

# Fitting Tester Yield Curves

José T. de Sousa

INESC/IST  
Technical University of Lisbon  
R. Alves Redol, 1000 Lisboa, Portugal  
*jose.desousa@inesc.pt*

Vishwani D. Agrawal

Bell Labs  
Lucent Technologies  
Murray Hill, NJ 07974, USA  
*va@research.bell-labs.com*

**Keywords:** *defect level, fault clustering, fault coverage, reject ratio, tester yield, yield*

## Abstract

*In this paper we present two case studies of fitting tester yield and defect level models to actual test data. The two case studies correspond to two different chips from two different manufacturers. Our results provide new evidence that the models that account for the defect/fault clustering effect are the most able to track the experimental data. We also present an improved numerical non-linear curve fitting procedure.*

## 1 Introduction

The Yield ( $Y$ ) of a given IC manufacturing process is the fraction of fault free chips that come out from production. It is during production test that the good chips are separated from the faulty ones and  $Y$  becomes known.

Defect level ( $DL$ ), also known as *reject ratio*, is the fraction of faulty chips among the chips that pass production test. These chips are taken as good devices and shipped to customers. They can later fail in the field, causing the manufacturer significant expense and adversely influencing important invisible parameters such as customer satisfaction, company prestige, etc.

The economical importance of  $Y$  and  $DL$  hardly needs to be highlighted. The problem is how to predict and control their values.

To cope with the complexities of physical defect phenomena, sophisticated yield models have been developed. Good yield and defect models must account for the defect clustering phenomenon, which has been shown indispensable to produce accurate predictions [1, 2, 3].

In this work we are particularly interested in expressing  $Y$  and  $DL$  as a function of the fault coverage  $f$ . As tests do not usually achieve 100% fault coverage ( $f = 1$ ), if we can find good  $Y(f)$  and  $DL(f)$  models we can predict the true yield  $Y_0 = Y(1)$ , which is used in the  $DL(f)$  model:

$$DL(f) = 1 - \frac{Y_0}{Y(f)}. \quad (1)$$

A strong assumption in this work is that  $DL(1) = 0$ , i.e.,  $DL$  will be zero if we manage 100% fault coverage. Therefore, to reflect realistic  $Y$  and  $DL$  values, the fault model used should be as accurate as possible. The stuck-at fault model is still the most widely used model and therefore is the one we use here. Our  $DL$  actually means the fraction of chips that pass the test while containing modeled faults. It cannot say anything about unmodeled faults. However, this is not a paper on fault modeling, and we will work with any fault coverage metric the user deems appropriate.

Most well known yield theories [1] distinguish between two types of contributions to the overall yield value: the *gross yield* ( $Y_g$ ) and the *random yield* ( $Y_r$ ):

$$Y = Y_g Y_r. \quad (2)$$

$Y_g$  pertains to gross defects (catastrophic phenomena) that cause immediate failure.  $Y_r$  refers to localized defects, which require more effort to be detected. It is natural to expect that  $Y_g$  contributions can be found in all testing phases, including the stuck-at fault testing one. However, the existing  $Y(f)$  models do not usually incorporate  $Y_g$ , seeming to imply that during this test stage a gross yield factor is unnecessary. Our results show that, on the contrary, the inclusion of a  $Y_g$  parameters in the models improves their ability to track the experimental data and improves the  $Y_0$  and  $DL$  predictions.

This paper is organized as follows. In Section 2 we discuss the  $Y(f)$  models to be used in the experiments. In Section 3, the curve fitting optimization procedure to find the parameters of the models is explained. Experimental results are reported in Section 4. There, it is shown how the introduction of the  $Y_g$  factor improves the curve fits. Section 5 concludes the paper and gives directions for future developments.

## 2 The models

In this section, the  $Y(f)$  models we used in the experiments are discussed. All these models disregard the gross yield factor  $Y_g$ , and thus assume that  $Y(f) = Y_r(f)$ .

In a lot containing  $N$  chips, if  $R_i$  new chips are rejected after application of test vector  $i$ , which corresponds to a fault coverage  $f_i$ , then  $Y_r(f)$  can be experimentally evaluated in a sample of  $N$  chips by

$$\hat{Y}(f_i) = \frac{N - \sum_{i=1}^n R_i}{N}. \quad (3)$$

We will require that a good  $Y(f)$  model is such that it can fit the experimental points  $\hat{Y}(f_i)$  as closely as possible with a small number of parameters. Note that  $N$  must be large enough so that the  $\hat{Y}(f_i)$  values from the sample become close to the real  $Y(f_i)$  values from the population. Once a good  $Y(f)$  model is found, we can predict the true yield as  $Y_0 = Y(1)$  and  $DL$  by Equation (1).

With a set of assumptions equivalent to considering a Poisson distribution for the number of faults in a chip, Williams and Brown [4] derived the following yield model

$$Y(f) = e^{-\lambda_f f} \quad (4)$$

where  $\lambda_f$  is the average number of faults per chip. Attempts to confirm the Williams-Brown model experimentally have not been very successful because the assumption that faults are Poisson distributed turned out to be not very accurate.

Another model based on the Williams-Brown formulation has been proposed by de Sousa et al. [5]. Its equation is

$$Y(f) = e^{-\lambda_f [1 - (1-f)^R]}, \quad (5)$$

where  $R$  is a parameter that was called the fault-to-defect susceptibility ratio. This model recognized that fault coverage must be mapped to defect coverage before it can be used in the Williams-Brown formulation. Compared to the Williams-Brown model it produces a lower fitting error.

Considering a defect model with two parameters, the true yield  $Y_0$  and the average number of faults in a faulty chip  $n_0$  (assumed to be a Poisson random variable), Agrawal et al. [6] proposed the model

$$Y(f) = Y_0 + (1-f)(1-Y_0)e^{-(n_0-1)f}. \quad (6)$$

The fact that  $n_0$  can be greater than 1 is a possible way of modeling fault clustering. Equation (6) proved much more accurate than the Williams-Brown formula at tracking experimental data.

The clustering effect was explicitly incorporated by Seth and Agrawal [2] in their negative binomial model

$$Y(f) = \left[ 1 + \frac{\lambda_d (1 - e^{-cf})}{\alpha_d (1 - e^{-c})} \right]^{-\alpha_d}, \quad (7)$$

where  $\lambda_d$  and  $\alpha_d$  are the average number of *defects* in the chip and the *defect* clustering parameter, respectively, and  $c$  is the average number of faults per defect, assumed to be Poisson distributed. The three parameters are positive real numbers. This model produced better fits to experimental data, and more realistic predictions of  $Y_0$  and  $DL$ .

Realizing that defects causing multiple faults can be modeled by the clustering parameter, de Sousa and Agrawal recently proposed a simplified version of the Seth-Agrawal model [3],

$$Y(f) = \left( 1 + \frac{\lambda_f}{\alpha_f} f \right)^{-\alpha_f}, \quad (8)$$

where  $\lambda_f$  and  $\alpha_f$  is the average number of *faults* in the chip and  $\alpha_f$  is the *fault* clustering parameter. From now on the clustering parameter and the average number of defects or faults will be represented simply by  $\alpha$  and  $\lambda$ , assuming that the reader will know when we are talking about defects or faults. For the chip studied in [3], this model was able to produce  $Y_0$  and  $DL$  values close to the values predicted by the Seth-Agrawal model, while exhibiting a smaller fitting error.

## 3 Optimization procedure

Having presented the  $Y(f)$  models, we must now address the problem of finding their parameters by non-linear curve fitting. The data has been collected while applying a set of test vectors to actual chip lots. For each applied test vector  $i$  we have the number of rejected chips  $R_i$ , and the cumulative stuck-at fault coverage  $f_i$ . The experimental yield  $\hat{Y}(f_i)$  is computed by Equation (3). Then, for each  $Y(f)$  model we will determine its parameters so that the square

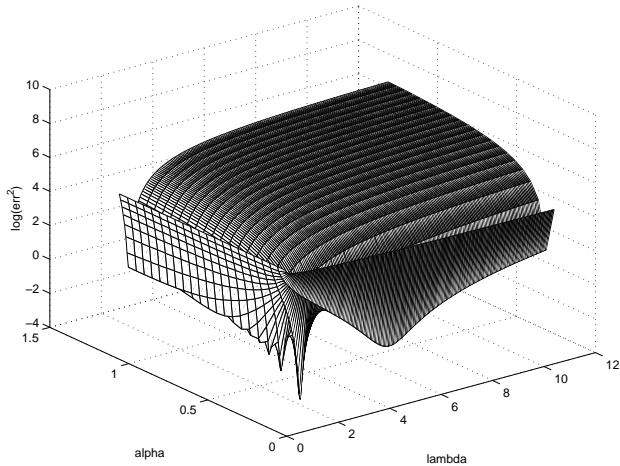


Figure 1: Optimization surface for the de Sousa-Agrawal model for an example chip.

error with respect to the experimental points  $\hat{Y}(f_i)$  is minimized. It is well known that the solution space for a nonlinear fitting problem may have multiple local minima. This forces us to use global optimization techniques, which however cannot guarantee an optimal solution. Figure 1 plots the logarithm of the square error as a function of  $\lambda$  and  $\alpha$  when applying the de Sousa-Agrawal model to one of the chips.

The technique we used initiates optimization procedures from various points in a region where we expect the solution to exist. The starting points form a grid over that region. The resolution of the grid is determined by the available computational time. However, there is a point where further increasing the resolution of the grid does not necessarily lead to better solutions.

The search region is determined by first computing a global starting point and then considering a vicinity of it. To determine the global starting point we will use the de Sousa-Agrawal example of Figure 1. The derivative of  $Y_r(f)$  at  $f = 0$  is  $-\lambda$ , according to Equation (8). Thus, by roughly measuring this derivative from the data we can determine a starting point  $\lambda_0$  for  $\lambda$ . We can use the number of chips that pass all vectors and divide it by the total number of chips  $N$  to get a rough  $Y_0$  estimate. Using  $Y_0$  and  $\lambda_0$ , we solve the yield equation numerically to get a starting point  $\alpha_0$  for  $\alpha$ . Next, given the starting point  $(\lambda_0, \alpha_0)$ , we empirically choose a vicinity beyond which local minima is unlikely to occur (e.g.,  $\alpha_0 \pm 1$  and  $\lambda_0 \pm 2$ ).

The complexity of this method is related to the size of the grid that represents the solution subspace. Considering a square grid with  $m \times n$  points the com-

plexity of the algorithm is  $O(m \times n \times C)$ , where  $C$  is the complexity of the curve fitting procedure used. Clearly, the complexity of the method increases fast with the number of parameters and resolution of the grid. Thus, it is important to keep the number of parameters small.

## 4 Experimental results

Preliminary experimental results were obtained for the 5 models featured in Section 2. In the final version of the paper more accurate results may be obtained, as there are small improvements yet to be made in our optimization script. The data examples come from two different chips: one is provided by the SEMATECH consortium, also used in [7, 3]; the other is a chip from DELCO. The maximum fault coverage for the SEMATECH chip is  $f_{max} = 0.9939$ , while for the DELCO chip  $f_{max} = 0.9889$ . The contents of the data is as described in the previous section. A preliminary version of the optimization procedure described in Section 3 was run to determine the parameters of the models. The procedure can optimize for 3 parameters and takes about 10 minutes in the worst case using a MATLAB script running on a Linux PC.

Table 1 and Table 2 give the parameter values for the 5 models on the SEMATECH and DELCO chips, respectively. It also gives the values of the predicted  $Y_0$ , the standard deviation  $\sigma_Y$  of the error in each  $Y(f_i)$  point, and the predicted  $DL$ . It should be said that  $\sigma_Y$  is not the standard deviation of the final  $Y_0$  estimate. In fact, the error in the  $Y(f_i)$  points tends to decrease with  $f_i$  and, therefore, this correlation must be analyzed before computing the accuracy of the  $Y_0$  prediction. Thus,  $\sigma_Y$  should be much higher than the actual standard deviation of  $Y_0$ , and merely serves as an indicator for the quality of the fit. For Table 1 and Table 2 the gross yield parameter  $Y_g$  has not been considered, which is equivalent to saying it has been set to  $Y_g = 1$ . For the SEMATECH chip, the results in Table 1 are slightly different from the results published in [3] because of the improvements introduced in the optimization procedure.

The results in Table 1 and Table 2 show that the de Sousa-Agrawal, Seth-Agrawal and Agrawal et al. models produce the closest fits, and similar predictions of  $Y_0$  and  $DL$ . A common feature of these models is that they all account for the defect/fault clustering effect, either implicitly (Agrawal et al.) or explicitly (Seth-Agrawal and de Sousa-Agrawal). The Williams-Brown and de Sousa et al. models produce the worst fits. In terms of the  $Y_0$  and  $DL$  predictions, if we

Table 1: Results for the SEMATECH chip with  $Y_g = 1$ 

Model	Model parameters					$Y_0$	$\sigma_Y$	$DL$
	$\alpha$	$\lambda$	$c$	$n_0$	R			
Williams-Brown [4]	-	0.31	-	-	-	0.7334	0.0269	0.0019
de Sousa et al. [5]	-	0.26	-	-	2.2	0.7711	0.0070	3.4719e-06
Agrawal et al. [6]	-	-	-	2.6	-	0.7628	0.0057	3.8579e-04
Seth-Agrawal [2]	0.29	1.98	1.46	-	-	0.7649	0.0039	4.7256e-04
de Sousa-Agrawal [3]	0.11	1.18	-	-	-	0.7628	0.0028	6.1394e-04

Table 2: Results for the DELCO chip with  $Y_g = 1$ 

Model	Model parameters					$Y_0$	$\sigma_Y$	$DL$
	$\alpha$	$\lambda$	$c$	$n_0$	R			
Williams-Brown [4]	-	0.31	-	-	-	0.7334	0.0168	0.0034
de Sousa et al. [5]	-	0.31	-	-	1.20	0.7334	0.0147	0.0014
Agrawal et al. [6]	-	-	-	2.2	-	0.7490	0.0067	0.0011
Seth-Agrawal [2]	0.37	1.92	0.96	-	-	0.7489	0.0054	0.0013
de Sousa-Agrawal [3]	0.15	0.88	-	-	-	0.7490	0.0046	0.0014

assume that the other three models are accurate, it can be said that the Williams-Brown model produces under estimates  $Y_0$  and underestimates  $DL$ . The de Sousa et al. model underestimates  $Y_0$  and overestimates  $DL$  for the SEMATECH chip. However, its  $DL$  estimate for the DELCO chip is quite accurate, despite the high fitting error and underestimated  $Y_0$  value.

Now we examine the case where the extra  $Y_g$  parameter is included. Our optimization procedure is still being upgraded to deal with the  $Y_g$  parameter. However, by manually inserting  $Y_g$  values in the models, we manage to decrease the overall fitting error. We used  $Y_g = 0.92$  and  $Y_g = 0.96$  for the SEMATECH and DELCO chips, respectively. The results are presented in Table 3 and Table 4. From the analysis of these tables it is clear that the introduction of the  $Y_g$  parameter decreases the fitting error  $\sigma_Y$ , indicating that we now have a better set of parameters. Only in one case has the fitting error increased (but not by much): the de Sousa et al. model on the SEMATECH chip. Another interesting effect is that after  $Y_g$  was introduced, the  $Y_0$  and  $DL$  predictions for different models agree much more. The de Sousa-Agrawal, Seth-Agrawal and Agrawal et al. models produced extremely close predictions. Again, the fact that they all account for the clustering effect may be responsible for their superiority. With the extra  $Y_g$  parameter the Seth-Agrawal model appeared slightly better

than the de Sousa-Agrawal model in terms of the fitting error  $\sigma_Y$ . This is a new result but it still has to be confirmed by the final version of our optimization procedure. The Agrawal et al. model has a slightly greater fitting error than the de Sousa-Agrawal and Seth-Agrawal models. The effect of the  $Y_g$  parameter can be better appreciated if we plot the  $\hat{Y}(f_i)$  data points and superimpose the curves of the various models on the same chart. This is done in Figure 2. As can be seen, after making  $Y_g = 0.96$  the model curves become much closer to the experimental points. Also note how the models that incorporate the clustering effect (de Sousa-Agrawal, Seth-Agrawal and Agrawal et al.) produce much closer fits than the models that do not (de Sousa et al. and Williams-Brown). This reaffirms the importance of having a clustering parameter.

## 5 Conclusion

We have fitted VLSI tester yield models to experimental data of two different chips from two different manufacturers: SEMATECH and DELCO. The models we used were the following: the Williams-Brown [4], the de Sousa et al. [5], Agrawal et al. [6], Seth-Agrawal [2] and de Sousa-Agrawal [3] models.

Our results present new and further evidence that the models that account for the defect/fault cluster-

Table 3: Results for the SEMATECH chip with  $Y_g = 0.92$ 

Model	Model parameters					$Y_0$	$\sigma_Y$	$DL$
	$\alpha$	$\lambda$	$c$	$n_0$	R			
Williams-Brown [4]	-	0.21	-	-	-	0.8106	0.0102	0.0013
de Sousa et al. [5]	-	0.21	-	-	1.05	0.8106	0.0099	9.8989e-04
Agrawal et al. [6]	-	-	-	1.7	-	0.8275	0.0027	6.3231e-04
Seth-Agrawal [2]	0.31	2.00	0.54	-	-	0.8285	0.0023	6.4884e-04
de Sousa-Agrawal [3]	0.15	0.38	-	-	-	0.8275	0.0025	6.5580e-04

Table 4: Results for the DELCO chip with  $Y_g = 0.96$ 

Model	Model parameters					$Y_0$	$\sigma_Y$	$DL$
	$\alpha$	$\lambda$	$c$	$n_0$	R			
Williams-Brown [4]	-	0.26	-	-	-	0.7711	0.0086	0.0029
de Sousa et al. [5]	-	0.26	-	-	1.15	0.7711	0.0061	0.0015
Agrawal et al. [6]	-	-	-	1.6	-	0.7762	0.0030	0.0018
Seth-Agrawal [2]	0.49	1.96	0.42	-	-	0.7773	0.0028	0.0018
de Sousa-Agrawal [3]	0.27	0.42	-	-	-	0.7762	0.0029	0.0018

ing effect (Agrawal et al. [6], Seth-Agrawal [2] and de Sousa-Agrawal [3]) are significantly better at tracking the data than the models that do not (Williams-Brown [4] and de Sousa et al. [5]).

We have also experimented with a gross yield ( $Y_g$ ) factor in the models. This new parameter represents catastrophic phenomena that fails a high number of chips on the first vector. Only after does stuck-at fault testing really begin. With the  $Y_g$  parameter the yield curves start from a value lower than 1 instead of from 1. The results show that practically all models benefit from the inclusion of  $Y_g$ , which is demonstrated by the lower fitting error achieved. After  $Y_g$  has been introduced, the models that account for clustering produced very similar  $Y_0$  and  $DL$  estimates. This better agreement among different models improves the confidence in the predictions. For the SEMATECH chip those models predicted a yield of about 82.8% and a defect level of about 650 parts per million (ppm). For the DELCO chip the models predicted a  $Y_0$  of about 77.6% and a  $DL$  of 1800 ppm.

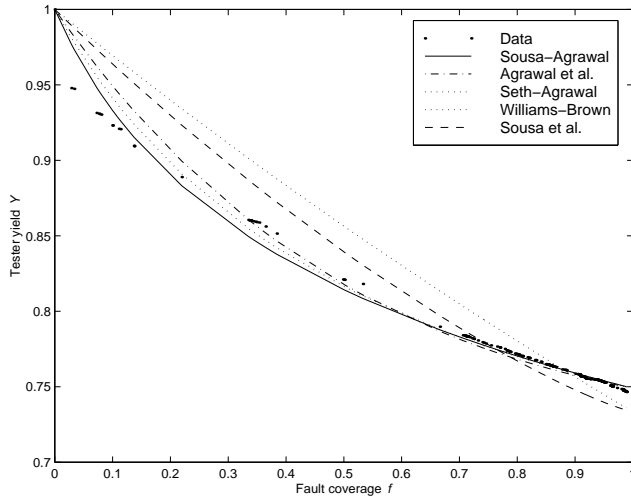
A strong assumption in this work, which is also implicit in many previous analyses, is that  $DL$  drops to zero when the stuck-at fault coverage is 100%. To be precise, we should mean the coverage of the “realistic” (i.e., those that actually occur) faults. The justification for this assumption lies in the fact that the measured tester yield depends on the capability of the tests for detecting the actual, so-called realistic,

faults. This argument needs further investigation.

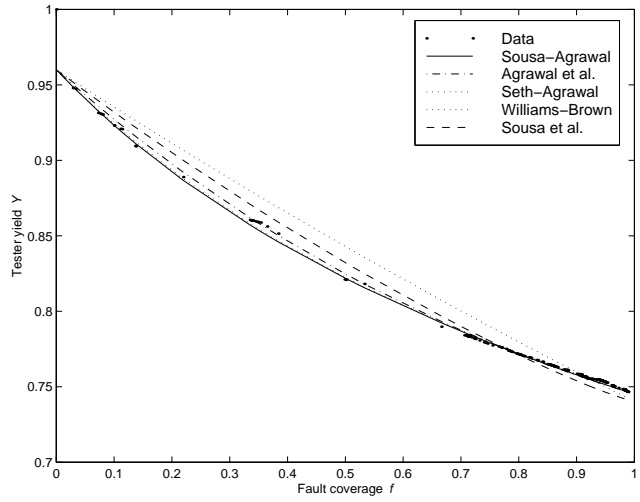
The paper only presents data on two chips and the applicability of the new model should be examined for a wide variety of chips.

## References

- [1] C. H. Stapper, F. Armstrong, and K. Saji. “Integrated Circuit Yield Statistics”. *Proc. IEEE*, 71:453–470, Mar. 1983.
- [2] S. C. Seth and V. D. Agrawal. “Characterizing the LSI Yield Equation from Wafer Test Data”. *IEEE Trans. on CAD*, CAD-3(2):123–126, April 1984.
- [3] J. T. de Sousa and V. D. Agrawal. Reducing the complexity of defect level modeling using the clustering effect. In *Proc. Design, Automation and Test in Europe (DATE) Conf.*, pages 640–644, March 2000.
- [4] T. W. Williams and N. C. Brown. “Defect Level as a Function of Fault Coverage”. *IEEE Transactions on Computers*, C-30(12):987–988, Dec. 1981.
- [5] J. T. Sousa, F. M. Goncalves, J. P. Teixeira, and T.W. Williams. “Fault Modeling and Defect Level Projections in Digital IC’s”. In *Proc. European Design and Test Conf. (EDTC)*, pages 436–442, March 1994.



(a)  $Y_g=1$



(b)  $Y_g=0.96$

Figure 2: Fitting models to SEMATECH test data.

- [6] V. D. Agrawal, S. C. Seth, and P. Agrawal. "Fault Coverage Requirement in Production Testing of LSI Circuits". *IEEE Journal of Solid State Circuits*, SC-17(1):57–61, Feb. 1982.
- [7] P. Nigh, W. Needham, K. Butler, P. Maxwell, and R. Aitken. "An Experimental Study Comparing the Relative Effectiveness of Functional, Scan, IDDQ and Delay-fault Testing". In *Proc. of IEEE 15th VLSI Test Symp. (VTS)*, pages 459–464, 1997.