IC Substrate Package Yield Prediction Model and Layer Level Risk Assessment by Design Analysis

Takeshi Nakazawa, Deepak V. Kulkarni, and Osborne A. Martin

***Abstract*—We present a method for quantifying a risk for killer defects at layer level and estimating yield for substrate packages using information from design files. To calculate risk ranks and predicted yield, we define a risk distance that is a key parameter extracted from designs using image processing techniques. In order to validate our model, we analyze two different designs, each having multiple layers, and compare with data from baseline lots. It is shown that there is an inverse correlation between risk layer ranks and yield. Estimated yield based on our model is compared with baseline yield for four layers of the second design. The model-to-baseline yield difference is less than 1% for three layers we tested.**

***Index Terms*—Yield prediction, yield estimation, metrology sampling, integrated circuit packaging, circuit analysis, assembly.**

I. Introduction

YIELD has been used as one of the most critical parameters in the manufacturing environment since it directly impacts the manufacturing profit. Yield is defined as a ratio of the number of good units that meets all manufacturing requirements to the total number of manufactured units [1]. The overall yield can be broken down into several components depending on the purpose of application. The most commonly used components are: line yield, die yield, assembly yield, and final test yield [2]. Having accurate yield model is important for the following reasons.

1) Give work priority for factory line: Any products performing less than expected yield need to be addressed for yield improvement activities.
2) Cost estimate: The model can be used to predict manufacturing costs under development.
3) Resource estimate: The model can be used to estimate how much human/tool resources are needed.
4) Schedule estimate: The model can be used to estimate the required time to achieve a yield target.
5) Feedback to new design: The model can be used to determine possible design changes to achieve a yield target within a desired timeframe.

Manuscript received December 14, 2015; revised March 14, 2016; accepted March 31, 2016. Date of publication April 14, 2016; date of current version August 2, 2016.

The authors are with Intel Corporation, Chandler, AZ 85226 USA (e-mail: takeshi.nakazawa@intel.com).

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/TSM.2016.2554105

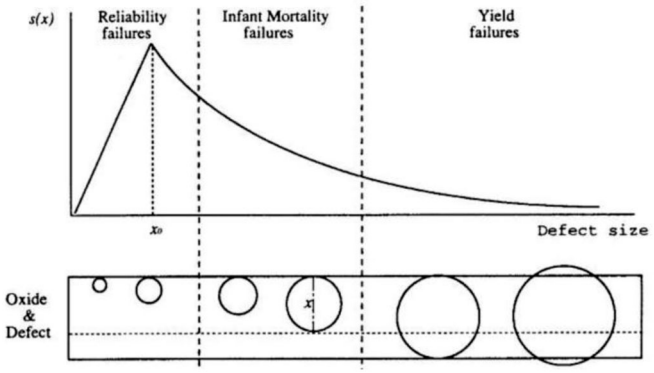


Fig. 1. The defect size PDF (reprinted from [2]).

In order to understand IC yield, a number of yield models have been proposed and the three commonly used yield models are the Poisson model, the Seed's model and the negative-binomial model [2], [3]. The first model used to predict IC yield was derived from the Poisson distribution function, which is given by the following form.

$$Y = e^{-DA}, \tag{1}$$

where A is the critical area (CA) [4] and D is defect density. The CA is defined as an area where particles must fall to cause open/short (O/S) and is given by the equation below.

$$A = \int_0^\infty A(x)s(x)dx. \tag{2}$$

$A(x)$ is the CA of defect size x. $s(x)$ is a defect size probability density function (PDF). The CA can be defined as the open critical area and the short critical area. The key to any yield model using the CA is to understand defect density distribution as a function of defect size. A commonly accepted PDF is shown in Fig. 1.

Historically, as chip sizes continued to increase, the Poisson model tended to underestimate yield for large die size. Because of this reason, a new model was developed by Murphy [5]. He assumed that Gaussian distribution would be a reasonable estimate for the defect density PDF. Seed took the model by Murphy but used exponential distribution rather than Gaussian to obtain the relationship below.

$$Y = \frac{1}{1 + DA}. \tag{3}$$

The negative binomial model uses the gamma distribution function and is given by Equation (4).

$$Y = (1 + DA/\alpha)^{-\alpha}, \tag{4}$$

where α is referred to as the cluster parameter. These three equations are a function of the CA which is the key for accurate yield prediction. There are mainly three different methods for critical area analysis (CAA): standard methods, statistical methods, and approximation methods [6]. The standard method includes the shape expansion method that computes a particle size which causes O/S. To get the average CA, the CA for each defect size needs to be integrated following the equation (2). The statistical method randomly places a pre-defined particle shape with different sizes on the IC layout and checks for O/S using Monte Carlo simulation [7], [8]. In general, previous two methods are CPU intensive, so the approximation method is proposed for improving computation speed [1].

In this paper, we propose layer level risk rank metric and a yield prediction model using extracted design parameters. We take a different approach for yield prediction. Instead of calculating the average CA, we defines a distance metric called "risk distance". One of the challenges for the CA calculation is to know the defect size distribution shown in Fig. 1. Depending on the manufacturing phase, one may not have reliable defect size distribution data because it might depend on tool capabilities to catch up with unexpected new defect modes during technology development phase and defect size extraction algorithms. Thus, instead of using defect size distribution, we use killer defect PDF as a function of risk distance group for yield model, which allows us to comprehend complex design structures and still extract meaningful design metrics that can be correlated to yield loss. One of the benefits for layer level risk assessment is that we can focus on risky layers for yield improvement activities. As the number of layers increase, more critical it is to have the layer priority, for example, for metrology sampling to increase a velocity of development and reduce manufacturing costs. Although we focus on only short killer defects in this paper, the same concept should be able to apply for open killer defects as well.

Our paper is organized as follows. In Section II, a design parameter extraction algorithm is described. We define risk metrics and the yield prediction model using the extracted risk distance. In Section III, we present model validation results. Yield from Design 1 baseline lots from our factory line is compared with layer risk ranks extracted from the design to validate the layer risk model. Then, estimated yield for Design 2 will be predicted from the model based on Design 1 data and compared with actual Design 2 yield. The conclusion is given in Section IV.

II. Method

A. Overview of Key Design Parameter Extraction

In order to evaluate risk levels for each net structure and layer, we analyze substrate design files. The key parameter is the minimum distance from one net to the closest net from each net boundary point. Fig. 2 illustrates the concept and it is

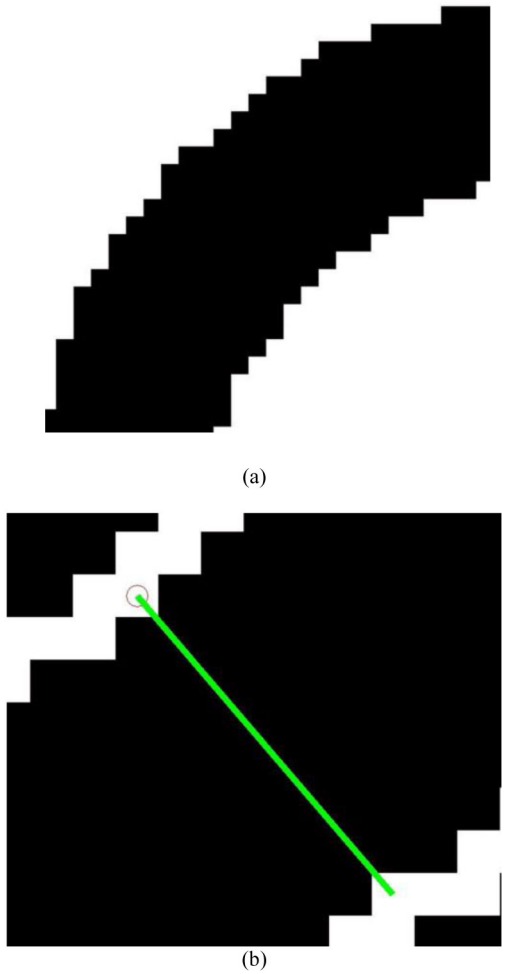


Fig. 2. Net-to-net minimum distance. (a) Two nets (top left – net 1, bottom right – net 2) and (b) edge boundary of each net and the minimum distance from certain edge boundary to the adjacent net.

the magnified image of a certain area in Design 1. In Fig. 2 (a), the top left white region is a net structure 1 and the bottom right white region is a net 2, which is the closest net structure from the net 1. Fig. 2 (b) illustrate the edge boundaries for each net. The red circle at the net 1 boundary indicates the edge location of the single boundary point. The green line shows the minimum distance to the corresponding net 2 edge boundary point. The minimum distance to the closest net point is calculated for all boundary points.

Fig. 3 shows the high-level flow of the risk metric calculations which can be divided into three parts. Firstly, we take a design file for a single layer and then do image preprocessing to identify object boundary locations. Secondly, geometrical information is calculated based on the preprocessed image. Lastly, the extracted data such as object numbers, a set of minimum distances, feature names defined in a printed circuit board (PCB) file, are combined to calculate the key risk metrics.

B. Risk Metrics Calculation – Image Pre-Processing

At first, we prepare a bitmap image from a design Gerber file and import it as an image file into our MATLAB program.

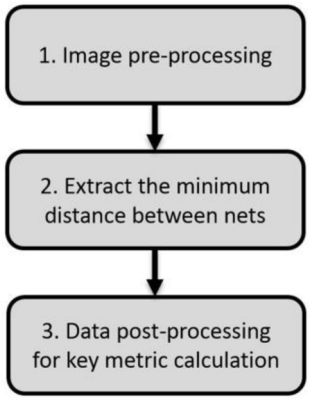


Fig. 3. High-level steps for key risk metric calculation.

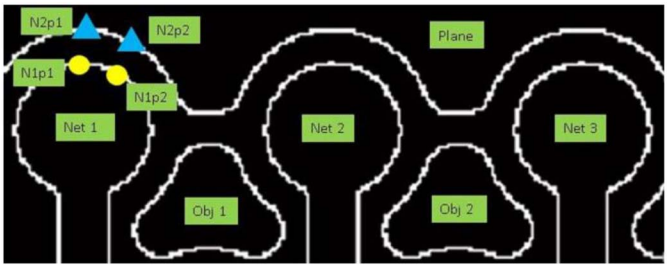


Fig. 4. Labeled object and its edge boundary.

Once we obtain the design file, each isolated object is labeled and this label information is also used to correlate with the actual net names defined in the PCB file. The Canny edge detector is used to obtain edge boundaries for tracing each edge pixel. An example labeled edge image is shown in Fig. 4. In this image, a plane structure is at the top and the other objects are pads/traces, some geometrical features. Each object has its corresponding labeling number (i.e., Plane = #1, Net1 = #2, Net2 = #3 and so on).

C. Risk Metrics Calculation – Geometrical Calculation for Short

Using labeled information, we start from the net 1's top left edge pixel point (N1p1 in Fig. 4) as an initial point to determine a set of minimum distances from the net 1 to closest net points. For finding the closest point of the other net from N1p1, we limit a search area with some threshold value to speed up our algorithm. If we cannot find any closest point within the predefined search area, we treat this net as an isolated structure and ignore it. If the closest point can be found, we store the minimum distance, label this point as N2p1 and store this point for defining a next search box. Now, by doing edge boundary tracing of the net 1 clockwise, we determine N1p2 and want to find a next set of distances from N1p2 to another closest point N2p2. We use the previously stored point N2p1 to define a search area, assuming that N2p2 should be closer from N2p1 (in general it is one pixel neighbor away from N2p1). If N2p2 cannot be found within the initial search area, we increase a search area with some amount to accommodate any object transition, i.e., in the example in Fig. 4, when object transition from the plane and the Obj 1 happens, the search area will be increased. We calculate the minimum distance and store N2p2 as a next search control point. By repeating these procedures, we will get a set of minimum distances between each net as shown in Fig. 5.

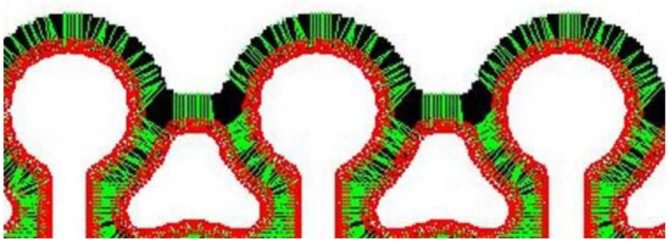


Fig. 5. Minimum distances from each net point to the other net.

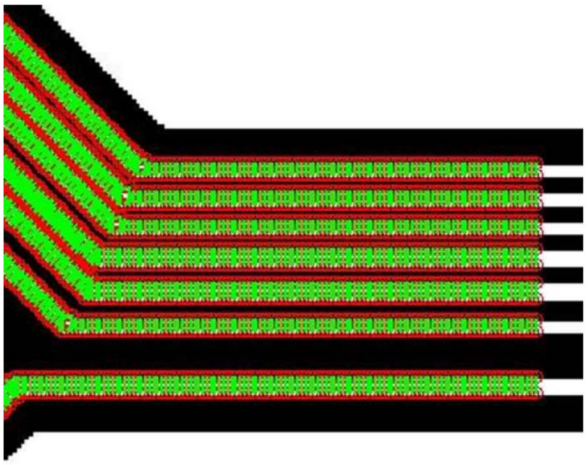


Fig. 6. Minimum distances for each net. To show design pattern clearly, the minimum distances at right side of the pattern are not shown.

D. Risk Metrics Calculation – Geometrical Calculation for Open

Since open happens within a single net as opposed to short which happens between nets, the approach for calculating minimum distances is a little bit different from the short distance calculation. At first, similar to the minimum distance calculation for short, we start from the top left edge pixel as an initial starting point and take the predefined search area from the edge boundary image. Then we label edge objects within the search area. For example, if we take a single horizontal metal pattern edge boundary, we have two parallel lines. Then the top line is labeled as the object 1 and the bottom one as the object 2. Using this labeled information, we calculate the minimum distance from the top edge to the bottom edge. By repeating the same procedure for each edge point, we can obtain a set of minimum distances for open as shown in Fig. 6.

E. Risk Metrics Calculation – Data Post-Processing

Once a set of minimum distances are obtained, we sum up these distances based on predefined distance groups. For example, any distance between 0 [um] and 0.9x [um] is summed and stored as a risk 1 distance group, and between x and 1.9x

into a risk 2 distance group and so on. We define such group as "risk distance group". Finally, we correlate object label information with actual net names using pad and trace connecting point coordinates and corresponding net name information extracted from the PCB file. Using these coordinates, we can select the object label number and correlate with the net name.

F. Layer Level Risk Assessment and Yield Model

We define a normalized layer level risk value *Rlayer#N* as follows.

$$R_{layer\#N} = \frac{\left[\sum_{n=1}^{N} w_n Dr_n\right]_{layer\#M}}{\max\left[\sum_{n=1}^{N1} w_n Dr_n\right]_{all\_layer}}. \quad (5)$$

w is a weighting factor for each risk distance group and Dr indicates a risk distance. The subscript n represents each risk bin. The reason we need the weighting factor is that the risk of short should be higher for net structures having a shorter net-to-net distance rather than ones having a longer net-to-net distance. The weighting factors are determined by defect occurrence rate at each risk distance group from any single design. In general, lots from a test vehicle or pilot lots from a revenue product are used to calculate the weighting factors. Then, these pre-assigned weighting factors are used to calculate expected yield for actual revenue product designs, as long as the process technology is identical. If process conditions change, we need to re-calculate these weighting factors to reflect the latest factory capabilities. *Rlayer#N* is normalized by the maximum value among the all layers.

We define our yield model as follows.

$$Y = e^{-\sum_{n=1}^{N1} Dr_n P_n D}. \quad (6)$$

P is killer defect probability density function for each risk group and D is killer defect density, given from the actual defect data.

III. Results

A. Layer Level Risk Assessment

Fig. 7 is the color map example of the short risk distance group. In this image the plane structure is excluded for illustration purpose. The red shows the highest risk distance (risk 1 distance group) and yellow is risk 2 group and so on.

Based on the risk distances, we can now calculate *Rlayer* using Equation (5). Fig. 8 shows the layer level risk *Rlayer* for each layer from Design 1, which has 9 layers. This figure illustrates the layer level relative risk rank.

In order to validate our layer risk model, *Rlayer* is compared with the actual yield values which are illustrated in Fig. 9. Fig. 9 shows yield versus *Rlayer* to validate our model. Due to our current layer sampling plan, yield data for some layers are not available. The actual yield values are not shown in this plot. We can observe the trend that as the layer level risk *Rlayer* decreases, yield increases. The result suggests that *Rlayer* can be used as a risk metric for making priority decision for any yield improvement activities or any metrology sampling. For example, if it is determined that automated optical inspection can support only up to 3 different operations

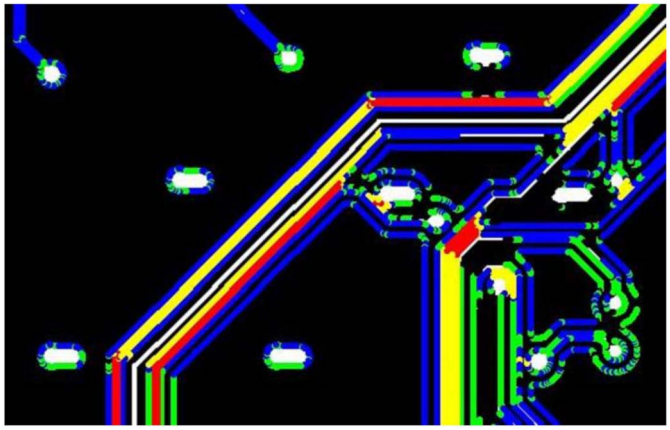


Fig. 7. Short risk distance group color map from Design 1.

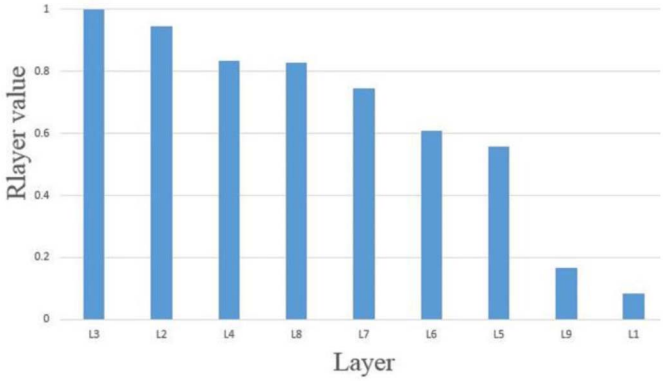


Fig. 8. Layer risk ranks for Design 1.

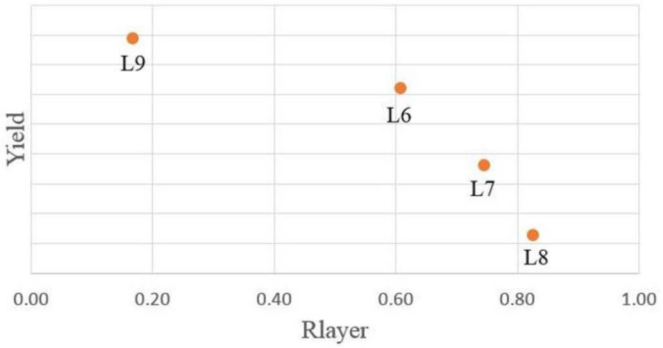


Fig. 9. Yield vs. *Rlayer* for Design 1.

based on the tool availability and the number of lots start per week, it is reasonable to choose the layer L2, L3 and L4 as the initial sampling layers.

B. Yield Prediction Model

Our yield model is a function of the killer defect PDF, the killer defect density and the risk distances as described in Equation (6). In order to calculate expected yield, we need to know the killer defect PDF as a function of the risk distance groups. Fig. 10 shows the actual short killer locations for Design 1. Color illustrates the corresponding risk distance groups. For example, the red dots indicate that these defects

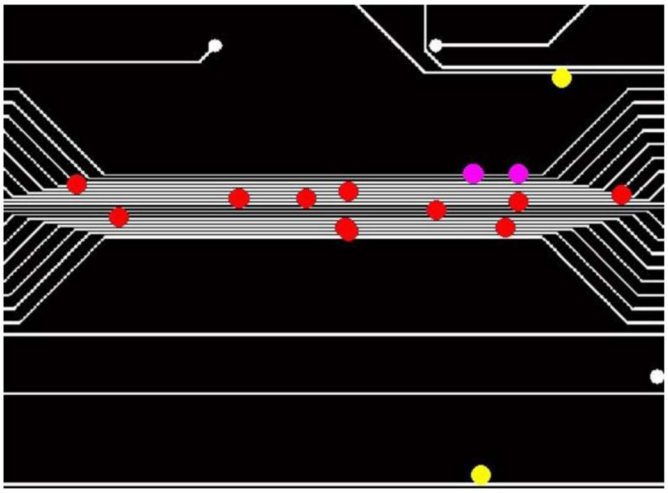


Fig. 10. Actual killer defect locations. Color illustrates corresponding risk level. Plane feature is excluded for illustration purpose.

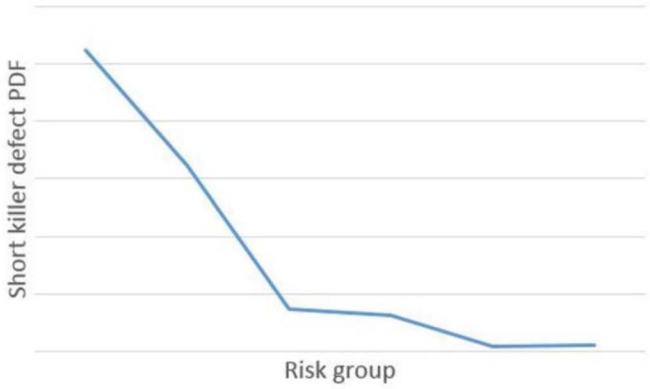


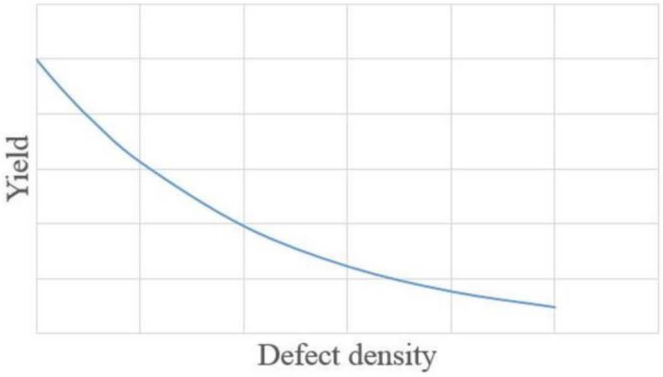
Fig. 11. Short killer defect PDF for Design1.

occurred at the risk 1 distance group areas, the pink dots at the risk 2 distance areas and so on.

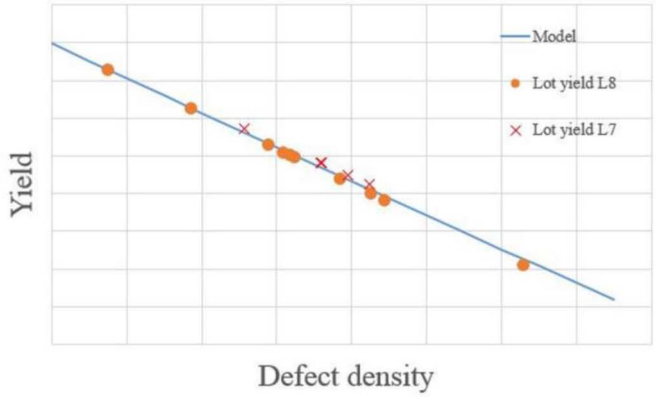
Once we know defect distribution at each risk distance group, we can calculate the killer defect PDF. Fig. 11 shows the PDF from Design 1 using the 14 baseline lots. The actual values are excluded from the plot.

Now, we can calculate yield based on our model, given defect density. Fig. 12 shows the yield model based on the short killer PDF given in Fig. 11. Again, the actual values are excluded from the plot. Fig. 12 (a) illustrates yield distribution across a large range of defect density, which follows the exponential function. Fig. 12 (b) shows the magnified region of Fig. 12 (a) with the some individual lot yield values for model validation purpose. The circles and crosses are yield from L8 and L7 respectively. As we can see from the plot, our model matches well with actual lot yield.

Finally, using the killer defect PDF from Design 1, estimated yield for Design 2 is calculated. Here the assumption is that when we run Design 2 through our factory line, the short killer defect PDF remains the same between Design 1 and Design 2. Using the risk distance groups calculated from the design files, we can calculate expected yield for each layer of Design 2. Since actual yield cannot be provided, we provide



(a)



(b)

Fig. 12. The yield model for Design 1. (a) Yield vs. defect density from our model, and (b) the magnified portion of (a). Line is the model. Circles are lot yields for L8 and crosses are lot yield for 7.

TABLE I
YIELD DIFFERENCE [%] BETWEEN OUR MODEL AND ACTUAL YIELD

Layer	Yield difference [%] (model – actual yield)
L1	0.24
L2	0.29
L3	0.69

the difference in yield [%] between the estimate based on our model and actual yield calculated from our factory line in TABLE I. For this specific product and layers, our model to actual yield difference is less than 1 %.

IV. CONCLUSION

In this paper, we present a method of quantifying layer level risks using the risk distance metric. The key design parameter extraction algorithm is presented. The layer level risk rank is compared with baseline yield using Design 1. We observe the trend that as *Rlayer* decreases, yield increases. Based on this result, *Rlayer* rank can be considered as one of the key metrics for making any priority decision in the factory line or yield improvement activities. Then, our yield model is validated using killer defect PDF from baseline lots from Design

1. We see that our model matches with actual yield for different layers. Finally, estimated yield for Design 2 is compared with actual yield. We observe a correlation with our prediction model and actual yield within less than 1% yield difference for given design and process conditions.

REFERENCES

[1] C. Chiang and J. Kawa, *Design for Manufacturability and Yield for Nano-Scale CMOS*. Dordrecht, The Netherlands: Springer, 2007.
[2] W. Kuo and T. Kim, "An overview of manufacturing yield and reliability modeling for semiconductor products," *Proc. IEEE*, vol. 87, no. 8, pp. 1329–1344, Aug. 1999.
[3] J. A. Cunningham, "The use and evaluation of yield models in integrated circuit manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 3, no. 2, pp. 60–71, May 1990.
[4] C. H. Stapper, "Modeling of defects in integrated circuit photolithographic patterns," *IBM J. Res. Develop.*, vol. 28, no. 4, pp. 461–475, Jul. 1984.
[5] B. T. Murphy, "Cost-size optima of monolithic integrated circuits," *Proc. IEEE*, vol. 52, no. 12, pp. 1537–1545, Dec. 1964.
[6] A. R. Dalal, P. D. Franzon, and M. J. Lorenzetti, "A layout-driven yield predictor and fault generator for VLSI," *IEEE Trans. Semicond. Manuf.*, vol. 6, no. 1, pp. 77–82, Feb. 1993.
[7] Y. Hamamura *et al.*, "Repair yield simulation with iterative critical area analysis for different types of failure," in *Proc. 17th IEEE Int. Symp. Defect Fault Tolerance VLSI Syst. (DFT)*, Vancouver, BC, Canada, 2002, pp. 305–313.
[8] G. A. Allan, "A comparison of efficient dot throwing and shape shifting extra material critical area estimation," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Syst.*, Austin, TX, USA, 1998, pp. 44–52.

Takeshi Nakazawa received the Ph.D. degree in optical sciences from the College of Optical Sciences, University of Arizona, in 2011.

He is currently a Senior Metrology Engineer with Intel Corporation, Chandler, AZ, for developing automated defect detection and classification system and yield prediction modeling. He was a recipient of several Intel divisional and department awards, the Best Paper Award for Intel technology journal, and several distinguished invention awards.

Deepak V. Kulkarni received the Ph.D. degree in mechanical engineering from the University of Illinois at Urbana–Champaign, in 2005. He currently serves as an Engineering Technology Development Manager with the Assembly and Test Technology Development Group, Intel Corporation, Chandler, AZ. His interests are in applying big data analysis techniques to improve manufacturing yield.

Osborne A. Martin received the B.S. degree from North Carolina A & T State University, in 1995, and the M.S. and Ph.D. degrees from the Center of Precision Manufacturing, University of North Carolina at Charlotte, in 1999 and 2003, respectively, all in mechanical engineering. He is currently a Metrology Manager with Intel Corporation, Chandler, AZ.