Analyzing Volume Diagnosis Results with Statistical Learning for Yield Improvement

Huaxing Tang, Sharma Manish, Janusz Rajski, Martin Keim, Brady Benware
Mentor Graphics Corporation, 8005 S.W. Boeckman Road, Wilsonville, Oregon, USA, 97070
Email: {huaxing_tang, sharma_manish, janusz_rajski, martin_keim, brady_benware}@mentor.com

Abstract— **A novel statistical learning algorithm is proposed to accurately analyze volume diagnosis results. This algorithm effectively overcomes the inherent ambiguities in logic diagnosis, to produce accurate feature failure probabilities, which are critical in understanding systematic yield limiters. The results of Monte-Carlo simulation are presented, which demonstrate the feasibility and impacts of various factors on this approach. Additional experiments based on injected defects are performed, which confirm the ability of this approach to generate accurate feature failure probabilities for an industrial design using actual diagnosis results.**

I. Introduction

The profitability of an IC company highly relies on a high and stable yield, which is defined as the ratio of the number of good dies to the total number of dies manufactured. After the first silicon of a new design comes out, the yield must be ramped up to a certain high profitable level before the volume production can be launched. However due to new technologies and materials introduced for $90nm$ and below nodes, the yield ramp becomes more difficult. It takes longer to achieve an acceptable stable yield compared to the previous processes, which will increase the time-to-market, missing the market window in the worst case.

Traditional methods for yield learning include test chips, memory bitmap, inline inspection, physical failure analysis, and etc. Each method has its own advantages and limitations. When the process moves into deep sub-micron domain, feature-related systematic defects, such as single vias, bridges over wide metal, cause more and more yield problems. Unfortunately, traditional yield learning approaches become less effective for them. Hence new yield learning methods based on volume diagnosis results have gained a lot of attentions recently [2], [5]–[12]. The basic idea is to use the manufactured dies as their own test chips. Thousands of failing dies, which typically go to the trash can, are a gold mine of defect mechanism information.

Although the accuracy and resolution of logic diagnosis is continuously improved, [1], [3], [4], [13], [14], it is still not perfect due to various limitations, such as logic equivalency or lack of layout and physical information. For example, the input pin and output pin(s) of a buffer are undistinguishable if only a gate-level netlist is available. Similarly all instances of potential physical bridges between a pair of nets can not be distinguished easily. Because of various ambiguities existing in the volume diagnosis results, statistical learning is needed to suppress the noise and to extract valuable defect mechanism information.

This paper proposes a statistical learning algorithm to effectively analyze volume diagnosis results. The algorithm employs an iterative solving procedure to handle the ambiguities in diagnosis results and to extract defect mechanism information. Monte-Carlo simulation is used to investigate the impact of various factors on the accuracy of learned results, such as the error of the initial estimate, the accuracy of diagnosis results, and the number of failing dies. Controlled experiments are also performed on an industrial design to further validate the proposed algorithm.

The paper is structured as follows: At first the problem is formulated in Section II. Section III explains the proposed algorithm. Experimental results are shown in section IV. A case study based on an industrial design is discussed in Section V. Finally, the paper is summarized and conclusions are drawn.

II. Problem Formulation

A. Terminology

Before we discuss the proposed algorithm, the terminology used in this work must be explained.

- A feature, f_i, is a layout configuration in the design with a set of unique characteristics, such as being a specific library cell, or being a side-to-side bridges on metal layer 3, etc. Each feature typically has more than one instance in the design.
- A feature instance, f_i^j, is the j-th instance of feature f_i in a manufactured die of a given design.
- Failure probability of feature f_i, $p_{fail}(f_i)$, is defined as the probability of an instance of f_i to be defective after manufacturing. A set of features should be properly defined to make sure all instances of a given feature have a similar failure probability, and thus an averaged failure probability can be used for all instances for this feaure.

Here are some terms used to describe diagnosis results:

- A faillog is the failure information collected by a tester for a given failing die. Although a die may fail different tests, such as memory test, chain test, and scan test, only scan test failures are considered in this work.
- A suspect is a diagnosis callout explaining some parts of a faillog, which may be associated with one or more features.

12th IEEE European Test Symposium (ETS'07)
0-7695-2827-9/07 $20.00 © 2007 IEEE

TABLE I

AN EXAMPLE OF VOLUME DIAGNOSIS RESULTS

Die	Symptom	Suspect List
1	1	f_5^2
2	1	f_2^{123}, f_3^{22}, f_3^{23}, f_1^{1001}
3	1	f_1^4, f_3^1
3	2	f_2^{34}, f_3^{23}, f_5^{61}
...	...	...
N_{fail}	1	f_i^j, f_s^t

- A symptom is a group of suspects which explain some failure information, and tends to be associated with the same physical defect.

Since more than one defect may occur in a single die, diagnosis results may report more than one symptom for each faillog. For each symptom, more than one suspect may be identified because of the limitations of diagnosis algorithms.

B. Problem Formulation

Table I shows an example of volume diagnosis results for N_{fail} failing dies. In the ideal case of diagnosis, every die has a single symptom and a single suspect, like Die_1, the failure probability of each feature can be trivially computed by dividing the total number of occurrences of a feature in the suspect table like Table I by the total number of manufactured instances of this feature. Unfortunately, the majority of diagnosis results have many suspects, like Die_2 or Die_3, and thus a method is needed to suppress the noise and to extract feature failure probabilities.

III. PROPOSED ALGORITHM

An iterative algorithm is proposed to address the ambiguities in diagnosis results and to compute the failure probability of each feature: The volume diagnosis results in Table I can be considered as a system of equations with feature failure probabilities $p_{fail}(f_i)$ as unknown variables. Assuming different features fail independently, the equations can be solved in an iterative fashion to obtain $p_{fail}(f_i)$.

Consider Die_2 in Table I, for which the diagnosis produces the following feature instances as its suspects: f_2^{123}, f_3^{22}, f_3^{23}, f_1^{1001}. The formulas below assume that the actual defective feature instance(s) of a failing die is in its suspect list. However, the proposed statistical algorithm itself can handle incorrect diagnosis results as we show later. The probability that, given the above diagnosis results, the actual cause of defect in Die_2 is f_2^{123} can be determined using probability theory by defining two events A and B as:

$A = f_2^{123}$ is the only defect in Die_2; and

$B =$ At least one of the feature instances f_2^{123}, f_3^{22}, f_3^{23}, f_1^{1001} is the cause of defective behavior of Die_2.

The conditional probability of A given B is then:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)}{P(B)} \tag{1}$$

since $A \subset B$. Now, assuming that all features fail independently, the probability of events A and B can be given by:

$$P(A) = \frac{p_{fail}(f_2)}{(1 - p_{fail}(f_2))} \prod_{i=1}^{K} (1 - p_{fail}(f_i))^{n_i} \tag{2}$$

and

$$P(B) = \frac{1 - (1 - p_{fail}(f_1))(1 - p_{fail}(f_2))(1 - p_{fail}(f_3))^2}{(1 - p_{fail}(f_1))(1 - p_{fail}(f_2))(1 - p_{fail}(f_3))^2} \cdot \prod_{i=1}^{K} (1 - p_{fail}(f_i))^{n_i} \tag{3}$$

where n_i is the total number of instances of f_i, and K is the total number of features.

By substituting (2) and (3) into (1), and ignoring the higher order terms which are much smaller than the first order terms, (1) becomes:

$$P(A|B) \approx \frac{p_{fail}(f_2)}{p_{fail}(f_1) + p_{fail}(f_2) + 2p_{fail}(f_3)} \tag{4}$$

In general, assuming there are x_i instances of f_i, $1 \le i \le K$, for a failing die, the probability that the actual cause of defect in this failing die is an instance of f_i can be given by:

$$P(f_i) \approx \frac{x_i p_{fail}(f_i)}{\sum_{j=1}^{K} x_j p_{fail}(f_j)} \tag{5}$$

The contribution of each failing die to the failure count of f_i can be added. An initial estimate of $p_{fail}(f_i)$ can be obtained by dividing the sum by $n_i N_{manuf}$, where N_{manuf} is the total number of manufactured dies.

In summary, assume that N_{fail} dies out of N_{manuf} fabricated dies fail scan test and are successfully diagnosed. For failing dies with more than one symptom, each symptom is treated as a separate defect. Let x_i^l denote the number of instances of feature f_i in the suspect list for failing die l, $1 \le l \le N_{fail}$. The failure probability of feature f_i can then be estimated according to this:

$$p_{fail}(f_i) = \frac{1}{n_i N_{manuf}} \sum_{l=1}^{N_{fail}} \left(\frac{x_i^l p_{fail}(f_i)}{\sum_{j=1}^{K} x_j^l p_{fail}(f_j)} \right) \tag{6}$$

for $1 \le i \le K$. Hence, volume diagnosis results can be treated as a system of non-linear equations in the unknown variables $p_{fail}(f_i)$. These equations can be solved in an iterative fashion, as shown in (6), starting from some initial guesses of the $p_{fail}(f_i)$ values and iteratively converging towards a solution. This technique can thus be characterized as an iterative learning procedure for feature failure probabilities.

The accurate learned failure probability of each feature can be used to draw defect paretos, identify systematic yield limiters, etc. In addition, the rankings of suspects of each failing die can be improved by assigning higher scores to suspects associated with features with higher failure probabilities.

IV. Experimental Results

In order to validate the proposed learning algorithm, several experiments are performed using either Monte-Carlo simulation or injected defects of a real industrial design. In this section we will focus on the experimental results based on Monte-Carlo simulation. The controlled experiment using an industrial design will be discussed in the next section.

A. Experiment Setup

Monte-Carlo simulation is used to emulate the processes of manufacturing a large number of dies and diagnosing the failing ones. For an abstract design with a total of K features, assume that each feature f_i has a total of n_i different instances and an arbitrary failure probability $p^{exp}_{fail}(f_i)$. The Monte-Carlo simulation to create N_{fail} failing dies is done as follows:

1) For each die, traverse all feature instances and determine which feature instance(s) will become defective based on $p^{exp}_{fail}(f_i)$. One die is considered as a good die if no instance fails.
2) For each failing instance, a list of suspects, i.e. a symptom, is created by adding some arbitrary feature instances as diagnosis noise. To emulate the case where the real defect is missed by diagnosis, a list of random suspects will be used instead.
3) Repeat the previous two steps until the total number of failing dies reaches the limit.

Due to the non-perfect randomness, the feature failure probability of actual injected defects $p^{inj}_{fail}(f_i)$ may be slightly different from $p^{exp}_{fail}(f_i)$. Therefore, $p^{inj}_{fail}(f_i)$ is used for comparison with the learned results.

The proposed statistical learning algorithm is then applied to the volume diagnosis results generated by the Monte-Carlo simulation to compute the learned failure probability of each feature $p^{lrn}_{fail}(f_i)$. To evaluate the quality of learned results, the individual error ε_i, average error ε_{avg}, and maximal error ε_{max} between $p^{inj}_{fail}(f_i)$ and $p^{lrn}_{fail}(f_i)$ of all features are computed as follows:

$$\varepsilon_i = |\frac{p^{lrn}_{fail}(f_i) - p^{inj}_{fail}(f_i)}{p^{inj}_{fail}(f_i)}|, 1 \leq i \leq K \quad (7)$$

$$\varepsilon_{avg} = \frac{1}{K}\sum_{i=1}^{K}\varepsilon_i \quad (8)$$

$$\varepsilon_{max} = max(\varepsilon_1, \varepsilon_2, ..., \varepsilon_K) \quad (9)$$

The square of correlation coefficient (R^2) between $p^{inj}_{fail}(f_i)$ and $p^{lrn}_{fail}(f_i)$ is also computed.

B. An Example

An abstract design with fifteen features is used to illustrate the experiment setup and demonstrate the proposed algorithm. Assume each feature has 10^5 instances and a failure probability of $p^{exp}_{fail}(f_i) = \lceil i/3 \rceil \cdot 10^{-7}$ for $1 \leq i \leq 15$. A total of $27,288$ dies are created by Monte-Carlo simulation, out of which $10,000$ dies fail. For these failing dies, diagnosis

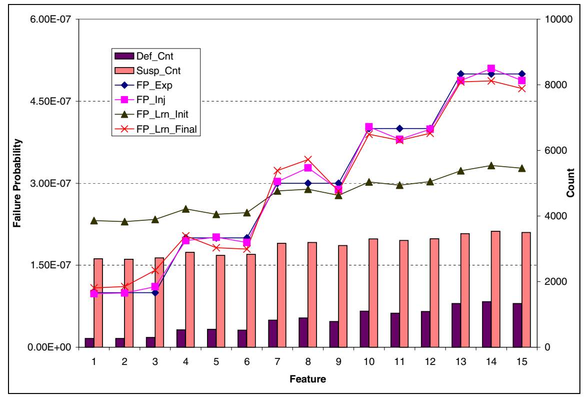


Fig. 1. An example of defect injection and statistical learning

results are created in which each symptom has on average 4 feature instances in its suspect list. In Figure 1, both the number of injected defects (Def_Cnt) and the number of diagnosed suspects ($Susp\_Cnt$) of each feature are reported as bars. We apply the proposed algorithm on them and compute feature failure probabilities. Four curves representing the expected, injected, initial and final learned failure probabilities are also plotted as FP_Exp, FP_Inj, FP_Lrn_Init, and FP_Lrn_Final .

As expected, the curve FP_Inj directly derived from Def_Cnt closely follows $p^{exp}_{fail}(f_i)$. The curve FP_Lrn_Init, derived by equally distributing credit to all suspects for each symptom, aggregating the credit for each feature, and dividing the sum by the total number of manufactured dies, simply follows the trend of $Susp\_Cnt$. Due to the ambiguities in diagnosis results, FP_Lrn_Init is quite far away from FP_Inj. Starting from the curve FP_Lrn_Init, the iterative algorithm continuously improves the results until they converge, shown as curve FP_Lrn_Final. It can be seen that the curve FP_Lrn_Final matches very well with the curve FP_Inj, confirmed with a very high value of $R^2 = 0.992$. The errors of learned results are significantly reduced by the iterative algorithm. The maximal error is reduced from 1.366 down to 0.272, and the average error is reduced from 0.435 down to 0.060 after only 25 iterations. The total CPU time is less than one second due to the fast converging of the proposed algorithm and the small number of features involved in this example. We believe that the run time of the proposed algorithm should not be an issue for real applications, and will be ignored in the following experiments.

This simple example demonstrates that the proposed statistical learning algorithm is necessary and effective to handle the ambiguities in volume diagnosis results, and is able to accurately recover the feature failure probability within a small number of iterations.

0-7695-2827-9/07 $20.00 © 2007 IEEE

TABLE II

IMPACT OF INITIAL ERROR OF FAILURE PROBABILITY

Δ	$N_{fail} = 10^4$			$N_{fail} = 10^5$		
	R^2	ε_{avg}	ε_{max}	R^2	ε_{avg}	ε_{max}
0.0	0.992	0.060	0.258	0.998	0.027	0.066
0.1	0.992	0.060	0.259	0.998	0.027	0.067
0.2	0.992	0.060	0.257	0.998	0.027	0.067
0.3	0.992	0.060	0.256	0.998	0.027	0.067
0.5	0.992	0.061	0.256	0.998	0.027	0.068
0.8	0.992	0.062	0.260	0.998	0.027	0.068

TABLE III

IMPACT OF DIAGNOSIS RESULT ACCURACY

Acc	$N_{fail} = 10^4$			$N_{fail} = 10^5$		
	R^2	ε_{avg}	ε_{max}	R^2	ε_{avg}	ε_{max}
100%	0.992	0.060	0.258	0.998	0.027	0.066
95%	0.989	0.079	0.262	0.998	0.052	0.236
90%	0.986	0.101	0.335	0.998	0.099	0.344
85%	0.989	0.126	0.527	0.998	0.133	0.431
80%	0.966	0.173	0.793	0.998	0.165	0.530

C. Impact of Initial Error of Failure Probability

An initial guess is needed for any iterative algorithm. For the proposed algorithm, the initial values could be obtained in several ways. One way is to use the historical yield data to estimate the failure probability of each feature. EDA tools, such as a critical area analysis tool, can also be used to estimate the feature failure probability. The initial values could also be derived from the volume diagnosis results. Different approaches may introduce larger or smaller errors into the initial values, which may impact the accuracy of the final learned results. The following experiment was performed to investigate such impacts.

The same design with the same fifteen features from the previous section is used in this experiment. In this experiment, errors are intentionally introduced into the initial guesses of feature failure probabilities like this: $p_{fail}^{exp}(f_i) \cdot (1 + r\Delta)$ for $1 \leq i \leq 15$, where Δ is a predefined coefficient to control the maximal error to be introduced, and r is a random number between -1 and 1. Based on these initial values, the learned failure probabilities $p_{fail}^{lrn}(f_i)$ are computed and correlated with the injected failure probabilities $p_{fail}^{inj}(f_i)$ as shown in Table II. The experiment is repeated for different values of Δ and different numbers of failing dies.

From Table II, it is easy to see that the initial errors have a minimal impact on the final results. Regardless of the initial errors, for the same set of diagnosis results, the proposed algorithm can virtually converge to the same final solution within a similar number of iterations. For the case with 10^4 failing dies, even with $\Delta = 0.8$, both ε_{avg} and ε_{max} increase just by 0.002, compared to the case where the expected failure probabilities are used as the initial values ($\Delta = 0.0$). It can also be seen that with more failing dies the negative impact of initial errors could be further reduced. For example, with 10^5 failing dies, the impact of initial errors on ε_{avg} becomes unnoticeable. In short, the proposed iterative algorithm can effectively tolerate the initial errors and converge to the same final results with good accuracy.

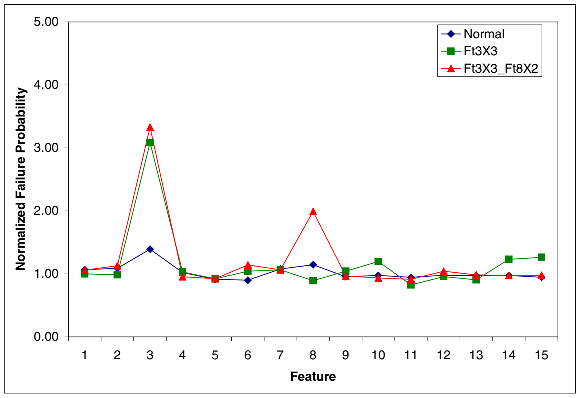
D. Impact of Diagnosis Result Accuracy

Intuitively the accuracy of volume diagnosis results has a direct impact on the accuracy of learned results. Therefore another experiment was performed to understand the significance of this impact. In previous experiments, it is assumed that the real defect is always covered by the suspect list, which implies a 100% diagnosis accuracy. To emulate the case where the real defect is missed by the diagnosis results, as explained in experiment setup, the original suspect list is simply replaced with a random suspect list of the same size during Monte-Carlo simulation. The same design with the same fifteen features is used again. Volume diagnosis results with different levels of accuracy are generated by Monte-Carlo simulation. For example, for a given accuracy of 90%, diagnosis results of 10% of failing dies are random feature suspects. The feature failure probabilities are computed and correlated with the actual values in Table III. The same set of experiments are repeated for 10^4 and 10^5 failing dies.

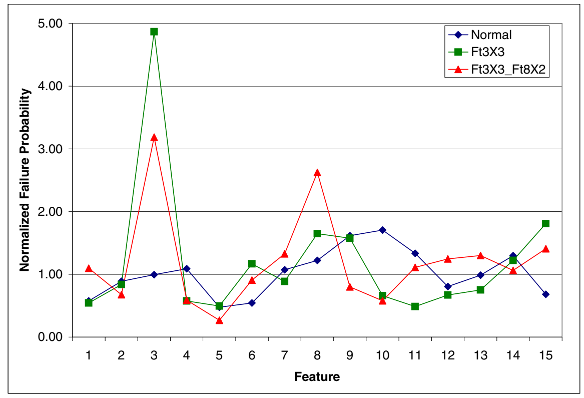
It can be seen that the accuracy of learned results keeps decreasing as the diagnosis result accuracy goes down. For 10^4 failing dies, the errors almost increase by three times when the accuracy drops from 100% down to 80%, which shows that the accuracy of diagnosis results is critical to the success of the statistical learning. Based on our experience, better than 90% accuracy can be achieved by current advanced diagnosis algorithms, which gives us a very good results with $R^2 = 0.986$ and $\varepsilon_{avg} = 0.101$ for 10^4 failing dies. Again, the accuracy of learned results can be further improved if more failing dies are available.

E. Impact of Number of Failing Dies

For better understanding of the impact of the number of failing dies on the accuracy of learned results and on identifying systematic yield limiters, the following three experiments were performed. The first experiment, called $Normal$, uses the same feature failure probabilities as the previous experiments: $p_{fail}^{expA}(f_i) = \lceil i/3 \rceil \cdot 10^{-7}$ for $1 \leq i \leq 15$. The second experiment, called $Ft3X3$, has the same values as $Normal$ for all features except for f_3. $p_{fail}^{expB}(f_3) = 3p_{fail}^{expA}(f_3)$ is used to emulate a systematic issue with f_3. Similarly, the third experiment, called $Ft3X3\_Ft8X2$, uses a higher failure probability for f_3 and f_8, $p_{fail}^{expC}(f_3) = 3p_{fail}^{expA}(f_3)$ and $p_{fail}^{expC}(f_8) = 2p_{fail}^{expA}(f_8)$. Monte-Carlo simulation is used to create a certain number of failing dies for each experiment, and the statistical learning algorithm is applied to compute the feature failure probabilities as before. These three experiments are done for $10,000$, $1,000$, 200, and 100 failing dies. Due to the limited space, only data of $10,000$ and 200 failing



(a) $N_{fail} = 10,000$



(b) $N_{fail} = 200$

Fig. 2. Impact of the number of failing dies on learned feature failure probabilities

dies are plotted in Figure 2(a) and 2(b). For easy comparison, the learned failure probabilities are normalized by dividing by $p_{fail}^{expA}(f_i)$.

Ideally, we should see a straight line $y = 1.0$ for curve $Normal$, which implies the learned failure probability of each feature perfectly matches the expected value. The curve $Normal$ in Fig. 2(a) closely follow the line $y = 1.0$ as expected. But in Fig. 2(b) it shows many fluctuations around $y = 1.0$, due to the very limited number of failing dies.

All features with systematic issues are expected to stick out due to their abnormal high failure probabilities. This is clearly demonstrated by curves $Ft3X3$ and $Ft3X3\_Ft8X2$ in Fig. 2(a), where spikes are observed for all features with systematic issues, f_3 and f_8. In addition, the heights of such spikes could be used to prioritize the investigation of features with systematic issues. Identifying features with systematic issues becomes difficult when the total number of failing dies is small, like in Fig. 2(b).

Another experiment for identifying systematic issues was done as follows. Let us consider all features with the normalized failure probability higher than a predefined threshold value C as victim candidates of systematic issues. A victim candidate is *good* if it is actually affected by some systematic issues, like f_3 in case $Ft3X3$. Otherwise, it is a $false$ victim candidate. For each experiment, the numbers of good and false victim candidates are counted with $C = 1.8$, and presented in Table IV. It can be seen that the systematic issue identification based on the normalized failure probabilities works very well when the number of failing dies is reasonable large. Both cases with $10,000$ and $1,000$ failing dies can identify all real victims and zero false victim. When N_{fail} decreases to 200, feature f_{15} becomes a false victim candidate for case $Ft3X3$, which may trigger unnecessary investigation on it. If N_{fail} drops further down to 100, good victim candidates are totally disguised by

TABLE IV
IDENTIFYING SYSTEMATIC ISSUE ($C = 1.8$)

	Normal		Ft3X3		Ft3X3_Ft8X2	
N_{fail}	#Good	#False	#Good	#False	#Good	#False
10,000	0	0	1	0	2	0
1,000	0	0	1	0	2	0
200	0	0	1	1	2	0
100	0	2	1	1	2	1

false victim candidates, and thus makes the results less useful. In all, with a few thousand failing dies from volume production, the proposed algorithm can accurately recover the feature failure probability and pinpoint the features with systematic issues. In the case where only several hundreds of failing dies are available, such as first silicon, the proposed algorithm is still helpful by providing a smaller list of victim candidates of systematic issues.

V. AN INDUSTRIAL CASE

The proposed algorithm was further validated on an industrial design as follows:

1) Bridge features are extracted from layout similar to [8].
2) A certain number of these bridges are randomly selected to be injected as defects.
3) A SPICE based simulator from [13] is used to simulate the injected bridge defects and derive more than ten thousand faillogs.
4) Run diagnosis on these faillogs.
5) Feature failure probabilities are computed by applying the statistical learning algorithm on the volume diagnosis results.

6) Compare the computed failure probabilities with the actual failure probabilities of injected bridges.

After excluding features with a very small number of instances, only ten features are studied in this work: side-to-side metal 1 through 5 ($S1$ to $S5$), corner-to-corner metal 1 through 4 ($C1$ to $C4$), and side-to-side over wide metal on metal 5 ($W5$). The actual injected failure probability of each feature is trivially computed by dividing the total number of injected defective instances by the total number of instances fabricated for each feature. Fig. 3 shows the correlation. As we can see, the curve FP_Lrn representing the learned failure probabilities is quite similar to the curve FP_Inj, the injected failure probabilities. More importantly, notice that the relative order, i.e. the ranking from the most important yield limiting feature down to the least important yield limiting feature is the same between the injected and the learned features. Only for $C3$ the ranking would be lower, which may be caused by the correlation between features and needs further investigation.

Without the absolute yield for the features studied, the proposed algorithm can only achieve a relative accuracy of the feature failure probability. With additional calibration, the absolute accuracy of learned feature failure probability may be achieved, which enables other applications based on learned feature failure probabilities, such as yield prediction and design-for-manufacture (DFM) rule calibration.

The proposed algorithm has been successfully applied to the volume diagnosis results of more than twenty-three thousand real faillogs of a large industrial design in [8], [11]. This design has more than two million gates, is manufactured in $110nm$ technology node. The failure probability for various features was computed from the volume diagnosis results using the proposed algorithm. It was shown that such information could be very valuable for understanding the defect mechanism of the manufacturing process, and also very helpful to identify the defect prone layout features.

VI. Conclusions And Future Works

In this paper, we have proposed a novel statistical learning algorithm to effectively analyze volume diagnosis results. We have demonstrated the effectiveness, robustness and accuracy of the proposed algorithm by using Monte-Carlo simulation and by performing controlled experiments on an industrial design. Feature failure probabilities can be accurately recovered from volume diagnosis results, and valuable defect mechanism information, such as systematic yield limiters, can be successfully identified. The accurate learned feature failure probabilities can also be used to improve the rankings of suspects for each single failing die. Further improvements can be made to address the correlations between features.

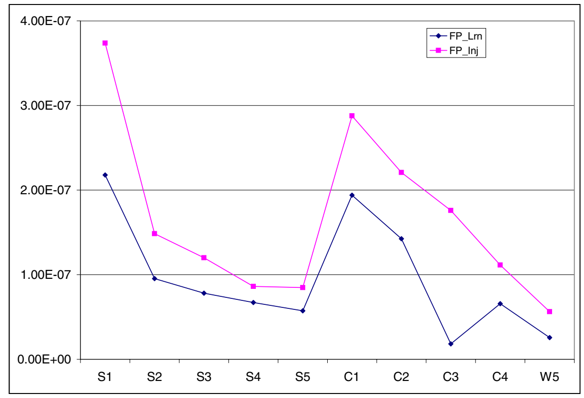


Fig. 3. Controlled experiment on an industrial design

Acknowledgment

The authors would like to thank Dr. Nagesh Tamarapalli for his instructive discussion during this project and Dr. Wei Zou for his help creating faillogs for the controlled experiments.

References

[1] M. E. Amyeen, D. Nayak, and S. Venkataraman, *Improving precision using mixed-level fault diagnosis*, Proc. of ITC, paper 22.3, 2006

[2] D. Appello, A. Fudoli, K. Giarda, E. Gizdarski, B. Mathew, and V. Tancorre, *Yield analysis of logic circuits*, Proc. of VTS, pp. 103-108, 2004

[3] T. Bartensten, B. Koenemann, L. Todd, and L. Valerie, *Integrating logical and physical analysis capabilities for diagnositics*, Proc. of ISTFA, pp. 521-526, 2004

[4] D. Bodoh, A. Blakely, and T. Garyet, *Diagnostic fault simulation for the failure analyst*, Proc. of ISTFA, pp. 181-190, 2004

[5] D. Chieppi, G. De Nicolao, P. Amato, D. Appelo and K. Giarda, *E⋆: A new statistical algorithm to enhance volume diagnostic effectiveness and accuracy*, IEEE Silicon Debug and Diagnosis Workshop, 2006

[6] H. Erb, C. Burmer, and A. Leininger, *Yield enhancement through fast statistical scan test analysis for digital logic*, Proc. of Adv. Semi. Manuf. Conf. and Workshop, pp. 250-255, 2005

[7] C. Hora, R. Segers, S. Eichenberger, and M. Lousberg, *An effective diagnosis method to support yield improvment*, Proc. of ITC, pp. 260-269, 2002

[8] M. Keim, N. Tamarapalli, H. Tang, M. Sharma, J. Rajski, C. Schuermyer, and B. Benware, *A rapid yield learning flow based on production integrated layout-aware diagnosis*, Proc. of ITC, paper 7.1, 2006

[9] B. Kruseman, A. Majhi, C. Hora, S. Eichenberger, and J. Meirlevede, *Systematic defects in deep sub-micron techologies*, Proc. of ITC, pp. 290-299, 2004

[10] A. Leininger, P. Muhmentaler, W.-T. Cheng, N. Tamarapalli, W. Yang, and H. Tsai, *Compression mode diagnosis enables high volume monitoring diagnosis flow*, Proc. of ITC, paper 7.3, 2005

[11] J. Rajski, G. Chen, M. Keim, N. Tamarapalli, M. Sharma, and H. Tang, *Determining and analyzing integrated circuit yield and quality*, USA patent application, No. 11/221,395, Sept. 6, 2005

[12] C. Schuermyer, K. Cota, R. Madge, and B. Benware, *Identification of systematic yield limiters in complex ASICS through volume structural test fail data visualization and analysis*, Proc. of ITC, paper 7.1, 2005

[13] W. Zou, W.-T. Cheng, and S. M. Reddy, *Bridge defect diagnosis with physical information*, Proc. of ATS, pp. 248-253, 2005

[14] W. Zou, W.-T. Cheng, S. M. Reddy, and H. Tang, *On methods to improve location based logic diagnosis*, Proc. of VLSI Design, pp. 181-187, 2006