EFFICIENT VLSI YIELD PREDICTION WITH CONSIDERATION OF PARTIAL CORRELATIONS

A Thesis

by

SRIDHAR VARADAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2007

Major Subject: Electrical Engineering

EFFICIENT VLSI YIELD PREDICTION WITH CONSIDERATION OF PARTIAL CORRELATIONS

A Thesis

by

SRIDHAR VARADAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Jiang Hu
Committee Members,	Gwan Choi
	Shankar Bhattacharyya
	Duncan Walker
Head of Department,	Costas N. Georghiades

December 2007

Major Subject: Electrical Engineering

ABSTRACT

Efficient VLSI Yield Prediction with Consideration
of Partial Correlations. (December 2007)
Sridhar Varadan, B.E, Anna University
Chair of Advisory Committee: Dr. Jiang Hu

With the emergence of the deep submicron era, process variations have gained importance in issues related to chip design. The impact of process variations is measured using manufacturing/parametric yield. In order to get an accurate estimate of yield, the parameters considered need to be monitored at a large number of locations. Nowadays, intra-die variations are an integral part of the overall process fluctuations. This leads to the difficult case where yield prediction has to be done while considering independent and partially correlated variations. The presence of partial correlations adds to the existing trouble caused by the volume of variables. This thesis proposes two techniques for reducing the number of variables and hence the complexity of the yield computation problem namely - *Principal Component Analysis (PCA)* and *Hierarchical Adaptive Quadrisection (HAQ)*. Systematic process variations are also included in our yield model. The biggest plus in these two methods is reducing the size of the yield prediction problem (thus making it less time complex) without affecting the accuracy in yield. The efficiency of these two approaches is measured by comparing with the results obtained from Monte Carlo simulations. Compared to previous work, the PCA based method can reduce the error in yield estimation from 17.1% - 21.1% to 1.3% - 2.8% with 4.6× speedup. The HAQ technique can reduce the error to 4.1% - 5.6% with 6× - 9.4× speedup.

To My Parents

ACKNOWLEDGMENTS

This thesis would not have been possible without the support of many people. Many thanks to my adviser, Dr. Jiang Hu, for all his support, and more importantly for reading my numerous revisions and helping in making some sense of the confusion. I would like to thank Dr. Janet Wang from the University of Arizona at Tucson, for her timely guidance. Also thanks to my committee members, Dr. Gwan Choi, Dr. Shankar Bhattacharyya and Dr. Hank Walker for their encouragement.

I thank God for giving me the will to work harder when things didn't go smooth. Many thanks to my parents for all the motivation and the confidence boosting talks we have had through the last year. Just the word thanks won't justify the support they gave me. Special thanks to Karuna and Krishna for giving me all the light hearted moments and helping me regain my spirits after some dull moments.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Today's chip designs are characterized by shrinking feature sizes. Almost all IC manufacturing processes use complex physical and chemical interactions to get the targeted parameters. Given the complexity of these interactions and constant decrease in feature sizes, it is no longer possible to neglect process variations. The presence of process variations makes it important to predict manufacturing and parametric yield in circuit design stages. Manufacturing yield may be defined as the probability that a manufacturing spec is satisfied. Parametric yield on the other hand is defined as the probability that performance measures like power, timing etc. are met.

In the past, process variations were analyzed in the form of lot to lot, wafer to wafer and die to die variations [1]. All these variations are independent of the circuit's design and the corresponding loss in yield was acceptable. These days, under the deep submicron era era, with shrinking feature sizes and tighter pitches being the order of the day, intra-die variations are becoming dominant and make significant contributions to manufacturing yield [2].

In general, process variations can broadly be categorized into two types - systematic variations (depend on circuit design and layout patterns) and random variations (depend on fluctuations). Random variations can be further sub-divided into inter-die and intra-die variations. This thesis focuses on a specific yield model that can handle manufacturing and parametric variations.

For a manufactured chip, the inter-die variations tend to be perfectly correlated and hence approximated into a single random component. Intra-die variations, on

The journal model is *IEEE Transactions on Automatic Control.*

the other hand, consist of both independent parts and partially correlated parts. In case there exist intra-die variations completely independent of each other (absence of partial correlations), the overall yield is the product of the individual probability of each independent variable meeting its spec. The presence of partial correlations between all intra-die variations makes the scenario complicated. The yield in such a case is computed through numerical integration over a joint probability distribution function [3].

The yield model discussed in this thesis can be used to find the probability that n number of random variables lie within a specific range. In the case where these n random variables represent metal thicknesses at different locations on a chip, we can use this model to predict *Chemical Mechanical Planarization (CMP)* yield [1]. In case of a CMP model, yield may be defined as the probability of interconnect thicknesses at all locations on the chip to fall within the upper and lower thickness specifications.

CMP is a key enabling process for advanced interconnect technology and is used for planarizing and patterning copper interconnects [3]. After depositing copper, the metal surfaces are polished to leave copper only in the desired vias and trenches. The resulting metal interconnect does not have an ideal flat surface across the entire chip after removal of the copper. Such non-ideal effects are caused by over-polishing. Non-idealities in metal interconnect thickness profiles are shown in Figure 1.

The term dishing indicates excess polishing of a copper interconnect. *Dishing* could lead to loss in cross-sectional area of the interconnect and thus an increase in resistance. *Erosion* may be defined as the loss of oxide or dielectric thickness across an array of copper interconnects [4]. Both dishing and erosion depend on the layout of the design. Dishing increases with width of the copper interconnects, while erosion increases as the interconnects get narrow.

The copper thickness at any location (x, y) on the wafer is affected by - (1)

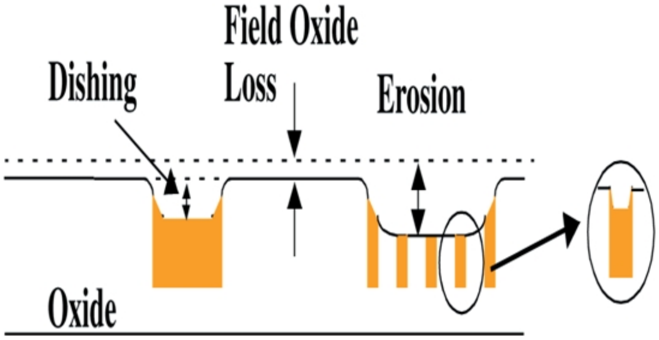

Fig. 1. Variations in CMP Process for Cu Interconnects

different layout patterns in the design, and (2) a number of process parameters [5]. As a result of these factors, the thickness of interconnects usually fluctuate around their nominal values. Depending on the range of variations, these thickness variations due to process fluctuations can lead to potentially serious issues such as open or short faults in interconnects.

The overall intra-die thickness at n locations on a chip may be described by the following equation -

$$p(n) = \mu(n) + \epsilon(n) \tag{1.1}$$

where $p(n)$, $\mu(n)$ and $\epsilon(n)$ represent the overall thickness, systematic and random components of intra-die variations respectively.

The yield (with respect to interconnect thickness) in case of partially correlated variations is obtained via numerical integration over a joint probability distribution function [6] as follows -

$$Yield = \int_L^U \int_L^U ... \int_L^U \phi(\overrightarrow{p}) \cdot dp_1 \cdot dp_2 \cdot ... \cdot dp_n \tag{1.2}$$

where, $\phi(\overrightarrow{p})$ - represents the joint distribution of thicknesses at n locations, U and L - represent the upper and lower thickness limits respectively.

The joint distribution function may be re-written as -

$$\phi(\overrightarrow{p}) = \frac{e^{-0.5 \cdot (\overrightarrow{p} - \overrightarrow{\mu})^T \cdot \Sigma^{-1} \cdot (\overrightarrow{p} - \overrightarrow{\mu})}}{\sqrt{(2\pi)^n |\Sigma|}} \tag{1.3}$$

where Σ is the covariance matrix representing correlations between the n different locations monitored on a chip. Typically the number of locations monitored in computing CMP yield is of the order of 10^5 or 10^6 [4], [3]. The presence of spatial correlation between such a large number of locations further adds to the complexity of the problem.

Although, the presence of partially correlated variations makes the scene a lot more complicated, we could use some divide and conquer based clustering algorithms to reduce the number of variables. One such method is discussed in [1] where the authors make use of perfect correlation clusters to group and reduce the number of variables used in computing yield from a very large number to a fewer number of variables. Upon reduction, the number of variables we are left with equals the number of perfect correlation clusters. Such methods ensure a reduction in the number of variables used in predicting yield and hence speed up the yield prediction problem.

However, there lies one trouble with the clustering algorithm proposed in [1]. Often, the size of each perfect correlation cluster used in grouping and reduction could affect the accuracy of yield. In order to eliminate such accuracy and time complexity issues, this thesis proposes the use two techniques - *Principal Component*

Analysis (PCA) and *Hierarchical Adaptive Quadrisection (HAQ)* in predicting yield. These algorithms are designed with the view of computing yield at a faster pace with minimal effect on accuracy.

The first method named *Principal Component Analysis* transforms a large set of correlated variables to an uncorrelated and reduced set of variables through an orthogonal base. Once an orthogonal base is found, reductions is achieved by discarding all redundant variables and others that are small in magnitude and thus make a less contribution to the yield.

The second method named *Hierarchical Adaptive Quadrisection* reduces a large number of variables in a given layout region to a reduced set of basic sub-regions. Each sub-region is group containing a specific number of variables. The variable with maximum or minimum thickness is used to characterize all variations within the sub-region. The size of each sub-region is decided based on how sensitive the other variables to the one with minimum or maximum thickness. Both these methods of reduction are explained in the subsequent chapters.

CHAPTER II

BACKGROUND WORK

Yield in general can be defined as the probability of a certain manufacturing parameter to stay within the specified boundaries. In [1], a novel method to reduce the size of the yield prediction problem is presented. [1] discusses a cluster and divide approach which decomposes the yield prediction equation 1.2 to -

$$Yield = Y_U + Y_L - 1 \tag{2.1}$$

where Y_U (or High Yield) represents the probability that the thickness values of all variables are below the maximum thickness limit and Y_L (or Low Yield) represents the probability that thickness values of all variables are above the minimum thickness limit. The equations defining these two components are written below -

$$Y_U = \int_{-\infty}^{U} \int_{-\infty}^{U} ... \int_{-\infty}^{U} \Phi(\vec{p}) dp_1 dp_2 ... dp_n \tag{2.2}$$

$$Y_L = \int_{L}^{\infty} \int_{L}^{\infty} ... \int_{L}^{\infty} \Phi(\vec{p}) dp_1 dp_2 ... dp_n \tag{2.3}$$

where U and L are the upper and lower thickness limits respectively.

The above written equations can be further reduced to a problem involving a smaller set of variables. Computing these new set of reduced number of variables is done using a clustering based divide and conquer approach [1]. This approach is explained in the subsequent paragraphs.

To simplify analysis, the chip is broken down to a number of equally sized tiles.

From now on, in this setup, each tile will indicate a separate variable. When the chip is seen as a bunch of equally sized tiles, the setup will look something similar to what is shown in Figure 2. In Figure 2 let us consider a chip with dimensions 100 $\mu m\times$ 90 μm. The chip is then divided into a number of tiles, each of size 10 $\mu m\times$ 10 μm. In all, the entire chip is covered using 90 tiles. These tiles will now form the initial set of variables. The distances between variables when accounting for correlation is calculated keeping the centres of all tiles in mind. As stated above, all this is done for simplicity sake.

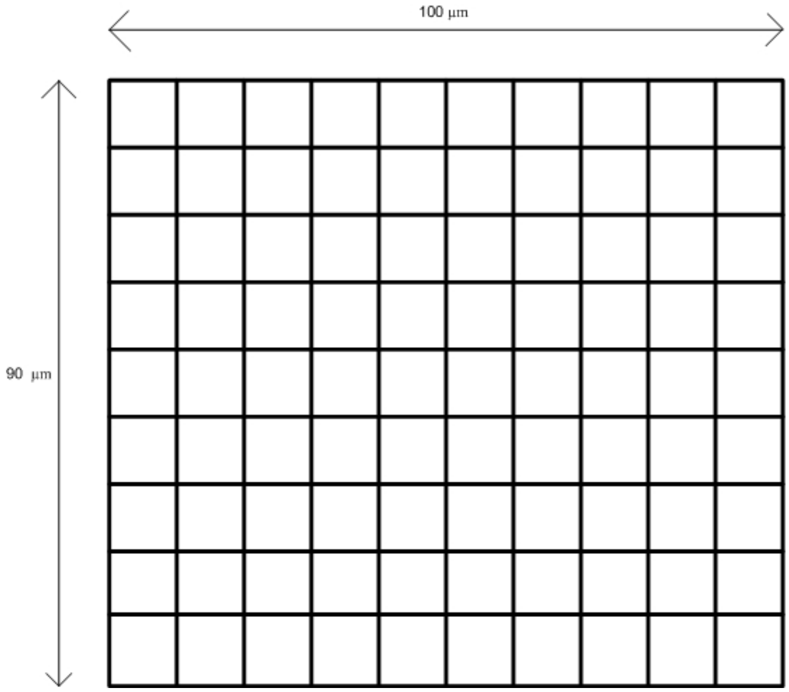

Fig. 2. Initial Setup - A Chip Consisting of a Number of Equally Sized Tiles

In reality, the number of tiles covering any chip is a large number. With such a setup, the size of the covariance will also be big, thus making yield prediction complex. In order to reduce the size of the correlation matrix, [1] makes use of a clustering based divide and conquer appoach. Each cluster is defined using a *Perfect Correlation Circle (PCC)* with a pre-fixed radius. The idea behind using perfect correlation circles is something similar to this all tiles that fall within the area covered by a perfect correlation circle are assumed to have a perfect correlation, so all tiles lying within a PCC can be represented by the tile at the centre of the PCC. This idea is used to bring about a reduction in the number of variables used in computing yield.

A. Computing High Yield

The nominal thickness values of n locations or tiles on a chip are given as input to the algorithm. To start with, we find the tile of maximum interconnect thickness in the chip, let the tile be labeled MAX_1. Now a circle (let this circle be called $CIRCLE_1$) is drawn with the tile MAX_1 as centre and a prefixed radius. All tiles that lie within the perfect correlation circle of MAX_1 are assumed to have a perfect correlation.

The next step is to find another tile MAX_2 which is the point of highest thickness interconnect thickness outside the perfect correlation circle $CIRCLE_1$. All tiles which come inside the perfect correlation circle $CIRCLE_2$ (drawn with tile MAX_2 as centre) are now assumed to have perfect correlation. Following this, a tile of maximum thickness (MAX_3) and its corresponding PCC $CIRCLE_3$ are formed and this procedure is repeated until all regions in the chip are covered by perfect correlation circles (let the total number of PCCs be m. After there are no regions left uncovered on the chip, the centres of all $PCCs$ (all the tiles, MAX_1, MAX_2, ..., MAX_m) form the new set of reduced variables. This reduction in problem size to a smaller number of variables (tiles at the centre of the m PCCs) is a very big win compared to what the original size of the problem (n tiles in the initial setup) .

After reduction, High Yield can be computed using the following equation -

$$Y_U = \int_{-\infty}^{U} \int_{-\infty}^{U} ... \int_{-\infty}^{U} \Phi(\overrightarrow{p}) dp_1 dp_2 ... dp_m \tag{2.4}$$

where m indicates the total number of PCCs covering the chip.

A diagrammatic representation of the heuristic discussed in [1] is shown in Figure 3. The diagram shows a reduction from a large number of variables (90 tiles) to the variables used as the centres of 14 clusters (circles labeled MAX_1 through MAX_{14}).

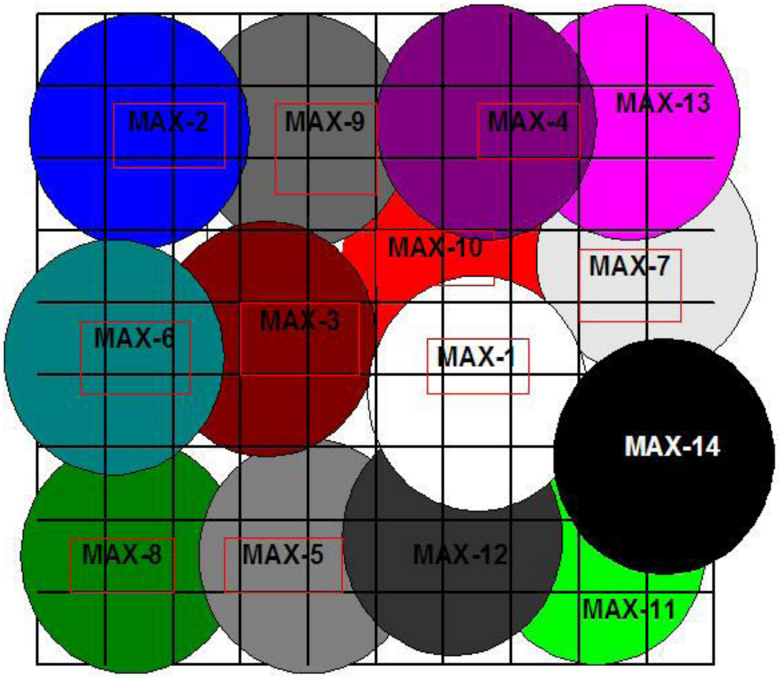

Fig. 3. Perfect Correlation Circles used to Reduce the Number of Variables in Yield Computation

A flowchart describing the procedure for computing High Yield is shown in figure 4.

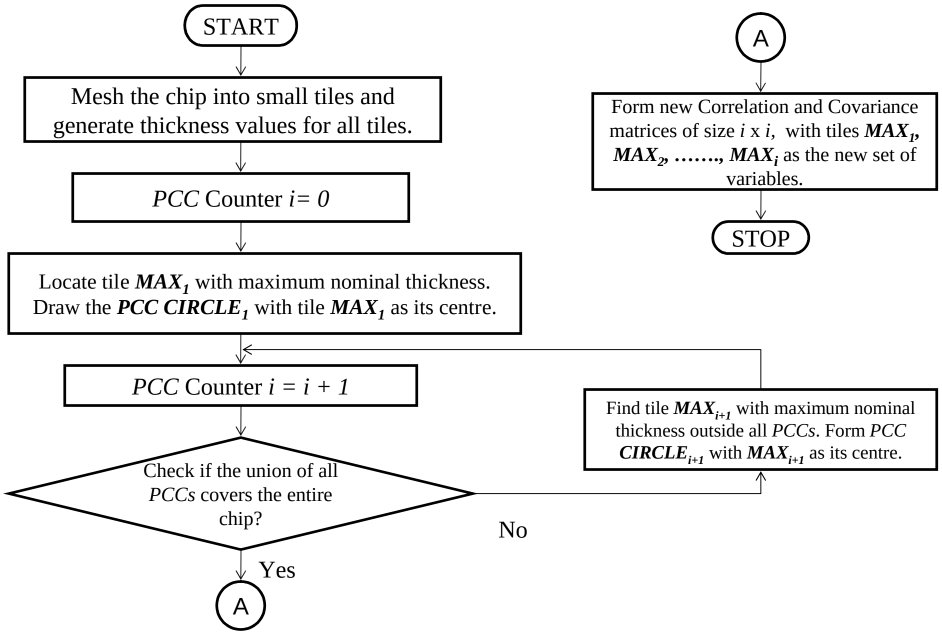
START
Mesh the chip into small tiles and generate thickness values for all tiles.
PCC Counter *i= 0*
Locate tile ***MAX$_1$*** with maximum nominal thickness. Draw the ***PCC CIRCLE$_1$*** with tile ***MAX$_1$*** as its centre.
PCC Counter *i = i + 1*
Check if the union of all *PCCs* covers the entire chip?
Yes
A
No
Find tile ***MAX$_{i+1}$*** with maximum nominal thickness outside all *PCCs*. Form *PCC* ***CIRCLE$_{i+1}$*** with ***MAX$_{i+1}$*** as its centre.
A
Form new Correlation and Covariance matrices of size *i* x *i*, with tiles ***MAX$_1$, MAX$_2$, ……., MAX$_i$*** as the new set of variables.
STOP

Fig. 4. Flowchart Describing the Procedure for Computing High Yield using the PCC Approach

B. Computing Low Yield

The procedure used in computing Low Yield is similar to the one used in computing High Yield with the one major difference being the use of tiles with lowest thickness to identify perfect correlation clusters. To begin with, the nominal thickness values of n tiles are fed as input to the algorithm. The tile with least thickness value if identified (let this tile be called MIN_1). With MIN_1 as centre, the PCC $CIRCLE_1$ is drawn with a pre-fixed radius. Following this, the PCC $CIRCLE_2$ is drawn with the tile MIN_2 (the tile with minimum thickness outside the PCC $CIRCLE_1$) as centre. This procedure is repeated until all regions on the chip are covered by PCCs. The tiles at the centre of all PCCs form the new set of reduced variables.

After reduction, the equation for computing Low Yield would be -

$$Y_L = \int_L^\infty \int_L^\infty ... \int_L^\infty \Phi(\overrightarrow{p}) dp_1 dp_2 ... dp_m \tag{2.5}$$

where m represents the number of reduced variables that indicate the tiles MIN_1, MIN_2, ..., MIN_m at the centre of m PCCs.

C. Computing Overall Yield

As stated in equation 2.1, the overall yield of a chip is given by subtracting one from the sum of the two integrals Y_U, Y_L. Though there is heavy reduction in the size of the correlation matrix by making use of perfect correlation circles, the reduced matrix could still be complex to compute using numerical integrations. The complexity of the resulting calculations may still be taxing as we would needs lots of memory and huge computation time. This situation is solved by making use of *Genz* algorithm [6], [1]. *Genz algorithm* further simplifies the integration problem by transforming the existing covariance matrix using *Cholesky decomposition.* Cholesky decomposition re-expresses the symmetric and positive-definite covariance matrix as a combination of one upper and one lower triangular matrix. The upper and lower triangular matrices also happen to be a transpose of each other. After performing Cholesky decomposition and a sequence of other transformations, the yield equation is reduced to this form -

$$Y = (U - L) \int_0^1 (U - L) \int_0^1 ....(U - L) \int_0^1 dw \tag{2.6}$$

The equation 2.6 helps in reducing complexity by employing a priority ordering in integration. Priority ordering means that the more dominant variables in integration (MAX_1, then MAX_2,.... etc.) are given importance in integration. Also, the new limits of integration are changed to 1 and 0, thus making integration simple. In equation 2.6, the new variable w is independent of the upper and lower thickness limits U and L limits respectively. In general, the Genz algorithm helps in reducing the complexity of numerical integration through techniques such as Cholesky decomposition, priority ordering etc. The algorithm also changes the limits of integration to much simpler values, thereby making the problem simpler as it is now reduced to a sequence of easy multiplications.

The following is a brief recap on the PCC approach. This approach aims at reducing the number of variables involved in yield prediction. Reduction in the number of variables is obtained by the use of perfect correlation circles (PCCs). PCCs are used to cluster a group of tiles under the assumption that the entire set of tiles are perfectly correlated and can from then on be represented by one single tile (the tile with maximum or minimum thickness). After covering the entire chip with such PCCs, the centres of all these PCCs are the new variables of interest. At this stage, the new set of reduced variables help decrease the complexity of the yield computation problem when compared to the large number of thickness variations present initially. Along with a reduction in the number of variables, the reduction also ensures reduced time complexity.

Although this approach ensures heavy reduction in the number of variables and the corresponding run time, there is one drawback in this approach.

In the case where the size of each PCC is big, we would end up having huge reduction in the number of variables. Even though using less variables would ensure a smaller run time it could also result in an over-estimation of yield. On the other hand, when the size of each PCC is small, we would end up having too many variables and not affect the accuracy by a great deal, however the problem will have slower run time. The main reason for these two critical factors to get affected is the use of homogeneously sized PCCs. This trade-off between accuracy in yield value and run time of the problem is caused by the use of homogenously sized clusters.

CHAPTER III

PROPOSED RESEARCH

The variation and yield model used in this work is based on the case of metal thickness like in [1]. Let the metal thicknesses at n locations be represented by a vector $\overrightarrow{p} = (p_1, p_2, ..., p_m)^T$. The thickness at each location can be decomposed as follows -

$$p_i = \mu_i + \delta_i \tag{3.1}$$

$$\mu_i = \overrightarrow{\mu} + \Delta_i \tag{3.2}$$

where $\overrightarrow{\mu}$, Δ_i and δ_i indicate the nominal value and systematic and random variations respectively. All the components of our yield model are shown in Figure 5.

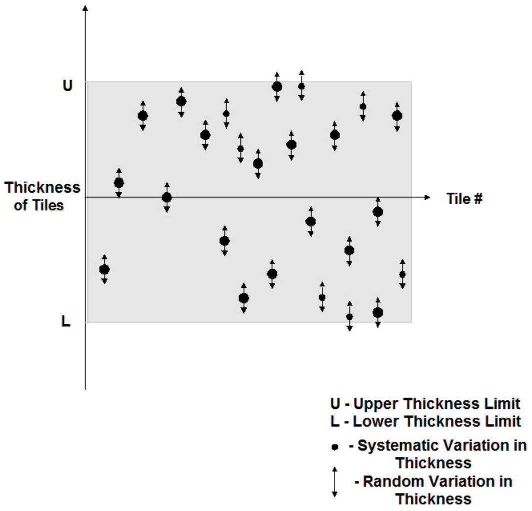
U
L
Thickness of Tiles
Tile #
U - Upper Thickness Limit
L - Lower Thickness Limit
● - Systematic Variation in Thickness
↕ - Random Variation in Thickness

Fig. 5. Systematic and Random Variations in CMP Yield Model

In the figure, the nominal thickness is a constant value and is shown using a central horizontal line. The systematic variation (Δ_i) at any location is dependent on the layout pattern around that location and is a deterministic value. Consequently, the term μ_i maybe termed as the deterministic variation. The deterministic variations are shown using black dots in Figure 5. The random variations (inter-die and intra-die variations included) are represented by double sided arrows. The random variations are assumed to follow a normal distribution with roughly equal variance (just the same way as in [1]).

The thickness vector $\overrightarrow{p}$ can be represented by a multivariate normal distribution $N(\overrightarrow{\mu};\Sigma)$ where $\overrightarrow{\mu} = (\mu_1, \mu_2, ..., \mu_n)^T$ and Σ is an $n \times n$ covariance matrix. The joint distribution function may now be written as -

$$\phi(\overrightarrow{p}) = \frac{e^{-0.5\cdot(\overrightarrow{p}-\overrightarrow{\mu})^T\cdot\Sigma^{-1}\cdot(\overrightarrow{p}-\overrightarrow{\mu})}}{\sqrt{(2\pi)^n\cdot|\Sigma|}} \tag{3.3}$$

According to Figure 5 CMP Yield may also be stated as the probability for thickness variations at all locations to stay within the shaded region (region within the upper and lower thickness limits). When monitoring thickness values on a chip, the entire chip is tesselated into a large number of tiles. The thickness variation at a tile Γ_i is characterized by a variable p_i. In order to get an accurate estimate of yield, a large number of locations need to be monitored.

The method of using perfect correlation circles to reduce the size of the yield prediction problem was discussed in the previous chapter. The objective of this approach is to reduce the complexity of the yield prediction problem (size and run time of the problem) by reducing the number of variables used in calculating yield. The accuracy in yield value depends on the size of the PCCs. Depending on the size of each PCC, the accuracy in yield and run time for the algorithm were seen to be inversely related.

This thesis attempts at eliminating the problem of accuracy in the predicted yield value by proposing two new techniques for reduction in the number of variables used for computing yield namely - *Principal Component Analysis (PCA)* and *Hierarchical Adaptive Quadrisection (HAQ)*. In the PCA approach, the presence of partial correlations between thickness variations at various locations is eliminated by transforming the correlated variations to an orthogonal base. After achieving orthogonality, the number of variables are reduced to ease the complexity of yield prediction. HAQ on the other hand is similar to the PCC approach where a divide and conquer based clustering approach is followed to bring about reduction. One difference between the two approaches is the use of heterogeneously sized cluster in out HAQ approach in order to reduce the number of variations with minimal compromise on the accuracy in yield value.

A. Principal Component Analysis

Intra-die variations add a large number of correlations to the yield model. The goal of PCA is to compute the most meaningful basis for re-expressing the correlated variables into an independent set of reduced number of variables through an orthogonal base. Determining an orthogonal base gives users the flexibility of discerning any number of variables with ease as all variables are now independent and no longer correlated. This means that that a user may now remove any redundant variable or other variables which are just noise [7].

1. Math behind PCA

In the CMP yield model, we consider n metal variations in thickness as n random variables. Let these variations be represented by a thickness vector $\delta_{n\times1}$. So we have

-

$$\overrightarrow{\delta(n)} = [\delta_1, \delta_2, ..., \delta_n] \tag{3.4}$$

where δ_1, δ_2, ..., δ_n - represent the thickness variations at n locations.

Let the correlations between these n locations be represented by a correlation matrix $\Gamma(\overrightarrow{\delta})$ of size $n \times n$.

$$\Gamma(\overrightarrow{\delta}) = \Gamma_{ij_{n\times n}} \tag{3.5}$$

Let the variance of each variable be σ_i^2. The covariance matrix $\Sigma_{n\times n}$ can now be obtained from the correlation matrix as follows -

$$\Sigma(\overrightarrow{\delta}) = \Gamma(\overrightarrow{\delta})_{n\times n} \times \sigma_i \cdot \sigma_j \tag{3.6}$$

The covariance matrix is symmetric and contains positive entries. By definition, the covariance is a measure of the linearity in relationship between two variables and the variance is a measure of the deviation of any variable from the mean value. Hence orthogonality in a given set of variables can be achieved by maximizing the main impact of the variables measured using variance and minimizing the redundancy in variables measured by covariance. This can be achieved by eigenvalue decomposition. Eigenvalue decomposition allows us to express a symmetric covariance matrix as follows -

$$\Sigma(\overrightarrow{\delta}) = Q \cdot \Lambda(\overrightarrow{\delta}) \cdot Q^T \tag{3.7}$$

where $\Lambda(\overrightarrow{\delta})$ is a diagonal matrix of size $n \times 1$ containing eigenvalues for the covariance matrix and Q is a $n \times n$ matrix with column vectors representing the corresponding eigenvectors. The diagonal matrix $\Lambda(\overrightarrow{\delta})$ will look like -

$$\Lambda(\overrightarrow{\delta}) = \begin{pmatrix} \lambda_1 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & \lambda_2 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & \lambda_n \end{pmatrix} \tag{3.8}$$

where $\lambda_1 \geq \lambda_2 \geq .... \geq \lambda_n$. Using eigenvalue decomposition helps us in two ways. Firstly, it gives us the dominant directions in the covariance relationship between the original set of variables through a diagonal matrix $\Lambda(\overrightarrow{\delta})$ and secondly, it gives us an idea of how to map the original set of variables in a new uncorrelated set. Let the new set of uncorrelated variables $\epsilon_{n\times 1}$ be related to the original set of variations $\delta_{n\times 1}$ through a $n \times n$ matrix B as follows -

$$\overrightarrow{\delta} = B \cdot \overrightarrow{\epsilon} \tag{3.9}$$

where B is a $n \times n$ matrix. Without loss of generality, it can be assumed that the transformed sources of variations follow a Gaussian distribution such that -

$$\mu(\overrightarrow{\epsilon}) = 0 \tag{3.10}$$

$$\Lambda(\overrightarrow{\epsilon}) = I \tag{3.11}$$

One can easily deduce the presence of matrix J with dimensions $n \times n$ such that -

$$\Lambda(\overrightarrow{\delta}) = J \cdot \Lambda(\overrightarrow{\epsilon}) \cdot J^T \tag{3.12}$$

It can easily be seen that J is a diagonal matrix and looks as follows -

$$J = \begin{pmatrix} \sqrt{\lambda_1} & 0 & \cdot & \cdot & 0 \\ 0 & \sqrt{\lambda_2} & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \sqrt{\lambda_n} \end{pmatrix} \tag{3.13}$$

J together with Q gives the map $B = Q \cdot J$ that is used to transform the original set of variables δ into the orthogonal vector ϵ.

$$\overrightarrow{\delta} = B \cdot \overrightarrow{\epsilon} = Q \cdot J \cdot \overrightarrow{\epsilon} \tag{3.14}$$

Similarly, the covariance matrix $\Sigma_{n\times n}$ can be rewritten as follows -

$$\Sigma(\overrightarrow{\delta}) = Q \cdot \Lambda(\overrightarrow{\delta}) \cdot Q^T = Q \cdot J \cdot \Lambda(\overrightarrow{\epsilon}) \cdot (Q \cdot J)^T \tag{3.15}$$

Hence, from the above set of equations we see how to transform a set of correlated variables into a new set of uncorrelated variables through an orthogonal basis. Now we shall see how to obtain a reduction in the number of variables.

2. Reduction in Number of Variables

In the previous sub-section we learnt to derive an independent set of orthogonal vectors from a correlated set through eigenvalue decomposition. Through eigenvalue decomposition we get a set of eigenvalues $\Lambda(\overrightarrow{\delta})$ for the correlated set of variables $\overrightarrow{\delta} = [\lambda_1, \lambda_2, ..., \lambda_n]$ such that $\lambda_1 \geq \lambda_2 \geq .....\lambda_n$. It is possible that many of the eigenvalues might be very small or some may even be redundant values. By neglecting such repeating or small eigenvalues, we can reduce the number of variables in the problem. Suppose, after reduction we are left with k variables, then the matrix $\Lambda(\overrightarrow{\delta})$ of size $k \times k$ will look like -

$$\Lambda(\overrightarrow{\delta}) = \begin{pmatrix} \lambda_1 & 0 & \cdot & \cdot & 0 \\ 0 & \lambda_2 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & \lambda_k \end{pmatrix} \tag{3.16}$$

Correspondingly, because $\Lambda(\overrightarrow{\delta}) = J \cdot J^T$, the matrix $J_{k\times k}$ becomes -

$$J = \begin{pmatrix} \sqrt{\lambda_1} & 0 & \cdot\ \cdot & 0 \\ 0 & \sqrt{\lambda_2} & \cdot\ \cdot & 0 \\ 0 & 0 & \cdot\ \cdot & 0 \\ 0 & 0 & \cdot\ \cdot & 0 \\ 0 & 0 & \cdot\ \cdot & 0 \\ 0 & 0 & \cdot\ \cdot & \sqrt{\lambda_k} \end{pmatrix} \tag{3.17}$$

Since the size of matrix J is $k \times k$, the corresponding sizes of matrices B and Q through the equation $B = Q \cdot J$ become $m \times k$. The matrix B of reduced size can be used to map the initial thickness vector $\overrightarrow{\delta}$ to a reduced set of uncorrelated variations $\overrightarrow{\epsilon}$. Thus orthogonality is obtained by using eigenvalue decomposition.

In short, the PCA approach consists of the following four important steps to be executed in the following sequence -

1. Form the vector composing of the correlated set of metal thickness variations and the corresponding correlation and covariance matrices.

2. Perform eigenvalue decomposition.

3. Calculate the mapping matrix B for transforming the correlated variations into a new set of uncorrelated variations.

4. Compute the new thickness vector after discerning unwanted eigenvalues.

B. Hierarchical Adaptive Quadrisection

In the previous chapter we read about the inaccuracy in yield value arising in the PCC method due to the use of homogeneously sized clusters. This thesis proposes the use of a reduction technique called Hierarchical Adaptive Quadrisection HAQ that helps in reducing inaccuracy through the use of a divide and conquer based clustering approach where all clusters are heterogeneously sized. Heterogeneous clustering helps in maintaining a sufficient amount of variables after reduction thus ensuring minimal compromise over yield accuracy. In case of the PCC approach, the size of each cluster was maintained at a pre-fixed value, in our approach, the size of each cluster is decided based upon the systematic variations inside the cluster. The extent of clustering is dependant upon the deviations in systematic variations of different tiles within a cluster. A cluster if further internally grouped into smaller clusters if the deviations in systematic variations inside the sub-groups are found sensitive with respect to a certain pre-fixed threshold thickness variation.

The HAQ approach is explained in detail in the subsequent paragraphs. Similar to the PCC approach, overall yield is computed from two separate functions, High Yield (probability of thicknesses at all locations being lesser than the upper thickness limit) and Low Yield (probability of thicknesses at all locations being greater than the lower thickness limit). We shall see a working model explaining the reduction process in computing High Yield. The same approach can be applied in computing Low Yield as well. Computing Low Yield using the same algorithm is explained in brief.

1. Hierarchical Adaptive Quadrisection for Computing High Yield

Clustering in our Hierarchical Adaptive Quadrisection approach is done using basic sub-regions. Similar to the PCC approach, basic sub-regions are made up of a group of tiles and each basic sub-region is represented by a single random variable. The size of each sub-region is not homogeneous as in the case of [1]. The sub-regions are heterogeneously sized and the size of each sub-region depends on the systematic variations within the cluster.

To begin with, the entire chip is divided into an array of relatively large sized sub-regions (or) a coarse-grained array. We then perform the HAQ procedure separately on each of these sub-regions. For any sub-region S, at first we find the tile Γ_i of maximum deterministic thickness in that sub-region. So, in sub-region S we have -

$$\mu_i = \mu_{max} \tag{3.18}$$

Next, we temporarily quadrisect the sub-region S into four equally sized plates P_1, P_2, P_3, P_4. One of these four plates will contain the tile Γ_i. Let this plate be called the Critical Plate. So the sub-region S is composed of *one critical plate* and *three non-critical plates*. In each non-critical plate, we identify one tile with maximum deterministic thickness $\mu_{j,max}$. We then compute a Difference Vector d which holds the difference in thickness between the tile μ_i and the tiles $\mu_{j,max}$. The vector d may be computed as follows -

$$d = \forall_{j=1,2,3,4,j\neq i} |\mu_i - \mu_{j,max}| \tag{3.19}$$

Following this, the Critical Difference Value (the minimum value in the vector d) is computed. Let the critical difference value be called d_{min}.

$$d_{min} = min(d) \tag{3.20}$$

After computing d_{min}, the possibility of any further clustering within the sub-region S is determined by the critical difference value d_{min} and a pre-fixed Threshold Thickness Value θ using the following condition.

The most important step in the algorithm is explained using equation 3.21. This condition dictates the further course of action in this algorithm by deciding if a sub-region needs further quadrisection or not. Both possibilities are evaluated based on the sensitivity between the maximum thickness variations in the critical and non-critical plates in a sub-region. The impact on sensitivity between thickness variations is explained the next few paragraphs.

$$d_{min} \leq \theta \tag{3.21}$$

If the above condition is satisfied, it means that the difference in thickness values between the tile μ_i and tiles of maximum thickness in the three non-critical plates, is less. This in turn indicates the possibility for other tiles in the sub-region S being close to the upper thickness limit and hence the chance of other tiles in the sub-region satisfying the upper thickness constraint. So, the tile Γ_i can no longer be used to represent the thickness variations of all other tiles in the sub-region S. Following such a result, we make the four temporarily quadrisected plates into four new sub-regions and repeat the HAQ procedure in each of these sub-regions to investigate the possibility of any further quadrisection.

On the other hand, in case the condition in equation 3.21 is not satisfied, the difference between the thickness at tile Γ_i and other tiles of maximum thickness $\mu_{j,max}$ is large. Such a large difference in thickness indicates the fact that none of the other tiles in the sub-region S have a chance of satisfying the upper thickness constraint. In other words, if the thickness of any tile $\mu_{j,max} <$ U, then we can safely assume that the thickness of all tiles in the non-critical plates are no greater than upper thickness limit. Therefore, the probability of satisfying the upper thickness constraint is approximately decided by the tile Γ_i and further quadrisection on the sub-region S is unnecessary.

This procedure is repeated until we reach a stage where there is no possibility of any further quadrisection. The set of sub-regions covering the chip form the reduced set of variables and the maximum thickness variation inside each sub-region is used to represent all variations inside the sub-region. This explains how the HAQ procedure may be used to bring about reduction from a large number of tiles to a smaller number of basic sub-regions.

As the algorithm progresses, we can clearly see a decrease in the size of every newly formed sub-region (fine-grained clustering). Varying cluster size helps in maintaining accuracy of the yield prediction. This fact is clearly illustrated in the next sub-section. Using the HAQ approach for computing High Yield is illustrated using the algorithm and flowchart shown in Figures 6 and 7.

2. Working Model for High Yield using HAQ

Let us consider a chip with dimensions $0.16\mu m \times 0.16\mu m$. The chip is broken into small tiles, each of size $10\mu m \times 10\mu m$. This would leave us with a total of *256 tiles* (initial set of variables before reduction) covering the chip. Let the pre-fixed threshold thickness value be 10. To begin with, let the entire chip be considered as a basic sub-region S. Let us temporarily quadrisect the sub-region S into four plates P_1, P_2, P_3, P_4. At this stage, the entire setup would appear as follows -

The sub-region S is shown in thick lines while the four temporarily quadrisected plates P_1, P_2, P_3 and P_4 are shown in dotted black lines. Let the maximum thickness values in the four plates be as shown in Figure 8. As written in Figure 8, let the maximum thickness variations in the plates P_1, P_2, P_3 and P_4 be 93, 97, 95 and 94 respectively. Based on the variations in Figure 8, the plate labeled P_2 is the critical plate and the other plates P_1, P_3 and P_4 are the non-critical plates. Given the

```
Procedure: $HierarchicalAdaptiveQuadrisection(S)$

Input: A layout region $S$ consisting of an array of tiles
Output: A set of sub-regions $P$ covering $S$

Find tile $\tau_i \in S$ with maximum deterministic thickness $\mu_{max}$
Temporarily quadrisect $S$ into plates $\{P_1, P_2, P_3, P_4\}$
Identify critical plate $P_k$ containing the tile $\tau_i$
Find the maximum deterministic tile thickness $\mu_{j,max}$
        for all plates except $P_k$
Compute Critical Difference $d_{min} = \min_{\forall j \in \{1,2,3,4\},\ j \neq k}(\mu_{max} - \mu_{j,max})$
If $d_{min} >$ Threshold $\theta, P \leftarrow S$
Else
    $P \leftarrow \emptyset$
    For $j = 1$ to $4$
        $P \leftarrow P \cup HierarchicalAdaptiveQuadrisection(P_j)$
Return $P$
```

Fig. 6. Algorithm of Hierarchical Adaptive Quadrisection.

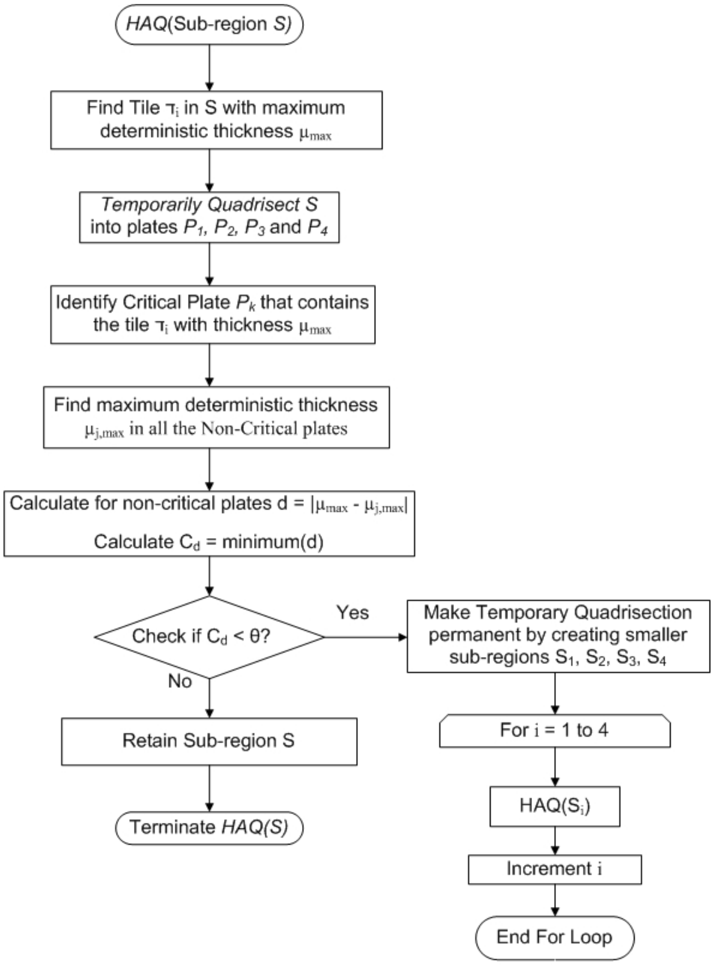
HAQ(Sub-region S)
Find Tile ㄱi in S with maximum deterministic thickness μmax
Temporarily Quadrisect S into plates P1, P2, P3 and P4
Identify Critical Plate Pk that contains the tile ㄱi with thickness μmax
Find maximum deterministic thickness μj,max in all the Non-Critical plates
Calculate for non-critical plates d = |μmax - μj,max|
Calculate Cd = minimum(d)
Check if Cd < θ?
Yes
No
Make Temporary Quadrisection permanent by creating smaller sub-regions S1, S2, S3, S4
For i = 1 to 4
HAQ(Si)
Increment i
End For Loop
Retain Sub-region S
Terminate HAQ(S)

Fig. 7. HAQ for High Yield

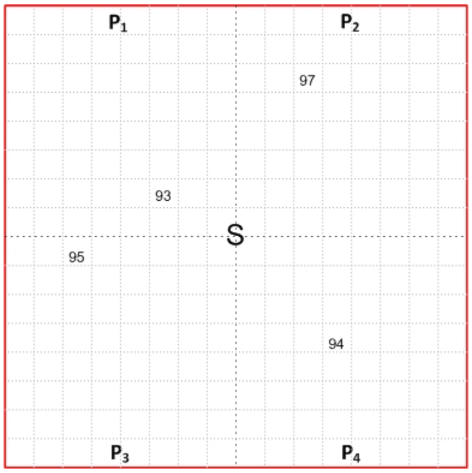

Fig. 8. Working Model to Compute High Yield using HAQ - Stage 1

thickness variations in all four plates, the critical difference value C_d in sub-region S can easily be calculated 2 ($97 - 95 = 2$). As stated in the previous paragraph, the threshold thickness value is 10. Since the critical difference value (2) is less than the threshold (10), according to the HAQ algorithm, we make the quadrisection permanent. This means the plates P_1, P_2, P_3 and P_4 from now on will become permanent sub-regions and will be identified as S_1, S_2, S_3 and S_4 respectively.

Table I illustrates all the conditions that led to the formation of sub-regions S_1, S_2, S_3 and S_4.

Table I. Working Model to Compute High Yield using HAQ - Stage 1

Sub-Region Monitored	Max Thickness		d_{min}	$d_{min} \leq \theta$	Next Action
	Critical	Non-Critical			
S	97	93, 95, 94	2	Yes	Quadrisect S

This completes Stage 1 of the HAQ algorithm. At the end of Stage 1, we have a *total of 4 sub-regions.* Now we need to monitor the sub-regions S_1, S_2, S_3 and S_4 independently. Let the temporary plates in these sub-regions be (P_{11}, P_{12}, P_{13}, P_{14}),

(P_{21}, P_{22}, P_{23}, P_{24}), (P_{31}, P_{32}, P_{33}, P_{34}) and (P_{41}, P_{42}, P_{43}, P_{44}) respectively. Our next action is to find the maximum thickness variations in all these 16 plates. Let these thickness values be (85, 78, 81, 93), (97, 83, 79, 86), (95, 76, 73, 80) and (94, 88, 84, 89). The newly formed sub-regions (S_1, S_2, S_3, S_4) and the corresponding temporarily quadrisected plates (P_{11}, P_{12}, P_{13}, P_{14}), (P_{21}, P_{22}, P_{23}, P_{24}), (P_{31}, P_{32}, P_{33}, P_{34}) and (P_{41}, P_{42}, P_{43}, P_{44}), along with their maximum thickness variations are all shown in Figure 9.

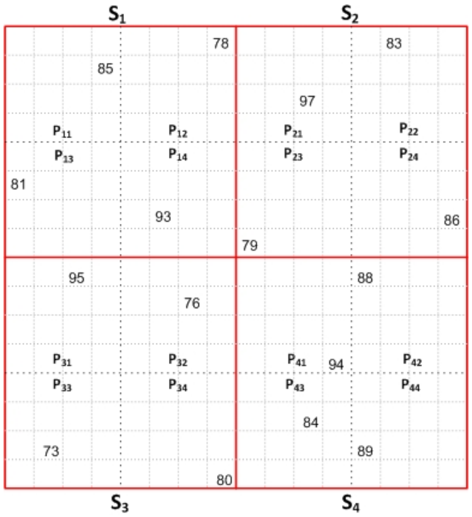

Fig. 9. Working Model to Compute High Yield using HAQ - Stage 2, Step 1

Based on the thickness variations in each of the 16 temporarily quadrisected plates, further course of action may be comprehended from Table II.

Table II. Working Model to Compute High Yield using HAQ - Stage 2

Sub-Region Monitored	Max Thickness		d_{min}	$d_{min} \leq \theta$	Next Action
	Critical	Non-Critical			
S_1	93	85, 78, 81	8	Yes	Quadrisect S_1
S_2	97	83, 79, 86	11	No	Retain S_2
S_3	95	76, 73, 80	15	No	Retain S_3
S_4	94	88, 84, 89	5	Yes	Quadrisect S_4

Based on the observations in Table III, we retain sub-regions S_2 and S_3, and do further quadrisection on sub-regions S_1 and S_4. Both these sub-regions will now be divided into sub-regions S_{11}, S_{12}, S_{13}, S_{14} and S_{41}, S_{42}, S_{43}, S_{44} respectively. Further, let these newly formed sub-regions be temporarily divided into *32 plates*. These newly formed sub-regions, the temporary plates and the maximum thickness values in each of the plates are all shown in Figure 10. This completes Stage 2 of the HAQ algorithm.

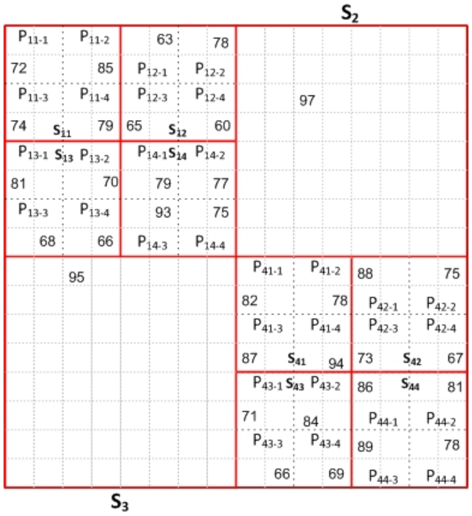

Fig. 10. Working Model to Compute High Yield using HAQ - Stage 2, Step 2

At the end of Stage 2, we have a total of *10 sub-regions*. Now we begin Stage 3 of the HAQ algorithm. Based on the thickness values shown in Figure 10, the following observations can be made -

Based on the values in Table III, we retain all sub-regions except for S_{11} and S_{44}. These two sub-regions need further quadrisection. This completes Stage 3 of the HAQ algorithm. At the end of Stage 3, the setup will look as in Figure 11.

At the end of Stage 3, we have a total of *19 sub-regions*. Whether we do further

Table III. Working Model to Compute High Yield using HAQ - Stage 3

Sub-Region Monitored	Max Thickness		d_{min}	$d_{min} \leq \theta$	Next Action
	Critical	Non-Critical			
S_{11}	85	72, 74, 79	6	Yes	Quadrisect S_{11}
S_{12}	78	63, 65, 60	15	No	Retain S_{12}
S_{13}	81	70, 68, 66	11	No	Retain S_{13}
S_{14}	93	79, 77, 75	14	No	Retain S_{14}
S_{41}	94	82, 78, 87	12	No	Retain S_{41}
S_{42}	88	75, 73, 67	13	No	Retain S_{42}
S_{43}	84	71, 66, 69	13	No	Retain S_{43}
S_{44}	89	86, 81, 78	3	Yes	Quadrisect S_{44}

quadrisection in sub-regions S_{11} and S_{44} or not depend on the maximum thickness variations inside these two sub-regions. We will stop further illustration of the working model at this point. The HAQ algorithm will run until there is no need for quadrisection. At this point, the total number of sub-regions indicate the reduced set of variables and the maximum thickness variations in each of the sub-regions will be the new set of variables.

3. Comparing the PCC and HAQ Approaches

We will use the working model discussed in the previous sub-section to compare the PCC and HAQ approaches. In Figure 10 we see the HAQ procedure clustering the chip into *19 clusters of different sizes.* If the PCC approach had been applied at the same stage of clustering, the result would have been *64 equally sized clusters.*

The use of clusters with varying sizes gives the user the flexibility of doing both fine and coarse grained clustering, and thus also helps in preserving accuracy of the computed yield value.

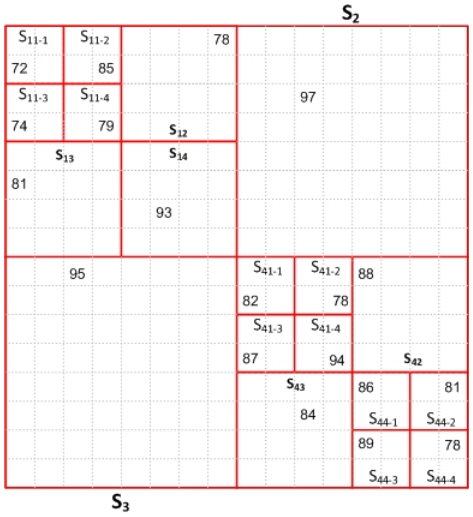

Fig. 11. Working Model to Compute High Yield using HAQ - Stage 3

4. Computing Low Yield using the HAQ Approach

The procedure to evaluate Low Yield using the HAQ approach is quite similar to the one described in the preceding paragraphs. To begin with, the chip is divided into an array of relatively large sized sub-regions and the HAQ procedure is performed separately on each of these sub-regions. For any sub-region S, we find the tile Γ_i of minimum deterministic thickness in that sub-region μ_{min}. Then, the sub-region is temporarily quadrisected into four plates (P_1, P_2, P_3, P_4). The minimum deterministic thickness variations in each of these four plates is identified as $\mu_{min,j}$. One of these four variations will coincide with the thickness variation in tile Γ_i. Let this plate be labeled as a critical plate and the others as non-critical plates. The next step is to compute the difference vector and the critical difference value. The difference vector holds the difference in thickness between the minimum thickness variation in the critical plate and the minimum thickness variations in the three non-critical plates. The critical difference value is the minimum value in the difference vector. Following this, we then check if the critical difference value is comparable to the pre-fixed threshold value or not. The outcome of this step decides the succeeding sequence of actions.

In case, the critical difference value is greater than the threshold value, it means that the thickness variations in the non-critical plates are greater than the minimum thickness variation in the critical plate. This indirectly points to a scenario where no tile in the non-critical plates is likely to satisfy the lower thickness bound. In such a case, the minimum thickness variation in the critical plate (the variation μ_{min} at tile Γ_i is sufficient to represent all variations in the sub-region S when computing low yield. So we retain the sub-region as it is and use the thickness at tile Γ_i to represent all thickness variation in the sub-region. On the other hand if the critical difference value is lesser than or equal to the threshold thickness value, it means that the minimum thickness variation in one or more non-critical plates is comparable with the minimum thickness variation in the critical plate. Comparable thickness variation in the four plates imply a possibility for the tiles in the non-critical plates to also satisfy the lower thickness constraint. In such a case, we do further quadrisection by converting the four temporary plates into four permanent sub-regions and the same procedure is repeated independently in all the four newly formed sub-regions.

The above mentioned procedure is repeated until there is no longer a need to perform further quadrisection. After completing the HAQ procedure, the reduced set of variables is the set of sub-regions covering the chip. The tile with minimum thickness variation in each sub-region is used to represent the thickness variations in the sub-region.

5. Working Model for Low Yield using HAQ

A chip with dimensions similar to the one used in the working model for High Yield is used here to describe HAQ for Low Yield. Let the pre-fixed threshold value be

10. To begin with, let the entire chip be considered as a basic sub-region S. Let the temporarily quadrisected plates be (P_1, P_2, P_3, P_4). At this stage, the entire setup would appear as follows -

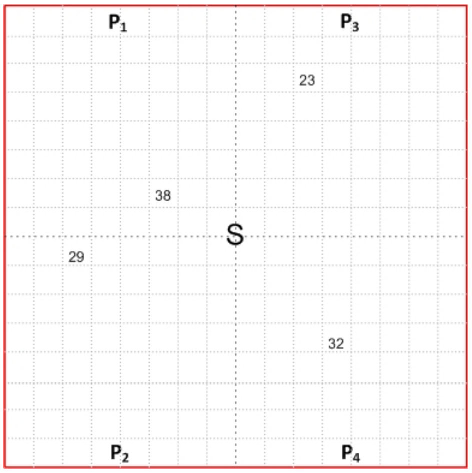

Fig. 12. Working Model to Compute Low Yield using HAQ - Stage 1

The sub-region S is shown in thick lines while the four temporarily quadrisected plates P_1, P_2, P_3 and P_4 are shown in dotted black lines. Let the minimum thickness values in the four plates be as shown in Figure 12. Based on these values, we make the temporary plates permanent and label the new sub-regions as S_1, S_2, S_3 and S_4. Table IV and Figure 13 illustrate all the conditions that led to the formation of sub-regions S_1, S_2, S_3 and S_4.

Table IV. Working Model to Compute Low Yield using HAQ - Stage 1

Sub-Region Monitored	Min Thickness		d_{min}	$d_{min} \leq \theta$	Next Action
	Critical	Non-Critical			
S	23	38, 29, 32	2	Yes	Quadrisect S

This completes Stage 1 of the HAQ algorithm. In the next stage, let the minimum thickness variations inside the sub-regions look as shown in Figure 13.

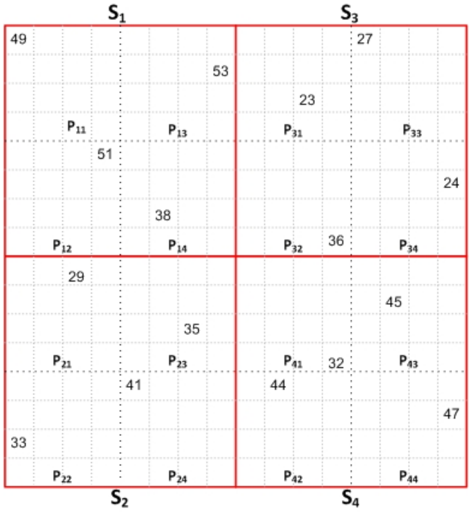

Fig. 13. Working Model to Compute Low Yield using HAQ - Stage 2, Step 1

Based on the values in Figure 13, the future course of action will be dictated by the observations in Table V.

Table V. Working Model to Compute Low Yield using HAQ - Stage 2

Sub-Region Monitored	Min Thickness		d_{min}	$d_{min} \leq \theta$	Next Action
	Critical	Non-Critical			
S_1	38	49, 51, 53	11	Yes	Retain S_1
S_2	29	33, 35, 41	4	No	Quadrisect S_2
S_3	23	36, 27, 24	1	No	Quadrisect S_3
S_4	32	44, 45, 47	12	Yes	Retain S_4

Based on the calculations in Table V, sub-regions S_1 and S_4 are retained while sub-regions S_2 and S_3 are further analyzed. This completes Stage 2 of the HAQ process. At the end of this stage we have *10 sub-regions*. Based on the thickness variations in each of the *16 temporarily quadrisected plates* in the newly formed sub-regions, the setup will look as follows -

The next action can be comprehended from Table MIN-Stage2.

Based on the observations made in Table VI, we end up with a total of *19 sub-*

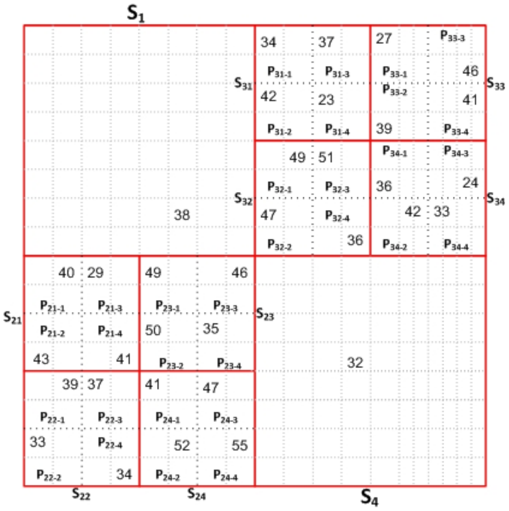

Fig. 14. Working Model to Compute Low Yield using HAQ - Stage 2, Step 2

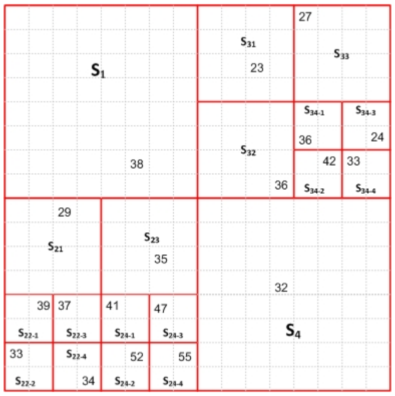

Fig. 15. Working Model to Compute Low Yield using HAQ - Stage 3

Table VI. Working Model to Compute Low Yield using HAQ - Stage 3

Sub-Region	Min Thickness		d_{min}	$d_{min} \leq$	Next
Monitored	Critical	Non-Critical		θ	Action
S_{21}	29	40, 43, 41	1 1	Yes	Retain S_{21}
S_{22}	33	39, 37, 34	1	No	Quadrisect S_{22}
S_{23}	35	49, 50, 46	11	No	Retain S_{23}
S_{24}	41	52, 47, 55	6	No	Quadrisect S_{24}
S_{31}	23	34, 42, 37	11	No	Retain S_{31}
S_{32}	36	49, 51, 47	11	No	Retain S_{32}
S_{33}	27	39, 46, 41	12	No	Retain S_{33}
S_{34}	24	36, 42, 33	9	Yes	Quadrisect S_{34}

regions. This completes Stage 3 of the HAQ algorithm. The setup after Stage 3 will look as shown in Figure 15. We will stop further analysis of the working model for Low Yield. The quadrisection will continue until we reach a point where there is no further possibility for quadrisection. This is the final step in the reduction process. The tiles with minimum thickness variations in all the sub-regions form the new set of variables.

CHAPTER IV

SIMULATION RESULTS

A. Experimental Setup

All algorithms discussed in previous chapters were carried out in MATLAB. As we did not have exact chip details of interconnect thickness profiles, the inputs are the same as given in [1]. The experimental setup consists of a chip with dimensions 4.8 $mm \times 7.5$ mm. The chip is tesselated into a 480 × 750 array of tiles where each tiles is of size 10 $\mu m \times 10$ μm. This means the size of our input thickness vector would be $360,000 \times 1$. For all experiments, the upper and lower thickness limits for interconnects in each layer are 0.4580 μm and 0.2580 μm respectively. So in short, the yield would be computed as the probability of thickness values in all tiles to lie within the above specified upper and lower thickness limits.

The nominal thickness values for all tiles were generated generated based on a normal distribution with a mean thickness value of 0.3580 μm and a standard deviation of 0.02 μm. For all experiments, the variance value used in computing the covariance matrix is 0.0009 μm^2. The input thickness values for all simulations were generated for specific seed values. Spatial correlation between different tiles was taken into account. A linear reduction in correlation was assumed with increase in distance. The distance between different tiles was calculated between their centers. Yield results were obtained for all algorithms using different cases of correlation equations.

B. Monte Carlo Experiments

In addition to the algorithms discussed in previous chapters, the yield is also computed using *Monte Carlo (MC)* methods (with and without spatial correlation). Monte

Carlo simulations were performed in order to compare and validate the results obtained using other methods. The random thickness values at each iteration are obtained using the nominal thicknesses as mean value and a standard deviation of 0.03 μm. In the first case where there is no spatial correlation, the experiment checks at each iteration whether thickness values of all tiles lie within the lower and upper thickness limits. The ratio of successful iterations to the total number of iterations is calculated as yield.

In case of Monte Carlo with spatial correlation, after getting the initial thickness values of all tiles at each iteration, the thickness vector is reconstructed from its principal components [8], [9]. No reduction was implemented after computing the principal components. Following reconstruction of the thickness vector, it is checked if the thickness vector falls within the upper and lower thickness limits. Again the ratio of successful iterations to the total number of iterations is calculated as yield. These simulations were performed for three different correlation equations, each with three different seed values in order to monitor variations in yield with initial seed for a given iteration count.

C. Experimental Results

Yield results were obtained for all algorithms using three different cases of correlation equations namely, $-3\times10^{-5}x+0.9958$, $-4\times10^{-5}x+0.9958$ and $-2\times10^{-5}x+0.9958$. These results for different correlation equations are presented in the subsequent subsections in the following order - 1. MC method, 2. PCC method, 3. PCA method 4. HAQ method and 5. a combination of the HAQ and PCA approaches. In order to facilitate comparison of accuracy and time complexity of all algorithms discussed in the previous chapters, interconnect thickness values are generated for each correlation case with the same initial seed.

1. Correlation Equation: $-3 \times 10^{-5}x + 0.9958$

Yield results obtained for this case of correlation are shown below. Table VII shows yield values computed using Monte Carlo methods (with and without correlation). In the case of Monte Carlo with correlation, the initial seed values used in generating nominal thickness values were varied. Table VIII shows the the yield computed using the PCC approach. Variations in yield value with radius of PCCs are shown. Table IX shows yield values obtained using PCA. The different eigenvalues showing reduction and its corresponding yield values are also shown in Table IX. Table X shows the results obtained using HAQ. The threshold thickness values were varied so that we could have cases where the order of the reduction is varied. The same changes in threshold value are made for other correlation equations as well. The accuracy of yield using both HAQ and PCA algorithms is observed in Table XI. The reduction in this case is done using the HAQ approach. Following reduction, the new set of variables are transformed into an un-correlated set through PCA and yield is then calculated using this new set orthogonal variables.

Genz algorithm was used in computing yield for all cases except Monte Carlo. In tables with results obtained using the PCA, HAQ and PCC approaches, the terms Y_{max} and Y_{min} refer to the sizes of covariance matrices keeping maximum and minimum interconnect thickness profiles in mind.

Table VII. Correlation = $-3 \times 10^{-5}x + 0.9958$. MC Simulations

Iterations	Monte Carlo without PCA (Initial Seed = 5)		Monte Carlo with PCA		
	Yield	CPU Run Time	Seed	Yield	CPU Run Time
10,000	61.17%	1296.517 secs	5	78.17%	1439.247 secs
			20	77.82%	1421.388 secs
			50	78.35%	1447.736 secs
30,000	60.39%	3727.310 secs	5	76.61%	4182.968 secs
			20	76.18%	4209.535 secs
			50	75.96%	4193.036 secs
50,000	59.83%	6191.005 secs	5	74.39%	6541.266 secs
			20	75.02%	6449.612 secs
			50	74.47%	6518.389 secs

Table VIII. Correlation = $-3 \times 10^{-5}x + 0.9958$. PCC Approach. Initial Seed = 5

Radius of PCCs	Size of Covariance Matrix (Y_{max}/Y_{min})	Yield	CPU Run Time
150 μm	432/427	87.58%	2237.944 secs
250 μm	305/310	88.37%	1619.172 secs
350 μm	194/197	89.18%	1193.237 secs
500 μm	93/98	89.73%	582.378 secs
600 μm	68/72	90.52%	366.241 secs
800 μm	43/41	91.06%	240.516 secs
1000 μm	29/30	91.82%	158.577 secs
2000 μm	10/9	92.75%	54.782 secs

2. Correlation Equation: $-4 \times 10^{-5}x + 0.9958$

Tables XII - XVI show the results obtained for the correlation equation shown above in the same order as the preceding tables.

3. Correlation Equation: $-2 \times 10^{-5}x + 0.9958$

Tables XVII - XXI show the results obtained for the correlation equation shown above in the same order as the preceding tables.

Table IX. Correlation = $-3 \times 10^{-5}x + 0.9958$. PCA Approach. Initial Seed = 5

Reduced No. of Eigenvalues	Yield	CPU Run Time
25	80.37%	441.186 secs
50	79.18%	447.253 secs
100	78.25%	456.640 secs
200	77.36%	468.839 secs
300	75.87%	481.927 secs

Table X. Correlation = $-3 \times 10^{-5}x + 0.9958$. HAQ Approach. Initial Seed = 5

Threshold	Final Size of	Yield			CPU Run
Thickness	Covariance Matrix				Time
	(Y_{max}/Y_{min})	Total Yield	High Yield	Low Yield	
0.015 μm	27/24	85.57%	93.08%	92.49%	162.410 secs
0.03 μm	35/33	83.61%	92.25%	91.36%	180.372 secs
0.045 μm	41/37	81.37%	91.13%	90.24%	193.315 secs
0.06 μm	44/47	80.12%	90.25%	89.87%	205.934 secs
0.075 μm	61/61	78.91%	89.63%	89.28%	221.063 secs
0.09 μm	80/79	77.45%	88.81%	88.64%	239.388 secs

D. Comparison of Results

In case of Monte Carlo simulations without PCA, neglecting the presence of correlations causes an under-estimation in yield value. Such under-estimation is avoided in case of Monte Carlo simulations with PCA. These simulations are used as a baseline to compare the accuracy of results obtained from the other algorithms.

Table XI. Correlation = $-3 \times 10^{-5}x + 0.9958$. HAQ and PCA Approaches. Initial Seed = 5

Threshold	Final Size of Covariance Matrix		Yield			CPU
Thickness	Before PCA	After PCA	Total	High	Low	Run
Value	(Y_{max}/Y_{min})	(Y_{max}/Y_{min})	Yield	Yield	Yield	Time
0.015 μm	27/24	27/24	85.08%	92.63%	92.44%	189.137 secs
0.03 μm	35/33	35/33	83.11%	91.61%	91.50%	206.744 secs
0.045 μm	41/37	41/37	80.87%	91.05%	89.82%	224.039 secs
0.06 μm	44/47	44/47	79.68%	90.17%	89.51%	245.428 secs
0.075 μm	61/61	61/61	78.53%	89.43%	89.10%	271.811 secs
0.09 μm	80/79	80/79	77.39%	88.73%	88.66%	292.340 secs

Table XII. Correlation = $-4 \times 10^{-5}x + 0.9958$. MC Simulations

Iterations	Monte Carlo without PCA (Initial Seed = 15)		Monte Carlo with PCA		
	Yield	CPU Run Time	Seed	Yield	CPU Run Time
10,000	62.36%	1285.841 secs	15	75.85%	1429.947 secs
			35	75.91%	1450.812 secs
			75	76.07%	1437.263 secs
30,000	60.75%	3756.729 secs	15	73.72%	4185.825 secs
			35	73.86%	4197.762 secs
			75	73.98%	4209.311 secs
50,000	59.96%	6158.836 secs	15	71.59%	6517.837 secs
			35	71.36%	6489.516 secs
			75	70.85%	6472.372 secs

Table XIII. Correlation = $-4 \times 10^{-5}x + 0.9958$. PCC Approach. Initial Seed = 15

Radius of PCCs	Size of Covariance Matrix (Y_{max}/Y_{min})	Yield	CPU Run Time
150 μ m	429/425	86.31%	2213.723 secs
250 μ m	307/308	87.16%	1649.256 secs
350 μ m	198/201	88.09%	1182.670 secs
500 μ m	95/94	88.87%	577.218 secs
600 μ m	63/66	89.53%	361.833 secs
800 μ m	43/41	90.26%	245.511 secs
1000 μ m	27/27	90.97%	161.452 secs
2000 μ m	11/12	91.62%	56.742 secs

In case of the PCA simulations, when variable reduction less, the yield value tends to be closer to the results obtained from Monte Carlo simulations and hence more accurate. Consequently, the run time for the algorithm is also more with lesser variable reduction.

As stated in [1], yield values obtained using the PCC approach show a increasing in trend with increase in size of each PCC. Greater, the size of each PCC, greater is the reduction in variables and the yield is also overestimated. The run time for the algorithm also decreases with an increase in size of PCCs. With smaller PCC sizes, there is lesser reduction in the number of variables and the resulting yield value is closer to the results obtained using Monte Carlo methods. This improved accuracy

Table XIV. Correlation = $-4 \times 10^{-5}x + 0.9958$. PCA Approach. Initial Seed = 15

Reduced No. of Eigenvalues	Yield	CPU Run Time
25	78.65%	445.362 secs
50	77.09%	449.728 secs
100	75.72%	455.107 secs
200	73.96%	463.175 secs
300	72.68%	475.623 secs

Table XV. Correlation = $-4 \times 10^{-5}x + 0.9958$. HAQ Approach. Initial Seed = 15

Threshold Thickness	Final Size of Covariance Matrix ($\mathrm{Y}_{max}/\mathrm{Y}_{min}$)	Yield			CPU Run Time
		Total Yield	High ield	Low ield	
0.015 μm	38/39	82.38%	91.53%	90.85%	171.113 secs
0.03 μm	69/72	80.63%	90.72%	89.91%	198.437 secs
0.045 μm	112/109	79.17%	89.83%	89.34%	231.725 secs
0.06 μm	127/125	77.81%	88.95%	88.86%	278.933 secs
0.075 μm	148/143	76.29%	88.13%	88.16%	315.548 secs
0.09 μm	172/170	74.62%	87.37%	87.25%	361.027 secs

is obtained at the expense of the algorithms run time.

In case of simulations for the HAQ approach, the threshold thickness value decides the extent of reduction. Keeping a greater threshold value results in a more refined covariance matrix or a fine-grained set of basic sub-regions and thus a more accurate yield estimate. Using a smaller threshold value gives an yield value which is

Table XVI. Correlation = $-4 \times 10^{-5}x + 0.9958$. HAQ and PCA Approaches. Initial Seed = 5

Threshold Thickness Value	Final Size of Covariance Matrix		Yield			CPU Run Time
	Before PCA ($\mathrm{Y}_{max}/\mathrm{Y}_{min}$)	After PCA ($\mathrm{Y}_{max}/\mathrm{Y}_{min}$)	Total Yield	High Yield	Low Yield	
0.015 μm	38/39	38/39	82.09%	91.37%	90.72%	189.529 secs
0.03 μm	69/72	69/72	80.12%	90.49%	89.63%	217.388 secs
0.045 μm	112/109	112/109	78.89%	89.41%	89.48%	255.947 secs
0.06 μm	127/125	127/125	77.41%	88.77%	88.64%	301.437 secs
0.075 μm	148/143	148/143	76.08%	88.36%	87.72%	384.822 secs
0.09 μm	172/170	172/170	74.56%	87.23%	87.33%	418.610 secs

Table XVII. Correlation = $-2 \times 10^{-5}x + 0.9958$. MC Simulations

Iterations	Monte Carlo without PCA (Initial Seed = 30)		Monte Carlo with PCA		
	Yield	CPU Run Time	Seed	Yield	CPU Run Time
10,000	61.68%	1278.647 secs	30	79.72%	1429.256 secs
			60	79.59%	1433.372 secs
			100	80.36%	1425.418 secs
30,000	60.92%	3741.239 secs	30	78.03%	4217.577 secs
			60	77.86%	4205.813 secs
			100	77.95%	4183.480 secs
50,000	59.77%	6165.471 secs	30	75.92%	6539.328 secs
			60	76.11%	6496.672 secs
			100	76.28%	6516.087 secs

Table XVIII. Correlation = $-2 \times 10^{-5}x + 0.9958$. PCC Approach. Initial Seed = 30

Radius of PCCs	Size of Covariance Matrix ($\mathrm{Y}_{max}/\mathrm{Y}_{min}$)	Yield	CPU Run Time
150 μ m	431/435	88.93%	2241.683 secs
250 μ m	305/310	89.86%	1635.835 secs
350 μ m	194/197	90.47%	1169.526 secs
500 μ m	93/98	91.05%	581.972 secs
600 μ m	68/72	91.71%	386.351 secs
800 μ m	43/41	92.48%	245.227 secs
1000 μ m	29/30	92.97%	159.539 secs
2000 μ m	10/9	93.56%	56.119 secs

much higher when compared with Monte Carlo simulations.

Based on the simulation results made in the previous section, comparisons in yield accuracy and run times between the different algorithms are as follows -

The observations made in Table XXII indicate the improvement in accuracy and increase in run time of the PCA and HAQ approaches discussed in this thesis.

Table XIX. Correlation $= -2 \times 10^{-5}x + 0.9958$. PCA Approach. Initial Seed = 30

Reduced No. of Eigenvalues	Yield	CPU Run Time
25	81.85%	446.174 secs
50	80.58%	452.726 secs
100	79.76%	459.603 secs
200	78.33%	470.418 secs
300	77.27%	481.347 secs

Table XX. Correlation $= -2 \times 10^{-5}x + 0.9958$. HAQ Approach. Initial Seed = 30

Threshold Thickness	Final Size of Covariance Matrix	Yield			CPU Run Time
	$(\mathrm{Y}_{max}/\mathrm{Y}_{min})$	Total Yield	High Yield	Low Yield	
0.015 μm	40/38	88.41%	94.63%	93.78%	166.729 secs
0.03 μm	71/70	86.69%	93.75%	92.94%	191.318 secs
0.045 μm	110/114	85.11%	92.81%	92.30%	235.528 secs
0.06 μm	133/131	83.37%	92.12%	91.25%	287.832 secs
0.075 μm	153/155	82.13%	91.70%	90.43%	311.618 secs
0.09 μm	175/178	80.42%	90.57%	89.85%	372.275 secs

Table XXI. Correlation $= -2 \times 10^{-5}x + 0.9958$. HAQ and PCA Approaches. Initial Seed = 5

Threshold Thickness Value	Final Size of Covariance Matrix		Yield			CPU Run Time
	Before PCA $(\mathrm{Y}_{max}/\mathrm{Y}_{min})$	After PCA $(\mathrm{Y}_{max}/\mathrm{Y}_{min})$	Total Yield	High Yield	Low Yield	
0.015 μm	40/38	40/38	88.07%	94.46%	93.51%	187.318 secs
0.03 μm	71/70	71/70	86.49%	93.28%	93.21%	213.722 secs
0.045 μm	110/114	110/114	84.87%	92.39%	92.48%	259.635 secs
0.06 μm	133/131	133/131	83.08%	91.77%	91.31%	304.168 secs
0.075 μm	153/155	153/155	81.85%	91.03%	90.82%	388.492 secs
0.09 μm	175/178	175/178	79.93%	90.07%	89.86%	423.833 secs

Table XXII. Comparison of Results

Correlation Equation	Method	Yield Error	Speed
$-3 \times 10^{-5}x + 0.9958$	PCC	18.9%	1×
	PCA	2.7%	4.6×
	HAQ	4.1%	9.4×
$-4 \times 10^{-5}x + 0.9958$	PCC	21.1%	1×
	PCA	2.8%	4.7×
	HAQ	5.6%	6.2×
$-2 \times 10^{-5}x + 0.9958$	PCC	17.1%	1×
	PCA	1.3%	4.7×
	HAQ	5.3%	6×

CHAPTER V

CONCLUSION

Yield prediction involves monitoring the possibility of any electrical/manufacturing spec to get satisfied at n locations on a chip. CMP yield concerns the probability of interconnect thicknesses at n locations staying within the upper and lower thickness limits. In order to get an accurate estimate of yield, the variations in interconnect thickness at a large number of locations need to be monitored.

With shrinking feature sizes, the presence of intra-die variations can no longer be ignored. Overall, the process variations may be divided into two components, inter-die variations (layout dependent component) and intra-die variations (further sub-divided into systematic and random variations). With a rise in dominance of intra-die variations, the inter-die variations can be assumed as independent and represented using a single random variable. Intra-die random variations consist of both independent and partially correlated components. The case of intra-die variations is a lot more complicated due to the difficulty in handling a large number random variables with partial correlations (all variables are spatially correlated). Such complications add to the computational complexity of the yield prediction problem.

The demand of monitoring a large number of locations for thickness variations when combined with the existence of partial correlations between different locations, makes the yield prediction problem very complex. This thesis attempts to ease the complexity by reducing the number of variables used in computing yield. The techniques discussed in this thesis compute yield for a CMP model where meeting the interconnect thickness specs decides yield. [1] predicts a mechanism to reduce the number of variables through the use of perfect correlation clusters (PCC). Although, the PCC approach reduces the number of variables by a significant margin, it suffers in the accuracy of the resulting yield. This thesis discusses the use of two new reduction methods namely -

- *Principal Component Analysis (PCA)* - Re-express a set of large and correlated variables into a new, reduced and uncorrelated set through an orthogonal base.
- *Hierarchical Adaptive Quadrisection (HAQ)* - Reduces a large number of variables to a reduced set of basic sub-regions.

The advantage of these two methods is the reduction in number of variables without much compromise on accuracy in yield.

CHAPTER VI

FUTURE COURSE OF RESEARCH

Since the advent of the deep submicron era, the dominance of local variations has resulted in a large number of independent and partial correlations in metal thickness values. Accounting for intra-die variations (systematic and random) in yield prediction gives rise to a large number of random variables in the CMP model. However, not all the random variables might have a significant contribution towards yield. Hence the presence of so many random variations and their allied partial correlations makes yield prediction very cumbersome.

Given this kind of a setup, this thesis aims at reducing the problem to a smaller set of variables and then computing yield using numerical integration methods like *Genz algorithm* [6]. This thesis aims at overcoming the shortcomings due to the trade-off between computation accuracy and computation run time by using two different reduction techniques in yield prediction namely *Principal Component Analysis (PCA)* and *Hierarchical Adaptive Quadrisection (HAQ)*. The main advantage in using these reduction techniques lies in improvement of yield accuracy.

These techniques which are used to predict CMP yield can also be extended for prediction of yield with respect to timing constraints. The same techniques discussed in this thesis can be used to predict timing yield for sequential circuits [10].

REFERENCES

[1] J. Luo, S. Sinha, Q. Su, J. Kawa, and C. Chiang. "An IC manufacturing yield model considering intra-die variations," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 749–754, 2006.

[2] S. R. Nassif. "Modeling and analysis of manufacturing variations," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 223–228, 2001.

[3] T. Tugbawa, T. Park, D. Boning, T. Pan, P. Li, and S. Hymes. "A mathematical model of pattern dependencies in Cu CMP process," *Proceedings of the Electrochemical Society CMP Symposium*, pp. 605–615, 1999.

[4] D. Boning, T. Tugbawa and T. Park. "Characterisation and Modelling of Pattern Dependencies in Copper CMP," *Semiconductor Fabtech*, Cambridge, USA, 13^{th} Edition, pp. 265–268, 2002.

[5] J. Luo and D. Dornfeld. "Integrated Modeling of Chemical Mechanical Planarization for Sub-Micron IC Fabrication: from Particle Scale to Feature, Die and Wafer Scales," *Springer-Verlag, Berlin, Germany*, 2004.

[6] A. Genz. "Numerical computation of multivariate normal probabilities," *Journal of Computational and Graphical Statistics*, Vol. 1, pp. 141–149, 1992.

[7] R. Jiang, W. Fu, J. M. Wang, V. Lin, and C. C.-P. Chen. "Efficient statistical capacitance variability modeling with orthogonal principle factor analysis," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 683–690, 2005.

[8] H. Chang and S. S. Sapatnekar. "Statistical timing analysis considering spatial correlations using a single pert-like traversal," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 621–625, 2003.

[9] J. Shlens. "Tutorial on principal component analysis," Systems Neurobiology Laboratory, University of California at San Diego, December 2005.

[10] M. Pan, C. C.-N. Chu, and H. Zhou. "Timing yield estimation using statistical static timing analysis," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 2461–2464, 2005.

VITA

Sridhar Varadan received his Bachelor of Engineering degree in Electrical and Electronics Engineering from Sri Venkateswara College of Engineering (affiliated to Anna University), India in 2005. He joined the Masters program in Electrical Engineering at Texas A&M University in August 2005 and received his degree in December 2007.

Sridhar can be contacted at sridhar@ece.tamu.edu.