Adjustment-Based Modeling for Statistical Static Timing Analysis with High Dimension of Variability

Lin Xie, Azadeh Davoodi, Jun Zhang†, Tai-Hsuan Wu
Department of Electrical and Computer Engineering, †Department of Statistics
University of Wisconsin, Madison, WI 53706

Abstract— **This paper presents an adjustment-based modeling framework for Statistical Static Timing Analysis (SSTA) when the dimension of parameter variability is high. Instead of building a complex model between the circuit timing and parameter variability, we build a model which adjusts an approximate variation-aware timing into an accurate one. The intuition is that it is simpler to build a model which adjusts an approximate estimate into an accurate one. It is also more efficient to obtain an approximate circuit timing model. The combination of these two observations makes the use of an adjustment-based model a good choice for SSTA with high dimension of parameter variability. To build the adjustment model, we use a simulation-based approach, which is based on Gaussian Process. Combined with intelligent sampling, we show that an adjustment-based model can more effectively capture the nonlinearity of the circuit timing with respect to parameter variability compared to polynomial modeling. We also show that with only 200 samples of the circuit timing and 42 independent parameter variations, adjustment-based modeling obtains higher accuracy than direct SSTA using quadratic modeling.**

I. Introduction

In today's sub-45nm era, we expect an increase in the dimension of parameter variability as well as in their ranges [7], [5]. Consequently, an efficient and accurate Statistical Static Timing Analysis (SSTA), which can handle high dimension of variability, becomes a crucial need. It might be necessary during the sign-off stage especially when accurate interconnect and device parameter variability should be incorporated. It can also be used to more effectively make optimization decisions or to perform sensitivity analysis. The majority of the existing SSTA techniques are based on polynomial model of the circuit timing such as [3]-[4], [6], [10]-[12]. Many of them experimented with only few number of parameter variabilities (for example one global V_t, L_{eff}, t_{ox} for all devices) and reported high modeling accuracy. With the increase in the dimension of parameter variability, we require complex polynomials to achieve the same accuracy.

In this paper, we introduce a new SSTA technique which is useful for high dimension of variability. We first study Gaussian Process (GP) as an alternative to model the nonlinear dependencies of the circuit timing as a function of parameter variability. GP models are built using a simulation-driven approach in which the models are constructed solely using the input-output behavior collected for a limited number of intelligently chosen variation samples. We show this type of modeling has a simpler form than high-order polynomials but the modeling complexity depends on the number of samples.

We then introduce a novel adjustment-based modeling framework for SSTA. Instead of building a complex model between the circuit timing and the parameter variability, we build a simple model which adjusts an approximate circuit timing into an accurate one. The adjustment model is built using GP, which is simulation-driven. However, we show that the number of required samples can be greatly reduced for an adjustment-based model. This is because approximate models have already captured some nonlinear dependencies in terms of the input parameter variability.

We introduce two ways to obtain approximate circuit timing. The first one obtains the circuit timing from an extracted subcircuit which is expected to be timing-critical. The second one is to use a cheap linear SSTA. We then adjust these approximate timings into accurate ones. Here, by accurate timing, we mean a timing that is obtained using Monte Carlo (MC) simulation over the entire circuit. Our simulation results show that an adjustment-based timing model can achieve higher accuracy compared to a quadratic model of the circuit timing, which is directly a function of parameter variables. We also show that for 42 independent parameter variables, we only need 200 samples to build an adjustment-based model while fitting-based quadratic modeling requires at least 10,000.

II. Previous Works and Shortcomings

Previous SSTA works [3]-[4], [6], [10]-[12] always express the gate delay and arrival times using polynomial model in terms of variations in transistor or interconnect parameters. Particularly, [4], [6], [12] used quadratic forms to express the circuit timing, in which high modeling accuracy are reported. However, the dimension of the parameter variability used for accuracy verification in all these works is low.

With the scaling into sub-45nm era, the dimension of variability will become high even after implementing efficient parameter order reduction techniques. [7] and [5] reported 25 and 30 sources of uncorrelated variations after reduction, respectively. **In addition, as illustrated in [2], low-order polynomial models become inaccurate with the increase in the dimension and the range of the variability**. We also would like to note that quadratic model in this case also have very high computation complexity. As an example, a quadratic expression with 30 uncorrelated variables has over 400 terms if cross-terms are not ignored. Therefore, existing polynomial-based SSTA approaches need to be revisited. To address the aforementioned problems, we propose to express the circuit timing using Gaussian Process (GP) model, which is simpler than polynomial model and can flexibly capture a large class of complex nonlinear relationships [2], [9]. To further reduce the complexity of GP model, we propose adjustment-based model in Section IV.

III. Overview of Gauss Process Model

Given the input uncertainty vector $\mathbf{x} = \{x_1, x_2, ..., x_d\} \in \mathbb{R}^d$, we model the deterministic circuit timing as [9]:

$$y(\mathbf{x}) = \boldsymbol{\beta}\mathbf{f}^T(\mathbf{x}) + \epsilon(\mathbf{x}) = \sum_{i=0}^{n} \beta_i f_i(\mathbf{x}) + \epsilon(\mathbf{x}), \tag{1}$$

where $\mathbf{f}(\mathbf{x}) \triangleq \{f_1(\mathbf{x}), ..., f_n(\mathbf{x})\}$ represents pre-specified functions, which are typically a constant or a linear expression.

The $\epsilon(\mathbf{x})$ in Eq. (1) is a stationary Gaussian Process (GP) with mean 0 and variance σ^2. Its correlation structure defined between any two input sample vectors $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ is typically modeled as

$$R(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = \prod_{i=1}^{d} \exp(-\theta_i |x_i^{(1)} - x_i^{(2)}|^2), \tag{2}$$

where $x_i^{(j)}$ denotes the i^{th} element of $\mathbf{x}^{(j)}$.

This research was supported by National Science Foundation under Grant 0811082.

978-1-4244-2820-5/08/$25.00 ©2008 IEEE

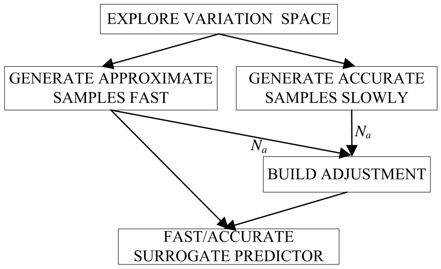
EXPLORE VARIATION SPACE
GENERATE APPROXIMATE SAMPLES FAST
GENERATE ACCURATE SAMPLES SLOWLY
N_a
N_a
BUILD ADJUSTMENT
FAST/ACCURATE SURROGATE PREDICTOR

Fig. 1. The flow of the adjustment-based modeling framework

In order to estimate the unknown parameters $\boldsymbol{\beta}$ in Eq. (1) as well as $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_d\}$ in Eq. (2) and σ^2 which defines $\epsilon(\mathbf{x})$, we first collect N input/output samples, and each sample refers to the input parameter variations and their corresponding output circuit timing. We denote the input/output of the i^{th} sample by $\mathbf{x}^{(i)}$ and $y^{(i)}$, respectively. We also denote the unknowns after estimation as $\widehat{\boldsymbol{\beta}}$, $\widehat{\boldsymbol{\theta}}$ and $\widehat{\sigma^2}$, respectively.

As shown in [9], we can obtain the estimation $\widehat{\boldsymbol{\theta}}$ via:

$$\max_{\boldsymbol{\theta}} \quad -\frac{1}{2}\left[N\log\widehat{\sigma}^2(\boldsymbol{\theta}) + \log\left(\det(\mathbf{R}(\boldsymbol{\theta}))\right) + N\right], \tag{3}$$

where

$$\widehat{\sigma}^2(\boldsymbol{\theta}) \triangleq \frac{(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})^T\mathbf{R}^{-1}(\boldsymbol{\theta})(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}})}{N}, \tag{4}$$

and $\mathbf{R}(\boldsymbol{\theta})$ has its ij^{th} entry $R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ as defined in Eq. (2).

In Eq. (4), $\mathbf{y}$ is a column vector with its i^{th} entry as the i^{th} observation (i.e., $y^{(i)}$), $\mathbf{F}$ is a matrix with its ij^{th} entry as $f_j(\mathbf{x}^{(i)})$, and $\widehat{\boldsymbol{\beta}}$ is defined as

$$\widehat{\boldsymbol{\beta}} \triangleq (\mathbf{F}^T\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{F}\mathbf{R}^{-1}\mathbf{y}. \tag{5}$$

The optimization problem in Eq. (3) can be *efficiently* solved using mlegp in “R” [1]. Using these estimated coefficients, we can compute the circuit timing for any given input variation vector $\mathbf{x}^\star$ using

$$y(\mathbf{x}^\star) = \widehat{\boldsymbol{\beta}}\mathbf{f}^T(\mathbf{x}^\star) + \widehat{\mathbf{r}}\mathbf{R}^{-1}(\mathbf{y}-\mathbf{F}\widehat{\boldsymbol{\beta}}), \tag{6}$$

where $\widehat{\mathbf{r}}$ is a vector with its i^{th} entry as $R(\mathbf{x}^\star, \mathbf{x}^{(i)})$.

Complexity of GP Model: As illustrated in Eq. (3), the number of unknowns to implement GP modeling is equal to $d+n+1$, where d is the dimension of the variability and n is the number of $f_i(\mathbf{x})$ (n is always very small). Note that these unknowns will maintain even when the Probability Density Function (PDF) of $\mathbf{x}$ changes.

As illustrated in Eq. (6), to predict the circuit timing with different $\mathbf{x}^\star$ only requires the computation of $f(\mathbf{x}^\star)$ and $\widehat{\mathbf{r}}$. All other components such as $\mathbf{R}$, $\mathbf{F}$, and $\widehat{\boldsymbol{\beta}}$ will not change, and thus can be precomputed after GP modeling is complete. Note that the dimension of $\widehat{\mathbf{r}}$ is equal to the number of samples (i.e., N) used in GP modeling. Therefore, decreasing N becomes necessary, such that efficient Latin Hypercube Design (LHD) is always used [9].

IV. Proposed Adjustment-Based Model

As illustrated in Section III, we can utilize LHD to reduce N, the required number of samples to obtain a desired modeling accuracy. However, N might be still large, especially when the dimension of variability is rather high. Moreover, larger N indicates higher evaluation complexity. Therefore, in this paper, we propose a novel adjustment-based model to characterize the circuit timing under high-dimension of variability. This adjustment-based model has recently been proposed in the statistics community and can significantly reduce the number of collected samples [8]. To the best of our knowledge, we are *the first* to introduce its usage and discuss its benefits in the CAD area for SSTA.

A. Overview

Fig. 1 shows the flow of the adjustment-based modeling framework. Instead of directly modeling the circuit timing with respect to the process variations using GP model, we build a model which can *adjust* an approximate circuit timing (subject to error) into an accurate one. The intuition behind this modeling framework is the following principle:

Principle of Accuracy: The approximate and accurate simulations should generate different output values for the same input sample. However, the approximate simulation can still reflect the relationship between the inputs and their corresponding accurate outputs. This reflection is subject to error but is sufficient to capture the complexities between the input and outputs to some extent.

After satisfying the above principle, building a model, which adjusts an approximate circuit timing into an accurate one, becomes a simpler task [8]. In other words, *the task of building an adjustment-based model requires much fewer samples than direct GP modeling, which indicates the reduction of the evaluation complexity*. Therefore, the adjustment-based model can be more suitable for modeling in presence of high dimension of variability and can perform much better than GP model.

As shown in Fig. 1, in order to build an adjustment-based model, apart from generating accurate reference samples for the circuit timing, we also need to generate samples for approximate circuit timing, which should be able to satisfy the principle of accuracy. These approximate samples can be obtained, for example, based on relaxing assumptions/models and can be generated much faster than the accurate ones, as we further illustrate. Moreover, even though the approximate samples are generated based on relaxed assumptions, they can still capture some nonlinear relationship between the parameter variations and the actual circuit timing.

B. Adjustment-Based Modeling Process

In order to build the adjustment-based model, we first generate a small number (i.e., N_a) of input samples, and obtain their corresponding approximate and accurate observations. For the convenience of notations, we denote the approximate and accurate observations by $y_I(\mathbf{x})$ and $y_A(\mathbf{x})$ for the input sample $\mathbf{x}$, respectively. Then, we utilize the GP modeling procedures introduced in Section III to build the adjustment function between y_I and y_A:

$$y_A(\mathbf{x}) = \rho(\mathbf{x})y_I(\mathbf{x}) + \delta_0 + \delta(\mathbf{x}), \tag{7}$$

where

$$\rho(\mathbf{x}) = \rho_0 + \sum_{i=1}^{d}\rho_i x_i, \tag{8}$$

x_i is the i^{th} element of $\mathbf{x} \in \mathbb{R}^d$ for $i = 1, 2, \ldots, d$, and $\delta(\cdot)$ is a Gaussian Process with variance σ_δ^2 and $\boldsymbol{\theta}_\delta$, as defined in Section III. Note that the parameters σ^2, $\boldsymbol{\theta}_\delta$, ρ_i and δ_0 in Eqs. (7) and (8) all need to be estimated.

Eq. (7) resembles the GP model in Eq. (1) in the following way

$$\begin{cases} f_i(\mathbf{x}) = y_I(\mathbf{x}),\ \beta_i = \rho_0, & \text{when} \quad i = 0 \\ f_i(\mathbf{x}) = y_I(\mathbf{x})\cdot x_i,\ \beta_i = \rho_i, & \text{when} \quad i = 1, 2, \ldots, d \\ f_i(\mathbf{x}) = 1,\ \beta_i = \delta_0, & \text{when} \quad i = d+1. \end{cases}$$

Consequently, we can similarly obtain the unknowns (i.e., σ^2, $\boldsymbol{\theta}_\delta$, ρ_i, δ_0) by following the estimation procedures for GP in Section III. We can then build the final adjustment-based model, and accurately compute the output performance for any new input vector using Eq. (6).

Remark: When Eq. (7) has all ρ_is equal to 0, it is equivalent to the original GP form in Eq. (1), where f_i is a constant. Therefore, we can expect not to loose accuracy by applying our adjustment-based model compared to building GP model based on accurate simulation.

V. Adjustment-Based Model for SSTA

We propose two methods to generate approximate variation-aware static timing analysis efficiently. We obtain accurate samples using Monte Carlo based SSTA on the Entire Circuit (MC-SSTA-EC).

A. Approximate Timing Using MC-SSTA-CS

The first way to obtain approximate circuit timing is to conduct MC simulation over a Critical Subcircuit, which we refer to as MC-SSTA-CS. In presence of process variations, the critical path will not be a constant path. However, we still extract the top critical paths of a circuit in the same way as under the *nominal case*. We first traverse the circuit topologically from the POs towards the PIs. At each gate, we consider the arrival times of all its fanins. Their arrival times are computed *under the nominal case*. The fanin with maximum arrival time is chosen to be part of the critical subtree. If the arrival times of other fanins are close to this maximum arrival time by a given threshold of $t\%$, we also include these fanins in this critical subtree. We apply Monte-Carlo based SSTA over the extracted critical subcircuit and record these as the approximate observations.

According to the *Principle of Accuracy*, the circuit delays obtained from MC-SSTA-CS should reflect the relationship between the parameter variations and the accurate circuit timing. To illustrate this principle, we selected circuit S35932 as an example. Fig. 2(a) shows the similarity between the circuit timing obtained using MC-SSTA-CS (i.e., Y-axis) and that using MC-SSTA-EC (i.e., X-axis). These samples were generated for 42 parameter variation cases as described in Section VI. We also plot the line $y = x$ to indicate the similarity between these two sets of circuit timing. As seen in this figure, in some variation cases, the approximate circuit timing is equal to accurate one (i.e. circuit timing is indeed determined by the critical subcircuit extracted under the nominal case). However, in other variation cases, the critical subcircuit does not determine the circuit timing, but they still exhibit a correct pattern of increase or decrease in the circuit timing. This is likely due to the common (global) source of variations which result in identical increase or decrease in the parameters of all the gates and interconnects for each variation sample. Furthermore, different gate types have similar range of sensitivities to different parameters.

B. Approximate Timing Using LG-SSTA

The second way to obtain approximate circuit timing, which can follow the Principle of Accuracy, is to use Linear Gaussian SSTA (i.e., LG-SSTA) [11]. In LG-SSTA, the arrival time at each gate is approximated to have Gaussian distribution. These arrival times, along with the individual gate delays are expressed as a linear combination of Gaussian random variables representing different device and interconnect parameter variations. Particularly, in computing the arrival times in the circuit, two atomic operations, namely summation and maximization should be computed. Summation operation is computed exactly. However, maximum of Gaussian arrival times is approximated as a Gaussian arrival time which can be represented in the same linear form. This approximation is based on Clark's approach. Given two Gaussian arrival times X and Y, we define Z as maximum of X and Y, and approximate it as:

$$Z \triangleq \max(X, Y) \approx tX + (1-t)Y + \Delta, \tag{9}$$

where t is the tightness probability defined as

$$t \triangleq \int_{-\infty}^{\lambda} \frac{1}{\sqrt{2\pi}} \exp\big(-\frac{x^2}{2}\big) dx, \tag{10}$$

and Δ is an additional random variable to ensure that this approximation has the same mean and covariance as Z.

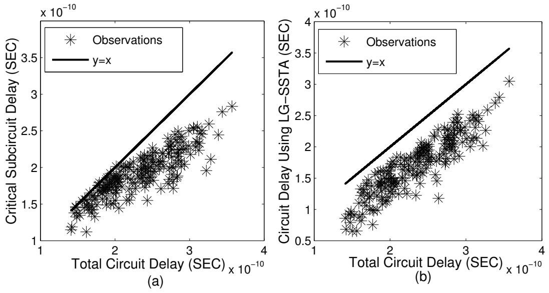

Fig. 2. Illustrations of Principle of Accuracy (a): approximate timing using MC-SSTA-CS; (b): approximate timing using LG-SSTA

In Eq. (10), λ is defined as $\frac{\mu_{X-Y}}{\sigma_{X-Y}}$, where μ_{X-Y} and σ_{X-Y} denote the mean and standard deviation of $X - Y$, respectively. The Gaussian approximation in Eq. (9) makes maximum operation easy to implement, but suffers from high error. The authors in [12] pointed that the Clark's approximation in LG-SSTA introduces high error when X and Y have very similar mean but very different variances, and when X and Y have very similar mean but highly negative correlation. Also, LG-SSTA is based on moment matching technique and can *only guarantee the correctness of the statistical information, such as mean and covariance at each maximization but the actual circuit timing is under variations and subject to error.*

Despite the intrinsic problems in LG-SSTA illustrated above, we can still use LG-SSTA to obtain approximate circuit timing. First, LG-SSTA is the fastest known SSTA with a complexity linear in the number of gates in the circuit; for each node with multiple fanins, it only needs to approximate the mean and variance of the maximum of their arrival times. Therefore, using LG-SSTA for approximate timing will not add much overhead for our modeling. Second, the LG-SSTA is subject to error; for example, the mean of the circuit timing obtained from LG-SSTA is proven to always be an underestimation of the exact mean. However, LG-SSTA can still capture the output/input relationship. Fig. 2(b) plots the circuit timing for S35932 using LG-SSTA in Y-axis and that using MC-SSTA-EC in X-axis. For LG-SSTA, we obtain the samples by doing Monte Carlo simulation on the linear Gaussian expression of timing obtained using LG-SSTA. Even though the circuit timing using LG-SSTA suffers from high error, it can still reflect the output/input relationship compared to the accurate simulation.

VI. Simulation Results

We synthesized ISCAS89 and ITC99 benchmark suites with 90nm TSMC library. Using the spatial correlation model in [5], we assumed 42 independent random variables (RVs) with Gaussian distribution to represent parameter variations in channel length and zero-body threshold voltage. All these RVs have their standard deviation to be 7% of their mean. We also assumed a linear gate delay model in terms of these RVs.

Evaluation of Adjustment-Based SSTA Using Critical Subcircuit Extraction: To evaluate the performance of our proposed adjustment-based SSTA using critical subcircuit, we implemented 3 types of SSTA, which are based on Quadratic model (QR), GP model, and Adjustment-based model (ADJ), respectively. *Note that by adjusting t, we extracted a small critical subcircuit in ADJ.* In all these SSTA approaches, we first generated a large number of samples for the 42 RVs and their corresponding circuit timing. We then fitted the samples to the model used in each approach to get a circuit timing expression in terms of these RVs. Particularly, we considered 4 variations of these 3 types of SSTA, which are listed as follows:

	CDF Percentile Error (%)					Speedup
BENCH	5%	25%	50%	75%	95%	$\frac{MC-SSTA-EC}{ADJ-CS-200}$
S35932	-1.554	-0.541	-1.159	-2.182	-3.795	23.50X
S38417	0.335	-0.567	-0.609	-0.551	-0.394	143.39X
S38584	-0.101	-0.207	-0.234	-0.283	-0.311	301.31X
b18	-1.504	-1.300	-1.167	-1.154	-1.136	11.72X
b20	2.000	2.221	2.573	2.611	2.273	38.31X
b21	1.049	1.149	1.186	1.202	1.093	62.43X
b22	-0.346	0.232	0.258	0.621	-2.289	31.62X
Ave $\|Err\|$	0.984	0.888	1.026	1.229	1.678	87.47X

TABLE I

CDF PERCENTILE ERROR COMPARED TO ACCURATE MONTE CARLO

1. QR model: We expressed the circuit timing as a quadratic function of the 42 RVs, which does not have cross-terms. It indicates that we only need to determine 85 coefficients (similar to [4]). Another reason we do not use the standard form of quadratic function is that, for 42 RVs, nearly 1,000 coefficients need to be estimated, which requires a large number of samples for modeling. Two variations of the QR approach are used to obtain the circuit timing samples depending on the sampling scheme and the number of samples. These two variations are (1.1) QR-UNI: quadratic fitting of circuit timing using 10,000 uniformly generated samples (using "rand" in Matlab), and (1.2) QR-LHD: quadratic fitting of circuit timing using 2,000 LHD samples (using "lhsdesign" in Matlab).

2. GP model: We generated 200 LHD samples for the parameter variations and built GP model to characterize the circuit timing in terms of these 42 RVs. For GP modeling, we used the mlegp package in "R" and denote this SSTA approach by GP-200.

3. ADJ model: We generated 200 LHD samples for parameter variations, and obtained the circuit timing samples using 200 MC samples over entire circuit and over the critical subcircuit. These are considered to be the accurate and approximate simulations, respectively. We used the mlegp package in "R" to build the adjustment-based timing model. We denote this approach which is based on critical subcircuit extraction by ADJ-CS-200.

In order to evaluate the accuracy of the above approaches, we first generated 10,000 test samples by assuming the parameter variations to have a standard normal distribution. We also implemented MC-SSTA-EC as our reference case.

First, we demonstrate the accuracy of ADJ-CS-200. Table I reports the percentage error at different percentiles of the circuit delay distribution using ADJ-CS-200 compared to MC-SSTA-EC. Columns 2-6 show the cumulative density function (CDF) error at different percentiles. The average of all percentile errors was smaller than 2%. Column 7 shows the speedup of our proposed ADJ-CS-200 over MC-SSTA-EC. The maximum and average speedup are 301.31X and 87.47X, respectively.

Next, we make comparisons among all these 4 SSTA approaches. Due to our linear gate delay model, all the approaches including the quadratic fitting SSTAs were highly accurate for un-balanced circuits. So we focused on S35932, which was a highly balanced benchmark.

Discussion for balanced S35932: Fig. 3 shows the histograms of the relative errors using these four SSTA approaches. The relative error is defined as the difference between the circuit delay using these SSTA approaches and that using MC-SSTA-EC. The average relative errors of QR-UNI, QR-LHD, GP-200, and ADJ-CS-200 were 18.27%, 11.87%, 10.67%, and 4.42%, respectively. We can thus conclude that, although ADJ-CS-200 used the smallest number of samples, which is 50 times lower than QR-UNI, it still achieved the minimum average relative error (i.e., 4.42%). We also observed that QR-LHD with 2,000 LHD samples is more accurate than QR-UNI with 10,000 uniform samples; therefore a good sampling scheme can result in significant improvement.

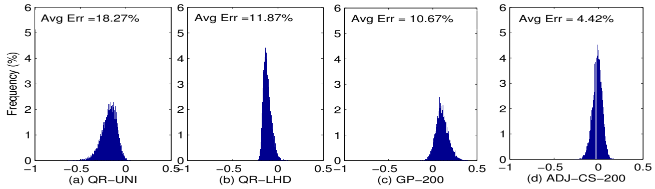

Fig. 3. Histograms of Relative error for S35932 (a): QR-UNI; (b): QR-LHD; (c): GP-200; (d): ADJ-CS-200

Evaluation of Adjustment-based SSTA Using LG-SSTA: We implemented the adjustment-based SSTA based on LG-SSTA as follows. We first generated 200 LHD samples for the process variations and obtained their corresponding approximate and accurate circuit delays using LG-SSTA and MC-SSTA-EC, respectively. We then built the adjustment model using "R" and denote this SSTA approach by ADJ-LG-200. In order to evaluate the accuracy of ADJ-LG-200, we make comparison with MC-SSTA-EC with 10,000 samples. For S35932 benchmark, ADJ-LG-200 had errors of 3.44%, 2.07%, 1.14%, 0.51% at CDF percentile points 25%, 50%, 75%, 95%, respectively.

Discussion on Efficiency: We draw the following conclusions:

1. The simulation runtime of ADJ-CS-200 was nearly proportional to the size of the extracted subcircuit which was much lower than the simulation runtime for the total circuit size. The runtime to collect the approximate circuit timing using MC-SSTA-CS was on average about 5% of that of MC-SSTA-EC. Also, we required collection of only 200 approximate and accurate samples. Similarly, the runtime of ADJ-LG-200 was very fast; this is because Clark's method is known to have linear runtime complexity in the number of nodes in the timing graph and is the fastest SSTA technique [11].
2. To build the adjustment-based model, Eq. (3) can be efficiently solved using "R". Note that this model can be directly used for robust analysis under other different PDFs of process variations.

VII. CONCLUSIONS

We proposed a novel adjustment-based model for SSTA which translates an approximate variation-aware circuit timing into an accurate one. We proposed two ways to generate approximate circuit timing to satisfy the principle of accuracy. For large circuits and with high dimension of variability, our improvement is mainly due to significant reduction in the number of samples using adjustment-based modeling over conventional quadratic and GP modeling.

REFERENCES

[1] The r project for statistical computing, http://www.r-project.org/.
[2] M. Bernardo and et al. Integrated circuit design optimization using a sequential strategy. *IEEE TCAD*, 1992.
[3] H. Chang and et al. Parameterized block-based statistical timing analysis with non-gaussian parameters, nonlinear delay functions. In *DAC*, 2005.
[4] L. Cheng and et al. Non-linear statistical static timing analysis for non-gaussian variation sources. In *DAC*, 2007.
[5] Z. Feng and et al. Fast second-order statistical static timing analysis using parameter dimension reduction. *DAC*, 2007.
[6] V. Khandelwal and et al. A general framework for accurate statistical timing analysis considering correlations. *DAC*, 2005.
[7] A. Mitev and et al. Principle hessian direction based parameter reduction with process variation. *ICCAD*, 2007.
[8] Z. Qian. *Computer Experiments: Design, Modeling and Integration*. Ph.D. Dissertation, Georgia Institute of Technology, 2006.
[9] T. Santner and et al. *The Design and Analysis of Computer Experiments*. Springer, New York, 2003.
[10] J. Singh and et al. Statistical timing analysis with correlated non-gaussian parameters using independent component analysis. In *DAC*, 2006.
[11] C. Visweswariah and et al. First-order incremental block-based statistical timing analysis. In *DAC*, 2004.
[12] L. Zhang and et al. Correlation-preserved non-gaussian statistical timing analysis with quadratic timing model. In *DAC*, 2005.