

Prediction of Multidimensional Spatial Variation Data via Bayesian Tensor Completion

Jiali Luan and Zheng Zhang^{ID}

Abstract—This paper presents a multidimensional computational method to predict the spatial variation data inside and across multiple dies of a wafer. This technique is based on tensor computation. A tensor is a high-dimensional generalization of a matrix or a vector. By exploiting the hidden low-rank property of a high-dimensional data array, the large amount of unknown variation testing data may be predicted from a few random measurement samples. The tensor rank, which decides the complexity of a tensor representation, is decided by an available variational Bayesian approach. Our approach is validated by a practical chip testing data set, and it can be easily generalized to characterize the process variations of multiple wafers. Our approach is more efficient than the previous virtual probe techniques in terms of memory and computational cost when handling high-dimensional chip testing data.

Index Terms—Bayesian statistics, data analytics, process variation, tensor, tensor completion, variation modeling.

I. INTRODUCTION

Today's nano-scale semiconductor manufacturing is subject to significant process variations (e.g., uncertain geometric and material parameters caused by imperfect lithography, chemical-mechanical polishing, and other steps) [1]. These process variations can propagate to circuit and system levels, and cause remarkable performance uncertainties and yield degradation. Therefore, extensive numerical modeling, simulation, and optimization techniques have been developed in the past decades to predict and control performance uncertainties of analog, digital, and mixed-signal design [2]–[13]. These numerical tools typically require a given detailed statistical model (e.g., a probability density function or a set of statistical moments) of the process variations. The statistical models of process variations are typically obtained by measuring and analyzing the performance data of a huge number of testing chips. The testing data can also be used for post-silicon yield analysis and performance tuning [14].

It is nontrivial to design and measure testing chips. First, one needs to carefully design and fabricate specialized circuits (e.g., ADC or ring oscillators) to measure or monitor the variation of certain parameters (e.g., V_{th}) [15]–[17]. For instance, a micro-processor may have hundreds of ring oscillators to monitor parametric variations, leading to large chip area overhead [18]. Second, one usually needs to

Manuscript received August 9, 2018; revised November 8, 2018; accepted December 31, 2018. Date of publication January 9, 2019; date of current version January 18, 2020. This work was supported in part by NSF-CCF under Grant 1763699 and Grant 1817037, and in part by Samsung Gift. This paper was recommended by Associate Editor X. Zeng. (Corresponding author: Zheng Zhang.)

J. Luan was with the Department of Mathematics, University of California at Santa Barbara, Santa Barbara, CA 93106 USA. She is now with the Department of Statistics, University of Michigan, Ann Arbor, MI USA (e-mail: luanjiali@hotmail.com).

Z. Zhang is with the Department of Electrical and Computer Engineering, University of California at Santa Barbara, Santa Barbara, CA 93106 USA (e-mail: zhengzhang@ece.ucsb.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCAD.2019.2891987

measure many testing circuits on each die in order to extract statistical distributions or to characterize intradie/interdie spatial correlations. Testing these circuits can consume a large amount of time. Finally, hardware measurement may also permanently damage the chips due to mechanical stress [19].

Instead of measuring all circuits, virtual testing techniques aim to reduce the cost by measuring only a small number of samples. A representative example is the “Virtual Probe” technique [20], [21] and its variants [22], [23], which employ compressed sensing [24] to estimate all performance data from a few measurement samples. In order to estimate all data on a die with an $n_1 \times n_2$ array of circuits, these techniques approximate the variation data by the linear combination of $n_1 n_2$ basis functions [e.g., 2-D discrete cosine transformation (DCT) bases] of spatial axes x and y . When the approximation is very sparse (which is generally true in practice), the $n_1 n_2$ coefficients can be estimated even if only $N < n_1 n_2$ measurement samples are available. These techniques have proved to be more efficient than traditional approaches, such as Kriging prediction [25] and k -LSE estimation [26]. Compressed sensing is effective for processing 2-D data, but it has some shortcomings: 1) it is inefficient to exploit the structure of multidimensional data and 2) it involves large-scale optimization to compute all DCT coefficients. More detailed analysis will be given in Sections III-C and V.

This paper presents an alternative tensor approach to reduce the cost of modeling variations across multiple dies or wafers. Tensor computation [27] can reveal more information that cannot be captured by matrix- or vector-based computations (e.g., compressed sensing). By stacking all 2-D chip data as a multidimensional data array, we estimate them simultaneously with a small number of random samples. The full unknown multidimensional data set is characterized by several low-rank tensor factors, and the unknown tensor factors are adaptively computed by employing the recently developed variational Bayesian approach [28] with an automatic rank determination process. We demonstrate the effectiveness of our approach by a realistic data set with 717080 data samples describing the contact resistivity of 20 dies, which is beyond the computational capability of Virtual Probe [20]–[23].

II. BASICS OF TENSOR

We first describe a few key definitions related to tensor, which are necessary to understand this paper. We refer the readers to [27] for a detailed introduction of tensor and [29] for tensor computation in electronic design automation.

Definition 1: A tensor is a high-dimensional generalization of a matrix. A matrix $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ is a second-order tensor, and its element indexed by (i_1, i_2) can be denoted as $x_{i_1 i_2}$. For a general d -th-order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_d}$, its element indexed by (i_1, \dots, i_d) can be denoted as $x_{i_1 \dots i_d}$.

Fig. 1 shows a matrix and a third-order tensor, respectively. In this paper, we denote scalars by lowercase letters (e.g., x), vectors (tensors of order one) by boldface lowercase letters (e.g., \mathbf{x}), matrices (tensors of order two) by boldface capital letters (e.g., \mathbf{X}), and higher-order

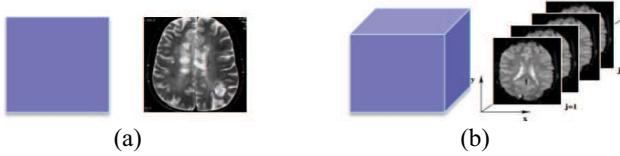


Fig. 1. (a) 2-D data array (e.g., one slice of MRI data) is a matrix. (b) 3-D data array (e.g., multiple slices of images) is a third-order tensor.

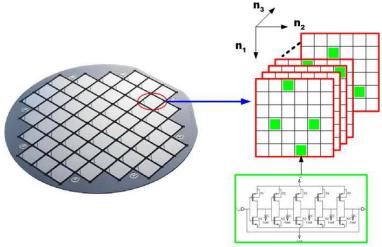


Fig. 2. Tensor completion for chip data testing. Each die is a $n_1 \times n_2$ slice (i.e., a matrix) of a tensor. The small green squares represent a small number of available measurement results.

tensors (order three or higher) by boldface calligraphic letters (e.g., \mathcal{X}), respectively.

Definition 2: Given any two tensors \mathcal{X} and \mathcal{Y} of the same size, their inner product is defined as

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1 \dots i_d} x_{i_1 \dots i_d} y_{i_1 \dots i_d}. \quad (1)$$

Definition 3: Given n tensors $\{\mathcal{X}^{(m)}\}$ of the same size, their generalized inner product is defined as

$$\langle \mathcal{X}^{(1)}, \dots, \mathcal{X}^{(n)} \rangle = \sum_{i_1 \dots i_d} \prod_{m=1}^n x_{i_1 \dots i_d}^{(m)}. \quad (2)$$

Since tensors are a generalization of matrices and vectors, the above definitions of (generalized) inner product apply to matrices and vectors as well.

Definition 4: The Frobenius norm of a tensor \mathcal{X} is further defined as $\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$.

Definition 5: A tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \dots \times n_d}$ is rank-1 if it can be written as the outer product of d vectors

$$\mathcal{X} = \mathbf{u}_1 \circ \dots \circ \mathbf{u}_d \Leftrightarrow x_{i_1 \dots i_d} = \mathbf{u}_1(i_1) \dots \mathbf{u}_d(i_d) \quad (3)$$

where $\mathbf{u}_k(i_k)$ denotes the i_k th element of vector $\mathbf{u}_k \in \mathbb{R}^{n_k}$.

III. TENSOR-BASED CHIP TESTING

Different from the previous virtual probe techniques [20]–[23] that employ compressive sensing, this section formulates the virtual testing as a tensor completion problem.

A. Problem Formulation

We consider the variations of n_3 dies on a wafer, and assume that each die has $n_1 \times n_2$ circuits (e.g., ring oscillators) which can capture spatial correlations. Instead of measuring all $n_1 n_2 n_3$ testing circuits, we aim to estimate their performance by measuring only N circuits, with $N \ll n_1 n_2 n_3$. In order to achieve this goal, we first stack all dies as a 3-D data array. As shown in Fig. 2, the whole data set of die i_1 is a matrix \mathbf{X}^{i_1} . We can see \mathbf{X}^{i_1} as i_1 th slice of a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Now the virtual testing problem can be formulated as tensor completion. Assume that N measurement

results $x_{i_1 i_2 i_3}$ are given, $(i_1, i_2, i_3) \in \Omega$, and Ω denotes the indices of measured samples. Then, we have the following problem:

$$\text{Given } x_{i_1 i_2 i_3} \text{ for } (i_1, i_2, i_3) \in \Omega, \text{ find } \mathcal{X}. \quad (4)$$

This formulation can be easily extended to handle multiple wafers: one can add another index i_4 to indicate a specific wafer, and the whole data set \mathcal{X} is a fourth-order tensor.

B. Low-Rank Tensor Completion

The problem in (4) has an infinite number of solutions, since we do not have any information about the un-given samples. Therefore, some constraints should be added. For instance, a ℓ_1 regularization is used in Virtual Probe [20], [21] because heuristic experience shows that a 2-D DCT transform of the data on each die has very sparse coefficients.

In this paper, we estimate the unknown variation data based on a different heuristic: we find that \mathcal{X} in the chip testing problem usually has a low-rank property in the high-dimensional space. Intuitively, this is because two reasons. First, there exist strong spatial correlations. Second, the fabrication data samples depend on the same fabrication process, and some fabrication process have much stronger influence in causing process variations. Similar to matrices, a low-rank tensor can be written as the sum of some rank-1 tensors

$$\mathcal{X} = \sum_{j=1}^r \mathbf{u}_1^j \circ \dots \circ \mathbf{u}_d^j. \quad (5)$$

This factorization is called the CANDECOMP/PARAFAC (CP) factorization, which is one of several popular factorization formats [27]. Having a few samples of \mathcal{X} , we attempt to compute the factors in (5) and to determine the rank r . Many tensor completion methods were introduced, but most approaches tend to have an inaccurate tensor rank and latent factors estimation, and eventually lead to the problem of over-fitting or weak predictive performance. In this paper, we choose to employ the variational Bayesian CP factorization model [28] to solve our problem. We will introduce the key ideas of variational Bayesian CP factorization in Section IV.

C. Comparison With Virtual Probe

The tensor completion approach can be considered as a more flexible generalization of the virtual probe [22]. In order to demonstrate this, we consider the problem of approximating a d -variable function $f(t_1, \dots, t_d)$, where $t_i \in [0, T_i]$ is a continuous variable for $i = 1, 2, \dots, d$.

We discretize $[0, T_i]$ into $n_i - 1$ segments of length $\Delta_i = T_i/(n_i - 1)$, then the (i_1, i_2, \dots, i_d) th element of \mathcal{X} can be regarded as the discretized value of $f(t_1, \dots, t_d)$

$$x_{i_1 \dots i_d} = f(t_1 = (i_1 - 1)\Delta_1, \dots, t_d = (i_d - 1)\Delta_d). \quad (6)$$

Consequently, the low-rank tensor factorization is equivalent to approximating f by some separable functions

$$f(t_1, \dots, t_d) = \sum_{j=1}^r u_1^j(t_1) u_2^j(t_2) \dots u_d^j(t_d) \quad (7)$$

and the vector \mathbf{u}_k^j in (5) includes n_k discretized function values of $u_k^j(t_k)$ on the grid points $\{t_k = (i_k - 1)\Delta_k\}$. In a low-rank tensor factorization, we adaptively find some *unknown* univariate functions $u_k^j(t_k)$ to approximate $f(t_1, \dots, t_d)$. Once some “good” univariate functions are found, a small number of product terms can be used to approximate f (and \mathcal{X} accordingly) with good accuracy.

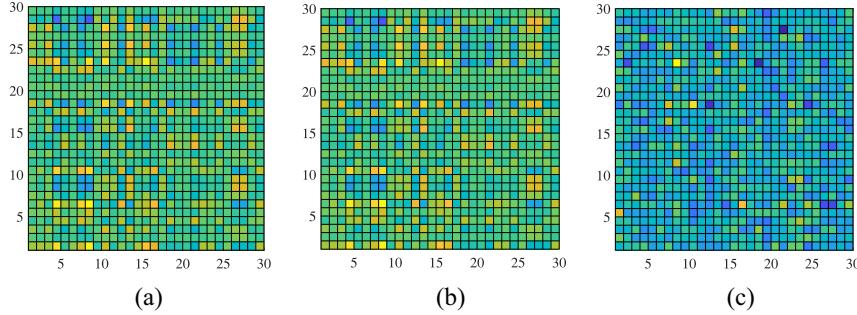


Fig. 3. (a) One slice of the original tensor. (b) One slice of the result from tensor completion with 10% samples. (c) One slice of the result obtained by virtual probe with 10% samples.

The virtual probe technique [20] is equivalent to approximating $f(t_1, \dots, t_d)$ by some *given* and *fixed* basis functions

$$f(t_1, \dots, t_d) = \sum_{i_1=1}^{n_1} \dots \sum_{i_d=1}^{n_d} c_{i_1 \dots i_d} \hat{u}_1^{i_1}(t_1) \hat{u}_2^{i_2}(t_2) \dots \hat{u}_d^{i_d}(t_d).$$

Here, $\{\hat{u}_k^{i_k}(t_k)\}$ are some *predefined* basis functions, and $\{c_{i_1 \dots i_d}\}$ are the unknown weights. In [20], the basis functions are specifically chosen as some Fourier basis functions. This choice of basis functions normally leads to a sparse representation; however, it is not guaranteed optimal. For instance, the Fourier basis function is not a good choice to approximate a nonsmooth function $f(t_1, \dots, t_d)$. Nonsmooth behaviors actually frequently appear in spatial variation modeling due to the random systematic variations [30]. In fact, $f(t_1, \dots, t_d)$ is rarely smooth with respect to t_k when dimension k is not an actual spatial dimension (e.g., when k is the additional dimension after stacking multiple 2-D dies as a 3-D array).

Our tensor-completion chip testing approach is more flexible and often more efficient than the virtual probe technique because of the following reasons.

- 1) In tensor completion, the univariate function $u_k^j(t_k)$ in (7) is chosen *adaptively* and is not limited to smooth functions (e.g., cosine functions as in [20]). Therefore, tensor completion is able to use “better” univariate functions and thus fewer product terms to approximate $f(t_1, \dots, t_d)$. For instance, we consider a random low-rank 3-D data array of size $30 \times 30 \times 15$. We use tensor completion to recover the whole 3-D data set with 10% entries, and use virtual probe to recover the data slice by slice. Fig. 3 shows the result of the first slice. The Virtual Probe has a huge error (and a relative error of 106%) for this data set due to the nonsmoothness. In contrast, tensor completion recovers the nonsmooth data perfectly with a relative error 2.23×10^{-9} .
- 2) The virtual probe technique forms an under-determined equation with $N = \prod_{k=1}^d n_i$ columns to compute the unknown weights $\{c_{i_1 \dots i_d}\}$. As d increases, the size of this linear equation grows exponentially. Consequently, importing this huge-size equation is beyond the capability of a computer’s physical memory, let alone performing numerical computation. Our tensor completion approach is a natural solution for large d , and it does not suffer from the curse of dimensionality.

IV. BAYESIAN CP FACTORIZATION

We employ the variational Bayesian approach [28] to solve our problem, due to its automatic rank determination and low computational cost. The key ideas are summarized below.

Let $\mathcal{Y} = \mathcal{X} + \varepsilon$ be a noisy tensor, and the true latent tensor \mathcal{X} is generated by a CP model

$$\mathcal{X} = \sum_{j=1}^r \mathbf{u}_1^j \circ \dots \circ \mathbf{u}_d^j = \mathbb{T}_{cp}(\mathbf{U}_1, \dots, \mathbf{U}_d). \quad (8)$$

The noise term ε is an independent identically distributed Gaussian distribution, $\varepsilon \sim \prod_{i_1, \dots, i_d} \mathcal{N}(0, \tau^{-1})$. Here, we have used \mathbf{u}_k^j to denote the factor vector of dimensionality k in the j th outer product; we use the matrix $\mathbf{U}_k = [\mathbf{u}_k^1, \dots, \mathbf{u}_k^r] \in \mathbb{R}^{n_k \times r}$ to denote all factor vectors associated with the k th dimension.

A. Probabilistic Model

Suppose Ω denotes the indices of some observed entries in \mathcal{Y} , the observed tensor \mathcal{Y}_Ω is defined as

$$\mathcal{Y}_\Omega = \begin{cases} y_{i_1 \dots i_d} & \text{if } (i_1, \dots, i_d) \in \Omega \\ 0 & \text{otherwise.} \end{cases}$$

We further denote $\mathbf{u}_{k, i_k} = [\mathbf{U}_k(i_k, :)^\top]^\top \in \mathbb{R}^{r \times 1}$, i.e., the transpose of the j th row of matrix \mathbf{U}_k . Combining the noise distribution and the CP model, we get the observation model that is factorized over the observed tensor entries

$$p(\mathcal{Y}_\Omega | \{\mathbf{U}_k\}_{k=1}^d, \tau) = \prod_{(i_1, \dots, i_d) \in \Omega} \times \mathcal{N}(y_{i_1 \dots i_d} | \langle \mathbf{u}_{1, i_1}, \dots, \mathbf{u}_{d, i_d} \rangle, \tau^{-1}).$$

The selection of a latent tensor rank, r , has been a challenging task. Previous probabilistic models rely on a predetermined tuning parameter chosen either by cross-validations or maximum likelihood. However, the Bayesian CP factorization method [28] is able to automatically determine the tensor rank as part of the Bayesian inference process. This approach uses a set of hyper-parameters ($\lambda = [\lambda_1, \lambda_2, \dots, \lambda_r]$) to control the rank. Each λ_j controls the magnitude of the j th column of each \mathbf{U}_k . Further, a sparsity-inducing prior distribution is placed over all the latent factors, given by

$$p(\mathbf{U}_k | \lambda) = \prod_{i_k=1}^{n_k} \mathcal{N}(\mathbf{u}_{k, i_k} | \mathbf{0}, \Lambda^{-1}), \forall k \in [1, d] \quad (9)$$

where $\Lambda = \text{diag}(\lambda)$ is the precision matrix. Note that the larger λ_j , the smaller elements in the j th column of \mathbf{U}_k . The hyper-priors over λ and τ are Gamma distributions, factorized over each dimensionality of the latent tensor \mathcal{X} , given by

$$p(\lambda) = \prod_{j=1}^r \text{Ga}(\lambda_j | \mathbf{c}_0(j), \mathbf{d}_0(j)) \quad (10)$$

$$p(\tau) = \text{Ga}(\tau | a_0, b_0) \quad (11)$$

where a_0, b_0 (scalars) and $\mathbf{c}_0, \mathbf{d}_0$ (vectors) are selected heuristically. Together, the overall probabilistic model can be expressed as the following joint distribution:

$$p(\mathcal{Y}_\Omega, \mathbf{U}_1, \dots, \mathbf{U}_d, \lambda, \tau) = p(\mathcal{Y}_\Omega | \{\mathbf{U}_k\}_{k=1}^d, \tau) \prod_{k=1}^d p(\mathbf{U}_k | \lambda) p(\lambda) p(\tau).$$

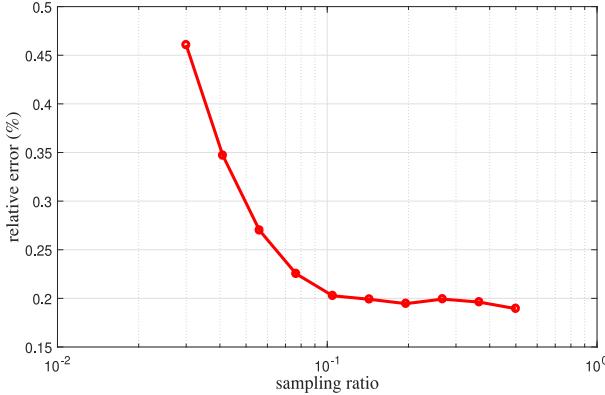


Fig. 4. Relative errors of tensor completion with various sampling ratios.

B. Variational Bayesian Solver

We need to compute the posterior distribution of all latent variables, including CP factors $\{\mathbf{U}_k\}$, rank-controlling hyper-parameters λ , and noise precision τ . In order to achieve this goal, the variational Bayesian method [28] seeks for a distribution function $q(\mathbf{U}_1, \dots, \mathbf{U}_d, \lambda, \tau)$ by minimizing the KL divergence of the two distribution p and q . By assuming that the posterior distributions of \mathbf{U}_k , λ , and τ are independent, one can use expectation–maximization steps to obtain the optimal solution in the following form:

$$q_k(\mathbf{U}_k) = \prod_{i_k=1}^{n_k} \mathcal{N}(\mathbf{u}_{k,i_k} | \tilde{\mathbf{u}}_{k,i_k}, \mathbf{V}_{k,i_k}), \quad \forall k \in [1, d] \quad (12)$$

$$q_\lambda(\lambda) = \prod_{j=1}^r \text{Ga}(\lambda_j | c(j), d(j)) \quad (13)$$

$$q_\tau(\tau) = \text{Ga}(\tau | a, b) \quad (14)$$

where the computed $c(i)$ and $d(j)$ determines the posterior mean of λ_j and thus the magnitude of the j th column of all CP factors $\{\mathbf{U}_k\}_{k=1}^d$.

The predicted tensor $\tilde{\mathbf{Y}}$ is calculated based on the posterior mean of the obtained latent factors

$$\tilde{\mathbf{Y}} = \sum_{j=1}^r \tilde{\mathbf{u}}_1^j \circ \dots \circ \tilde{\mathbf{u}}_d^j \quad (15)$$

where $\tilde{\mathbf{u}}_k^j$ is the posterior mean of vector \mathbf{u}_k^j . The relative error of the predicted data can be measured as

$$\text{relative error} = \|\tilde{\mathbf{Y}} - \mathbf{Y}\|_F / \|\mathbf{Y}\|_F. \quad (16)$$

V. NUMERICAL RESULTS

In order to verify our tensor completion-based chip variation data prediction method, we have implemented the algorithm in MATLAB on a Windows Desktop Workstation with 8-GB RAM and a 3.4-GHz CPU. We test our codes by a data set describing the variability of contact plug resistance in a 90-nm CMOS process [15]. The data set has the measurement results of multiple dies, and each die has $256 \times 144 = 35\,854$ testing circuits. We stack the data of 20 dies as a third-order tensor, which has 717 080 data samples in total. In the numerical experiments, we assume that only a small number of samples (which are randomly picked with a uniform distribution) are given, and we aim to accurately estimate the whole 3-D data set based on the given measurement samples.

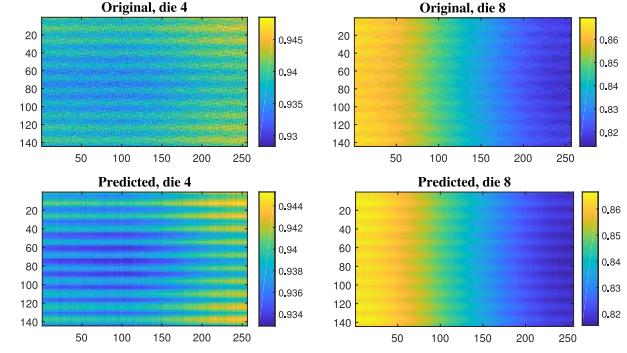


Fig. 5. Top: exact spatial variation patterns of die 4 and die 8. Bottom: predicted variation patterns by our approach with a 15% sampling rate.

TABLE I
CPU TIME FOR VIRTUAL PROBE AND OUR METHOD WITH 15% SAMPLES

Methods	Slice-By-Slice	3-D Array
Virtual Probe	62.947 s (17.5 h)	Out of Memory
Proposed Method	325 s (5.5 min)	67 s

A. Numerical Accuracy

In order to check the numerical accuracy of our approach on the chip data set, we fix the maximum tensor rank as 15. Then, we perform tensor completion repeatedly for different sampling ratios. The sampling ratios are chosen as ten logarithmically spaced points between 3% and 50%. For each experiment, we compute the relative error of the predicted results by $\|\tilde{\mathbf{Y}} - \mathbf{Y}\|_F / \|\mathbf{Y}\|_F$. As shown in Fig. 4, our approach can predict the spatial variation data with a very small sampling ratio: the relative error decreases to around 0.2% as the sampling ratio is greater than 10%; the relative errors are below 1% for all ten experiments. As shown in [15], the chip variation data typically has some certain patterns in the spatial domain. However, these patterns are not easy to capture, since they depend on very small variations across a die or a wafer. Our approach is capable of predicting the spatial patterns of the multiple-die data set simultaneously. We show the results of tensor completion by using a 15% sampling ratio and by fixing the maximum rank as 15. The top part of Fig. 5 shows the exact variation patterns of two chips obtained by measuring all testing circuits. The bottom of Fig. 5 shows the predicted variation pattern by tensor completion using only 15% random measurement data.

B. Comparison With Virtual Probe

We further compare our approach with the virtual probe on this realistic data set. Our data set has $144 \times 256 \times 20$ entries. Suppose we observe 15% of the data, the Virtual Probe approach then has to solve a linear equation with 110 592 rows and 737 280 columns. Computing such a large-scale matrix and importing it into the physical memory is far beyond the capability of our desktop computer. In contrast, the Bayesian tensor completion takes only 1 min to predict such a large-scale 3-D data array with an accuracy of 0.2%, as shown in Table I. Since the Virtual Probe technique is unable to directly process the high-volume 3-D data, we perform another round of comparison by predicting the 3-D data array slice-by-slice. Specifically, we use Virtual Probe and tensor completion to predict the 20 individual slices of 144×256 matrices based on 15% given samples. The Virtual Probe approach can work in this case, and it generates one model for each individual slice. However, Virtual Probe is extremely time-consuming: it takes 17.5 h to predict all slices as shown in Table I. In contrast, our Bayesian tensor completion finishes

TABLE II
PREDICTED TENSOR RANKS UNDER DIFFERENT MAXIMUM RANKS

Maximum Rank	Predicted Rank	Relative Error
5	4	0.214%
10	8	0.206%
15	9	0.200%
20	14	0.198%
25	17	0.196%

the prediction in only 5.5 min and with the similar level of accuracy. This is because that Virtual Problem has to solve a large-scale under-determined equation, whereas tensor completion only needs to compute a small number of unknown low-rank factors.

Remarks: Our proposed approach employs the variational Bayesian method [28] to estimate the tensor rank probabilistically. Once the algorithm converges, we can compute the expected value of each λ_j : a large λ_j indicates very small \mathbf{u}_k^j for all $k = 1, \dots, d$, thus the j th outer product in (8) will vanish, and a tensor rank deficiency is detected. We should choose a maximum rank that is greater than the true rank; otherwise, some tensor factors cannot be captured. However, if the selected maximum rank is too large, extensive data will be required to infer the latent variables, causing higher computational cost. Table II has shown the predicted ranks with respect to different maximum ranks when the sampling ratio is fixed as 15%. The predicted rank remains below 20 with the relative errors around 0.2%; the relative error decreases as the predicted rank increases to capture some small variations. However, a large maximum rank may cause over-fitting, and may overestimate the true rank.

VI. CONCLUSION

This paper has presented a tensor framework to predict the spatial variation data of semiconductor fabrications. Our key idea is to estimate the data of multiple dies simultaneously by performing tensor completion in a higher-dimensional data space. The approach has been implemented with a recently developed variational Bayesian approach which automatically determines the tensor rank in a probabilistic way. The numerical experiments on a contact plug resistivity variation data set has shown excellent performance. High accuracy (e.g., a 0.2% relative error) has been achieved with a small (e.g., 10%) sampling ratio. The proposed approach has also correctly predicted the spatial patterns of multiple dies simultaneously. Our proposed approach have easily handled a huge 3-D data set in 1 min, whereas the Virtual Probe technique failed to work due to its huge cost of physical memory and computational resources.

REFERENCES

- [1] D. S. Boning *et al.*, "Variation," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 63–71, Feb. 2008.
- [2] T. El-Moselhy and L. Daniel, "Stochastic dominant singular vectors method for variation-aware extraction," in *Proc. DAC*, Anaheim, CA, USA, Jun. 2010, pp. 667–672.
- [3] W. Yu, C. Hu, and W. Zhang, "Variational capacitance extraction of on-chip interconnects based on continuous surface model," in *Proc. Design Autom. Conf.*, Jun. 2009, pp. 758–763.
- [4] A. Singhee and R. A. Rutenbar, "Statistical blockade: Very fast statistical simulation and modeling of rare circuit events and its application to memory design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 8, pp. 1176–1189, Aug. 2009.
- [5] X. Li, "Finding deterministic solution from underdetermined equation: Large-scale performance variability modeling of analog/RF circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 11, pp. 1661–1668, Nov. 2011.
- [6] Z. Zhang, T. A. El-Moselhy, I. A. M. Elfadel, and L. Daniel, "Stochastic testing method for transistor-level uncertainty quantification based on generalized polynomial chaos," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 10, pp. 1533–1545, Oct. 2013.
- [7] Z. Zhang, T.-W. Weng, and L. Daniel, "Big-data tensor recovery for high-dimensional uncertainty quantification of process variations," *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 7, no. 5, pp. 687–697, May 2017.
- [8] J. Wang, P. Ghanta, and S. Vrudhula, "Stochastic analysis of interconnect performance in the presence of process variations," in *Proc. DAC*, San Jose, CA, USA, 2004, pp. 880–886.
- [9] X. Li, P. Gopalakrishnan, Y. Xu, and T. Pileggi, "Robust analog/RF circuit design with projection-based posynomial modeling," in *Proc. ICCAD*, San Jose, CA, USA, 2004, pp. 855–862.
- [10] Z. Zhang, I. Osledets, X. Yang, G. E. Karniadakis, and L. Daniel, "Enabling high-dimensional hierarchical uncertainty quantification by ANOVA and tensor-train decomposition," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 1, pp. 63–76, Jan. 2015.
- [11] Z. Zhang *et al.*, "Stochastic testing simulator for integrated circuits and MEMS: Hierarchical and sparse techniques," in *Proc. CICC*, San Jose, CA, USA, Sep. 2014, pp. 1–8.
- [12] C. Cui and Z. Zhang, "Stochastic collocation with non-Gaussian correlated process variations: Theory, algorithms and applications," *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 9, no. 7, pp. 1362–1375, Jul. 2019.
- [13] C. Cui and Z. Zhang, "Uncertainty quantification of electronic and photonic ICs with non-Gaussian correlated process variations," in *Proc. Int. Conf. Comput.-Aided Design*, 2018, pp. 1–97.
- [14] 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," in *Proc. Int. Conf. Comput.-Aided Design*, 2013, pp. 795–802.
- [15] K. Balakrishnan and D. Boning, "Measurement and analysis of contact plug resistance variability," in *Proc. CICC*, 2009, pp. 415–422.
- [16] A. H. Chang, K. Zuo, J. Wang, D. Yu, and D. Boning, "Test structure, circuits and extraction methods to determine the radius of influence of STI and polysilicon pattern density," in *Proc. Int. Symp. Qual. Electron. Design*, 2012, pp. 185–192.
- [17] K. Balakrishnan, K. A. Jenkins, and D. Boning, "A simple array-based test structure for the AC variability characterization of MOSFETs," in *Proc. Int. Symp. Qual. Electron. Design*, 2011, pp. 1–6.
- [18] M. Bhushan, A. Gattiker, M. B. Ketchen, and K. K. Das, "Ring oscillators for CMOS process tuning and variability control," *IEEE Trans. Semicond. Manuf.*, vol. 19, no. 1, pp. 10–18, Feb. 2006.
- [19] W. R. Mann, F. L. Taber, P. W. Seitzer, and J. J. Broz, "The leading edge of production wafer probe test technology," in *Proc. Int. Test Conf.*, 2004, pp. 1168–1195.
- [20] W. Zhang *et al.*, "Virtual probe: A statistical framework for low-cost silicon characterization of nanoscale integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 12, pp. 1814–1827, Dec. 2011.
- [21] W. Zhang, X. Li, and R. A. Rutenbar, "Bayesian virtual probe: Minimizing variation characterization cost for nanoscale IC technologies via Bayesian inference," in *Proc. DAC*, Anaheim, CA, USA, 2010, pp. 262–267.
- [22] W. Zhang, X. Li, E. Acar, F. Liu, and R. Rutenbar, "Multi-wafer virtual probe: Minimum-cost variation characterization by exploring wafer-to-wafer correlation," in *Proc. ICCAD*, 2010, pp. 47–54.
- [23] H.-M. Chang, K.-T. Cheng, W. Zhang, X. Li, and K. M. Butler, "Test cost reduction through performance prediction using virtual probe," in *Proc. Int. Test Conf.*, Anaheim, CA, USA, 2011, pp. 1–9.
- [24] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [25] F. Liu, "A general framework for spatial correlation modeling in VLSI design," in *Proc. Design Autom. Conf.*, San Diego, CA, USA, 2007, pp. 817–822.
- [26] A. N. Nowroz, R. Cochran, and S. Reda, "Thermal monitoring of real processors: Techniques for sensor allocation and full characterization," in *Proc. Design Autom. Conf.*, 2010, pp. 56–61.
- [27] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009.
- [28] Q. Zhao, L. Zhang, and A. Cichocki, "Bayesian CP factorization of incomplete tensors with automatic rank determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1751–1763, Sep. 2015.
- [29] Z. Zhang, K. Batselier, H. Liu, L. Daniel, and N. Wong, "Tensor computation: A new framework for high-dimensional problems in EDA," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 4, pp. 521–536, Apr. 2017.
- [30] B. E. Stine, D. S. Boning, and J. E. Chung, "Analysis and decomposition of spatial variation in integrated circuit processes and devices," *IEEE Trans. Semicond. Manufact.*, vol. 10, no. 1, pp. 24–41, Feb. 1997.