S^2-PM: Semi-Supervised Learning for Efficient Performance Modeling of Analog and Mixed Signal Circuits

Mohamed Baker Alawieh, Xiyuan Tang and David Z. Pan
ECE Department, University of Texas at Austin, Austin, TX, USA
{mohdbaker,xitang,dpan}@utexas.edu

Abstract— As integrated circuit technologies continue to scale, variability modeling is becoming more crucial yet, more challenging. In this paper, we propose a novel performance modeling method based on semi-supervised co-learning. We exploit the multiple representations of process variation in any analog and mixed signal circuit to establish a co-learning framework where unlabeled samples are leveraged to improve the model accuracy without enduring any simulation cost. Practically, our proposed method relies on a small set of labeled data, and the availability of no-cost unlabeled data to efficiently build accurate performance model for any analog and mixed signals circuit design. Our numerical experiments demonstrate that the proposed approach achieves up to 30% reduction in simulation cost compared to the state-of-the-art modeling technique without surrendering any accuracy.

I. Introduction

With the continuous scaling of integrated circuit (IC) technologies, the challenges associated with retaining robustness of state-of-art designs continue to exacerbate [1]. At deep sub-micron technologies, process variation prevails among the most prominent factors limiting the product yield of analog and mixed-signal (AMS) circuits [2]. Thus, it is indispensable to consider this variation in the design flow of modern ICs [1]-[2]. Conventionally, performance modeling has been adopted to capture this variability through analytical models that can be used in various applications such as yield estimation [5]-[7] and design optimization [8]-[9].

In practice, achieving highly-accurate performance models is associated with high simulation cost, and as the complexity and size of the ICs increase, so does this cost [10]. Hence, it is becoming prohibitively expensive to collect enough simulation data to be used in traditional performance modeling frameworks.

ASPDAC '19, January 21-24, 2019, Tokyo, Japan ©2019 Association for Computing Machinery. ACM ISBN 978-1-4503-6007-4/19/01$15.00
https://doi.org/10.1145/3287624.3287657

In literature, different performance modeling approaches have been proposed to address this challenge [11]-[17]. Towards reducing the required number of samples, and hence, the simulation cost, recent performance modeling frameworks have incorporated knowledge about model coefficients based on prior assumption and/or historical data into the modeling framework. For instance, sparse regression exploits the assumption that most coefficients are close to zero to effectively build accurate models [11]-[13]. On the other hand, Bayesian model fusion (BMF) takes advantage of an early-stage model to efficiently build a model for a later stage [14]-[17]. Moreover, co-learning BMF proposed in [18] leverages performance side information to further reduce modeling cost. In this approach, a set of partially labeled data, where simulation samples are collected for an alternative performance metric, are fused with labeled samples collected for the performance of interest (PoI) to build an accurate model of the PoI.

From the data learning perspective, all the aforementioned techniques can be categorized as supervised learning. Particularly, the performance models are built using labeled, or partially labeled samples only. Recently, an approach to incorporate unlabeled data into the performance modeling framework was proposed in [19]. This approach makes use of the hierarchical structure of ICs to incorporate unlabeled data via Bayesian co-learning. The key idea is to partition the circuit into multiple blocks where block level performance metrics are defined. Then, low-dimensional performance models for block-level metrics can be combined with a set of unlabeled samples, and a small number of labeled samples to efficiently build accurate performance models.

Despite the fact that this approach can significantly reduce the modeling cost, its application is limited to hierarchical circuits where a block-level partitioning is possible. In practice, even for some hierarchical circuits, such partitioning may not be intuitive. Hence, and motivated by the important role the unlabeled data has proven to play, we will show in this paper that a novel general-purpose semi-supervised performance modeling framework can be derived without any assumption about the AMS circuit structure.

We propose a novel co-learning technique that leverages multiple views of the process variability to efficiently build a performance model. In fact, one of the interesting features of this modeling task is the availability of two *views* or

representations of the variability for each device in the circuit. The first is the device level variations such as $\Delta_{V_{TH}}$ or $\Delta_{w_{eff}}$, while the second view is the underlying set of independent random variables, referred to as process variables (PV), that are the seed for the device level variation. In particular, performance modeling targets expressing the PoI as an analytical function of PV because of the appealing independence characteristics and the sparsity feature these variables possess. In our approach, we capitalize on the information provided by the device level variability as an alternative view to efficiently build the performance model for the PoI.

Our proposed performance modeling starts from the bootstrapping co-learning approach originally proposed in [23] for semi-supervised classification to derive a novel semi-supervised regression framework that makes use of the two redundant data representation available at hand. The key idea is to use a small number of labeled samples to build an initial model for each of the views of the data, then attempt to iteratively bootstrap from the initial models using unlabeled data. In other words, initial models can be used to give pseudo labels for unlabeled data, then the most confident predictions from a particular model are used as pseudo samples for the other model. In each iteration step, *highly-confident* pseudo samples are fused with the small number of available labeled samples to build a new model.

A key component of the proposed framework is estimating the confidence metric for choosing pseudo samples. While this task is relatively easy in a classification framework, it is more challenging in regression. In this work, we derive a mathematical formulation to compute the confidence metric through propagation of modeling error [20]-[22].

- We propose an efficient performance modeling framework for AMS circuits based on semi-supervised co-learning.
- We leverage the multiple views of the variability in the AMS circuits to generate pseudo sample at almost zero cost to reduce the number of needed labeled samples.
- We propose a statistical metric for selecting pseudo samples based on the modeling error propagation.
- The experimental results demonstrates achieving up to 30% reduction in the simulation cost compared to conventional methods [21] without surrendering any accuracy.

The remainder of this paper is organized as follows. In Section II we review the technical background and then present the proposed approach in Section III. Section IV presents numerical results demonstrating the efficacy of our method, and conclusions are presented in Section V.

II. Background

A. Performance Modeling

Mathematically, a performance model approximates a circuit-level PoI (e.g. gain, power) as an analytical function of the process variables:

$$y \approx f_1(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m . b_m(\mathbf{x}) \tag{1}$$

where y is the PoI, $\mathbf{x}$ is a vector containing the PV, $f_1(\mathbf{x})$ is the modeling function, $\{\alpha_m; m = 1, 2, \ldots, M\}$ contains the model coefficients, $\{b_m; m = 1, 2, \ldots, M\}$ contains the basis functions, and M denotes the total number of basis functions.

Given a set of samples, the model coefficients in (1) are usually obtained through least-squares fitting by solving the following optimization problem [20], [22]:

$$\min_{\boldsymbol{\alpha}} ||\mathbf{y} - \mathbf{B}.\boldsymbol{\alpha}||_2^2 \tag{2}$$

where $||\bullet||_2$ is the L_2-norm of a vector, and

$$\mathbf{B} = \begin{bmatrix} b_1(\mathbf{x}^{(1)}) & b_2(\mathbf{x}^{(1)}) & \ldots & b_M(\mathbf{x}^{(1)}) \\ \vdots & \vdots & \ddots & \vdots \\ b_1(\mathbf{x}^{(N)}) & b_2(\mathbf{x}^{(N)}) & \ldots & b_M(\mathbf{x}^{(N)}) \end{bmatrix} \tag{3}$$

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & \ldots & \alpha_M \end{bmatrix}^T \tag{4}$$

$$\mathbf{y} = \begin{bmatrix} y^{(1)} & y^{(2)} & \ldots & y^{(N)} \end{bmatrix}^T . \tag{5}$$

In (3)-(5), N is the total number of samples, and $\mathbf{x}^{(n)}$ and $y^{(n)}$ are the values of $\mathbf{x}$ and y at the $n-$th sample respectively.

However, least-squares can build accurate models only when the number of samples is much greater than the number of unknown coefficients. Thus, given the high dimensionality of the performance models in complex AMS circuit designs, the simulation cost for building accurate models can be exorbitant. Hence, most recent performance modeling techniques incorporate additional information about the model to reduce the number of simulations needed to build accurate models [11]-[18].

While using unlabeled samples has shown significant reduction in modeling cost [19], the application of the proposed method is limited to AMS circuits with hierarchical structure. In this paper, we propose a generalized semi-supervised learning framework for AMS circuits without any assumption about their structures.

B. Multiple Variability Representation

Our main objective is to exploit the existence of two representations of the data to make use of the unlabeled samples in the performance modeling task. While the target is to build a performance model for the PoI as a function of PV, we suggest simultaneously building an alternative model that can be used to assist the original learning task. This is possible by the virtue of having two representations of variability for each device: (i) process variables and (ii) device-level variations.

For each device in the design, a set of independent random variables are used as a seed to generate all device level variation for the particular device. Examples of these variables

include change in oxide thickness (Δ_{tox}) and random dopant fluctuations (RDF) among others. In practice, device level variations are highly dependent and sampling directly from their joint distribution can be computational expensive especially that the existence of a closed form of such distribution is not guaranteed. Hence, it is more convenient to sample from the joint distribution of independent random variables, then express the device level variations as a function of these independent variables.

To further elaborate on this, we consider the process of generating the $n-$th simulation sample. First, the vector $\mathbf{x}^{(n)}$ containing the independent PV is sampled from its simple known distribution (typically a standard multivariate Gaussian distribution), then mapping functions are used to map the variability to the device level. As an example, the change in threshold voltage can be expressed as:

$$\Delta_{V_{TH}}^{(n)} = g(\mathbf{x}^{(n)}) \tag{6}$$

where $\Delta_{V_{TH}}^{(n)}$ is the threshold voltage variation at the $n-$th sampling point and $g(\mathbf{x})$ is mapping function that maps the PV variations to the change in threshold voltage.

Therefore, one or more device level variation parameters can be used to form an alternative model in the proposed co-learning process. Our main objective is to build the performance model in (1) that expresses PoI as a function of the process variables that fully describe the device variability. However, the alternative model need not be comprehensive especially considering the tradeoff between model accuracy and dimensionality when labeled data is scarce. In other words, the model can consider only a subset of the device level variation parameters that are expected to be the most important. In practice, we define our alternative model for the PoI as a function of only one device level variation parameter, which is the $\Delta_{V_{TH}}$:

$$y \approx f_2(\mathbf{v}) = \sum_{k=1}^{K} \beta.c_k(\mathbf{v}) \tag{7}$$

where $\mathbf{v}$ is a vector containing the change in threshold voltage for all devices in the circuit, $f_2(\mathbf{v})$ is the modeling function, $\{\beta_k; k = 1, 2, \ldots, K\}$ contains the model coefficients, $\{c_k; k = 1, 2, \ldots, K\}$ contains the basis functions, and K denotes the total number of basis functions.

The choice of the alternative model in (7) is motived by two key points. The first is that the dimensionality of this model is much lower than that of the model in (1). In fact, instead of representing each device by a set of PVs in (1), only one variable is used per device in (7). The second point is that, given the high correlation between the device level variation parameters, including only one of them in the model guarantees the independence of all variables in the vector $\mathbf{v}$. Meanwhile, it is important to note that the proposed framework can be easily extended to include multiple device-level variation parameters.

Therefore, we can define two models for the PoI starting from the two representations of data. Also, it is important to note that while obtaining labeled samples for the PoI in an AMS circuit requires expensive simulation, the two representations for unlabeled samples can be obtained at almost no cost by simply populating sample points using the simulator without performing any simulations.

III. Proposed Approach

A. Co-Learning Framework

The semi-supervised co-learning approach proposed in this work is based upon three main pillars: (i) a diverse representation of the data, (ii) a small set of labeled samples, and (iii) the availability of cheap unlabeled samples. Initially, the set of labeled data points are used to build initial models while considering the prior information about the model coefficients. At each iteration, using the two available models, pseudo predictions for unlabeled samples can be obtained. Then, using an appropriate confidence metric, the highly-confident samples from one model are used by the other model as pseudo samples in the next iteration [23]-[24].

As a first step, two regression problems are formulated to obtain the two initial models using the set of labeled data $\{\mathbf{y}_L, \mathbf{v}_L, \mathbf{x}_L\}$ while incorporating the prior information about the model coefficients. In this work, we assume the models coefficients are sparse, hence, the modeling tasks can be formulated mathematically as two constrained optimization problems:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & ||\mathbf{y}_L - \mathbf{B}_L.\boldsymbol{\alpha}||_2^2 \\ \text{subject to} \quad & ||\boldsymbol{\alpha}||_0 \leq \lambda_x \end{aligned} \tag{8}$$

and,

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & ||\mathbf{y}_L - \mathbf{C}_L.\boldsymbol{\beta}||_2^2 \\ \text{subject to} \quad & ||\boldsymbol{\beta}||_0 \leq \lambda_v \end{aligned} . \tag{9}$$

In (8)-(9), $||\bullet||_0$ is the "L_0-norm" of a vector, $\mathbf{B}_L$ is the evaluation of (3) based on $\mathbf{x}_L$, $\mathbf{C}_L$ is defined for $\mathbf{v}_L$ in a way analogous to $\mathbf{B}_L$, and λ_x and λ_v are upper bounds on the number of non-zero confections in $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ respectively. We denote by $\Omega_x = \{\mathbf{y}_L, \mathbf{x}_L\}$ and $\Omega_v = \{\mathbf{y}_L, \mathbf{v}_L\}$ the initial sets of training data used in fitting the two models.

The optimization problems in (8)-(9) are NP-hard [21]; hence, several heuristics have been proposed to efficiently find the sub-optimal solutions $\boldsymbol{\alpha}^*$, and $\boldsymbol{\beta}^*$ respectively. Among them, Orthogonal Matching Pursuit (OMP) is one of the sparse regression methods that has been extensively used in literature [9], [12], [13], [19], [21]. OMP assumes that there exists few dominant basis functions and most coefficients corresponding to other non-dominant basis functions are close to zero. Hence, it iteratively chooses a small number of important basis functions to include in the model by examining the correlation between the basis function and the performance values. Practically, the sparsity metrics λ_x and λ_v can be obtained using cross-validation [22], [20], [21]. Details about OMP are not shown due to page limit, however, reader can refer to [13], [19], [21] for more details.

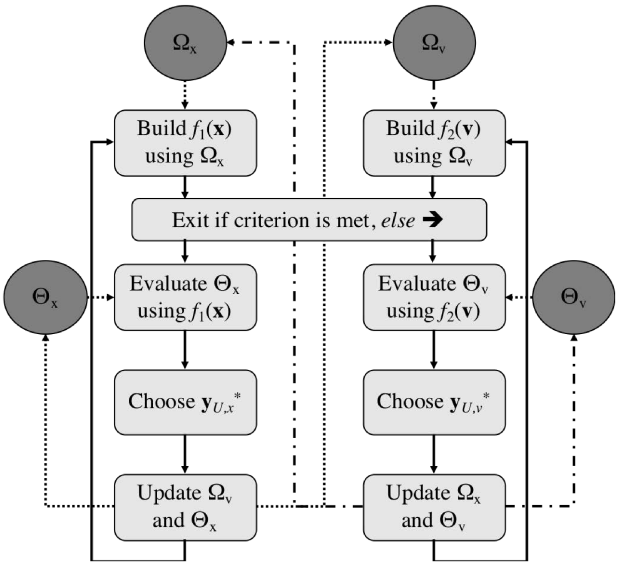

Fig. 1. The flow of one co-learning iteration is shown. Starting from training sets Ω_x and Ω_v, performance models are built. Then, the training data set is updated to include confident pseudo samples for the next iteration.

Using the initial models, pseudo labels for the unlabeled data set $\Theta_x = \{\mathbf{x}_U\}$ and $\Theta_v = \{\mathbf{v}_U\}$ are obtained as follows:

$$\mathbf{y}_{U,x} = \mathbf{B}_U.\boldsymbol{\alpha}^* \tag{10}$$

$$\mathbf{y}_{U,v} = \mathbf{C}_U.\boldsymbol{\beta}^* \tag{11}$$

where $\mathbf{y}_{U,x}$ and $\mathbf{y}_{U,v}$ are the pseudo labels for unlabeled samples Θ_x and Θ_v using models $f_1(\mathbf{x})$ and $f_2(\mathbf{v})$ respectively, and $\mathbf{B}_U$ and $\mathbf{C}_U$ are analogous to $\mathbf{B}_L$ and $\mathbf{C}_L$ respectively for unlabeled samples $\mathbf{x}_U$ and $\mathbf{v}_U$.

In the next step, the sets of training data Ω_x and Ω_v are updated to include the most confident samples among $\mathbf{y}_{U,v}$ and $\mathbf{y}_{U,x}$ respectively. Meanwhile, these samples are removed from the unlabeled data sets Θ_x and Θ_v. The update equation are given by:

$$\Omega_x = \Omega_x \cup \{\mathbf{y}^*_{U,v}, \mathbf{x}^*_U\}; \quad \Theta_v = \Theta_v - \{\mathbf{v}^*_U\} \tag{12}$$

$$\Omega_v = \Omega_v \cup \{\mathbf{y}^*_{U,x}, \mathbf{v}^*_U\}; \quad \Theta_x = \Theta_x - \{\mathbf{x}^*_U\} \tag{13}$$

where $\mathbf{y}^*_{U,x}$ and $\mathbf{y}^*_{U,v}$ are the most confident pseudo labels from models $f_1(\mathbf{x})$ and $f_2(\mathbf{v})$ respectively, and $\mathbf{x}^*_U$ and $\mathbf{v}^*_U$ are their corresponding $\mathbf{x}$ and $\mathbf{v}$ values.

After this update, new models can be obtained by solving the optimization problems in (8)-(9) with the updated training data sets. This process continues iteratively until reaching a user defined stopping criterion. Fig.1 shows one iteration of the co-learning method. Starting from the training data sets Ω_x and Ω_v, OMP is used to solve for the models $f_1(\mathbf{x})$ and $f_2(\mathbf{v})$. If the exit criterion is not yet met, the unlabeled data sets Θ_x and Θ_v are evaluated using the trained model and the most confident samples $\mathbf{y}^*_{U,x}$ and $\mathbf{y}^*_{U,v}$ are selected. Finally, the selected samples are moved from Θ_x and Θ_v alongside their pseudo labels to Ω_v and Ω_x respectively.

The intuition behind this approach comes from the very basic concept of learning from noisy data; which is the case in most regression problems. In [23], the authors show that, given a function that is learnable from noisy labeled data $f_2(\mathbf{v})$, with two redundant representations $\mathbf{x}$ and $\mathbf{v}$, then a weak initial model for f_2 can generate noisy labels for $\mathbf{x}$. In other words, given an unlabeled sample $\mathbf{v}^{(1)}$, the data point $\{\mathbf{x}^{(1)}, f_2(\mathbf{v}^{(1)})\}$ can be used to learn $f_1(\mathbf{x})$ and vice versa.

B. Confidence Estimation

One critical point in this approach is deciding upon the metric used to choose the pseudo samples to be included in the training set. In classification, this is relatively straightforward because confidence about class label is provided along the label by many classifiers [23]-[24]. However, such confidence is not readily available to be used in regression. Therefore, a key contribution of this work is estimating the confidence level for pseudo labels corresponding to unlabeled samples.

To illustrate the approach used for confidence estimation, we will start from the simple regression problem defined in (2). This approach can be easily extended to the OMP framework used in this work since, at its core, OMP is simply solving a similar problem for a select subset of basis functions.

While confidence level has been extensively studied in the field of statistics [20]-[22], most contributions address the challenge of estimating the confidence level on the model coefficients, usually referred to as confidence on estimator. However, the proposed co-learning framework presented in the previous subsection requires computing the confidence on each sample in the unlabeled data set to choose the most confident ones.

To address this challenge, we propose propagating the confidence from the estimator level to the sample level. To do so, we start by rewriting the expression for the performance model in (1) as:

$$y = f_1(\mathbf{x}) + \epsilon = \sum_{m=1}^{M} \alpha_m.b_m(\mathbf{x}) + \epsilon \tag{14}$$

where ϵ is a a zero mean residual term with variance equal to σ^2. Intuitively, one can see that the confidence on the model coefficients in (14) increases as the value of σ^2 moves closer to zero.

In practice, the confidence level on the model coefficients can be obtained starting from the closed form solution, $\hat{\boldsymbol{\alpha}}$, of the optimization problem in (2) [20]-[22]:

$$\hat{\boldsymbol{\alpha}} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{y}. \tag{15}$$

In statistics community, the standard deviation of the model coefficients, referred to as *standard error*, is the most widely used confidence metric. Using (14)-(15), the covariance matrix of the model coefficients can be expressed as:

$$cov(\hat{\boldsymbol{\alpha}}) = cov(\boldsymbol{\alpha} + (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\epsilon). \tag{16}$$

In (16), $\boldsymbol{\alpha}$ and $\mathbf{B}$ are both constants. Therefore, $cov(\hat{\boldsymbol{\alpha}})$ can be expressed as:

$$cov(\hat{\boldsymbol{\alpha}}) = (\mathbf{B}^T\mathbf{B})^{-1}\sigma^2. \tag{17}$$

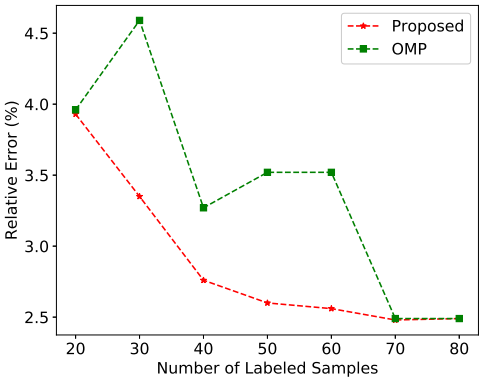

Fig. 2. The relative modeling error of the circuit is shown as a function of the labeled samples.

As defined in (14), σ^2 represents the variance of the residuals, hence it can be approximated by the unbiased estimator $\hat{\sigma^2}$:

$$\hat{\sigma^2} = \frac{||\mathbf{y} - \mathbf{B}.\boldsymbol{\alpha}||_2^2}{N - M}, \tag{18}$$

and eventually, $cov(\hat{\boldsymbol{\alpha}})$ can be computed as:

$$cov(\hat{\boldsymbol{\alpha}}) = (\mathbf{B}^T\mathbf{B})^{-1}\hat{\sigma^2}. \tag{19}$$

Once the standard error on the model coefficients is computed, it can be propagated to the regression prediction for the unlabeled samples. Using a prediction equation similar to (10), the variance of the predicted label for a the i-th unlabeled sample can be expressed as:

$$var(\hat{y}^{(i)}) = var(\mathbf{b}^{(i)}\hat{\boldsymbol{\alpha}}) = (\mathbf{b}^{(i)})^T cov(\hat{\boldsymbol{\alpha}})\mathbf{b}^{(i)}, \tag{20}$$

where $\hat{y}^{(i)}$ and $\mathbf{b}^{(i)}$ are respectively the prediction and vector of basis functions of the i-th sample.

The variance value computed in (20) represents a measure of the confidence on the prediction for all samples. Particularly, in each iteration of the co-learning framework, the samples with the smallest variance values, i.e., highest prediction confidence, are included in the training sets for the next iteration.

IV. Experimental Results

In this section, two circuit examples implemented using TSMC-40nm technology are used to demonstrate the efficacy of the proposed method. All numerical experiments are performed on a server with 3.4GHz and 32GB memory.

A. Comparator Design

To demonstrate the proposed approach we consider a StrongARM latch comparator circuit with power being the PoI. This comparator, which serves as an interface between the analog and digital domains, is among the most used designs thanks to its positive feedback which enables fast decisions, and the fully dynamic structure which eliminates static power.

TABLE I
Modeling error and cost for comparator

	OMP [13]	Proposed
Number of Simulations	70	50
Relative Error	2.50%	2.53%
Simulation Cost (Sec.)	84000	60000
Modeling Cost (Sec.)	2	123
Total Cost (Sec.)	84002	60123

In total, 1280 random variables are used to model the process variations for the circuit, whereas 44 device level parameters ($\Delta_{V_{TH}}$) are used in the alternative model.

To generate the labeled dataset, circuit simulations are performed based on Monte Carlo sampling. However, unlabeled data is obtained without running any simulations. To show the efficacy of the proposed method, two performance modeling approaches are implemented and compared: (i) OMP [13], and the proposed method. Fig. 2 shows the modeling error as a function of the number of labeled samples. Throughout this section, the error metric used is the relative absolute error (%).

In the proposed framework, 20 unlabeled samples are used in addition to the labeled samples to build the performance model. In practice, the co-learning is implemented such that only one sample is added to the training set in each iteration; i.e., only the most confident sample is used in each iteration. Table I further summarizes the computational cost for the two approaches which includes both simulation cost and modeling cost. In fact, the total cost is dominated by the simulation cost which is the time needed to collect the labeled samples. The results show that the proposed approach achieves 30% reduction in the modeling cost compared to OMP for the same target accuracy.

One important observation from Fig. 2 is that the behavior of the modeling error versus the number of labeled samples when using the proposed approach is more consistent with the expected behavior compared to OMP. The modeling error is expected to decrease with the increase of labeled samples. However, when the number of samples is small, the random fluctuation associated with random sampling can cause a non-monotonic pattern in modeling error. Hence, it is clear from the two trends in Fig. 2 that the use of the unlabeled samples can help in reducing such random fluctuations resulting in a monotonic behavior.

B. Voltage Controlled Oscillator Design

In this example, a voltage controlled oscillator (VCO) is considered where the PoI is the frequency. VCO is a common block in a frequency synthesizer. In recent years, ring VCO has replaced traditional Operational Transconductance Amplifier (OTA) where it demonstrated superior power efficiency. In total, 4620 random variables are used to model the process variations for this circuit and 148 device level parameters ($\Delta_{V_{TH}}$) are used in the alternative model.

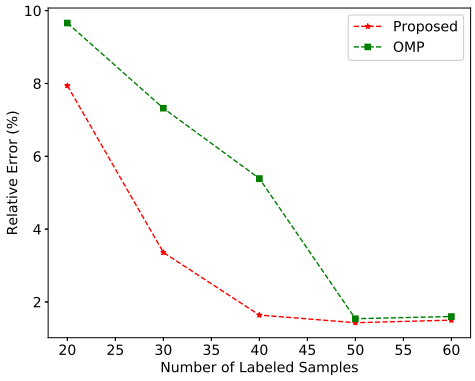

Fig. 3. The relative modeling error of the VCO is shown as a function of the labeled samples.

TABLE II
MODELING ERROR AND COST FOR VCO

	OMP [13]	Proposed
Number of Simulations	50	40
Relative Error	1.55%	1.6%
Simulation Cost (Sec.)	45000	36000
Modeling Cost (Sec.)	1.5	95
Total Cost (Sec.)	45002	36095

Using a setup similar to the previous example, performance modeling is performed and the modeling accuracy of both OMP and the proposed approach are summarized in Fig. 3. The error trends in Fig. 3 demonstrate the superiority of the proposed approach when compared to OMP. Moreover, Table II presents the modeling cost for both approaches which shows 20% reduction in the modeling cost when using the proposed method.

V. CONCLUSION

In this paper, a novel semi-supervised performance modeling approach is presented. The proposed approach relies on a co-learning framework that makes use of the two redundant data representation available at hand to incorporate unlabeled samples in the modeling process. Iteratively, two models are built based on the two representations and the most confident samples from one model are used as pseudo samples for the other in a co-learning scheme while taking into account the prior information of the model coefficients. As shown by our experimental results, the proposed approach can achieve up to 30% reduction in the stimulation cost when compared to state-of-the-art method.

REFERENCES

[1] Semiconductor Industry Associate, *International Roadmap for Devices and Systems*, 2016.

[2] X. Li. J. Le, and L. Pileggi, *Statistical Performance Modeling and Optimization* , Now Publishers, 2007.

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

[4] T. McConaghy and G. Gielen, "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.

[5] X. Li , J. Le, P. Gopalakrishnan and L. Pileggi, "Asymptotic probability extraction for nonnormal performance distributions,"*IEEE TCAD*, vol. 26, no. 1, pp. 16-37, Jan. 2007.

[6] F. Gong, Y. Shi, H. Yu and L. He, "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.

[7] H. Zhang, H. Zhang, T. Chen, M. Ting and X. Li, "Efficient design-specific worst-case corner extraction for integrated circuits," *DAC*, pp. 386-389, 2009.

[8] Y. Wang, M. Orshansky and C. Caramanis, "Enabling efficient analog synthesis by coupling sparse regression and polynomial optimization," *DAC*, pp. 1-6, 2014.

[9] M. B. Alawieh, F. Wang, R. Kanj, X. Li and R. Joshi, "Efficient analog circuit optimization using sparse regression and error margining," *ISQED*, pp. 410-415, 2016.

[10] V. Natarajan, A. Banerjee, S. Sen, S. Devarakond and A. Chatterjee, "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.

[11] X. Li, W. Zhang and F. Wang, "Large-scale statistical performance modeling of analog and mixed-signal circuits," *CICC*, 2012.

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

[13] 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.

[14] F. Wang, W. Zhang, S. Sun, X. Li and C. Gu, "Bayesian model fusion: large-scale performance modeling of analog and mixed-signal circuits by reusing early-stage data," *DAC*, pp. 1-6, 2013.

[15] X. Li, F. Wang, S. Sun and C. Gu, "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.

[16] Q. Huang, C. Fang, F. Yang, X. Zeng, and X. Li, "Efficient multivariate moment estimation via Bayesian model fusion for analog and mixed-signal circuits," *DAC*, 2015.

[17] C. Fang,Q. Huang, F. Yang, X. Zeng, X. Li, and C. Gu, "Efficient bit error rate estimation for highspeed link by Bayesian model fusion," *DATE*, pp. 1024- 1029, 2015.

[18] F. Wang, M. Zaheer, X. Li, J. Plouchart and A. ValdesGarcia, "Co-learning Bayesian model fusion: efficient performance modeling of analog and mixed-signal circuits using side information," *ICCAD*, pp. 575-582, 2015.

[19] M. B. Alawieh, F. Wang and X. Li, "Efficient hierarchical performance modeling for analog and mixed-signal circuits via Bayesian co-learning," *IEEE TCAD*, 2018.

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

[21] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Information Theory*, vol. 53, no. 12, pp. 46554666, Dec. 2007.

[22] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[23] A. Blum and T. Mitchell., "Combining labeled and unlabeled data with co-training," *COLT*, pp. 92100, 1998.

[24] Z. H. Zhou and M. Li, "Semi-Supervised Regression with Co-Training", *IJCAI*, pp. 1-6, 2005.