag{31}$$

Since we have no prior knowledge for $f_c(\mathbf{x})$ and its coefficients $\boldsymbol{\alpha}$, the contribution of $\boldsymbol{\alpha}$ to the joint distribution is represented as a "1" at the end of (29).

3.4 Maximum-A-Posteriori Estimation

Once the prior distribution $pdf(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha})$ is defined by (29), we can combine it with K late-stage physical samples $\{(\mathbf{x}^{(r)}, y^{(r)});\ r = 1, 2, ..., K\}$ and estimate the optimal values of late-stage model coefficients $\boldsymbol{\alpha}$ by maximum-a-posteriori (MAP) estimation.

Based on Bayes' theorem, the posterior distribution is proportional to the prior distribution $pdf(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha})$ multiplied by the likelihood function $pdf(\mathbf{y}_L|\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha})$:

$$pdf(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}|\mathbf{y}_L) \propto pdf(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha})\cdot pdf(\mathbf{y}_L|\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}). \tag{32}$$

MAP attempts to find the optimal values of $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}$ to maximize the posterior distribution $pdf(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}|\mathbf{y}_L)$. Namely, it aims to find the solutions $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}$ that are most likely to occur according to the posterior distribution, although we actually only care about the value of $\boldsymbol{\alpha}$ as it contains the coefficients of the target model.

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

$$\max_{\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}}\ pdf(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}|\mathbf{y}_L). \tag{33}$$

Substituting (32) by (26) and (29) and taking the logarithm for the posterior distribution, we can convert (33) to the following equivalent optimization problem:

$$\min_{\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}}\ h(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}), \tag{34}$$

where

$$h(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha})=\frac{\|\mathbf{X}\cdot\boldsymbol{\alpha}_1-\mathbf{X}\cdot\boldsymbol{\alpha}\|_2^2}{\sigma_1^2}+\frac{\|\mathbf{X}\cdot\boldsymbol{\alpha}_2-\mathbf{X}\cdot\boldsymbol{\alpha}\|_2^2}{\sigma_2^2}+\frac{\|\mathbf{y}_{\mathrm{L}}-\mathbf{X}\cdot\boldsymbol{\alpha}\|_2^2}{\sigma_c^2}$$
$$+k_1\cdot(\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_{\mathrm{E},1})^T\cdot\mathbf{D}_1\cdot(\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_{\mathrm{E},1})+k_2\cdot(\boldsymbol{\alpha}_2-\boldsymbol{\alpha}_{\mathrm{E},2})^T\cdot\mathbf{D}_2\cdot(\boldsymbol{\alpha}_2-\boldsymbol{\alpha}_{\mathrm{E},2}) . \quad (35)$$

In this cost function, the first two terms penalize the discrepancy between the prediction results of the target model $f_c(\mathbf{x})$ and the other two single-prior models $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. The third term represents the error of traditional least-squares fitting. The last two terms penalize the differences between the model coefficients of $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and their prior knowledge $\boldsymbol{\alpha}_{\mathrm{E},1}$, $\boldsymbol{\alpha}_{\mathrm{E},2}$, respectively. We can see the cost function aims to compromise among three parts: $f_c(\mathbf{x})$'s prediction error, its similarity to the other two single-prior models and the similarity of the single-prior models to their perspective prior knowledge.

Although our target is to get the solution of $\boldsymbol{\alpha}$ in (34), the model coefficients $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}$ are solved together. By taking partial derivatives of $h(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha})$ with respect to $\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$, $\boldsymbol{\alpha}$ and then setting them to zero, we can get the MAP estimation of the target model coefficients $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha}_{\mathrm{L}}=\mathbf{M}^{-1}\cdot\mathbf{b} , \quad (36)$$

$$\mathbf{M}=\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}+\frac{1}{\sigma_c^2}\right)\cdot I-\frac{1}{\sigma_1^4}\cdot\left(\frac{\mathbf{G}^T\cdot\mathbf{G}}{\sigma_1^2}+k_1\cdot\mathbf{D}_1\right)^{-1}\cdot\mathbf{G}^T\cdot\mathbf{G}$$
$$-\frac{1}{\sigma_2^4}\cdot\left(\frac{\mathbf{G}^T\cdot\mathbf{G}}{\sigma_2^2}+k_2\cdot\mathbf{D}_2\right)^{-1}\cdot\mathbf{G}^T\cdot\mathbf{G} , \quad (37)$$

$$\mathbf{b}=\frac{1}{\sigma_1^2}\cdot\left(\frac{\mathbf{G}^T\mathbf{G}}{\sigma_1^2}+k_1\cdot\mathbf{D}_1\right)^{-1}\cdot k_1\cdot\mathbf{D}_1\cdot\boldsymbol{\alpha}_{\mathrm{E},1}$$
$$+\frac{1}{\sigma_2^2}\left(\frac{\mathbf{G}^T\mathbf{G}}{\sigma_2^2}+k_2\cdot\mathbf{D}_2\right)^{-1}\cdot k_2\cdot\mathbf{D}_2\cdot\boldsymbol{\alpha}_{\mathrm{E},2}+\frac{1}{\sigma_c^2}\cdot(\mathbf{G}^T\mathbf{G})^{-1}\cdot\mathbf{G}^T\cdot\mathbf{y}_{\mathrm{L}} . \quad (38)$$

From (36)-(38) we can see the estimated result is controlled by five hyper-parameters σ_1, σ_2, σ_c, k_1 and k_2. These parameters control the balance in two aspects: (i) the balance between trusting prior knowledge ($\boldsymbol{\alpha}_{\mathrm{E},1}$ and $\boldsymbol{\alpha}_{\mathrm{E},2}$) and trusting late-stage samples in $\mathbf{y}_{\mathrm{L}}$; (ii) the balance between the confidence in two sources of prior knowledge. To calculate the coefficient vector α_{L} in (36), these hyper-parameters must be carefully determined. In practice, only three of them are independent and we can determine their values by a two-dimensional cross-validation process. In Section 4.1, we will discuss in detail about the respective influence of these hyper-parameters and the method to find their optimal values.

4. IMPLEMENTATION ISSUES

To make the proposed DP-BMF method practically efficient, we also need to consider several implementation issues. In this section, we discuss these issues in detail, including (i) the influence of hyper-parameters and how to determine their optimal values, and (ii) the detection method of two highly biased sources of prior knowledge.

4.1 Hyper-Parameters

The hyper-parameters σ_1, σ_2, σ_c, k_1 and k_2 in (37)-(38) control the trust in prior information $\boldsymbol{\alpha}_{\mathrm{E},1}$, prior information $\boldsymbol{\alpha}_{\mathrm{E},2}$ and the late-stage data samples in $\mathbf{y}_{\mathrm{L}}$. They should be carefully determined so that we can properly exploit the information from prior knowledge and data samples to get an accurate estimation result.

In fact, σ_1, σ_2, σ_c are not independent. As shown in Figure 2, since we suppose the distributions of differences between $<f_1, f_c>$ and $<f_c, y>$ are all zero-mean Gaussian with variance σ_1^2 and σ_c^2 respectively, the distribution of difference between f_1 and y is then zero-mean Gaussian with variance γ_1:

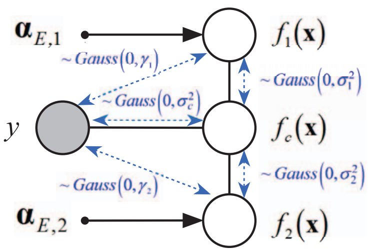
$\boldsymbol{\alpha}_{E,1}$
$\sim Gauss(0,\gamma_1)$
$\sim Gauss(0,\sigma_c^2)$
y
$\sim Gauss(0,\gamma_2)$
$\boldsymbol{\alpha}_{E,2}$
$f_1(\mathbf{x})$
$\sim Gauss(0,\sigma_1^2)$
$f_c(\mathbf{x})$
$\sim Gauss(0,\sigma_2^2)$
$f_2(\mathbf{x})$

Figure 2. The distribution of the differences between models and observed data.

$$\gamma_1=\sigma_1^2+\sigma_c^2 . \quad (39)$$

By running the conventional single-prior BMF in Section 2 with prior information $\boldsymbol{\alpha}_{\mathrm{E},1}$ and late-stage samples, we can then estimate the value of γ_1 from the variance of modeling error. Similarly, the distribution of difference between f_2 and y is another zero-mean Gaussian distribution with variance γ_2:

$$\gamma_2=\sigma_2^2+\sigma_c^2 . \quad (40)$$

The value of γ_2 can be estimated by running another single-prior BMF with prior information $\boldsymbol{\alpha}_{\mathrm{E},2}$. Once we get the estimated values of γ_1 and γ_2, both σ_1 and σ_2 can be uniquely determined by σ_c. Thus, overall we only need to find the optimal values for three independent hyper-parameters σ_c, k_1 and k_2.

Now we discuss the influence of the hyper-parameters σ_c, k_1 and k_2 by considering several extreme cases:

***Case* 1:** k_1 and k_2 are small enough (close to zero)

In this case, the terms concerning k_1 and k_2 in (37)-(38) can all be eliminated and we can then reduce (36) to:

$$\boldsymbol{\alpha}_{\mathrm{L}}\approx(\mathbf{G}^T\cdot\mathbf{G})^{-1}\cdot\mathbf{G}^T\cdot\mathbf{y}_{\mathrm{L}} , \quad (41)$$

which is exactly the least-squares estimation. It indicates that both sources of prior knowledge are inaccurate, so the model coefficients are fitted simply by the late-stage data samples.

***Case* 2:** $k_1 >> k_2$, and k_2 is close to zero

In this case, (37)-(38) can be reduced and transformed to:

$$\mathbf{M}=\left(\frac{\sigma_c^2}{\gamma_1-\sigma_c^2}+1\right)\cdot I-\frac{\sigma_c^2}{(\gamma_1-\sigma_c^2)^2}\cdot\left(\frac{\mathbf{G}^T\cdot\mathbf{G}}{\gamma_1-\sigma_c^2}+k_1\cdot\mathbf{D}_1\right)^{-1}\cdot\mathbf{G}^T\cdot\mathbf{G} , \quad (42)$$

$$\mathbf{b}=\frac{\sigma_c^2}{\gamma_1-\sigma_c^2}\cdot\left(\frac{\mathbf{G}^T\cdot\mathbf{G}}{\gamma_1-\sigma_c^2}+k_1\cdot\mathbf{D}_1\right)^{-1}\cdot k_1\cdot\mathbf{D}_1\cdot\boldsymbol{\alpha}_{\mathrm{E},1}+(\mathbf{G}^T\cdot\mathbf{G})^{-1}\cdot\mathbf{G}^T\cdot\mathbf{y}_{\mathrm{L}} , \quad (43)$$

which then yields:

$$\boldsymbol{\alpha}_{\mathrm{L}}\approx\boldsymbol{\alpha}_{\mathrm{E},1}, \qquad \text{when } \frac{\sigma_c^2}{\gamma_1-\sigma_c^2}\gg 1 ; \quad (44)$$

$$\boldsymbol{\alpha}_{\mathrm{L}}\approx(\mathbf{G}^T\cdot\mathbf{G})^{-1}\cdot\mathbf{G}^T\cdot\mathbf{y}_{\mathrm{L}}, \quad \text{when } \frac{\sigma_c^2}{\gamma_1-\sigma_c^2}\ll 1 . \quad (45)$$

From these cases, we can interpret k_1 and k_2 as the trust in prior information $\boldsymbol{\alpha}_{\mathrm{E},1}$ and $\boldsymbol{\alpha}_{\mathrm{E},2}$ respectively. With larger value of k_1 (or k_2), more weight is assigned to $\boldsymbol{\alpha}_{\mathrm{E},1}$ (or $\boldsymbol{\alpha}_{\mathrm{E},2}$) in estimation. The ratio of k_1 and k_2 then controls the balance between two sources of prior knowledge. That is why we have (41) when k_1 and k_2 are both small and have (44) in the case that k_1 is obviously larger than k_2. On the other hand, σ_c^2 can be interpreted as the distrust in late-stage samples. Intuitively, large σ_c^2 implies small σ_1^2 (or σ_2^2), which then indicates that f_c is much closer to f_1 (or f_2) than to the observed samples of y. Therefore, the estimation result is similar to the prior information, as shown in (44). Small σ_c^2 then implies that the estimation based on observed samples of y is accurate, so the estimation result tends to largely rely on late-stage samples as (45) shows.

It is important to find the optimal values of the hyper-parameters σ_c, k_1 and k_2 to minimize the modeling error. Towards this goal, we first set the value of σ_c^2 as:

$$\sigma_c^2 = \lambda \cdot \min(\gamma_1, \gamma_2) \tag{46}$$

where λ is a scale factor between 0 and 1, since we can see from (39)-(40) that σ_c^2 should be no more than γ_1 or γ_2. In practice, we set λ close to 1. It is because the number of late-stage samples is far less than the number of model coefficients, so that simply estimating from late-stage samples would be very inaccurate, which then leads to large value of σ_c^2.

Then we use two-dimensional cross-validation [15] to find the optimal values for k_1 and k_2 based on few late-stage samples. All combinations of k_1 and k_2 within a pre-defined range are chosen as candidates. For each combination, we apply a Q-fold cross-validation strategy. We divide the entire set of data samples into Q groups and modeling error is estimated from Q independent runs. At each run, Q-1 groups are used to calculate the model coefficients and the remaining group is used to estimate the modeling error. Different groups are selected for error estimation in different runs. After a complete cross-validation process of Q runs, we then calculate the average error of the Q modeling errors and use it to indicate the estimation accuracy of the given hyper-parameters k_1 and k_2. The combination with the least modeling error is then selected as the optimal values of k_1 and k_2.

4.2 Highly Biased Prior Knowledge

Given two sources of early-stage information each of which can facilitate the late-stage performance modeling, we still need to consider the scenario when there is great competence disparity between the two sources of prior knowledge. Namely, when one source of prior knowledge provides far more useful information for coefficient estimation than the other one does.

Ideally, the two sources of prior knowledge are equally competent and complementary to each other. Thus, we can extract more useful information for modeling and achieve higher estimation accuracy than using single source. With two highly biased sources, however, we can expect that cross-validation will automatically assign an extremely small weight value to the less useful prior information. When $k_1 >> k_2$, for example, the estimation result is then similar to (44). In this case, modeling by borrowing two sources of prior knowledge cannot achieve better result than using single prior knowledge $\boldsymbol{\alpha}_{E,1}$, since our result is always a compromise between two sources and one of them now becomes a pure hindrance.

Fortunately, such situation can be easily detected from two signs. The first sign is the values of γ_1 or γ_2 after applying two times of single-prior BMF. For example, if γ_1 is much larger than γ_2, then obviously its corresponding prior knowledge $\boldsymbol{\alpha}_{E,1}$ is far less useful than $\boldsymbol{\alpha}_{E,2}$. The second sign is the ratio of k_1 to k_2 after the two-dimensional cross-validation process. If the ratio is extremely high, for instance, then it implies we should trust much more in $\boldsymbol{\alpha}_{E,1}$ than in $\boldsymbol{\alpha}_{E,2}$ because $\boldsymbol{\alpha}_{E,2}$ is useless. If both signs show that the two sources of prior knowledge are highly biased, then DP-BMF cannot do any better than traditional single-prior BMF with the more competent source as prior knowledge.

4.3 Summary

Algorithm 1 summarizes the major steps of our proposed DP-BMF method for performance modeling from two sources of prior knowledge. Starting from two groups of given early-stage model coefficients as prior knowledge and a set of late-stage samples, we first determine the values of hyper-parameters by the method in Section 4.1. The hyper-parameters control the weights of each source of prior knowledge and information from late-stage samples. Once the optimal values of hyper-parameters are obtained, we estimate the late-stage model coefficients $\boldsymbol{\alpha}_L$ based on MAP.

Algorithm 1:Dual-Prior BMF for Performance Modeling

1. Start from two groups of existing coefficients $\{\alpha_{E,1,m};\ m = 1, 2, \ldots, M\}$ and $\{\alpha_{E,2,m};\ m = 1, 2, \ldots, M\}$ and a set of late-stage samples $\{(\mathbf{x}^{(r)}, y^{(r)});\ r = 1, 2, ..., K\}$.
2. Run single-prior BMF method as defined in Section 2 twice, with $\{\alpha_{E,1,m};\ m = 1, 2, \ldots, M\}$ and $\{\alpha_{E,2,m};\ m = 1, 2, \ldots, M\}$ as prior knowledge respectively. Then estimate γ_1 and γ_2 from the fitting error of each time.
3. Determine the value of hyper-parameter σ_c by (45) and the values of k_1 and k_2 by two-dimensional cross-validation. Then calculate the values of σ_1 and σ_2 according to (39) and (40) respectively.
4. Estimate the late-stage model coefficients α_L by (36)-(38).

5. NUMERICAL EXAMPLES

In this section, we use two circuit examples to demonstrate the efficiency of our proposed DP-BMF method. Our objective is to build performance models for pre-silicon verification of these circuits. To illustrate the improvement by using multiple sources of prior knowledge, we compare three different performance modeling methods: (i) single-prior BMF using one source of prior information, (ii) single-prior BMF using another source of prior information, and (iii) the proposed DP-BMF using both sources of prior information. All experiments are performed on a server with 2.5GHz dual-core CPU and 16GB memory.

5.1 Operational Amplifier

In this example, we use a two-stage operational amplifier (Op-amp) designed in a 45nm CMOS process. We consider the post-layout verification as late-stage and generate the data samples by post-layout simulation. The first source of prior knowledge is from least-squares fitting from many existing data samples of schematic-level simulations. On the other hand, by exploiting the underlying sparsity of model coefficients, we apply the sparse regression method [8] on a small set of obtained post-layout samples to get a group of coefficients as the second source of prior knowledge. We aim to accurately estimate the model coefficients of certain performance metrics in post-layout simulation by borrowing the prior knowledge of two groups of the given coefficients. It is noted that the sources of prior knowledge are not restricted to these we use. Other correlated information from simulation/measurement data of different working modes, different environment corners or previous time can also be reused as prior knowledge.

In this example, we use 581 independent random variables to model the device-level process variations, including both inter-die variations and random mismatches. Our objective is to approximate the offset of the Op-amp as a linear function of these 581 random variables.

For testing and comparison purpose, we generate a set of Monte-Carlo samples by both schematic-level and post-layout simulations, in which the device-level variations of all transistors are considered. We use Monte-Carlo samples of schematic-level simulation to calculate the model coefficients as prior information 1, and apply the sparse regression method on 80 samples of post-layout simulation to get another group of coefficients as prior information 2. A group of another 2000 post-layout simulation samples is used as test group to measure the modeling error.

Figure 4 shows the modeling error as a function of the number of late-stage samples. The errors are calculated from 50 repeated runs based on independent samples to average out random fluctuations. Note that the DP-BMF method achieves higher accuracy than traditional BMF using single source of prior information. In particular, DP-BMF only needs 120 samples to achieve high modeling accuracy. However, even the single-prior BMF with better performance (denoted as Single-prior 1) takes about 220 samples to achieve the same accuracy. Studying the plot reveals that DP-BMF achieves more than 1.83× cost reduction over single-prior BMF.

We also examine whether DP-BMF can effectively adjust the weights assigned to each source of prior knowledge. In Figure 4, traditional BMF achieves higher modeling accuracy with the first source of prior knowledge, which means that the first source provides more useful information than the second one. This is reflected in the values of hyper-parameters k_1 and k_2 determined by cross-validation. The optimized ratio of k_2 to k_1 is relatively small for all the different sample numbers (e.g., $k_2/k_1 = 0.1$ when post-stage sample number is 140). This implies that we trust the first source of prior knowledge more because it provides more useful information.

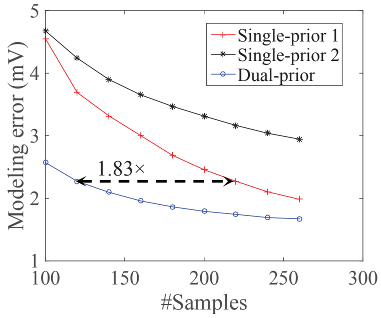

Figure 4. The modeling error is plotted as a function of the number of late-stage samples.

5.2 Analog to Digital Converter

In this example, we consider a flash analog to digital converter (ADC) in a 0.18μm CMOS process. The sources of prior information are the same as the previous example. We use 132 independent random variables to model the device-level process variations. The objective is to approximate the power of the ADC as a linear function of these 132 random variables.

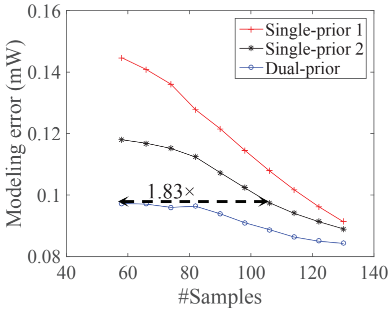

Figure 5. The modeling error is plotted as a function of the number of late-stage samples.

We use Monte-Carlo samples of schematic-level simulation to calculate the model coefficients as prior information 1, and apply the sparse regression method on 50 samples of post-layout simulation to get another group of coefficients as prior information 2. A group of another 2000 post-layout simulation samples is used as test group to measure the modeling error.

Figure 5 shows the modeling error as a function of the number of late-stage samples. Studying the plot reveals that DP-BMF achieves more than 1.83× cost reduction over single-prior BMF. In addition, in Figure 5 we can see that the second source of prior knowledge provides more useful information than the first one. That is why the optimized ratio of k_2 to k_1 is obviously larger than 1 for all the different sample numbers (e.g., $k_2/k_1 = 4.42$ when post-stage sample number is 58). This implies that we trust the second source of prior knowledge more because it is more useful.

6. CONCLUSIONS

In this paper, we propose a novel performance modeling algorithm DP-BMF. To achieve high modeling accuracy with low modeling cost, an efficient Bayesian inference is developed to fuse two different prior models and combine the prior information with a small number of training samples. Several circuit examples demonstrate that the proposed method can achieve up to 1.83× cost reduction over the traditional method without surrendering any accuracy.

7. ACKNOWLEDGEMENTS

This work is supported partly by National Natural Science Foundation of China research project 61574046, 61474026, 61376040 and 61574044, partly by the Recruitment Program of Global Experts (the Thousand Talents Plan), partly by Chen Guang project supported by Shanghai Municipal Education Commission and Shanghai Education Development Foundation and National Science Foundation under contract CCF-1316363 and grant 1115556.

8. REFERENCES

[1] Semiconductor Industry Associate. (2011). International Technology Roadmap for Semiconductors [Online]. Available: www.iers.net/Links/2011ITRS/Home2011.htm.

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

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

[4] T. McConaghy, et al., “Template-free symbolic performance modeling of analog circuits via canonical-form functions and genetic programming,” *IEEE Trans. on CAD*, vol. 28, no. 8, pp. 1162-1175, 2009.

[5] X. Li, et al., “Asymptotic probability extraction for nonnormal performance distributions,” *IEEE Trans. on CAD*, vol. 26, no. 1, pp. 16- 37, 2007.

[6] M. Sengupta, et al., “Application-specific worst case corners using response surfaces and statistical models.” Computer-Aided Design of Integrated Circuits and Systems, *IEEE Trans. on CAD*, vol. 24, no. 9, pp. 1372- 1380, 2005.

[7] V. Natarajan, et al.,“Yield recovery of RF transceiver systems using iterative tuning-driven power-conscious performance optimization,” *IEEE Design & Test of Comp.*, vol. 32, no. 1, pp. 61-69, 2015.

[8] X. Li, “Finding deterministic solution from underdetermined equation: large-scale performance variability modeling of analog/RF circuits,” *IEEE Trans. on CAD*, vol. 29, no. 11, pp. 1661-1668, 2010.

[9] T. McConaghy, “High-dimensional statistical modeling and analysis of custom integrated circuits,” *IEEE CICC*, pp. 1-8, 2011

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

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

[12] F. Wang, et al., “Co-learning Bayesian model fusion: efficient performance modeling of analog and mixed-signal circuits using side information,” *IEEE ICCAD*, pp. 575-582, 2015.

[13] S. Yu, et al., “Bayesian co-training,” NIPS, pp. 1-8, 2008.

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

[15] Q. Huang, et al, “Efficient multivariate moment estimation via Bayesian model fusion for analog and mixed-signal circuits,” *IEEE DAC*, pp. 169:1-169:6, 2015.