Determining a Failure Root Cause Distribution From a Population of Layout-Aware Scan Diagnosis Results

Brady Benware, Chris Schuermyer, and Manish Sharma
Mentor Graphics Corp.

Thomas Herrmann
GLOBALFOUNDRIES

Editor's note:
A framework for making root cause inferences from a population of logic-test-failing die is presented. The foundation of this methodology is a layout-aware diagnosis method and a Bayes net model. An "expectation maximization" algorithm is used to derive estimated root cause summaries even with the inherent ambiguity in the diagnosis calls.

—Anne Gattiker and Phil Nigh, IBM

The yield of an integrated circuit (IC) is well known to be a critical factor in the success of an IC in the market place. Achieving high stable yields helps ensure that the product is profitable and meets quality and reliability objectives. When a new manufacturing process is introduced, or a new product is introduced on a mature manufacturing process, yields will tend to be significantly lower than acceptable. The ability to meet profitability and quality objectives, and perhaps more importantly, time-to-market and time-to-volume objectives depend greatly on the rate at which these low yields can be ramped up. While the yield ramp depends on both the yield learning and yield enhancement cycle times, this work focuses on significantly increasing the value of test data and the yield learning rate.

Several recent works have investigated methods to extract meaningful information about the failure mechanisms causing yield loss from volume scan diagnosis results [1]–[7]. Scan diagnosis uses design information and tester failing cycle data to identify specific locations in the design that are likely to explain these failing cycles. The primary challenge for yield analysis of diagnosis data is dealing with the ambiguity present in the diagnosis results. The ambiguity in this process is twofold. First, in diagnosis, it is often possible that more than one location can explain the defective logical behavior observed in the failing cycles. Second, each suspect location will often have multiple possible root causes associated with it. Many of the aforementioned works have proposed aggregation of the raw diagnosis results, including the ambiguity, and still extract some meaningful information. In practice, these techniques have limited applicability. In contrast, a subset of these works has focused on statistical techniques aimed at eliminating the ambiguity from the aggregation of results. This work fits within this subset.

Iterative learning algorithms used to eliminate the diagnosis ambiguity and thus estimate the root cause distribution have been studied in [2] and [3].

Digital Object Identifier 10.1109/MDT.2011.2178386
Date of publications: 07 February 2012; date of current version: 30 April 2012.

0740-7475/12/$31.00 © 2012 IEEE Copublished by the IEEE CEDA, IEEE CASS, and IEEE SSCS **IEEE Design & Test of Computers**

These studies have demonstrated the feasibility of using maximum-likelihood parameter estimation to extract root cause distribution information, but have two specific limitations. First, neither of these works deals with metal open defects as possible root causes. This class of defects is much more problematic for root cause learning than bridge or cell internal defects due to the correlation among the metal and via layers in the diagnostic callouts. The techniques proposed in those works will not produce accurate results for open defects. Open defects have been investigated in [5], but the approach is limited to situations where only one root cause is present and therefore has limited applicability. Second, these prior works do not address the issue of overfitting, which is of real practical concern as the number of possible root causes can be on the order of hundreds or thousands, and in some case, significantly outnumber the total number of failing devices being analyzed.

This work addresses the aforementioned challenges and fundamentally extends the current body of work in two ways. First, previous works have used specific learning algorithms designed with a very narrow focus, while this paper presents a general probabilistic framework for the learning process based on the concept of a Bayes net. This is advantageous as there is a large body of work in the machine learning community that is directly applicable with this framework and alternative learning algorithms could easily be employed. In addition, extending the Bayes net model to take into account additional dependencies can be done without loss of generality or applicability of current or future work.

The second contribution of this work is a learning algorithm based on the well-known expectation–maximization (EM) principle that has been extended to allow for penalization of complexity to address the problem of overfitting. With this framework, it is possible to consider an extremely large set of possible root causes, which will be required in practice.

In addition to these fundamental extensions to existing work, this work incorporates a representation for all major defect classes including cell defects, interconnect short defects, and interconnect open defects (both metal layers as well as specific via configurations). This is the first time that a complete enough set of possible root causes has been employed such that this form of yield analysis can be of practical use as demonstrated by the silicon experiments presented herein.

The remainder of this paper is organized as follows. The second section presents the details of the complete learning and inference process, which we refer to as root cause deconvolution (RCD). The third section presents the results from a series of simulation experiments, followed by the results from a 28-nm yield ramp investigation. The paper is concluded with a discussion of limitations and possible future work to address these challenges.

Root cause deconvolution

The objective of RCD is as follows. Given a population of failing devices with their associated failing test cycles and given a set of possible root causes, determine how many failing devices were the result of each root cause. To accomplish this, the process can be conceptualized as having two main phases. The first phase operates on a single failing device and consists of determining the probability of getting the observed failing cycles if one were to know what the real root cause is. This is done for each root cause under consideration and requires a probabilistic model for how a design will fail for a given root cause. The second phase uses the probabilities for each failing device and is focused on finding a root cause distribution that yields the largest total probability of seeing the results from all the failing devices. One can imagine a very simplistic approach to this being to calculate the collective probability of seeing all the failing cycles from all the failing devices for every possible root cause distribution and selecting the distribution that produces the maximum total probability. This simple approach is of course not practical, but does accurately depict the intent that the algorithm presented herein aims to achieve.

The actual RCD process has four basic steps as outlined in Figure 1 and details of each step are provided below.

Layout-aware diagnosis

Layout-aware diagnosis [8] is the process that analyzes scan failures observed on the tester for a defective die to produce a list of suspects that potentially contain the real defect. If a die contains multiple defects, diagnosis arranges the suspects into multiple symptoms, each symptom corresponding to one defect in the die, and each suspect within

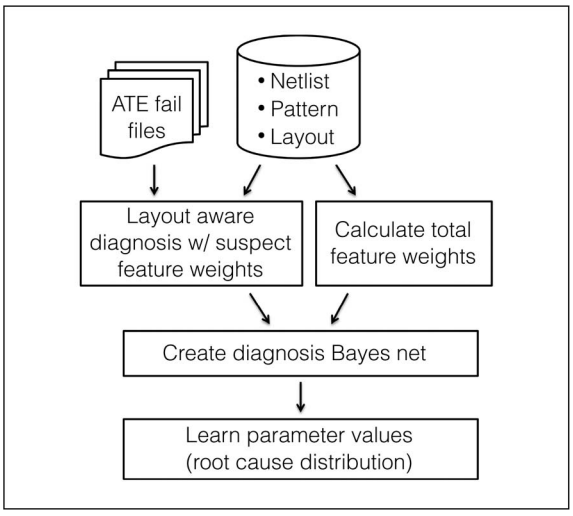


Figure 1. The basic flow for the complete RCD process.

that symptom representing a potential location and type for the defect. The diagnosis process uses a logic level model of the design along with physical layout information to perform this analysis. Three basic types of suspects are produced by layout-aware diagnosis: opens, bridges, and cell internal. These three correspond to interconnect open defects, interconnect short defects, and defects inside library cell boundaries.

For an open suspect, the potential defect location is resolved to a segment of a particular net. A net segment is defined as a physical portion of the net that drives the same net branches. In addition to the identification of a specific net segment location, the metal open critical area and the via counts for each via type and layer are also identified in the diagnosis callout. These features represent the potential physical root cause of failure for that location.

Bridge suspect locations are identified by a net pair which are considered electrically connected by a short defect. All line of site neighbors are considered and the diagnosis callout identifies the layer and critical area associated with these physical regions. Suspects are also classified as having a side-to-side, corner-to-corner, or end-of-line configuration for consideration of some systematic effects in addition to random-particle-type defects.

A cell internal suspect has two key features. First, in addition to the location of the cell, the associated library cell type is identified, which can be used to model systematic issues in the library. Second, the physical area of the cell is specified and used to model random-particle-like defects.

Calculating total feature weights

Determining the probability of observing a specific suspect given the root cause mechanism and given that it is the correct suspect requires the feature weights from the diagnosis callout, but additionally requires the computation of the total feature weight for all suspects that could potentially be observed in a diagnosis report. The computation of the total feature weight has two parts: 1) physical extraction of all possible suspects and their associated feature weights and 2) determination of which potential suspect can actually show up as suspects when diagnosing scan failures. The physical extraction of features for potential suspects is done in exactly the same fashion as during diagnosis itself. A potential suspect can be a net segment, a bridge pair, or a cell instance, and each will have the same features as described in the previous section.

Before adding the feature weight of a potential suspect to the total, it must be determined whether the suspect could be diagnosed under the current testing conditions. A suspect can only show up in a scan diagnosis report if it meets two criteria. First, the associated defect must be detected by the scan test pattern set applied to the part. Because the defect behavior can be highly arbitrary and hard to predetermine, heuristics based on multiple detection of stuck-at faults are used to determine when a potential suspect is highly likely to be detected by the scan test. Second, the defect associated with a potential suspect must *not* be detected by any tests applied before the scan test. Typically, the main defects that fall into this category are those affecting proper operation of the scan chain. It should be noted that diagnosis of chain failures is possible, but produces significantly different suspects that are not currently being modeled for the statistical analysis described herein and are therefore excluded. In general, the detection of any test that precedes the scan test, such as functional or memory bist test, must also be considered. In this work, this was not necessary as scan tests were the first to test the logic.

With the feature weights and detection information determined for each potential suspect, calculation of the total feature weight is a simple sum over all the diagnosable suspects for each suspect feature.

Creating the diagnosis Bayes net

A Bayes net is a graphical model that represents a set of random variables and their corresponding conditional dependencies. Each node in the graph (circles) represents a random variable, while each edge (arrow) represents dependence. The arrow points toward the dependent random variable and indicates that the value of the random variable at the tail of the arrow depends on the value of the random variable at the head. In this paper, filled circles represent observed random variables whereas empty circles represent hidden (or unobserved) random variables. The enclosing boxes are called tiles and indicate that the enclosed contents are repeated the number of times indicated by the variable in the lower right-hand corner of the tile.

For the purposes of this work, it is desired to construct a Bayes net that captures the probabilistic relationship between the unknown root cause (RC), the unknown correct suspect (CS), and the observed set of suspects (S) in a diagnosed symptom. Figure 2 shows the generic form of the Bayes net developed in this work. When performing learning for a specified population of failing devices, a specific Bayes net is constructed in this form.

To help illustrate this point, Figure 3 shows a fully enumerated example for a population of two failing devices, each of which has a diagnosis result. The first diagnosis result has one symptom containing two suspects. The second diagnosis result has two symptoms with three suspects and one suspect, respectively. The primary index for the random variables has a range from one to the total number of symptoms contained in all diagnosis reports (denoted as I in Figure 2).

The details of the model are as follows:

- RC_i: The root cause of the ith diagnosis symptom. This is a hidden variable and can take on any value in the list of L valid root causes being considered.
- CS_i; The correct suspect of the ith diagnosis symptom. This is a hidden variable and can take on any value in the list of possible suspects for the ith diagnosis symptom.

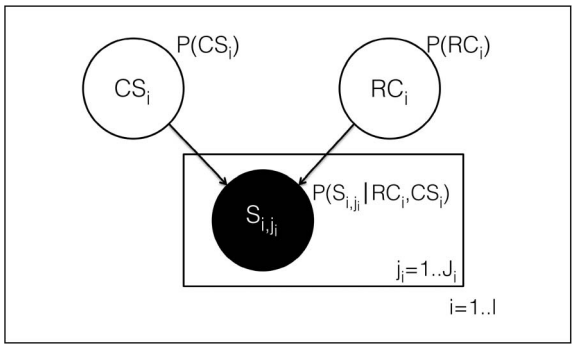


Figure 2. Bayes net model for RCD. Unfilled nodes are hidden variables, filled nodes are observed variables, and boxes specify that the enclosed items are repeated.

- S_{i,j_i}: The observed suspect list of the ith diagnosis symptom. This is an observed variable and has the value of the complete list of suspects in the ith symptom.
- $P(RC_i)$: It specifies the unknown probability distribution of root causes. This unknown distribution is parameterized in the following way:

$$P(RC_i = rc_l) = \frac{\theta_l}{\sum_{m=1}^{L} \theta_m}, \qquad l = 1, \ldots, L$$

where L is the total number of possible root causes under consideration and θ_l is the total number of defects in the population of failing devices corresponding to root cause rc_l. The vector $\boldsymbol{\theta}$ is the set of parameters of the model that will be learned from the diagnosis data. Notice that by definition, $P(RC_1) = P(RC_2) = \cdots = P(RC_l)$,

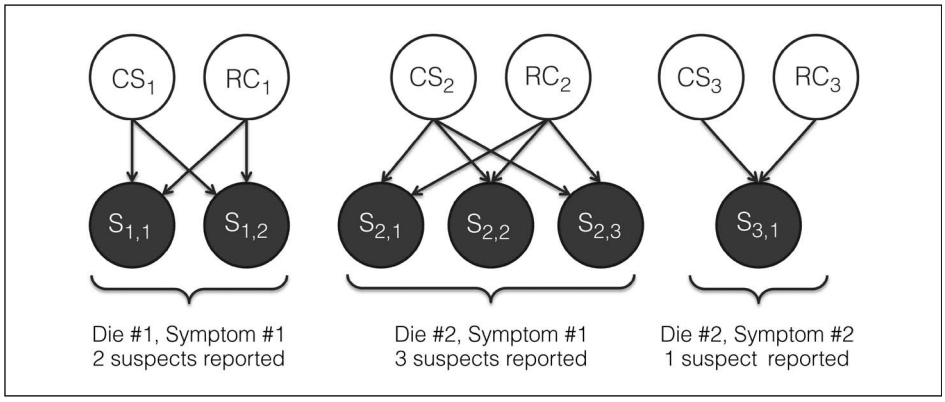


Figure 3. Example Bayes net for two failing devices in the considered population.

which implies that there is a single underlying defect distribution for a specified set of failing devices. With this model, care should be taken to separate failing devices into different populations if they are known to have significantly different processing conditions.

- $P(CS_i)$: It specifies the probability for each suspect in the ith symptom that it is the correct suspect. This represents the prior knowledge about the likelihood that a suspect may be the correct one. The actual probabilities represented by this prior knowledge will depend greatly on the diagnosis tools that generate the results, and the details of this calculation will therefore not be presented. However, the reader may assume that this is a function of diagnosis attributes such as how well the diagnosis suspect's behavior matches the ideal fault model, number of suspects on the report, number of symptoms on the report, and fault type, and can be expressed as

$$P(CS_i = s_{i,k_i}) = f(\text{attributes of } s_{i,k_i} \text{ and symptom } i), \quad k_i = 0, \ldots, J_i$$

where J_i is the number of suspects in the ith symptom and s_{i,k_i} is the k_ith suspect in the ith symptom. The special case $s_{i,0}$ corresponds to the scenario in which none of the reported suspects are correct for that symptom. In the event that prior knowledge about the likelihood of each suspect being correct is not known, then an equal probability distribution can be used.

- $P(S_{i,j_i}|RC_i, CS_i)$: It specifies the probability of observing a specific suspect as the j_ith suspect in the ith symptom given knowledge of the root cause and knowledge of the correct suspect. This probability can be broken into two different scenarios. In the first scenario, the suspect under consideration is the correct suspect. In this case, the probability of observing a specific suspect will depend on that suspect's susceptibility to the root cause. For example, if the root cause under consideration is a random particle open defect on metal layer two, then the probability of observing a specific suspect will be equal to the suspect's critical area for open defects on metal layer two, divided by the total critical area for all possible suspects.

  The second scenario is when the suspect under consideration is not the correct suspect. The model presented herein assumes that the observation of an incorrect suspect is independent of the correct suspect and independent of the root cause. The probability of observing a specific incorrect suspect is therefore only related to how many suspects there were to choose from. This further assumes that each potential suspect has the same probability of showing up on a diagnosis report. Therefore, the complete conditional probability can be expressed as

$$P(S_{i,j_i} = s_{i,j_i}|CS_i = s_{i,k_i}, RC_i = rc_l) = \begin{cases} \dfrac{w_{j_i,l}}{W_l}, & \text{if } s_{i,j_i} = s_{i,k_i} \\ \dfrac{1}{cnt(\text{suspects of type } s_{i,j_i})}, & \text{otherwise.} \end{cases}$$

  For the first condition, $w_{j_i,l}$ is the suspect feature weight for root cause l of the ith symptom and j_ith suspect. W_l is the total feature weight (as described in the "Calculating total feature weights" section) for root cause l. For the second condition, suspect type is one of cell suspect, bridge suspect, or open net segment suspect. Because each suspect type can have a different number of total occurrences in the design, each suspect type must be considered separately. Note that the special case where the correct suspect is not on the diagnosis report (i.e., $CS_i = s_{i,0}$) is also handled by this second condition.

Learning the parameter values

The primary objective of the learning phase is to determine the unknown parameter values of the model, in this case $\boldsymbol{\theta}$, which leads to an understanding of the root cause distribution. If the random variables RC were known (which they are not) for each diagnosis symptom, this would be a trivial task amounting to counting the occurrence for each root cause. Similarly, if one could accurately compute the likelihood of each root cause for each symptom, one could again determine the parameter $\boldsymbol{\theta}$ by summing the likelihood values for each root cause and thus determining $P(RC)$. As will be shown, to accurately compute these likelihood values, $P(RC)$ must be known. This dilemma of circular dependence can be resolved using the well-known expectation-maximization (EM) algorithm. The algorithm breaks the circular dependence by first making

up the parameter values, then iterating between an expectation step and a maximization step until convergence of the parameters.

- Expectation step: In this step, the expected frequencies of the possible values for the missing data are determined from the probability distribution from the previous step. This can be calculated through the posterior probability of observing a specific root cause given the observed suspect list

$$c_{i,l}^{(t)} = P(RC_i = rc_l | S_{i,1} = s_{i,1}, S_{i,2} = s_{i,2}, \ldots, S_{i,j_i} = s_{i,j_i}, \boldsymbol{\theta}^{(t)}).$$

  This can then be transformed using Bayes rule into an equation allowing direct computation by substituting the probability functions described above (see the equation at the bottom of the page).
- Maximization step: In this step, the maximum-likelihood parameters of the model are estimated from the expected frequencies of the missing data and the root cause probability distribution is updated

$$\theta_l^{(t+1)} = \sum_i c_{i,l}^{(t)}$$

$$P(RC_i = rc_l | \boldsymbol{\theta}^{(t+1)}) = \frac{\theta_l^{(t+1)}}{\sum_{m=1}^{L} \theta_m^{(t+1)}}.$$

  In the above equations, the superscript (t) and $(t+1)$ identify the variables associated with the current step and the next step, respectively. In each iteration, the new parameter values increase the likelihood function and the algorithm is run until the change in the parameter values is lower than some threshold. Because each step is guaranteed to increase the likelihood, convergence of the EM algorithm is assured. A detailed discussion of the convergence properties of the EM algorithm can be found in [9]. It is possible to apply the algorithm as stated above to learn the parameter values, however there are two issues that prevent this approach from producing accurate results.

First, the EM algorithm, while ensuring convergence, does not guarantee that it will find the global maximum of the likelihood function. A common practice to overcome this issue is to use a random restart technique. In this technique, the algorithm is repeated many times, each time starting from a different random initial guess. For each initial guess and EM convergence, the final likelihood is computed. The final parameter values are chosen from the EM run that produced the highest likelihood. The random restart method has been used for the experiments in this paper.

The second issue is the potential of this approach to overfit the data. Most modern semiconductor devices are fabricated with a number of routing layers, a number of different types of via configurations, a number of different standard cell variants, and a very large number of possible layout geometries that can all be associated with a root cause. This leads to hundreds of possible root causes to be considered in the learning process. In addition, there will very often be fewer failing devices in a population of interest than potential root causes. This creates the possibility that the learned parameter values will overfit the observations and not lead to a correct understanding of the real root cause distribution. Therefore, it becomes necessary to limit the complexity of the model and reduce the number of possible root causes to just those that are relevant to the population under consideration. However, the relevant root causes are not known *a priori*. To overcome this situation, a heuristic is added to the EM algorithm to limit the complexity of the learned model. In the maximization step, the calculation of the parameter values is changed to the following:

$$\theta_l^{(t+1)} = \begin{cases} \sum_i c_{i,l}^{(t)}, & \text{if } \sum_i c_{i,l}^{(t)} > \theta_{\min} \\ 0, & \text{otherwise} \end{cases}$$

$$c_{i,l}^{(t)} = \frac{\prod_{j_i} \sum_{k_i} P(S_{i,j_i} = s_{i,j_i} | CS_i = s_{i,k_i}, RC_i = rc_l) \cdot P(CS_i = s_{i,k_i}) \cdot P(RC_i = rc_l | \boldsymbol{\theta}^{(t)})}{\sum_l \prod_{j_i} \sum_{k_i} P(S_{i,j_i} = s_{i,j_i} | CS_i = s_{i,k_i}, RC_i = rc_l) \cdot P(CS_i = s_{i,k_i}) \cdot P(RC_i = rc_l | \boldsymbol{\theta}^{(t)})}.$$

Table 1 Summary of Results for Single Root Cause Simulation Experiments.

	Design 1	Design 2
Gate count	100k	10k
Metal layers	6	5
Via macros	75	4
Cell types	318	44
Possible root causes	406	59
Count of experiments	20	30
Count of experiments with perfect root cause identification	11	16
Average % of learned distribution for correct root cause	93%	94%
Worst case % of learned distribution for correct root cause	58%	69%
Worst case root cause experiment	Critical Area Open M1	Count Via Macro Vial

where $\theta_{\min}$ is the minimum allowed value that any root cause parameter can be before it is eliminated from consideration in the model. By setting a specific θ_l to zero, the lth root cause is effectively removed from consideration because it cannot recover from that value. This heuristic provides a mechanism to control the complexity and cross validation is used to determine the value of $\theta_{\min}$, which yields a model with the best predictive capacity.

Results from simulated defect experiments

Experiments based on simulated defect responses in an IC have been carried out to evaluate the accuracy of RCD. In each experiment, the following steps are followed.

1) Specify a root cause distribution.
2) Randomly sample a root cause based on the specified root cause distribution.
3) Randomly sample a correct suspect based on the suspect feature weight distribution corresponding to the sampled root cause.
4) Create a simulated tester datalog for the sampled defective suspect.
5) Perform layout-aware diagnosis.
6) Repeats steps 2)–7) to create a population of diagnosis reports.
7) Perform RCD and compare the specified root cause distribution to the learned one.

In every experiment performed, there is a consistent set of root cause models used as the complete set of possible root causes. The full set in these experiments includes:

- short critical area model for each metal layer;
- open critical area model for each metal layer;
- open via macro count model for each via macro defined in the layout;
- cell-type count model for each library cell;
- one cell area model.

The critical area models use a $1/r^3$ defect size distribution. The count-based models use the count of the feature within the suspect to indicate the probability of observing that suspect given it is the root cause. For example, if a net segment contains three single vias at via layer 2, it is more likely to be defective than a net segment with only one single via 2 if the root cause is single via 2.

Table 1 summarizes the design attributes and results of many simulation experiments involving only a single root cause for two different designs. In each experiment, a population of diagnosis reports is created and analyzed with RCD for a root cause distribution with only a single root cause specified. Only a subset of possible root causes was used as the injected root cause, however, each root cause model type (e.g., critical area shorts) is represented in the results.

The results show that the learned distribution from RCD correctly identifies the simulated root cause as the only root cause in just over 50% of the cases for each design. On average, the correct root cause accounts for around 93% of the learned distribution in each experiment. The worst case for design 1 was for injected open defects on the lowest routing layer of metal. In the learned distribution for this case, the correct root cause contributed only 58% of the distribution. Among the other root causes identified, 37% of the distribution was identified to be critical area short defects at the same metal layer

and the remaining 5% was distributed among a few other root causes. A brief investigation of this result revealed that there is a high degree of correlation between open suspects and bridge suspects at the lower metal layers. As can be seen from the model of the previous section, it is assumed that fake suspects showing up on a report do not depend on the real root cause. However, this result shows that taking this correlation into account could improve the success rate of this approach, especially for lower layers of metal where the correlation is the highest.

The demonstrated single root cause experiments do a good job representing excursion wafers and other situations where there is a single dominant defect mechanism. However, in many situations, such as process and product yield ramps, there will be multiple defect types that are causing the yield loss.

Figure 4 shows a comparison between the root cause distribution as learned by RCD and that of the one specified for a simulated defect experiment containing three root causes. As can be seen in the figure, the learned distribution is reasonably accurate and shows the correct ranking of root causes. The vast majority of diagnosis ambiguity is effectively eliminated, with less than 4% of the distribution attributed to nuisance root causes.

Results from 28-nm yield ramp

To better understand the practical aspects of RCD, this section presents the results from applying the methodology to the early stages of a 28-nm yield ramp. This scenario represents not only some of the most challenging conditions for diagnosis and yield analysis, but also it represents one of the most valuable yield challenges to solve. The design used for this case study is a 28-nm yield learning vehicle consisting of 24 logically identical, isolated, and independently tested cores. Each core has about 750-k gates and has one of six different possible layouts. Each layout is instantiated multiple times to make up the 24 total cores on the design. A single ATPG pattern set providing 100% stuck-at coverage is used to test and perform data collection for every core.

RCD was performed on all cores for four lots manufactured on a 28-nm bulk process. These four lots were manufactured at the early stages of the yield ramp and were expected to be manufactured under different process conditions as a result of implementing yield improvements. Therefore, the data were processed with RCD as independent populations. A separate population of failing devices was created for each layout configuration and each manufactured lot, making for a total of 24 populations for RCD analysis. After all the analyses were

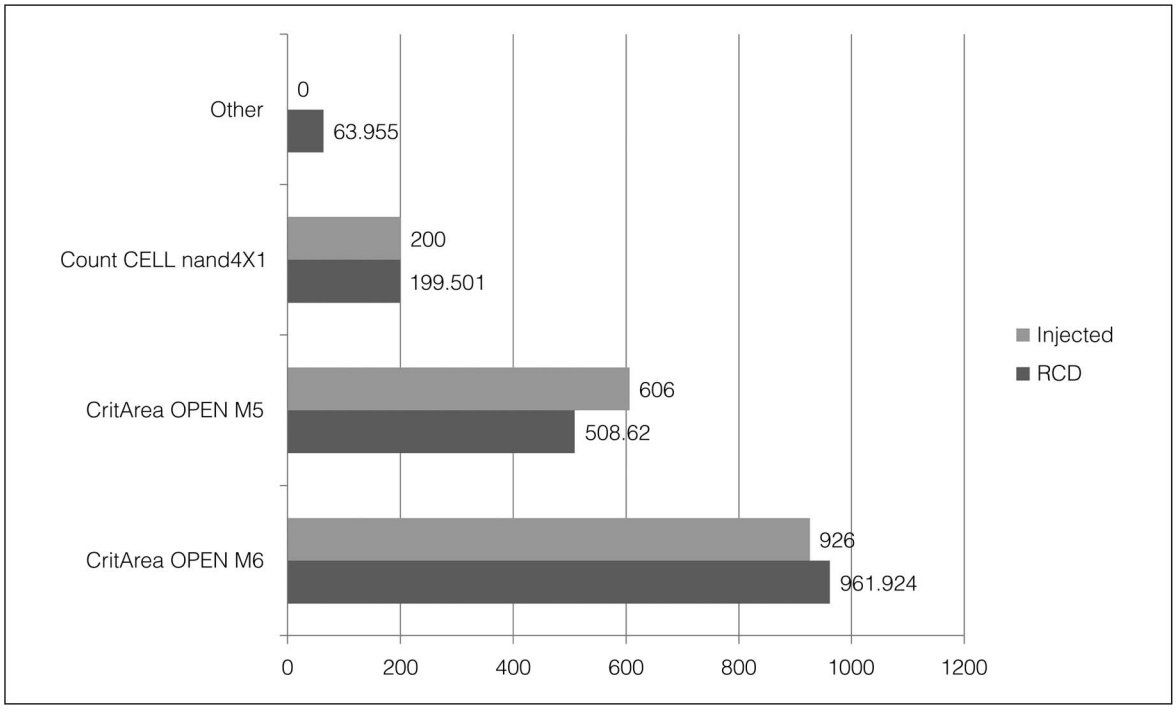
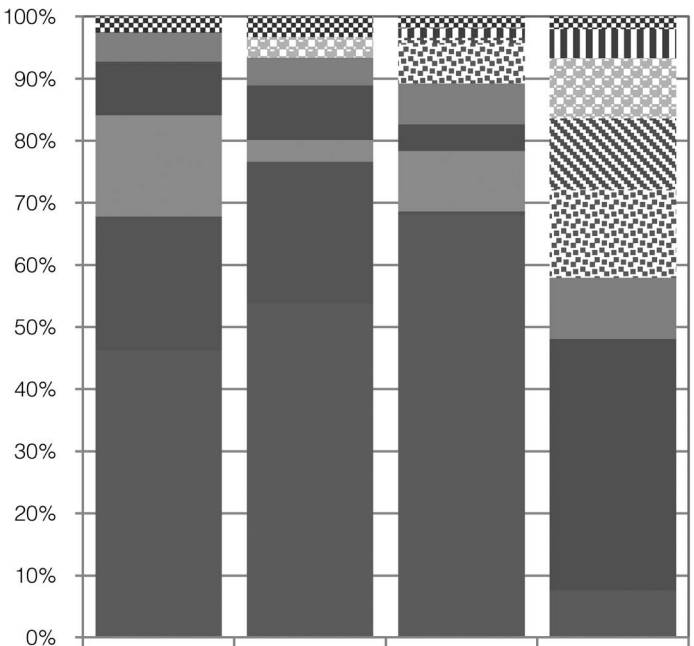


Figure 4. Results of RCD analysis with three different root causes being injected.



	Lot1	Lot2	Lot3	Lot4
Other	3%	3%	2%	2%
CritArea OPEN M4	0%	0%	2%	5%
Count ViaMacro VIA05	0%	3%	0%	10%
CritArea OPEN M3	0%	0%	1%	11%
CritArea SHORT M1	0%	0%	6%	14%
CritArea OPEN M6	5%	4%	7%	10%
CritArea OPEN M2	9%	9%	4%	40%
Count ViaMacro VIA01	16%	3%	10%	0%
CritArea SHORT M4	22%	23%	1%	0%
CritArea SHORT M6	46%	54%	68%	8%

Figure 5. Normalized RCD results summarized per lot.

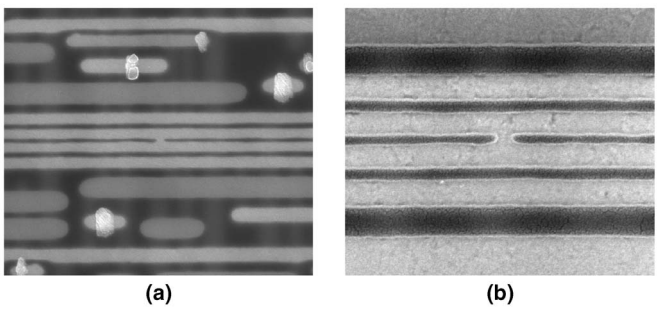


Figure 6. Top–down SEM images of (a) M4 short and (b) M6 short defects observed in Lot1 and Lot2.

completed, total root cause estimates per lot were obtained by summing the results obtained from each layout configuration. Figure 5 shows the lot summaries. The results have been normalized to the total number of symptoms for each lot. For example, for Lot1, RCD is estimating that 46% of the defects in the population of failing devices have a root cause of short type defects at metal layer M6.

The RCD results from Lot1 and Lot2 show a similar set of yield detractors, which is consistent with the fact that these two lots saw similar process conditions. These results further suggest that the yield loss for these lots is dominated by M4 and M6 shorts. Physical failure analysis (PFA) from Lot1 and Lot2 confirms this result. Figure 6 shows scanning electron microscope (SEM) images of the M4 and M6 bridges found on these lots. This systematic defect situation arises between the inner pair of four minimum spaced metal lines when surrounded by wide metal. This specific geometry causes the inner pair of metal lines to become closer to one another and increase its susceptibility to small particle defects. Improving the wafer clean process successfully eliminated the M4 shorts, which is accurately represented in the RCD results of Lot3 and Lot4.

A change to the dielectric material used for the interconnect layers was implemented in the processing of Lot4 to produce smaller metal CDs, which significantly reduced the M6 shorts. However, this process change had an adverse effect on the lower metal level interconnect open yields. RCD results again accurately reflected these changes with a significant decrease in M6 shorts as well as the emergence of interconnect open defects, which are most prevalent on metal layer M2. PFA results from Lot4

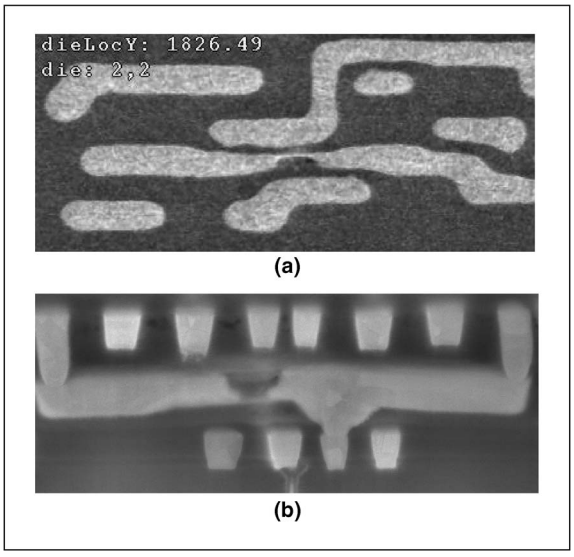


Figure 7. (a) Top–down and (b) cross-section SEM images of M2 open defects, which represent the majority of yield loss in Lot4.

confirm M2 opens as the dominant yield detractor, and revealed that this defect mechanism was sensitive to specific layout geometries.

Figure 7 shows SEM images of one M2 open failure. The top–down image shows the open occurs on a long metal layer M2 route at a point where short length surrounding metal lines reside. The RCD results showed that open defects on M3 and M4 were also a significant contributor to the yield loss on Lot4. PFA results confirmed the presence of these defects and revealed that the failure mechanism showed the same layout sensitivity.

BEING ABLE TO determine the underlying root causes represented in a population of failing devices from test data alone provides significant value to the yield and failure analysis process. This paper presents a generalized framework called RCD for learning the distribution of root causes of a population of failing devices from production test results. The key technologies for this approach are layout-aware diagnosis for each individual die and a Bayes net model for learning from the population of diagnosis results. Application of this learning framework to four lots manufactured during a yield ramp of 28-nm demonstrated very good correlation to the conclusions reached through inline inspection, failure analysis, and known process changes. The dominant yield detractors from these lots all demonstrated some sensitivity to layout geometries, which were not perfectly modeled in the possible list of root causes when performing RCD. Despite the potential limitation of imperfect defect modeling, these results demonstrate that the approach presented herein can be used successfully in practice. ■

■ References

[1] C. Hora, R. Segers, S. Eichenberger, and M. Lousberg, "An effective diagnosis method to support yield improvement," in *Proc. Int'l Test Conf.*, pp. 260–269, 2002.

[2] D. Chieppi, G. De Nicolao, P. Amato, D. Appelo, and K. Giarda, "E* A new statistical algorithm to enhance volume diagnostic effectiveness and accuracy," in *IEEE Silicon Debug and Diagnosis Workshop*, 2006, Paper 7.

[3] H. Tang, M. Sharma, J. Rajski, M. Keim, and B. Benware, "Analyzing volume diagnosis results with statistical learning for yield improvement," in *Proc. Eur. Test Symp.*, pp. 145–150, 2007.

[4] X. Yu and R. D. Blanton, "Estimating defect-type distributions through volume diagnosis and defect behavior attribution," in *Proc. Int'l Test Conf.*, pp. 1–10, 2010, Paper 22.3.

[5] M. Sharma, B. Benware, L. Ling, D. Abercrombie, L. Lee, M. Keim, H. Tang, W.-T. Cheng, T.-P. Tai, Y.-J. Chang, R. Lin, and A. Man, "Efficiently performing yield enhancements by identifying dominant physical root cause from test fail data," in *Proc. Int'l Test Conf.*, pp. 1–9, 2008, Paper 14.3.

[6] L.-C. Wang, P. Bastani, and M. S. Abadir, "Design-silicon timing correlation: A data mining perspective," in *Proc. 44th Annual Design Automation Conf.*, pp. 384–389, 2007.

[7] T. W. Chiu, O. Poku, and R. D. Blanton, "Systematic defect identification through layout snippet clustering," in *Proc. Int'l Test Conf.*, pp. 1–10, 2010, Paper 13.2.

[8] Y.-J. Chang, M.-T. Pang, M. Brennan, A. Man, M. Keim, G. Eide, B. Benware, and T.-P. Tai, "Experiences with layout-aware diagnosis—A case study," *Electronic Device Failure Analysis*, vol. 12, no. 12, pp. 12–18, 2010.

[9] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. New York: John Wiley & Sons, 1996.

Brady Benware is an engineering director in the Silicon Test Solutions group at Mentor Graphics with responsibility for debug, diagnosis and yield learning related products. Brady has a PhD, MS, and BS in electrical engineering from Colorado State University.

Chris Schuermyer is the product lead for the emerging area of Diagnosis Driven Yield Analysis at Mentor Graphics. His research interests include designing software solutions for yield learning, DFM analysis in silicon, defect modeling, yield and quality learning, and data science. He is a member of IEEE.

Manish Sharma is the product lead for the diagnosis tool in the Silicon Test Solutions group at Mentor Graphics. Manish Sharma has PhD and MS degree in electrical engineering from University of Illinois at Urbana-Champaign and a BS in electrical engineering from Indian Institute of Technology, Delhi.

Thomas Herrmann is MTS Product Engineer at GLOBALFOUNDRIES Yield Analysis Systems team focusing on Yield Learning through Scan Diagnosis. Previously, at AMD, Thomas was a member of the Dresden Design Center taking care of DFT, Test and Product Engineering as well as yield and defect analysis of a number of complex digital products. He earned his MS in computer science from the University at Jena, Germany.

■ Direct questions and comments about this article to Brady Benware, Mentor Graphics Corp. Wilsonville, OR 97070, USA; brady_benware@mentor.com.