

Accounting for Non-linear Dependence Using Function Driven Component Analysis *

Lerong Cheng

Puneet Gupta

Lei He

Department of Electrical Engineering
 University of California, Los Angeles
 Los Angeles, CA 90095
 e-mail: {lerong, puneet, lhe}@ee.ucla.edu

ABSTRACT

Majority of practical multivariate statistical analyses and optimizations model interdependence among random variables in terms of the linear correlation among them. Though linear correlation is simple to use and evaluate, in several cases non-linear dependence between random variables may be too strong to ignore. In this paper, We propose polynomial correlation coefficients as simple measure of multi-variable non-linear dependence and show that need for modeling non-linear dependence strongly depends on the end function that is to be evaluated from the random variables. Then, we calculate the errors in estimation which result from assuming independence of components generated by linear de-correlation techniques such as PCA and ICA. The experimental result shows that the error predicted by our method is within 1% error compared to the real simulation. In order to deal with non-linear dependence, we further develop a target function driven component analysis algorithm (FCA) to minimize the error caused by ignoring high order dependence and apply such technique to statistical leakage power analysis and SRAM cell noise margin variation analysis. Experimental results show that the proposed FCA method is more accurate compared to the traditional PCA or ICA.

1. INTRODUCTION

With the CMOS technology scaling down to the nanometer regime, process as well as operating variations have become a major limiting factor for integrated circuit design. These variations introduce significant uncertainty for both circuit performance and leakage power. Statistical analysis and optimization, therefore, has generated lot of interest in the VLSI design community.

Existing work has studied statistical analyses and optimization for timing [1, 2, 3, 4, 5, 6, 7, 8, 9] and power [10, 11, 12, 9], and spatial correction extraction [13]. Most of these papers assume independence between random variables when performing statistical analysis. In order to obtain independence, most of the existing works use linear transformations, such as principle component analysis (PCA) or independent component analysis (ICA), to de-correlate the data. However, when there is non-linear dependence between the random variables under consideration, both PCA

and ICA cannot completely remove the dependence between random variables. PCA can only remove linear correlation between random variables but can not remove the high order dependence. Independent random variables must be uncorrelated, but uncorrelated random variables are not necessarily independent. If we assume the uncorrelated random variables are independent (as is done by most VLSI statistical analyses techniques), errors in the statistical calculations can be significantly large. ICA tries to minimize the mutual information between the random variables. When $I(X_1, X_2)$ exists, X_1 and X_2 are independent if and only if $I(X_1, X_2) = 0$. Since it is still a linear operation, it cannot completely remove the dependence between random variables.

In practice, the dependence between different variation sources is rarely linear (e.g., V_{th} is exponentially related to L_{eff}). Therefore, ignoring such non-linear dependences can cause significant error in analyses. There are some existing techniques for handling arbitrary dependence, such as Copula [14] and total correlation [15]. However, the complexity of using Copula is exponentially related to the number of random variables. Mutual information [15] and total correlation [15] measures the dependence between random variables, however, it is not easy to apply them in the statistical analysis. Moreover, there is little work in *removing* dependence using such measures as is readily done using PCA for linear correlation.

There exists some nonlinear algorithms to decomposed nonlinear dependent variation sources to independent components, such as nonlinear PCA [16] (or Kernel PCA) and nonlinear ICA [17]. Applying such algorithm may completely (or almost completely) remove dependence between variation sources and results independent components. However, such algorithms either express the variation sources as a very complicate function of independent components or even do not give close form formulas to express variation source using independent components. Hence, such nonlinear algorithms are not easy to be applied in statistical analysis and optimization.

In this paper, we analyze the impact of nonlinear dependence on statistical analyses. Key contributions of this work are as follows:

- We propose *polynomial correlation coefficients* as a simple measure of non-linear dependence among random variables.
- We show that importance of modeling non-linear de-

*This paper is supported in part by an NSF CAREER award 0401682, a UC MICRO grant sponsored by Actel and Fujitsu, and UC Discovery Grant ele07-10283 under the IMPACT program. Address comments to lhe@ee.ucla.edu and puneet@ee.ucla.edu

pendence strongly depends on what is to be done with the random variables, i.e., the end function of random variables that is to be estimated.

- We develop closed form expressions to calculate error in estimation of arbitrary moments (e.g., mean, variance, skewness) of the to-be estimated function as a result of assuming true independence of components generated by PCA or ICA techniques.
- We develop a target function driven component analysis algorithm (we refer to as *FCA*) which minimizes the error caused by ignoring non-linear dependence without increasing the computational complexity of statistical analysis.

The methods developed in this paper can be used to check whether linear de-correlation techniques like PCA will suffice for particular analysis problem. To the best of our knowledge, this is the first work to propose a systematic method to evaluate the need for complex non-linear dependence modeling for statistical analysis in VLSI design or otherwise. We apply our error estimation formula to the typical example from computer aided VLSI design: and leakage analyses. Experimental result shows that the our estimation is within 1% error of simulation. Further we give two example applications of *FCA* algorithm: statistical leakage analysis and SRAM cell noise margin variation analysis. The experimental results also show that the *FCA* is more accurate than regular PCA or ICA.

The rest of the paper is organized as follows: Section 2 theoretically calculates the impact of high order correlation, Section 3 applies the formulas to statistical leakage analyses and presents some experimental results, finally Section 4 presents target function driven ICA algorithm to minimize the error caused by ignoring non-linear dependence and Section 5 concludes this paper.

2. ANALYSIS OF IMPACT OF NONLINEAR DEPENDENCE

As discussed above, commonly used PCA and ICA techniques cannot provide fully independent random variable decomposition. In this section, we are going to study the impact of non-linear dependence on statistical analyses. We define the ij^{th} order polynomial correlation coefficient between two random variables X_1 and X_2 as:

$$\rho_{ij} = \frac{E[X_1^i X_2^j] - E[X_1^i]E[X_2^j]}{\sqrt{E[(X_1^i - E[X_1^i])^2] \cdot E[(X_2^j - E[X_2^j])^2]}} \quad (1)$$

ρ_{ij} 's provide us with simple and good measures to estimate the impact of nonlinear dependence. Note that $-1 \leq \rho_{ij} \leq 1$ and that ρ_{11} is simply the linear correlation coefficient. In rest of this paper, we assume that the ρ_{ij} 's are known. In practice, ρ_{ij} can be computed from the samples of variation sources.

With the above definition, we will show how to evaluate the impact of non-linear dependence on statistical analysis. Let us consider the two random variable case first. Let f be a polynomial function (or Taylor expansion of an arbitrary function) of two random variables $X = (X_1, X_2)^T$:

$$f(X) = \sum_{ij} a_{ij} X_1^i X_2^j \quad (2)$$

Then

$$E[f(X)] = \sum_{ij} a_{ij} m_{ij} \quad (3)$$

where $m_{ij} = E[X_1^i \cdot X_2^j]$ is the ij^{th} joint moment of X_1 and X_2 . If we ignore m_{ij} , then the error of mean estimation will be $a_{i,j}(m_{i,j} - m_{i,0}m_{0,j})$. That is, the importance of the ij^{th} joint moment depends on the coefficient of the ij^{th} joint moment in the Taylor expansion, $a_{i,j}$ and $m_{i,j} - m_{i,0}m_{0,j}$. We define:

$$Q_{ij} = a_{i,j} \cdot \sqrt{m_{2i,0} \cdot m_{0,2j}}. \quad (4)$$

Then the mean can be expressed as:

$$E[f(X_1, X_2)] = \sum_{ij} \rho_{i,j} \cdot Q_{i,j} \quad (5)$$

where $\rho_{i,j}$ is the ij^{th} order polynomial correlation coefficient between X_1 and X_2 as defined in (1). From the above equation, we find that the importance of the ij^{th} order dependence depends on $Q_{i,j}$. The above equations illustrates the two random variable case.¹

In practice, people usually apply principle component analysis (PCA) or independent component analysis (ICA) to obtain principle components or independent components. Assume that

$$P = (P_1, P_2)^T = W \cdot X \quad (6)$$

are the principle components (or independent components) obtained from PCA (or ICA), where W is the transform matrix. Then the function f can be written as the function of P_1 and P_2 :

$$f(X) = f(W^{-1} \cdot P) = \sum_{ij} c_{ij} P_1^i P_2^j \quad (7)$$

Because P is a linear combination of X , it is easy to obtain the coefficients c_{ij} , from a_{ij} and the transform matrix W .

In practice, when high order dependence exists, P_1 and P_2 are not completely independent. In this section, we try to estimate the error caused by ignoring the high order dependence, we mainly focus on the most important characteristics that people concern in statistical analysis, the mean, variance, and skewness calculation.

We express mean of f as:

$$E[f(X)] = f(W^{-1} \cdot P) = \sum_{ij} \rho_{p,i,j} \cdot T_{i,j}^\mu \quad (8)$$

$$T_{i,j}^\mu = c_{i,j} \cdot \sqrt{m_{2i,0}^P \cdot m_{0,2j}^P}$$

where m_{ij}^P is the ij^{th} joint moment of P_1 and P_2 , and ρ_{ij}^P is the ij^{th} order correlation coefficient between P_1 and P_2 . Since P is a linear combination of X , it is easy to obtain joint moments m_{ij}^P and correlation coefficients ρ_{ij}^P can be easily calculated from the moments of X_i 's m_{ij} and the transform matrix W . If we assume that these components are independent, i.e., we assume all the ρ_{ij}^P to be zero, then total error in mean estimation is:

$$\Delta_\mu = \sum_{i \geq 1, j \geq 1} \rho_{ij}^P \cdot T_{i,j}^\mu \quad (9)$$

¹In this paper, we discuss the case of two variation sources for simplicity and brevity. This method can be easily extended to multiple variation sources.

Similar to the estimation of the error in mean, we may estimate the error in variance calculation. We first estimate the error of second order raw moment of $f(\cdot)$. $f^2(\cdot)$ can be expressed as a polynomial function of P_i 's as:

$$f^2(P_1, P_2) = \sum_{ij} d_{ij} P_1^i P_2^j \quad (10)$$

where the coefficients d_{ij} can be calculated from c_{ij} 's. Then we may estimate the error of the second order raw moment of $f(\cdot)$:

$$\Delta_2 = E[f^2] - E[f'^2] = \sum_{i \geq 1, j \geq 1} \rho_{ij}^P \cdot T_{i,j}^\sigma \quad (11)$$

$$T_{i,j}^\sigma = d_{i,j} \cdot \sqrt{m_{2i,0} \cdot m_{0,2j}}. \quad (12)$$

where f' is the function ignoring the dependence. Then the error of variance calculation if high order dependence is ignored is:

$$\Delta_{\sigma^2} = \Delta_2 - 2\mu' \Delta_\mu - \Delta_\mu^2 \quad (13)$$

where μ' is the mean calculated by ignoring the high order dependence and Δ_μ is the error of mean calculation which is calculated in (9). In practice Δ_μ is much smaller compared to μ' , therefore, we have:

$$\Delta_{\sigma^2} \approx \Delta_2 - 2\mu' \Delta_\mu \quad (14)$$

With the error of variance, we may also calculate the error of standard deviation:

$$\Delta_\sigma = \sqrt{\sigma'^2 + \Delta_{\sigma^2}} - \sigma' \approx \frac{\Delta_{\sigma^2}}{2\sigma'} \quad (15)$$

Besides mean and variance, skewness is also an important characteristic of statistical distributions. In order to estimate the error of skewness calculation, we first estimate the error of the third order raw moment Δ_3 in a similar way as (16):

$$\Delta_3 = E[f^3] - E[f'^3] = \sum_{i \geq 1, j \geq 1} \rho_{ij}^P \cdot T_{i,j}^\gamma \quad (16)$$

$$T_{i,j}^\gamma = u_{i,j} \cdot \sqrt{m_{2i,0} \cdot m_{0,2j}}. \quad (17)$$

where the coefficients u_{ij} can be calculated from c_{ij} . Then the error of skewness can be calculated as:

$$\Delta_\gamma = \frac{E[f'^3] + \Delta_3}{(\sigma' + \Delta_\sigma)^3} - \frac{E[f'^3]}{\sigma'^3} \approx \frac{\Delta_3}{\sigma'^3} \quad (18)$$

3. CASE STUDY OF STATISTICAL LEAKAGE ANALYSIS

Statistical analysis is widely used in integrated circuit design. In the section, we apply our error estimation techniques on the statistical leakage power analysis.

3.1 Single Cell leakage

Generally, the leakage variation of a single cell is expressed as an exponential function of variation sources: [18, 11, 10]

$$P_{leak} = P_0 \cdot e^{c_{11}X_1 + c_{12}X_1^2 + c_{21}X_2 + c_{22}X_2^2} \quad (19)$$

where X_1 and X_2 are variation sources, P_0 is the nominal leakage value, c_{ij} 's are sensitivity coefficients for variation sources X_1 and X_2 , respectively. Performing N_{th} order Taylor expansion to the above equation, we have:

$$P_{leak} = P_0 \sum_{i,j=0}^{\infty} a_{i,j} X_1^i X_2^j \approx P_0 \sum_{i,j=0}^N a_{i,j} X_1^i X_2^j \quad (20)$$

Now we have the to-be estimated function in a polynomial form of variation sources. Then we may apply the method in Section 2 to estimate the error of mean, variance, and skewness when ignoring the high order dependence.

3.2 Full chip leakage

Full chip leakage power is calculated as:

$$P_{chipleak} = \sum_{r \in C} P_{leak}^r \approx \sum_{i,j=0}^N q_{i,j} X_1^i X_2^j \quad (21)$$

$$q_{i,j} = \sum_{r \in C} a_{i,j}^r \quad (22)$$

where C is the set of all circuit elements in the chip and $a_{i,j}^r$ is the ij^{th} order coefficient for the r_{th} circuit element. From the above equation, we can see that the full chip leakage can be expressed as the Taylor expansion of the variation sources. Therefore, we may estimate the error of mean, variance, and skewness calculation as previously.

3.3 Experiments

In this section, we show experimental results on some small benchmark circuits to validate our estimation.

3.3.1 Dependent variation sources generation

In our experiment, we assume two variation sources effective channel length L_{eff} and threshold voltage V_{th} . Since these two variation sources are dependent, to generate the dependent variation sample, we assume the variation of gate length L_{gate} and dopant density N_{bulk} are independent. ² We first generate samples of L_{gate} and N_{bulk} then we use ITRS 2005 MASTAR4 (Model for Assessment of cmoS Technologies And Roadmaps) tool [19, 20, 21] to obtain dependent samples of L_{eff} and V_{th} from the samples of L_{gate} and N_{bulk} . By applying PCA (or ICA) to the samples of L_{eff} and V_{th} , we obtain the marginal distribution for each principle component (or independent component).

In the experiment, we use the samples of L_{eff} and V_{th} with the exact dependence to perform Monte-Carlo simulation to calculate the exact distribution of leakage power, which is the golden result for comparison. We also assume that each principle component (or independent component) from PCA (or ICA) to be independent. Then we calculate the leakage power under such assumption and compare the result to that of the Golden case.

3.3.2 Experimental results

In our experiments, for L_{gate} and N_{bulk} , we assume a Gaussian distribution with 3σ of 5% of the nominal value. We use 10,000 Monte-Carlo simulations to calculate the golden case leakage power. Since leakage power is mainly affected by inter-die variation, in our experiment, we only consider inter-die variation.

In the Table 1, we compare the result of Monte-Carlo simulation (MC), the result after fitting (After fitting), and result after applying PCA (PCA). Then we calculate the error caused by curve fitting (Fitting error), the error when

²Notice that in practice, L_{gate} and N_{bulk} can not be easily measured in silicon. The only parameters we can measure is L_{eff} and V_{th} . That is, we can only extract the dependence between L_{eff} and V_{th} from the measured samples without knowing the exact variation of L_{gate} and N_{bulk} .

ignoring the nonlinear dependence (PCA error), and the error predicted by our algorithm above (predicted error). In the table, we also compare the result for two different leakage models, linear model (Lin) and quadratic leakage model (Quad). For the linear leakage model, we fit leakage power as an exponential of the linear function of variation sources, that is, no the square term in the power in Equation (19).

From the table, we see that, as expected, the linear leakage model leads to larger fitting error but almost does not depend on high order correlation. However, the quadratic leakage model has smaller fitting error, but there is error (about 5%) of standard deviation if we ignore the non-linear correlation. Moreover, we see that error predicted by our algorithm (predicted error) is very close to the experimental result (PCA error).

We also show full chip leakage analysis for a few ISCAS85 benchmarks in Table 2. In the tables, we compare the result of Monte-Carlo simulation (MC), the result of statistical leakage analysis (stat leak), and leakage after applying PCA (PCA). Then we calculate the error of statistical leakage analysis (stat leak error), the error when ignoring the nonlinear dependence (PCA error), and the error predicted by our algorithm (predicted error). Notice that the statistical leakage analysis error are caused by both curve fitting and analysis algorithm. Similar to the single gate case, we see that error predicted by our algorithm (predicted error) is very accurate compared to the experimental result (PCA error). From the tables, we see that the error caused by non-linear dependence is less than 5% for full chip leakage power analysis.

4. TARGET FUNCTION DRIVEN COMPONENT ANALYSIS

In the previous section, we introduced the method to estimate the error caused by ignoring non-linear dependence and showed that it depends on the target function being estimated. It is more important to reduce the error caused by non-linear dependence. As discussed in Section 1, linear operations can not completely remove the dependence between variation sources. However, due to simplicity of application, linear operation is preferred. Therefore, in this section, we try to find an optimum linear transform to minimize the error of ignoring the non-linear dependence. The proposed algorithm, function driven component analysis (*FCA*) decomposes dependent variation sources into components so as to minimize error in estimation of certain statistical measures of the target function.

In the rest of this section, we first present our algorithm and then apply it to statistical leakage analysis and SRAM cell noise margin variation analysis.

4.1 FCA Algorithm

Let $f(X)$ be a polynomial function (or Taylor expansion of an arbitrary function) of an n -dimensional random vector $X = (X_1, X_2, \dots, X_n)^T$. The objective of the *FCA* is to find an $n \times n$ transfer matrix W and independent components $P = (P_1, P_2, \dots, P_n) = W \cdot X$ to minimize the error of $f(WP)$ when assuming all P_i 's are independent. In statistical analysis, the error of $f(WP)$ is usually measured by mean, variance, and skewness. Moreover, because usually mean is the most important characteristic of statistical

Exact	PCA	ICA	FCA
9	5.01	7.42	8.89

Table 3: Mean of f .

analysis, we try to match the mean of $f(X)$. That is:

$$W = \arg_{\Delta_{\mu}=0} \min \Delta \quad (23)$$

$$\Delta = \Delta_{\sigma} + \epsilon \Delta_{\gamma} \quad (24)$$

$$\Delta_{\mu} = \mu_f - \mu_{f'} \quad (25)$$

$$\Delta_{\sigma} = \sigma_f - \sigma_{f'} \quad (26)$$

$$\Delta_{\gamma} = \gamma_f - \gamma_{f'} \quad (27)$$

where μ_f , σ_f , and γ_f are the mean, standard deviation, and skewness of $f(X)$, respectively, $\mu_{f'}$, $\sigma_{f'}$, and $\gamma_{f'}$ are the mean, standard deviation, and skewness of $f(WP)$ when assuming all P_i 's are independent, ϵ is the weight factor for the skewness error. In practice, the value of ϵ can be set by users. Because $f(X)$ is a polynomial function of X , similar to (9), (15), and (18), it is easy to find that μ_f , σ_f , and γ_f can be expressed as a function of joint moments of X_i 's, which are known, and $\mu_{f'}$, $\sigma_{f'}$, and $\gamma_{f'}$ can be expressed as a function of joint moments of P_i 's. Considering $P = WX$, the joint moments of P_i 's can be expressed as functions of W and the joint moments of X_i 's. Hence, the error Δ can be expressed as a function of W and joint moments of X_i 's. Notice that joint moments of X_i 's are known, therefore (23) becomes a non-linear programming problem. We use a non-linear programming solver to obtain the transfer matrix W .

Notice that in practice the minimization objective Δ can be any error that the users wants to minimize. In this paper, we choose this type of Δ because we try to minimize the error of variance and skewness.

Unlike the regular PCA or ICA, our *FCA* algorithm presented above tries to minimize the error for a target function f . That is, for different target function f , we may have different transfer matrix W . In *FCA*, we need to obtain an $n \times n$ transfer matrix W , that is, we need to solve a n^2 variable non-linear programming problem. However, for any statistical analysis, *FCA* needs to be run only once. Moreover, *FCA* still uses linear operation to decompose the variation sources. Therefore, applying *FCA* does not increase the computational complexity of the statistical analysis compared to regular PCA or ICA.

In order to validate our algorithm, let's first take a look at the simple example we introduced in Section 1: Let S_1 and S_2 be two independent random variables with standard normal distributions and $X_1 = S_1 + S_2$, $X_2 = S_1 S_2$. Estimate the mean of $f(X_1, X_2) = X_1^2 + X_1 X_2 + X_1^2 X_2^2 + X_2^2$. Table 3 shows the exact mean, and the mean estimated after PCA, fast kernel ICA [22], and *FCA*. From the table, we can see that *FCA* works better than PCA and ICA.

4.2 Experimental results

In order to validate the *FCA* algorithm, we show two examples of *FCA* in VLSI design: statistical leakage analysis and SRAM noise margin variation analysis.

4.2.1 Statistical leakage analysis

We first discuss statistical leakage analysis. Similar to Section 3.3, we assume two variation sources, effective L_{eff} and V_{th} and we only consider inter-die variation for the variation sources. We generate dependent variation samples of

Gate	Fitting type	MC			After fitting			PCA			Fitting error			PCA error			predicted error		
		μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ
Inv	Lin	7.12	2.55	0.35	6.44	2.13	0.31	6.32	2.02	0.27	-0.68	-0.42	-0.04	-0.08	-0.11	-0.04	-0.11	-0.13	-0.03
	Quad	7.12	2.55	0.35	7.15	2.61	0.36	7.05	2.49	0.31	0.03	0.06	0.01	-0.10	-0.12	-0.05	-0.12	-0.14	-0.04

Table 1: Cell leakage. Note: leakage value is in nW.

Gate	Fitting type	MC			Stat Leak			PCA			Stat Leak error			PCA error			predicted error		
		μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ	μ	3σ	γ
C17	Lin	430.2	120.3	0.28	415.3	113.3	0.25	413.2	109.2	0.23	-14.9	-7.0	-0.03	-2.1	-4.1	-0.02	-2.3	-3.9	-0.02
	Quad	430.2	120.3	0.28	437.2	126.3	0.32	431.5	122.2	0.30	7.0	6.0	0.02	-5.7	-4.1	-0.02	-5.3	-3.8	-0.03

Table 2: Chip leakage. Note: leakage value is in mW.

L_{eff} and V_{th} in the same way as Section 3.3.1. With the dependent samples, we use FCA (PCA or ICA) to decompose the variation sources and obtain the marginal distribution of each component. Then we generate sample of each component according to its marginal distribution. Assuming the components are independent, we generate the samples of L_{eff} and V_{th} . Finally, we use these sample to run SPICE Monte-Carlo simulation to obtain leakage power. We use ITRS 65nm technology in the experiment and assume supply voltage to be 1.0V. For L_{gate} and N_{bulk} , we assume that they follow Gaussian distribution and the 3-sigma value is 5% of the nominal value. Similar to Section 3.3, we consider only inter-die variation in this experiment.

In order to validate the accuracy of FCA we define three comparison cases: 1) samples generated from Mastar4 with the exact dependence, which is the golden case for comparison 2) samples generated from PCA, 3) samples generated from fast kernel ICA [22].

Table 4 illustrates the mean, standard deviation, skewness, 90%, 95%, and 99% percentile point of leakage of an inverter. From the table, we see that the value obtained from FCA is closer to the exact value than PCA and ICA. ³

Table 5 illustrates the exact error and the estimated error (using the method in Section 3) of mean, standard deviation, and skewness. From the table, we can find the the estimation error is close to the exact error. Moreover, notice that in FCA, we try to fit leakage power to a polynomial of variation sources. Therefore, part of the FCA error comes from fitting error.

	μ	σ	γ	90%	95%	99%
Exact	762	253	1.65	1285	1321	1395
PCA	710	228	1.44	1210	1262	1315
ICA	713	219	1.29	1221	1268	1345
FCA	751	235	1.51	1235	1275	1363

Table 4: Inverter leakage power comparison. Note: leakage value is in pW.

	Exact error			Est error		
	μ	σ	γ	μ	σ	γ
PCA	-52	-25	-0.21	-45	-28	-0.19
ICA	-49	-34	-0.36	-42	-29	-0.24
FCA	-11	-18	-0.14	0	-14	-0.11

Table 5: Estimated error for the inverter leakage power. Note: leakage value is in pW.

³The run time for PCA and ICA is less than 0.1s, and the run time for FCA is 0.4s. However, because FCA needs to be run only once in the statistical analysis, such run time overhead is a non-issue.

4.2.2 SRAM noise margin variation analysis

The second application example for FCA is the 6T-SRAM cell noise margin (SNM). We use similar setting as the statistical leakage analysis in Section 4.2.1. In order to highlight the flexibility of FCA, in this experiment, we consider only within-die variation. That is, each transistor has its own variation. In this case, because there are 6 transistors in an SRAM cell, there are 12 variation sources in an SRAM. Notice that PCA and ICA provide the same transfer matrix for L_{eff} and V_{th} for all transistors, however because FCA tries to handle 12 variation sources together, it may provide different transfer matrix for different transistors.

Table 6 illustrates the mean, standard deviation, skewness, 90%, 95%, and 99% percentile point of noise margin of an SRAM. From the table, we find the same trend as the leakage power case, that is, the value obtained from FCA is closer to the exact value than PCA and ICA.

	Nominal value			0.1678		
	μ	σ	γ	90%	95%	99%
Exact	0.1612	0.0312	0.0912	0.1215	0.1015	0.0821
PCA	0.1552	0.0335	0.0774	0.1242	0.1061	0.0867
ICA	0.1542	0.0331	0.0791	0.1261	0.1048	0.0859
FCA	0.1616	0.0307	0.0941	0.1226	0.1022	0.0841

Table 6: SNM comparison. Note: SNM margin is in V.

With noise margin variation analysis, we may further estimate number of redundant SRAM cells needed to ensure error correct SRAM array. We assume that the variation of all SRAM cells in the array are independent and an SRAM cell is faulty when the noise margin is less than a cut off value. For non-ECC architecture, for simplicity, we calculate the number of redundant SRAM cells needed to achieve a certain percent yield. For ECC scheme, the number of redundant SRAM cells depends on the coding. For simplicity, we estimate the Shannon Channel limit [23], which is the lower bound of the redundancy required to achieve no error coding.

Figure 1 illustrates the percentage SRAM redundancy under different cut off SNM value. In the figure, the x-axis is the cut off SNM value, which is calculated as a certain percentage of the nominal value. The y-axis is the percentage redundancy. For the non-ECC scheme, we assume that the redundancy is to achieve 99% yield rate. ⁴ From the figure, we see that FCA predicts the redundancy more accurate than the PCA or ICA.

We also ran experiments for different variation settings. In stead of assuming L_{gate} and N_{bulk} to be Gaussian. We

⁴This is just a simple estimation. In practice, because redundancy is needed for each row and column of SRAM array, the actual redundancy may be much higher.

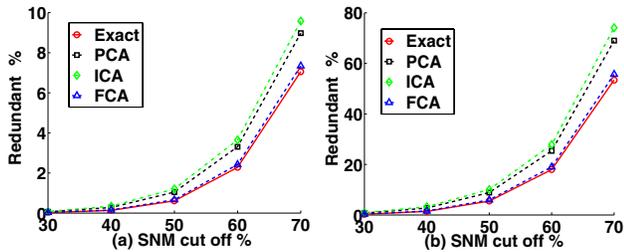


Figure 1: Redundancy for different cut off SNM. (a) Redundancy for Non-ECC scheme to achieve 99% yield rate. (b) Lower bound of redundancy for ECC scheme to achieve no error coding.

assume that they follows skew-normal distribution [24]. Table 7 illustrates the mean, standard deviation, skewness, 90%, 95%, and 99% percentile point of noise margin of an SRAM under such setting. Figure 2 illustrates redundancy. From the table and figure, we find that *FCA* works better than *PCA* and *ICA* under different variation settings.

	μ	σ	γ	90%	95%	99%
Exact	0.1723	0.0342	0.3214	0.1290	0.1120	0.0931
PCA	0.1758	0.0327	0.2856	0.1366	0.1177	0.0987
ICA	0.1742	0.0332	0.3102	0.1322	0.1172	0.0945
FCA	0.1736	0.0349	0.3156	0.1321	0.1147	0.0940

Table 7: SRAM cell noise margin comparison assuming L_{gate} and N_{bulk} to be with skew-normal distribution. Note: noise margin is in V.

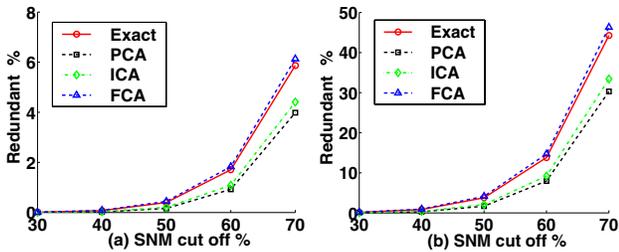


Figure 2: Redundancy for different cut off SNM assuming L_{gate} and N_{bulk} to be with skew-normal distribution. (a) Redundancy for Non-ECC scheme to achieve 99% yield rate. (b) Lower bound of redundancy for ECC scheme to achieve no error coding.

5. CONCLUSION

In this paper, we have proposed the first method to estimate the error of statistical analysis when ignoring the non-linear dependence using *polynomial correlation coefficients*. Such a method can be used to evaluate the accuracy the linear de-correlation techniques like *PCA* for a particular analysis problem. As examples, we apply our technique to statistical power analysis. Experimental result shows that the error predicted by our method is within 1% compared to the real simulation. We have further proposed a novel target function driven component analysis (*FCA*) algorithm to minimize the error caused by ignoring high order dependence. We apply such technique to two applications of

statistical analysis, statistical leakage power analysis and SRAM cell noise margin variation analysis. Experimental results show that the proposed *FCA* method is more accurate compared to the traditional *PCA* or *ICA*.

6. REFERENCES

- [1] A. Ramalingam, A. K. Singh, S. R. Nassif, G.-J. Nam, M. Orshansky, and D. Z. Pan, "An accurate sparse matrix based framework for statistical static timing analysis," in *ICCAD*, Nov 2006.
- [2] C. Visweswariah, K. Ravindran, K. Kalafala, S. Walker, and S. Narayan, "First-order incremental block-based statistical timing analysis," in *DAC 04*, June 2004.
- [3] Y. Lin, M. Hutton, and L. He, "Placement and timing for fpgas considering variations," in *Proc. Intl. Conf. Field-Programmable Logic and its Application*, August 2006.
- [4] L. Cheng, J. Xiong, and L. He, "FPGA performance optimization via chipwise placement considering process variations," in *Proc. Intl. Conf. Field-Programmable Logic and its Application*, August 2006.
- [5] S. Bhardwaj, P. Ghanta, and S. Vrudhula, "A framework for statistical timing analysis using non-linear delay and slew models," in *ICCAD*, Nov 2006.
- [6] L. Cheng, J. Xiong, and L. He, "Non-linear statistical static timing analysis for non-gaussian variation sources," in *DAC*, Jun 2007.
- [7] J. Singh and S. Sapatnekar, "Statistical timing analysis with correlated non-gaussian parameters using independent component analysis," in *ACM/IEEE International Workshop on Timing Issues*, pp. 143–148, Feb. 2006.
- [8] L. Cheng, J. Xiong, and L. He, "Non-gaussian statistical timing analysis using second-order polynomial fitting," in *Proc. Asia South Pacific Design Automation Conf.*, Jan 2008.
- [9] L. Cheng, Y. Lin, and L. He, "Tracebased framework for concurrent development of process and FPGA architecture considering process variation and reliability," February 2008.
- [10] C. H. Tan and J. Allen, "Minimization of power in VLSI circuits using transistor sizing, input ordering, and statistical power estimation," in *Proc. Int. Workshop on Low Power Design*, pp. 75–79, Apr. 1994.
- [11] H.-Y. Wong, L. Cheng, Y. Lin, and L. He, "FPGA device and architecture evaluation considering process variations," in *ICCAD*, Nov 2005.
- [12] Y. Lin and L. He, "Statistical Dual-Vdd assignment for FPGA interconnect power reduction," in *Proc. Design Automation and Test in Europe*, Apr 2007.
- [13] J. Xiong, V. Zolotov, and L. He, "Robust extraction of spatial correlation," in *Proc. Int. Symp. on Physical Design*, Apr 2006.
- [14] R. B. Nelsen, *An Introduction to Copulas*. Springer, 2007.
- [15] J. A. T. Thomas M. Cover, *Elements of information theory*. John Wiley and Sons, 2006.
- [16] A. Azzalini, "Nonlinear principal component analysis using autoassociative neural networks," *AICHE Journal*, 1991.
- [17] C. Jutten, "Advances in blind source separation (bss) and independent component analysis (ica) for nonlinear mixtures," *International Journal of Neural Systems*, Apr 2000.
- [18] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impacts on circuits and microarchitecture," in *DAC 03*, Jun 2003.
- [19] International Technology Roadmap for Semiconductor in <http://public.itrs.net/>, 2005.
- [20] International Technology Roadmap for Semiconductor, "A user's guide to MASTAR4," in <http://www.itrs.net/models.html>, 2005.
- [21] T. Skotnicki, "Heading for decanometer CMOS - Is navigation among icebergs still a viable strategy?," in *Solid-State Device Research Conference*, pp. 19–33, September 2000.
- [22] H. Shen, S. Jegelka, and A. Gretton, "Fast kernel ICA using an approximate newton method," in *11th International Conference on Artificial Intelligence and Statistics*, 2007.
- [23] R. E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge University Press, 2002.
- [24] A. Azzalini, "The skew-normal distribution and related multivariate families," *Scandinavian Journal of Statistics*, vol. 32, pp. 159–188, Jun 2005.