RC23386 (W0410-131) October 21, 2004
Mathematics

IBM Research Report

Factorial Analysis and Forecasting of Integrated-Circuit Yield

Michael Baron
University of Texas at Dallas
Richardson, TX 75083-0688

Asya Takken
IBM Microelectronics Division
Hopewell Junction, NY 12533

Emmanuel Yashchin, Mary Lanzerotti
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

FACTORIAL ANALYSIS AND FORECASTING OF INTEGRATED-CIRCUIT YIELD

Michael Baron[1], Asya Takken[2], and Emmanuel Yashchin[3], and Mary Lanzerotti[3]

[1]University of Texas at Dallas, Richardson, Texas 75083-0688

[2]IBM Microelectronics Division, Hopewell Junction, New York 12533

[3]IBM Research Division, Yorktown Heights, New York 10598

ABSTRACT

A detailed cause-and-effect stochastic model is developed to relate the type, size, location, and frequency of observed defects to the final yield in IC manufacturing. The model is estimated on real data sets with a large portion of unclassified defects and uninspected layers, and in presence of clustering of defects. Results of this analysis are used for evaluating kill ratios and effects of different factors, identifying the most dangerous cases and the most probable causes of failures, forecasting the yield, and designing optimal yield-enhancement strategies.

Key words: diagnostics, EM algorithm, factors, goodness of fit, inspection, likelihood, outliers

1 Introduction

Our main objective is building a cause-and-effect model explaining the patterns of failing chips in terms of observable defects. Fitting such a model to training data sets allows further factorial analysis, i.e., estimation and comparison of effects of different factors, detection of influential cases and the most probable causes of failures, etc. Given a detailed estimated model, forecasting of the yield at any time during the manufacturing cycle becomes straightforward, and also, accurate yield predictions can be made for future modifications of the production process, resulting in the optimal choice of yield-enhancing strategies.

A number of models for failing chips on a wafer has been proposed, concentrating on modeling the total number of failures (e.g., [6], [7], [17], [18], [19], [20], [24]), spatial dependence of failing chips on a wafer ([1], [5], [8], [10], [21]), modeling the yield per each produced layer ([22], [23]), modeling the yield based on critical area summary curves ([4], [11], [14], [22]), and defect counts per each defect type ([12], [13], [15], [16]).

Comparing with its predecessors, the model proposed here incorporates detailed information on the observed defects, in order to predict and explain the final yield. Defect types (codes), sizes, frequencies, and locations (layers or operations) play the role of covariates. The study involving nearly 1,000 lots and millions of chips of different grades and designs showed significance of each mentioned factor. Further, the standard chi-square analysis of the log-likelihood showed significance of *interaction terms* between the defect type and the layer where the defect occurred. To avoid over-parameterization of the model, similarly composed layers were combined into groups, and only interactions between defect types and groups of layers were included.

As a result, the proposed model contains *a large number of parameters*: effect of each defect type and effect of each layer, interactions, defect frequencies, and also, effects of other causes. Apparently, some failed chips had no defects on any of their layers. Such chips were killed by causes other than observable defects. The corresponding effect is lot-specific because all wafers in a lot are produced simultaneously (unless it is decided to split a lot).

An *EM algorithm* ([9], [25]) is proposed, with some modifications, to handle this multi-scale estimation problem. The introduced modification is essentially an extra step during each cycle, in addition to the standard E-step and M-step, that accelerates the numerical routine and prevents its convergence to possible local extrema. It is shown that the new step can only improve the algorithm's performance.

It is important to notice that a bulk of information gathered on chips is always missing. For reasons of economy, vast majority of detected defects is unclassified. In addition, only selected layers are inspected on each wafer. No usable information is available for the remaining uninspected layers, although they may certainly contain fatal defects, i.e., the chip killers. Nevertheless, a carefully applied formula of total probability allows to include all the collected pieces of information (e.g., sizes and locations of unclassified defects) into the likelihood.

As mentioned in [16], due to extremely complex designs and delicate technology, the defect information gathered on chips is not perfectly clean and not perfectly reliable. Realizing this issue, the estimation routine is accompanied by the *diagnostics* module aimed to assess the goodness of fit and to detect probable outliers and influential single chips and whole wafers.

After cycles of data cleaning, parameter estimation, and diagnostics, one obtains a set of parameter estimates that explains the impact of various factors on the final yield. The estimated model is then used

- to forecast the yield at any time during the production process,
- to evaluate *kill ratios*, that is, probabilities for defects of a certain types on certain layers to be chip killers,
- to compute and compare effects associated with each defect type and each layer,

– to identify the most dangerous combinations of defect type, size, and location,

– to find, on low-yield wafers, the most probable sources of chip killers and the most probable causes of failures,

– to compare the influence of layers,

– to evaluate significance of other causes,

– to predict results of any yield-improving modification of the manufacturing process, in terms of the expected change in the final yield,

– to find the optimal strategies to increase the yield.

The paper is organized as follows. The stochastic model relating observable defects and chip failures is developed in Section 2. Parameter estimation and model diagnostics tools are derived in Section 3. Applications of the model for yield forecasting and yield improvement are discussed in Section 4. Proofs and lengthy derivations are given in Appendix.

All the results are stated for applications in semiconductor industry. However, the proposed methods of estimation, forecasting, factorial and influential data analysis can be applied to any manufacturing environment with a large number of parameters of different types and under a significant portion of missing information.

Acknowledgment. This project was completed during the first author's one-year visit to IBM Research Division. We are thankful to our colleagues at IBM who made this collaboration possible, for their support and encouragement.

2 Model Building

Let us introduce the following notation. Throughout the paper, index k represents a defect, j is a defect type, s is a defect size, l is a layer, i is a chip, w is a wafer, and m is a lot. Thus, j_k is the type of the k-th defect, l_k is a layer on which it occurred, etc. The number of chips, wafers, layers, etc. is denoted by the corresponding capital letters, I, W, L, and so on.

Next, C and U will denote the set of classified and the set of unclassified defects, respectively. Likewise, C_{lw} is the set of classified defects on layer l of wafer w. The number of classified defects (of type j on layer l) is denoted by d (d_{jl}) whereas the number of unclassified defects (on layer l) is u (u_l). The total number of defects on layer l is then $N_l = u_l + \sum_j d_{jl}$.

Also, let ξ_i be a binary variable representing the quality of chip i; $\xi_i = 1$ if the chip is good, 0 otherwise. At the same time, ϕ_i will be the probability for chip i to survive, given all its defects and other causes. Thus, each ξ_i is a Bernoulli random variable with parameter ϕ_i.

Finally, let L_w be a set of layers that were inspected on wafer w. For any layer $l \in L_w$, all the defects are counted and measured, although only a small portion of them is classified. No information is available about defects on the remaining, uninspected layers $l \notin L_w$. Layers are inspected by wafer, hence, under normal circumstances, each layer either inspected on all chips $i \in w$ or uninspected on all chips.

Parameters of the proposed model are:

- $r(j)$ for $j = 1, \ldots, J$ is the effect of defect type j,
- $a(l)$ for $l = 1, \ldots, L$ is the effect of layer l,
- $b(m)$ for $m = 1, \ldots, M$ is the effect of other causes for lot m,
- $\lambda(j, l, m)$ is the frequency of defects of type j on layer l of lot m, i.e., the expected number of such defects per chip.

We start building the likelihood from a single defect. Suppose a defect of type j occurred on layer l of chip i. What is the probability for the chip to survive this defect? A number of competing models can be proposed, such as:

$$\begin{aligned}
&\boldsymbol{P}\{\text{survival}\} = \exp\left\{-r(j)a(l)g(s)\right\} \ \text{(multiplicative model)}, && (1)\\
&\boldsymbol{P}\{\text{survival}\} = \exp\left\{-\left[r(j)+a(l)\right]g(s)\right\} \ \text{(additive model)},\\
&\boldsymbol{P}\{\text{survival}\} = \exp\left\{-r(j)g(s)-a(l)\right\} \ \text{(simplified additive model)},
\end{aligned}$$

and others. According to our experiments, model (1) dominated and provided the best fit and the most accurate forecasts. Therefore, the rest of the paper is based on the multiplicative model.

However, no theory can guarantee that the multiplicative model will continue to dominate for future chip designs. Thus, it is natural to keep *a bank of plausible models* that can be compared for each new mode of production, so that the model with the best fit can always be chosen. All the proposed methods and the implemented routines can be used for each proposed model, and only minor changes in the computer code will be necessary.

Further, the *interaction terms*, or the *joint effects* of a defect type and a layer, were found significant and significantly improving the fit, based on a large number of lots spanning several types of production. To avoid overparameterization of the model that would immediately impede its predictive power, we only include interactions of each defect type with each *group of layers*. Layers are grouped by similarity into brightfield, darkfield, light metal, and other groups. Such a grouping of layers appears sufficient to improve the accuracy of yield prediction and the overall fit. In the sequel, we will not change the introduced notation and will let the index j run through the set of observed pairs of defect types and groups of layers. Only a portion of such pairs actually occurs.

The function $g(s)$ in (1) represents the transformation of the defect size, most suitable to enter the model equation. Comparison of various functions of size showed that the *logarithmic* transformation

$$x = g(s) = \log(1+s) \tag{2}$$

provides the best fit. Hence, we will use it here and below.

First, consider an idealized situation where all layers are inspected on all wafers, and all the detected defects are classified. Then, assuming independence of effects, as in [22] and [23], and including the effect of other causes, the survival probability for chip i of lot m is

$$\phi_i = e^{-b(m)} \prod_{l=1}^{L} \prod_{j=1}^{J} \prod_{k \in C_{ijl}} e^{-r(j)a(l)x_k}.$$

Further, assume independence of ξ_i which only means that each chip failure is caused by its own defects or other causes, but not by the condition of other chips. Then, one immediately constructs the binomial-type likelihood

$$\mathcal{L}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{r}) = \prod_{m=1}^{M} \prod_{w \in m} \prod_{i \in w} \phi_i^{\xi_i} (1-\phi_i)^{1-\xi_i}, \tag{3}$$

and, for example, maximizes it with respect to unknown parameters $\boldsymbol{r} = (r(1), \ldots, r(J))$, $\boldsymbol{a} = (a(1), \ldots, a(L))$, and $\boldsymbol{b} = (b(1), \ldots, b(M))$. Although (3) in fact represents only a portion, or a conditional likelihood function, given the number of defects of each type on each layer, its omitted terms do not contain the parameters of interest, hence, they drop as constant coefficients.

However, the situation is different due to a large number of **unclassified defects**. Only the size s and location (layer or operation) l are known for such defects. Therefore, a chip survives an unclassified defect k with probability

$$\boldsymbol{P}\{\xi = 1 \mid x_k\} = \sum_{j=1}^{J} \boldsymbol{P}\{j_k = j \mid x_k\} \, \boldsymbol{P}\{\xi = 1 \mid j_k = j, x_k\}, \tag{4}$$

according to the formula of total probability. Essentially, the expectation is taken with respect to an unknown defect type j. In (4), the conditional survival probabilities, given $j_k = j$, are obtained from (1), and probabilities of different defect types can be computed using the Bayes rule,

$$\boldsymbol{P}\left\{j_k = j \mid x_k = x\right\} = \frac{\boldsymbol{P}\left\{j_k = j\right\}\pi(x \mid j_k = j)}{\sum\limits_{j'} \boldsymbol{P}\left\{j_k = j'\right\}\pi(x \mid j_k = j')},$$

where

$$\boldsymbol{P}(j_k = j) = \frac{\lambda(j, l_k, m_k)}{\lambda(l_k, m_k)}$$

is the corresponding proportion of defect frequencies,

$$\lambda(l_k, m_k) = \sum_j \lambda(j, l_k, m_k),$$

and

$$\pi_j(x) = \pi(x \mid j_k = j)$$

is the distribution of (transformed) defect sizes which of course differs from one defect type to another. Given a large data set of sizes of classified defects, we decided to estimate the sizes non-parametrically (we could mention that a parametric option is, in principle, also possible).

Given a large database of classified defects, we estimated the distributions π_j nonparametrically by computing histogram density estimates ([3]), however, a parametric approach is, in principle, also possible. Dependence of the size distribution on the defect type was evident.

As a result, the probability of surviving an unclassified defect is now expressed as

$$\boldsymbol{P}\{\xi = 1 \mid x_k\} = \frac{\sum\limits_j \lambda(j, l_k, m_k)\pi_j(x_k)e^{-r(j)a(l_k)x_k}}{\sum\limits_j \lambda(j, l_k, m_k)\pi_j(x_k)}. \tag{5}$$

The other source of missing information relates to the practice of selective inspection schemes that leave a large number of **uninspected layers**. Many wafers have only one inspected layer, and only a few have more than 75% of their layers inspected. At the same time, an uninspected layer may contain fatal defects that cause the chip failures and affect the final yield.

The effect of uninspected layers can also be included into the likelihood through the formula of total probability. Since all the information on such layers is hidden, expectations should be taken over the number of defects of each type as well as their sizes. A Poisson($\lambda(j,l,m)$) number of defects N_{ijlm} of type j on layer l of chip i of lot m is assumed. Then,

$$\begin{aligned}
&\boldsymbol{P}\left\{\xi = 1 \mid \text{uninspected layer } l\right\} \\
&\quad = \prod_j \boldsymbol{P}\left\{ \text{ all type } j \text{ defects on layer } l \text{ are not fatal}\right\} \\
&\quad = \prod_j \boldsymbol{E}^{N_{ijlm}} \left(\boldsymbol{P}\left\{\text{a defect of type } j \text{ is not fatal}\right\}\right)^{N_{ijlm}} \\
&\quad = \prod_j \exp\left\{-\lambda(j,l,m)\left(1-\psi_{jl}\right)\right\}, \qquad (6)
\end{aligned}$$

where

$$\psi_{jl} = \boldsymbol{E}_j^x e^{-r(j)a(l)x} = \int e^{-r(j)a(l)t} d\pi_j(t) \qquad (7)$$

is the moment generating function of size x, by defect type and layer.

Notice that (7) is the probability of surviving a defect j (of a random size) on layer l. Thus, $(1-\psi_{jl})$ represents **the kill ratio**, the probability for such a defect to kill a chip, which is the quantity of primary interest to practitioners.

As we notice, probabilities (5) and (6) of surviving unclassified defects and uninspected layers contain unknown defect frequencies $\boldsymbol{\lambda} = \{\lambda(j,l,m)\}$ that are now included into the overall likelihood as parameters. As a result, parts of the likelihood characterizing occurrence of defects of each type are no longer constant; now they contain unknown parameters. Hence, now

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{a},\boldsymbol{b},\boldsymbol{r},\boldsymbol{\lambda}) \qquad (8) \\
&\quad = \boldsymbol{P}\left\{\text{defects}\right\}\boldsymbol{P}\left\{\text{classified defects} \mid \text{defects}\right\}\boldsymbol{P}\left\{\text{ failures} \mid \text{defects}\right\} \\
&\quad = \prod_m \prod_{w\in m} \prod_{l\in L_w} \prod_{i\in w} e^{-\lambda(l,m)} \frac{\lambda^{N_{il}}(l,m)}{N_{il}!} \prod_j \left(\frac{\lambda(j,l,m)}{\lambda(l,m)}\right)^{d_{ijl}} \phi_i^{\xi_i}(1-\phi_i)^{1-\xi_i},
\end{aligned}$$

where

$$\log\phi_i = -b(m) - \sum_{l\in L_w}\sum_{k\in C_{ijl}} r(j_k)a(l)x_k$$

$$+\sum_{l\in L_w}\sum_{k\in U_{il}}\log\frac{\sum_j \lambda(j,l)\pi_j(x_k)e^{-r(j)a(l)x_k}}{\sum_j \lambda(j,l)\pi_j(x_k)} - \sum_{l\notin L_w}\sum_j \lambda(j,l)\left(1-\psi_{jl}\right).(9)$$

This survival probability consists of four parts representing four sources of failing chips. It reflects the fact that in order to function, a chip needs to survive other causes (the first term), all classified defects on it (second term), all unclassified defects (the third term), and all the uninspected layers (fourth term). Then, equation (9) represents a *cause-and-effect relation* between defects and chip failures.

3 Parameter estimation, goodness of fit, and diagnostics

This section proposes the parameter estimation and model adequacy evaluation routines. In practice, one would apply this scheme to most recent training data sets, first, to update the parameter estimates that are used for effect comparison and yield prediction, and second, to test whether the chosen model continues to be adequate for the current production.

3.1 Modified EM algorithm

Given the explicit form of the likelihood function (8), it is the first impression that maximum likelihood estimation is natural and straightforward. The problem is in a very large dimension of the parameter space. Indeed, besides tens of defect type effects $r(j)$, tens of layer effects $a(l)$, and hundreds of lot-specific other causes effects $b(m)$, one has to estimate the defect frequencies $\lambda(j,l,m)$ for $j=1,\ldots,J$, $l=1,\ldots,L$, $m=1,\ldots,M$. Because of the latter, the total number of parameters often approaches 100,000, immediately making the "brute-force" optimization of the likelihood computationally infeasible.

On a side note, let us mention that estimation of defect frequencies is the problem of its own keen interest. Reduction of the frequencies $\lambda(j,l)$ of the most dangerous defects is a viable yield increasing strategy. On the other hand, a strategy directed towards elimination of the most significant defects may not be yield efficient if these defects have negligibly low frequencies. For this reason, in addition to kill ratios, practitioners often consider *weighted defect densities*

$$\lambda(j,l)\boldsymbol{P}\{\text{ defect of type } j \text{ on layer } l \text{ is fatal }\}$$

that evaluate effects of defect types taking into account both their probabilities to kill and their frequencies.

The *EM algorithm* ([9], [25]) offers an iterative computational method that converges to the maximum likelihood estimator (see [2]). It allows to split the set of parameters into two groups and estimate each group separately during each iteration, by an *M-step* and an *E-step*, conditionally on the other group. Each M-step represents usual maximum likelihood estimation and involves a moderate group of parameters that can be handled by the chosen optimization routine. During each E-step, the remaining parameters are treated as missing values. Being such, they are estimated by conditional expectations given their old values, obtained from the previous cycle, and the refined first group of parameters. A large number of parameters can be re-estimated by means of the E-step. In view of this, a natural split for model (8) is to estimate effects of defect types, layers, and other causes during the M-step, and to estimate the defect frequencies during the E-step.

3.1.1 Initialization of parameter estimates

A meaningful initial point in the iterative numerical routine may accelerate the entire scheme and prevent it from converging to local extrema. Here we propose simple choices for the initial values of parameter estimates, $\boldsymbol{a}^{(0)}, \boldsymbol{b}^{(0)}, \boldsymbol{r}^{(0)}$, and $\boldsymbol{\lambda}^{(0)}$.

Initially, it is natural to set intensities of defects of different types to be proportional to the corresponding numbers of classified defects, i.e.,

$$d_{jlm}/d_{lm} = \lambda^{(0)}_{jlm}/\lambda^{(0)}_{lm}.$$

Then, the cumulative frequency of all defects on each inspected layer l can be estimated as

$$\lambda^{(0)}_{lm} = \frac{d_{lm} + u_{lm}}{|\{i \in m : l \in L_i|\}},$$

and thus, all the frequencies are initialized as

$$\lambda^{(0)}_{jlm} = \left(\frac{d_{jlm}}{d_{lm}}\right) \lambda^{(0)}_{lm}.$$

Next, without any additional information at the initial step, suppose that $r(j) \equiv r$, $b(m) \equiv b$, and $a(l) \equiv 1$ (we notice that a_l are multipliers in model (1), hence they are determined only up to a constant coefficient). Replacing, for a rough approximation, transformed defect sizes x_k by their average $\bar{x}$, we obtain from (9) that

$$\sum_i \phi_i \dot{\approx} Ie^{-b} \prod_{j,l} \exp\left\{-r(j)a(l)\bar{x}\bar{\lambda}\right\} = I \exp\left\{-b - JLr\bar{x}\bar{\lambda}\right\}.$$

Equating, by the method of moments, the expected and the actual yield, $\sum \phi_i$ and $\sum \xi_i$, one obtains,

$$r^{(0)} = -\frac{\log(\sum \xi_i/I) + b^{(0)}}{JL\bar{x}\bar{\lambda}^{(0)}} = -M\frac{\log(\sum \xi_i/I) + b^{(0)}}{\bar{x}\sum\sum\sum \lambda^{(0)}(j,l,m)}, \tag{10}$$

an equation connecting the initial choice of the averaged effect of a defect, and the averaged effect of other causes, where M is the number of lots, and I is the number of chips.

3.1.2 M-step

Available optimization routines can handle the optimization problem arising during the M-step. Certainly, one can equivalently maximize the log-likelihood function, where only a part containing $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{r}$ needs to be maximized. Thus, the problem is to maximize

$$\sum_i \left\{\xi_i \log \phi_i + (1-\xi_i)\log(1-\phi_i)\right\}, \tag{11}$$

with $\log \phi_i$ given in (9). The speed and accuracy of the algorithm depends on the chosen optimization routine and convergence criteria. Also, the following two remarks allow to reduce the number of elementary operations significantly.

Remark 1. Since only a few layers are inspected on each wafer, there is a large number, often a vast majority, of chips without a single detected defect. Within each wafer, such chips share the same value of ϕ_i. Thus the corresponding terms of (11) can be computed only once for each wafer (but distinguish between good chips with no defects and bad chips with no defects).

Remark 2. It is easy to compute and supply the analytic gradient of (11); for details, see Section 5.1. Thus, it is recommended to include it into the routine instead of forcing its estimation by finite differences. Supplying the Hessian would be efficient too although it is cumbersome.

Even with these recommendations, the M-step is the most computer intensive in the entire scheme.

3.1.3 E-step

An *unbiased* estimator of defect frequencies $\lambda(j,l,m)$ is

$$\hat{\lambda}(j,l,m) = I_m^{-1}\boldsymbol{E}\left\{N_{jlm} \mid \boldsymbol{d},\boldsymbol{u},\boldsymbol{x},\boldsymbol{\xi},\boldsymbol{a},\boldsymbol{b},\boldsymbol{r},\boldsymbol{\lambda}\right\}, \tag{12}$$

where I_m is the total number of chips on lot m, and vectors $\boldsymbol{d}$, $\boldsymbol{u}$, $\boldsymbol{x}$, $\boldsymbol{\xi}$, $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{r}$, and $\boldsymbol{\lambda}$ represent, respectively, the number of classified and unclassified defects, their sizes, the quality of chips, and the current refined values of all estimated parameters. Frequencies can be re-estimated for each lot separately, thus we will omit the lot index m. As a result, we obtain the refining equation (for details, see Section 5.2)

$$\begin{aligned}\hat{\lambda}(j,l) &= I^{-1}\Bigg\{d_{jl} + \sum_{i:l\in L_i}\left(\frac{1-\xi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{v_{jk}}{v_k} + \frac{\xi_i-\phi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{w_{jk}}{w_k}\right) \\ &\qquad + \lambda(j,l)\sum_{i:l\notin L_i}\left(\frac{1-\xi_i}{1-\phi_i} + \frac{\xi_i-\phi_i}{1-\phi_i}\psi_{jl}\right)\Bigg\}, \end{aligned} \tag{13}$$

where

$$v_{j,k} = \lambda(j,l_k)\pi_j(x_k), \quad v_k = \sum_j v_{jk}, \tag{14}$$

and

$$w_{j,k} = v_{j,k}e^{-r(j)a(l)x_k} = \lambda(j,l_k)\pi_j(x_k)e^{-r(j)a(l)x_k}, \quad w_k = \sum_j w_{jk}. \tag{15}$$

During the E-step, each frequency is recomputed once, and no iterations are involved. Therefore, the E-step is much faster and computationally cheaper than the M-step, where a numerical optimization routine is used to maximize the likelihood under fixed $\boldsymbol{\lambda}$.

3.1.4 Modification

The EM algorithm posesses a number of appealing properties ([9], [25]), however, in a wide range of practical problems (specifically, those dealing with a large number of parameters and large data sets) its performance can typically be improved via suitable modifications. The problem described in this paper is not an exception.

Under the conditions of our problem, it is beneficial to terminate the M-step under rather mild convergence criterion, since high-precision optimization is inefficient in intermediate stages.

We also introduced an additional step that enables one to achieve a sizeable improvement in the speed of convergence. This step essentially tries to guess the correct search direction for the maximum of the likelihood function $\mathcal{L}(\boldsymbol{a},\boldsymbol{b},\boldsymbol{r},\boldsymbol{\lambda})$. When it succeeds, it starts making increasingly larger steps in that direction preventing the routine from too many iterations in the area where the likelihood is increasing slowly. Otherwise, it is skipped at this cycle, and the standard EM algorithm is followed.

The step is described as follows. It starts by analyzing results of the latest E-step and M-step. Let $\boldsymbol{\theta}_0$ be the vector of parameter estimates $(\hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{r}}, \hat{\boldsymbol{\lambda}})$ obtained as a result of the previous cycle, and $\boldsymbol{\theta}_1$ be the refined vector. That is, the latest E-step and M-step transformed $\boldsymbol{\theta}_0$ into $\boldsymbol{\theta}_1$. If the chosen global convergence criterion is met, then the last cycle failed to improve the value of $\mathcal{L}(\boldsymbol{\theta})$ by more than ε, and the entire routine stops. In all other cases, we obtain that $\mathcal{L}(\boldsymbol{\theta}_1) > \mathcal{L}(\boldsymbol{\theta}_0) + \varepsilon$, hence, the likelihood function is seemingly increasing in the direction of

$$\Delta\boldsymbol{\theta} = \boldsymbol{\theta}_1 - \boldsymbol{\theta}_0.$$

We now follow this direction and check if the likelihood continues to increase. Also, each time we increase the step, therefore, shaking the system and disallowing it to converge to a local extremum. That is, we consider a sequence of vectors $\{\boldsymbol{\theta}_n\}$ defined recursively as

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \gamma(n)\Delta\boldsymbol{\theta}, \quad n \geq 2,$$

where $\gamma(n)$ is a chosen increasing function of n (polynomial or even exponential). The value of $\mathcal{L}(\boldsymbol{\theta}_n)$ is calculated for each $n = 2, 3, ...$, the algorithm proceeds if it this value is improved and stops at the time

$$T = \min\left\{n \geq 2 : \ \mathcal{L}(\boldsymbol{\theta}_n) < \mathcal{L}(\boldsymbol{\theta}_{n-1})\right\}.$$

Then, $\boldsymbol{\theta}_{T-1}$, the best set of parameter estimates obtained so far, serves as an initial point of the next EM-iteration.

Clearly, this step is activated only if it leads to larger values of $\mathcal{L}(\boldsymbol{\theta})$. Otherwise, it is skipped, and the routine proceeds to the next EM-iteration. Thus, it will generally result in an equal or higher value of the likelihood.

And above all, it is computationally the cheapest of all three steps, requiring only computation of the likelihood function, but no optimization or gradient evaluation.

Based on our experience, insertion of this step into the EM algorithm always resulted in the same or, even more often, higher value of the maximum likelihood. It always accelerated the EM algorithm in the beginning by making aggressive steps and saving a considerable number of EM-iterations. During the late iterations, it rarely went farther than $\boldsymbol{\theta}_2$.

3.2 Assessment of the goodness of fit

Repeatedly applying the described three steps until the convergence criterion is met (which is inevitable because each cycle improves the likelihood by at least ε), we obtain the set of parameter estimates $(\hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{r}}, \hat{\boldsymbol{\lambda}})$. How good are these estimates, and how adequate is our model? From a practical standpoint, how useful is it for yield prediction, evaluation of kill ratios, designing new strategies, and other objectives? We propose two general goodness-of-fit assessment tools.

1. Standard goodness-of-fit tests (e.g., the chi-square test) compare the expected and observed values. In our case, it is a comparison of

$$\hat{Y}_m = \text{actual yield} = \sum_{i \in m} \xi_i$$

and

$$Y_m = \text{predicted yield} = \sum_{i \in m} \boldsymbol{E}_{\phi_i}(\xi_i) = \phi_i$$

for each lot or each wafer. One can evaluate the closeness of $\hat{Y}_m$ to Y_m by the standard chi-square statistic, or even by the correlation coefficient between Y and $\hat{Y}$. Along with the graph of actual and predicted *proportional yield* $(y_m, \hat{y}_m) = I_m^{-1}(Y_m, \hat{Y}_m)$ for $m = 1, \ldots, M$, it provides a simple illustration of the predictive power.

2. Even if the actual and predicted yields above are found close to each other, it may happen that the right yield was predicted by wrong reasons. Say, one could (at least, theoretically) predict a high yield on actually failed chips and a low yield on good functioning chips that in combination returned a prediction close to Y.

Therefore, it is elucidating to compute predicted yield separately for good chips and for bad chips. The actual numbers of good and bad chips are different, therefore, the only fair comparison is based on *proportional yields*

$$\hat{y}_g = \hat{\boldsymbol{P}}\{\text{ predicted good } \mid \text{ actually good }\} = \frac{\sum_i \phi_i \xi_i}{\sum_i \xi_i}$$

and

$$\hat{y}_b = \hat{\boldsymbol{P}}\{\text{ predicted good } \mid \text{ actually bad }\} = \frac{\sum_i \phi_i (1-\xi_i)}{\sum_i (1-\xi_i)}.$$

The model obtains predictions $\hat{y}_g$ and $\hat{y}_b$ from the observed defects only, without seeing the actual failures. Ideally, we would certainly wish to predict a 100% yield on good chips and a 0% yield on bad chips. However, this is not possible. One reason for that is existence of pairs of chips with an identical defect situation, whereas one chip in a pair is good and the other is bad. Moreover, there are failed chips that are exposed to other causes only and chips that survived not only the other causes but also many defects on different layers.

Then, how well can the model separate $\hat{y}_g$ and $\hat{y}_b$, and what difference between them should be considered satisfactory, or a good fit? The following theoretical result answers this questions.

Lemma 1 *Let $\{\phi_i,\ i = 1, \ldots, I\}$ be independent identically distributed random variables with the distribution $F(\phi)$. For each i, consider ξ_i, a Bernoulli variable with parameter ϕ_i, and let ξ_i be independent.*

Let $\hat{y}_g = \dfrac{\sum_i \phi_i \xi_i}{\sum_i \xi_i}$ and $\hat{y}_b = \dfrac{\sum_i \phi_i (1-\xi_i)}{\sum_i (1-\xi_i)}$.

Then the strong law of large numbers holds for $\hat{y}_g$ and $\hat{y}_b$, and

$$\lim_{I\to\infty} \hat{y}_g = \frac{\boldsymbol{E}_F \phi^2}{\boldsymbol{E}_F \phi} \qquad \textit{and} \qquad \lim_{I\to\infty} \hat{y}_b = \frac{\boldsymbol{E}_F \phi - \boldsymbol{E}_F \phi^2}{1 - \boldsymbol{E}^\pi \phi},$$

with probability 1, where $\boldsymbol{E}_F$ represents the expectation over the distribution F. Also,

$$\lim_{I\to\infty} (\hat{y}_g - \hat{y}_b) = \frac{Var_F(\phi)}{\boldsymbol{E}_F \phi (1 - \boldsymbol{E}_F \phi)}, \ \ a.s. \tag{16}$$

Applying this lemma to defects and failures, we define ϕ_i as the probability that chip i is good, given its defects, and $\xi_i \sim$ Bernoulli(ϕ_i) as the binary variable that equals 1 if the chip is good. For each i, the value of ϕ_i is a function of the number, types, locations, and sizes of defects occurring on chip i, as in (9). In turn, all these factors are random variables that collectively determine the distribution F of ϕ_i.

Lemma 1 explains the limitations in the desirable separation of predicted yield among good and among bad chips. Under no circumstances can we achieve $\hat{y}_g \approx 100\%$ and $\hat{y}_b \approx 0\%$. As an extreme situation, suppose that all the defects are getting eliminated by constant modification and improvement of the manufacturing line. Then $\phi \uparrow 1$, and both $\hat{y}_g$ and $\hat{y}_b$ approach 1. Conversely, if $\phi \downarrow 0$ then both $\hat{y}_g$ and $\hat{y}_b$ approach 0, and in both cases the difference between them vanishes.

The proof of Lemma 1 is given in Appendix, Section 5.3.

3.3 Clustering and rare defects

In practical implementations it is important to take into account a number of special properties of defects that require adjustments of the modeling and estimation procedures. In this section we discuss two properties of this type: a tendency of defects to cluster and defect rarity.

1. Clustering. Consider a chip with 8,000 detected defects. Under any plausible model, including three models mentioned in Section 2, the probability for such a chip to fail is practically 1. However, it is not unusual to see such a chip that is recorded as *good*! Proportion of such chips is low, but keeping them in the overall likelihood without any correction has a strong influence on final results. The model tries to explain their yield, and the only way this yield can be positive is when each defect type seen on a chip so many times has zero effect.

Investigating situations involving thousands to hundreds of defects observed on the same chip, we found that the vast majority of these defects occur on the same layer and belong to the same defect type or remain unclassified. Also, in all such cases, these defects were marked as *clustered.* Survival of chips containing such clusters indicates that *the effect of a cluster of defects is weaker than the mutual effect of the same number of individual defects.* Clusters of different sizes, from just a few defects to thousands of defects, are registered on a large portion of chips, thus such chips cannot be ignored or deleted.

There are different ways the effect of a cluster can be modeled. A plausible way is to treat a cluster as a single defect whose size (before the transformation (2) is applied) equals the sum of their individual sizes. In all the cases such modeling provided better fit comparing with the scheme that treats chips with unusually high number of defects as outliers, deletes them, and applies the uncorrected model to the remaining chips.

2. Rare defects. A considerable number of defect types are seen rather rarely, say, 1-5 times per 10,000 chips. Even if their effect on the quality of a chip is strong, they do not affect the remaining vast majority of chips, and therefore, the final yield is not affected by such "rare" defect types.

Since for some chips rare defects appear to be the cause of failure, they cannot be simply ignored. Other defects would then be classified as chip killers, introducing a bias in parameter estimates. In view of their small effect on the final yield, all such defect types can be combined, so that only their average effect and their cumulative frequency is estimated. This reduces the overall number of estimated parameters, leading not only to acceleration of the numerical routine, but also to a higher predictive power of the model.

3.4 Diagnostics and outlier detection

Besides situations described in the previous section, large data sets may contain chips, wafers, or even lots, that "do not belong there" and should be treated as outliers. Several outlier detection methods are proposed in this section.

Typically, a certain portion of data can be deleted from the study immediately by means of a simple inspection. This includes wafers with approximately zero yield, chips with hundreds of non-clustered defects, inspected layers with no detected defects on an entire lot (paradoxically!), etc. Such cases are usually results of various errors in data records.

After this inevitable data cleaning, we identify "suspicious" wafers and lots by answering the following two questions:

1. How different would the results be if obtained from this wafer (lot) only?
2. How different would the results be if obtained *without* this wafer (lot)?

To address these questions, we use four types of diagnostics.

1. Likelihood-based diagnostics. The general model (8) is estimated separately for each lot m, providing its own maximum value $\mathcal{L}_m(\boldsymbol{\theta}^{(m)})$ of the likelihood of this lot. It is always higher than the value of $\mathcal{L}_m(\boldsymbol{\theta})$ based on the global estimators $\boldsymbol{\theta}$ obtained by means of the modified EM algorithm described above. If the difference is significant, it means that the global estimators do not fit to lot m; its likelihood can be increased significantly given its own parameters.

To find a fair measure of significance of $\left\{\log \mathcal{L}_m(\boldsymbol{\theta}^{(m)}) - \log \mathcal{L}_m(\boldsymbol{\theta})\right\}$, we notice that letting each lot have its own parameters increases the total number of parameters in the model from $(M+L+J+JLM)$ to $(M+LM+JM+JLM)$, where M is the number of lots, J is the number of defect types, and L is the number of layers. This constitutes $(J+L)(M-1)$ additional parameters. According to [26], the test statistic

$$2\sum_{m=1}^{M} \log \mathcal{L}_m(\boldsymbol{\theta}_m) - 2\log \mathcal{L}(\boldsymbol{\theta}) \tag{17}$$

has asymptotically χ^2 distribution with $(J+L)(M-1)$ degrees of freedom if expanding the parameter space is not significant. Hence, if a lot is not an outlier, and the global parameters fit it well, one should expect its two-log-likelihood to increase approximately by a χ^2 variable with $(J+L)(1-1/M)$ degrees of freedom. Exceeding the critical value of χ^2_α automatically puts a lot into the list of suspicious ones.

Further, the test statistic (17) consists of the sum of differences, by wafer, and thus, the largest summand points to the most outlying wafer that is responsible for the large difference.

Remark. Even for large data sets, it is typically feasible to conduct separate estimation for each lot. In the process of such estimation, the global parameter estimates $\boldsymbol{\theta}$ are very helpful, because because they can serve as initial values in estimation for each lot. Notice that the purpose of this analysis is to find significant deviations from the global results. If a lot is well explained by the global model, such a deviation is small, and starting from $\boldsymbol{\theta}$, the algorithm will converge quickly.

2. Parameter-based diagnostics. Continuing the likelihood-based diagnostics, one can compare the lot-specific estimates $\boldsymbol{\theta}_m$ with the global estimates $\boldsymbol{\theta}$. Under the null hypothesis, or in the absence of outliers, vectors $\boldsymbol{\theta}_m$, obtained from similar lots with the same number of inspected layers, are i.i.d., and their mean vector and covariance matrix can be estimated by standard methods. This provides a null multivariate normal distribution (according to the asymptotic normality of maximum likelihood estimators), against which the differences $(\boldsymbol{\theta}_m - \boldsymbol{\theta})$ can be compared.

3. Prediction-based diagnostics.

A different way to identify lots that do not agree with the global model is to analyze the yield predictions for each lot. A good model should separate the predicted yield on good and on bad chips $\hat{y}_g$ and $\hat{y}_b$, as shown in Lemma 1. According to (16), the difference between $\hat{y}_g$ and $\hat{y}_b$ should be positive, provided a good fit. Otherwise, the model appears not to have a good predictive power on such a lot. Then it should be deleted from the estimation routine, as long as the parameter estimates are used for prediction on future lots. Typically, however, most of these lots appear to be already deleted by the two diagnostics tools described above.

4. Cross-validation.

In cross-validation, each lot is deleted, one at a time, and the model parameters are estimated without it. Again, the global parameter estimates $\boldsymbol{\theta}$ can be used as initial values for the EM-algorithm. Then, predictions are made on the deleted lot and compared with its actual yield.

All the proposed methods can be applied to either lots or wafers. The latter is recommended for wafers with sufficiently many inspected layers. We have seen a number of cases where an outlying lot was classified as such only due to one outlying wafer on it. Applying the diagnostics tools on a wafer level would in general delete fewer units. On the other hand, wafers with only one or two layers inspected may provide a rather small sample of defects. Inference made on such a wafer separately from other wafers in diagnostics tools 1–3 will not be reliable.

After the "suspicious" lots (wafers) are identified, it is very useful and strongly recommended to conduct an in-depth analysis of each case. In our application, it resulted in a number of "discoveries". Almost every critical case was attributed to a special cause, such as error in data collection and data recording, spontaneous change in the sensitivity of the inspection instrument, scrapped wafer or reworked layer.

4 Yield forecasting, kill ratios, and other applications

The described cycle of data cleaning, model fitting, model diagnostics, outlier detection, and most likely, model refitting and refinement results in a estimated model and a set of parameter estimates. This section concerns immediate applications of this analysis, usable information and interpretation that can be drawn from it.

1. Yield forecasting. One of the obvious practical by-products of our modeling is the possibility to predict the yield for each lot and each wafer. Indeed,

$$\phi_i = \boldsymbol{E}\left\{\xi_i \mid \text{defects on chip } i)\right\}$$

is the expected yield, or number of good chips, out of one chip i. Then

$$\hat{Y}_{\mathcal{I}} = \sum_{i \in \mathcal{I}} \phi_i \tag{18}$$

is the expected yield that can be computed for any set of chips $\mathcal{I}$, which may be a wafer, a lot, a number of lots, or an entire grade. Thus $\hat{Y}_{\mathcal{I}}$ serves as the yield forecast, and it is based on the observed defects on $\mathcal{I}$ and the parameter estimates obtained from the training data.

Using this method, the yield can be predicted at the end of the production cycle, after all layers have been processed but before the final testing. It can also be used to predict the yield at earlier stages of the production line. Layers that have not been processed are then treated as uninspected. If a low yield is predicted on some wafer or lot, a decision may be made about terminating its production at an early stage, or a layer at fault can be reworked.

2. Kill ratios and weighted defect densities. Next, we will derive the kill ratios and related probabilities, such as the probability for a certain defect to be fatal, the probability for a certain defect type or layer to contain the chip killer, etc. These probabilities are computed for a generic typical chip, and therefore, the chip index i will be omitted. For the sake of simplicity, we will also omit the conventional "hat" and keep in mind that all the used parameters are in practice replaced by the computed estimates.

Let D_{jl} be the total number of (classified and unclassified, including unobserved ones) defects of type j occurring on a layer l of some chip. By $D_j = \cup_l D_{jl}$, $D_l = \cup_j D_{jl}$, and $D = \cup_j D_j$ we will denote the total number of type j defects on a chip, the total number of defects of all types on layer l, and the overall number of defects on that chip, respectively.

Also, let A_k be the event that a defect k is fatal, and let $A^{(0)}$ be the event that the chip is killed by other causes. Then, for example, $\cup \{A_k, k \in D_l\}$ is the event of at least one fatal defect occurring on layer l, and $\cup \{A_k, k \in D\} \setminus A^{(0)}$ is the event that a chip is killed by observable defects but not by other causes. Notice that a chip may contain more than one fatal defect, and thus, the kill ratios considered here cannot be treated as probabilities of a failure *due to defect k only*. The latter probabilities are derived further below.

We start by computing the *kill ratio*, or the probability that a type j defect on layer l is fatal,

$$p_{jl}^{KR} = \boldsymbol{P}\left\{A_k \mid k \in D_{jl}\right\} = 1 - \psi_{jl} = 1 - \boldsymbol{E}_j^x \exp\left\{-r(j)a(l)x\right\}. \qquad (19)$$

The expectation in (19) can be estimated if a parametric model is assumed for the defect sizes x for each defect type j. For an alternative nonparametric method of estimating ψ_{jl}, see (39) in Section 5.1.

Further, one computes the probability for a defect of type j to be fatal,

$$p_j^{KR} = \boldsymbol{P}\left\{A_k \mid k \in D_j\right\} = \sum_l \frac{\lambda(j,l)}{\lambda(j)} p_{jl}^{KR}, \tag{20}$$

by the formula of total probability. This is a *proportion of type j defects that appear fatal* for a chip. Analysis of large data sets showed that estimator (20) is more accurate than the kill ratios computed from defect counts only ([12], [15]). Similarly, one computes the probability for a defect on layer l to be a chip killer,

$$p_l^{KR} = \boldsymbol{P}\left\{A_k \mid k \in D_l\right\} = \sum_j \frac{\lambda(j,l)}{\lambda(l)} p_{jl}^{KR}, \tag{21}$$

which is a *proportion of fatal defects on layer l*. Finally,

$$p^{KR} = \boldsymbol{P}\left\{A_k \mid k \in D\right\} = \sum_j \sum_l \frac{\lambda(j,l)}{\lambda} p_{jl}^{KR} \tag{22}$$

is *the overall proportion of fatal defects.* Generalizations used in p_l^{KR}, p_j^{KR}, and p^{KR}, are helpful to compare kill ratios of across defect types and across layers.

Expressions (20)–(22) are based on the products $\lambda(j,l)p_{jl}^{KR}$ that are called *weighted defect densities* by practitioners. Combining the probability to kill with the defect frequency, these quantities are often used to measure the adverse effect of each group of defects on the final yield.

The next four probabilities, or *group kill ratios*, describe the kill ratios of certain groups of defects. We consider grouping by defect type and by layer, but other interesting groups can be considered similarly, for instance, a group of large type j defects, all defects on the lowest 5 layers, etc.

As above, we assume Poisson distribution of the number of defects $N_{jl} = |D_{jl}|$. The probability of *at least one chip killer among all type j defects on layer l* is

$$p_{jl}^{GKR} \;=\; \boldsymbol{P}\left\{\cup A_k,\ k \in D_{jl}\right\} = 1 - \boldsymbol{E}^{N_{jl}} \boldsymbol{P}\left\{\cap \bar{A}_k,\ k = 1, \ldots, N_{jl}\right\}$$

$$= \quad 1 - \boldsymbol{E}^{N_{jl}} \psi_{jl}^{N_{jl}} = 1 - \exp\left\{-\lambda(j,l)(1-\psi_{jl})\right\}. \tag{23}$$

Using this equation, the probability of *at least one chip killer among all type j defects* is

$$p_j^{GKR} = \boldsymbol{P}\left\{\cup A_k,\ k \in D_j\right\} = \boldsymbol{P}\{\bigcup_l \bigcup_{k \in D_{jl}} A_k\} = 1 - \prod_l (1 - p_{jl}^{GKR}), \tag{24}$$

the probability of *at least one fatal defect on layer l* is

$$p_l^{GKR} = \boldsymbol{P}\left\{\cup A_k,\ k \in D_l\right\} = \boldsymbol{P}\{\bigcup_j \bigcup_{k \in D_{jl}} A_k\} = 1 - \prod_j \left(1 - p_{jl}^{GKR}\right), \tag{25}$$

and the probability of *at least one fatal defect* on a chip is

$$p^{GKR} = \boldsymbol{P}\left\{\cup A_k,\ k \in D\right\} = 1 - \prod_l \prod_j \left(1 - p_{jl}^{GKR}\right). \tag{26}$$

The latter is also the probability for a chip to fail not (or not only) because of other causes.

3. Causes of failures.

In this section, we consider chips that are known to have failed. Can the model pinpoint the cause of their failure?

We start by computing the probability for a randomly selected chip to survive all its defects and other causes,

$$p_1 = \boldsymbol{P}\left\{\xi = 1\right\} = \boldsymbol{P}\{\bar{A}^{(0)} \bigcap (\bigcap_{k \in D} \bar{A}_k)\} = e^{-b}\left(1 - p^{GKR}\right).$$

This probability is different from our predicted yield $\hat{Y}$ that is computed for the given dataset based on its observed defects. In comparison, p_1 estimates the proportional yield for the entire mode of production (grade). Similarly, $p_0 = 1 - p_1$ estimates the overall proportion of failed chips.

Then, *the probability for a failed chip to be killed by a type j defect on layer l* is

$$p_{jl}^{CF} = \boldsymbol{P}\{\bigcup_{k \in D_{jl}} A_k \mid \xi = 0\} = \frac{\boldsymbol{P}\left\{\cup A_k, k \in D_{jl}\right\}}{p_0} = \frac{p_{jl}^{GKR}}{1 - e^{-b}\left(1 - p^{GKR}\right)}, \tag{27}$$

and similarly, *the probability for a failed chip to be killed by a type j defect* is

$$p_j^{CF} = \frac{p_j^{GKR}}{1 - e^{-b}\left(1 - p^{GKR}\right)}, \tag{28}$$

the probability for a failed chip to be killed by some defect on layer l is

$$p_l^{CF} = \frac{p_l^{GKR}}{1 - e^{-b}\left(1 - p^{GKR}\right)}, \tag{29}$$

and *the probability for a failed chip to be killed by some defect* (and not, or not only, by other causes) is

$$p^{CF} = \frac{p^{GKR}}{1 - e^{-b}\left(1 - p^{GKR}\right)}. \tag{30}$$

The last probability also represents the proportion of failed chips that contain fatal observable defects, and $(1 - p^{CF})$ is *the proportion of failed chips that are killed by other causes.*

4. Single cause of failure. Based on the probabilities computed above, should defects with the highest chance of being fatal be regarded to as most dangerous, and should the reduction of such defects be given the highest priority?

Noticeably, even having such a defect on some layer, a failed chip may have been killed by other defects. In view of this, for example, it would be wrong to attribute the proportion of p_j^{CF} of failed chips to defects of type j only. And if so, then what proportion of failed chips is due to defects of type j?

Here we compute probabilities for a group of defects on a failed chip *and nothing else* to cause its failure. Thus, such probabilities are also proportions of failed chips that can be attributed to the considered defects *only*, making such a group a *single cause of failure.*

The probability that a chip failed due to defects of type j on layer l *only* is

$$p_{jl}^{SCF} \;=\; \boldsymbol{P}\left\{\left(\bigcup_{D_{jl}} A_k\right) \bigcap \left(\bigcap_{(j',l') \neq (j,l)} \bigcap_{D_{j'l'}} \bar{A}_k\right) \bigcap \bar{A}^{(0)} \mid \xi = 0\right\}$$

$$
\begin{aligned}
&= p_0^{-1} p_{jl}^{GKR} e^{-b} \prod_{(j',l')\neq(j,l)} \left(1 - p_{j'l'}^{GKR}\right) \\
&= \left(\frac{p_1}{p_0}\right)\left(\frac{p_{jl}^{GKR}}{1 - p_{jl}^{GKR}}\right) = \frac{p_{jl}^{CF} - p_{jl}^{GKR}}{1 - p_{jl}^{GKR}}. \qquad (31)
\end{aligned}
$$

Similarly, we compute the probability that *all* fatal defects on a failed chip are of type j,

$$p_j^{SCF} = \left(\frac{p_1}{p_0}\right)\left(\frac{p_j^{GKR}}{1 - p_j^{GKR}}\right) = \frac{p_j^{CF} - p_j^{GKR}}{1 - p_j^{GKR}}, \qquad (32)$$

the probability that the chip failed due to defects on layer l *only*,

$$p_l^{SCF} = \left(\frac{p_1}{p_0}\right)\left(\frac{p_l^{GKR}}{1 - p_l^{GKR}}\right) = \frac{p_l^{CF} - p_l^{GKR}}{1 - p_l^{GKR}}, \qquad (33)$$

and the probability that *a failed chip was killed by defects but not by other causes*,

$$p^{SCF} = e^{-b} p^{CF}. \qquad (34)$$

Notice that the sum of probabilities in each equation (31)–(33) is less than 1. Each equation deals with probabilities of mutually exclusive but not exhuastive events because several types of defects or several layers may be at fault for the chip failure.

5. Influential layers. If an unusually low or unusually high yield is predicted for some lot, a simple method can be proposed to find the main reason of the unusual prediction. Typically, the reason for an unusual prediction is an unusual situation that occurred on some inspected layer (of course, no surprise can be found on uninspected layers). It is then straightforward to evaluate the contribution of each layer into the prediction.

Moving sequentially through all the inspected layers on a lot or a wafer, consider one inspected layer at a time. Recompute predicted yield $\hat{y}_m$ with this layer being hidden, or uninspected. This operation does not require additional computer code or much of the computer time. As a result, we obtain the difference in predicted yield which shows how much of the yield is lost or gained due to the defects observed on this layer.

Not only this measures the influence of each inspected layer on the final yield, but also it points to practitioners a good direction of their possible yield improving effort.

6. Sensitivity analysis. When choosing an efficient yield improving strategy, one would be interested to predict, what changes in the final yield such a strategy will bring. For example, if one manages to reduce the number of type j defects by 10%, how strongly will this affect the yield? Is it worth the effort, and is there a more efficient strategy? In other words, how sensitive is the yield to certain changes in frequencies or sizes of defects?

To answer these questions for relatively small percentage reduction in frequencies and/or sizes of defects, we compute the corresponding *elasticities*, or derivatives of logarithms. Since $(\log f)' = df/f$ is the proportional change of a function f, each computed elasticity will show the proportional yield change caused by the given proportional change in the parameters.

We recall that the proportional yield is computed as

$$y = \boldsymbol{P}\{\xi = 1\} = \exp\left\{-b - \sum_j \sum_l \lambda_{jl}(1 - \psi_{jl})\right\}, \tag{35}$$

so that

$$\frac{\partial \log y}{\partial \lambda_{jl}} = -(1 - \psi_{jl}),$$

Hence, *sensitivity of yield to the frequency of type j defects on layer l* is measured by the following elasticity,

$$e_{jl} = \frac{\partial \log y}{\partial \log \lambda_{jl}} = \frac{\partial \log y}{\partial \lambda_{jl}/\lambda_{jl}} = -\lambda_{jl}(1 - \psi_{jl}). \tag{36}$$

For example, a 5% reduction in the number of type j defects on layer l results in the yield improvement by $5e_{jl}$%.

Next, suppose that a certain modification of the production process can reduce the size s of type j defects on layer l by $100(\Delta s)$%. As a result, the transformed size $x = g(s)$ reduces by $\Delta x \approx sg'(s)\Delta s$, the corresponding kill ratio changes by $(-\Delta\psi_{jl})$, and the log-yield, according to (35), changes by

$$\begin{aligned}
\Delta \log y &= \lambda_{jl}\Delta\psi_{jl} \\
&= \lambda_{jl}\boldsymbol{E}_j^x \left(\exp\left\{-r(j)a(l)[x - sg'(s)\Delta s]\right\} - \exp\left\{-r(j)a(l)x\right\}\right) \\
&= \lambda_{jl}\boldsymbol{E}_j^x \exp\left\{-r(j)a(l)x\right\}\left(\exp\left\{r(j)a(l)sg'(s)\Delta s\right\} - 1\right) \\
&= \lambda_{jl}r(j)a(l)\Delta s\boldsymbol{E}_j^x \exp\left\{-r(j)a(l)x\right\}sg'(s) + o(\Delta s), \text{ as } \Delta s \to 0.
\end{aligned} \tag{37}$$

Hence, *the proportional change in yield due to a small reduction of sizes of type j defects on layer l* is

$$\frac{\partial \log y}{\partial s} = \lambda_{jl}r(j)a(l)\boldsymbol{E}_j^x \exp\left\{-r(j)a(l)x\right\}sg'(s).$$

For the chosen in (2) transformation $x = g(s) = \log(s+1)$ and for relatively large sizes s, it equals

$$\frac{\partial \log y}{\partial s} = \lambda_{jl}r(j)a(l)\boldsymbol{E}_j^x \left(\frac{s}{s+1}\exp\left\{-r(j)a(l)x\right\}\right) \approx \lambda_{jl}r(j)a(l)\psi_{jl}. \tag{38}$$

7. Summary. In general, one computes (18) to forecast the yield under the current conditions. Then, (19)–(34) show the most dangerous defects and defect type–layer combinations, the most probable causes of failures, and influential layers that had the highest impact on the final yield. When choosing an efficient strategy, modifying the production process and resulting in reduced numbers or reduced sizes of such defects, one uses equations (36)–(38) to predict possible outcomes. In practice, the expected gain from such a strategy will then be weighed against its expected costs, and based on this balance, a business decision regarding its implementation will be made.

5 Appendix

5.1 Gradient of the log-likelihood for the M-step

Here we derive analytic expressions for $\nabla\mathcal{L}$ that are used by the optimization routine during the M-step. It is seen from (8) and (11) that for any parameter $\theta \in \{a(1),\ldots,a(L);\ b(1),\ldots,b(M);\ r(1),\ldots,r(J)\}$,

$$\frac{\partial \log \mathcal{L}}{\partial\theta} = \sum_i \left(\frac{\xi_i - \phi_i}{1-\phi_i}\right)\frac{\partial \log\phi_i}{\partial\theta}.$$

Thus, it remains to compute the partial derivatives of ϕ_i for each chip i. One has

$$\frac{\partial \log\phi_i}{\partial b(m)} = \begin{cases} -1 & \text{if chip } i \text{ belongs to lot } m \\ 0 & \text{otherwise;} \end{cases}$$

$$\frac{\partial \log\phi_i}{\partial a(l)} = -\sum_{k\in C_{il}} r(j_k)x_k - \sum_{k\in U_{il}} x_k \sum_j r(j)w_{jk}/w_k$$

for any layer l *inspected* on the chip i (the quantities w_{jk} and w_k are defined in (15));

$$\frac{\partial \log\phi_i}{\partial a(l)} = \sum_j \lambda(j,l,m_i)\frac{\partial\psi_{jl}}{\partial a(l)}$$

for all layers l that are *not inspected* on chip i; and

$$\frac{\partial \log\phi_i}{\partial r(j)} = -\sum_{k\in C_{ij}} a(l_k)x_k - \sum_{k\in U_{il}} a(l_k)x_k w_{jk}/w_k + \sum_{l\in L_i}\lambda(j,l,m)\frac{\partial\psi_{jl}}{\partial r(j)}.$$

The last two expressions contain partial derivatives of the moment generating function (7) of the distribution of defect sizes. They will certainly depend on the model used for this distributions. However, simple *nonparametric estimators* for these partial derivatives are available, as well as for ψ_{jl} itself.

Indeed, since $\psi_{jl} = \boldsymbol{E}_j^x e^{-a(l)r(j)x}$, it has partial derivatives

$$\frac{\partial\psi_{jl}}{\partial a(l)} = -r(j)\boldsymbol{E}_j^x x e^{-a(l)r(j)x} \quad\text{and}\quad \frac{\partial\psi_{jl}}{\partial r(j)} = -a(l)\boldsymbol{E}_j^x x e^{-a(l)r(j)x}.$$

All three sets of quantities can be estimated by the method of moments from the classified defects,

$$\widehat{\psi_{jl}} = \frac{1}{d_{jl}}\sum_{k\in C_{jl}} e^{-a(l)r(j)x_k}; \tag{39}$$

$$\frac{\widehat{\partial\psi_{jl}}}{\partial a(l)} = -r(j)\frac{1}{d_{jl}}\sum_{k\in C_{jl}} x_k e^{-a(l)r(j)x_k};$$

$$\frac{\widehat{\partial\psi_{jl}}}{\partial r(j)} = -a(l)\frac{1}{d_{jl}}\sum_{k\in C_{jl}} x_k e^{-a(l)r(j)x_k}.$$

This completes the computation of the analytic gradient that is supplied to the optimization routine that maximizes the log-likelihood function with respect to $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{r}$ during the M-step.

5.2 Derivation of the E-step

In this section, we derive the rules for refinement of estimators of $\lambda(j, l, m)$ during the E-step and prove equation (13).

The expected number of (j, l)-defects in (12) consists of three parts. Namely, these are all detected defects classified to type j on layer l, a suitable portion of unclassified defects that "should" be attributed to type j, and a portion of (j, l)-defects that is expected on wafers where layer l is uninspected. That is,

$$\hat{\lambda}_{j,l} = I^{-1}\left(d_{j,l} + \sum_{k\in U_l}\boldsymbol{P}\left\{j_k = j \mid x_k, \xi_i\right\} + \sum_{i:l\notin L_i}\boldsymbol{E}\left\{N_{ijl} \mid \xi_i\right\}\right). \qquad (40)$$

We compute these three terms separately. The first term is simply the observed number of classified type j defects observed on layer l. For the second term, consider two cases, when the chip containing defect k is good and when it is bad.

Suppose for a moment that an unclassified defect k in fact has type j, and consider the conditional probability

$$\phi_i(j, k) = \boldsymbol{P}\left\{\xi = 1 \mid j_k = j, x_k\right\}. \qquad (41)$$

The only difference between $\log\phi_i(j, k)$ and $\log\phi_i$ in (9) is caused by this defect appearing in the set C_{ijl} instead of U_{il}. Hence,

$$\log\phi_i(j, k) - \log\phi_i = -r(j)a(l_k)x_k - \log\frac{\sum_{j'}\lambda(j', l)\pi_{j'}(x_k)e^{-r(j')a(l_k)x_k}}{\sum_{j'}\lambda(j', l)\pi_{j'}(x_k)},$$

so that

$$\phi_i(j,k) = e^{-r(j)a(l_k)x_k}\phi_i/\rho_k,$$

where

$$\rho_k = \frac{w_k}{v_k} = \frac{\sum_{j'} w_{j',k}}{\sum_{j'} v_{j',k}}$$

is the probability for a chip to survive an unclassified defect k, which is independent of the (in fact, unknown) defect type j, and v_{jk}, v_k, w_{jk}, w_k are defined in (14) and (15).

Then, for an *unclassified* defect $k \in U_l$ occurring on a *good* chip i,

$$\begin{aligned}
&\boldsymbol{P}\{j_k = j \mid x_k, \xi_i = 1\} \\
&= \frac{\boldsymbol{P}\{j_k = j\}\,\pi(x_k \mid j_k = j)\boldsymbol{P}\{\xi_i = 1 \mid j_k = j, x_k\}}{\sum_{j'} \boldsymbol{P}\{j_k = j'\}\,\pi(x_k \mid j_k = j')\boldsymbol{P}\{\xi_i = 1 \mid j_k = j', x_k\}} \\
&= \frac{[\lambda(j,l_k)/\lambda_{l_k}]\,\pi_j(x_k)\phi_i(j,k)}{\sum_{j'} [\lambda(j',l_k)/\lambda_{l_k}]\,\pi_{j'}(x_k)\phi_i(j',k)} \\
&= \frac{v_{jk}e^{-r(j)a(l_k)x_k}\phi_i/\rho_k}{\sum_{j'} v_{j',k}e^{-r(j')a(l_k)x_k}\phi_i/\rho_k} \;=\; \frac{w_{jk}}{w_k}.
\end{aligned}$$

Similarly, for an *unclassified* defect k occurring on a *bad* chip i,

$$\begin{aligned}
\boldsymbol{P}\{j_k = j \mid x_k, \xi_i = 0\} &= \frac{[\lambda(j,l_k)/\lambda_{l_k}]\,\pi_j(x_k)\,[1-\phi_i(j,k)]}{\sum_{j'} [\lambda(j',l_k)/\lambda_{l_k}]\,\pi_{j'}(x_k)\,[1-\phi_i(j',k)]} \\
&= \frac{v_{jk}\left[1 - e^{-r(j)a(l_k)x_k}\phi_i/\rho_k\right]}{\sum_{j'} v_{j'k}\,[1 - e^{-r(j')a(l_k)x_k}\phi_i/\rho_k]} \\
&= \frac{v_{jk} - w_{jk}\phi_i/\rho_k}{v_k - w_k\phi_i/\rho_k} \\
&= \frac{v_{jk} - w_{jk}\phi_i/\rho_k}{v_k(1-\phi_i)} \\
&= \frac{v_{jk}/v_k - \phi_i w_{jk}/w_k}{1-\phi_i}.
\end{aligned}$$

Hence, the second term of (40), the expected number of type j defects among unclassified defects on layer l, equals

$$\begin{aligned}
\sum_{k\in U_l} \boldsymbol{P}\{j_k = j \mid \boldsymbol{x}, \boldsymbol{\xi}\} &= \sum_{i:l\in L_i}\sum_{k\in U_{il}} \left\{\xi_i\frac{w_{jk}}{w_k} + (1-\xi_i)\frac{v_{jk}/v_k - \phi_i w_{jk}/w_k}{1-\phi_i}\right\} \\
&= \sum_{i:l\in L_i}\left\{\frac{1-\xi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{v_{jk}}{v_k} + \frac{\xi_i-\phi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{w_{jk}}{w_k}\right\}. \qquad (42)
\end{aligned}$$

Finally, we compute the expected number of type j defects on an *uninspected layer* l. This expectation is not just the ratio of corresponding defect frequencies. Although the defect situation on an uninspected layer is hidden, the quality of a chip (ξ_i) is still known, and it should be used in our computation.

Similarly to (41), we define $\phi_{in}(j,l)$ to be the probability for chip i to be good, despite of its n type j defects on an uninspected layer l. Sizes of these defects are hidden and thus replaced by the corresponding expectation as in (7). Then, $\log \phi_{in}(j,l)$ can be obtained from $\log \phi_i$ by moving the effect of all n type j defects from the set $\{k \in i,\ l \notin L_i\}$ of defects on uninspected layers to the set C_{ijl} of classified defects, replacing, by the formula of total probability, their missing sizes by expectations ψ_{jl}. That is,

$$\log \phi_{in}(j,l) = \log \phi_i + \lambda(j,l)(1-\psi_{jl}) + n \log \psi_{jl}.$$

Then, the expected number of type j defects on an *uninspected layer* of a *good* chip i equals

$$\begin{aligned}
\boldsymbol{E}\left\{N_{ijl} \mid \xi_i = 1\right\} &= \sum_{n=0}^{\infty} n \boldsymbol{P}\left\{N_{ijl} = n \mid \xi_i = 1\right\} \\
&= \sum_{n} n \frac{\phi_{in}(j,l) e^{-\lambda(j,l)} \lambda^n(j,l)/n!}{\phi_i} \\
&= \sum_{n=0}^{\infty} n \left(e^{\lambda(j,l)(1-\psi_{jl})} \psi_{jl}^n\right) \left(e^{-\lambda(j,l)} \lambda^n(j,l)/n!\right) \\
&= \sum_{n=0}^{\infty} n e^{-\lambda(j,l)\psi_{jl}} \left(\lambda(j,l)\psi_{jl}\right)^n /n! \\
&= \lambda(j,l)\psi_{jl}. \qquad (43)
\end{aligned}$$

Similarly, for a bad chip i,

$$\begin{aligned}
&\boldsymbol{E}\left\{N_{ijl} \mid \xi_i = 0\right\} \\
&\quad = \sum_{n=0}^{\infty} n \boldsymbol{P}\left\{N_{ijl} = n \mid \xi_i = 0\right\} \\
&\quad = \sum_{n} n \frac{\left[1-\phi_{in}(j,l)\right] e^{-\lambda(j,l)} \lambda^n(j,l)/n!}{1-\phi_i}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{1-\phi_i}\sum_{n=0}^{\infty} n\left(1-\phi_i e^{\lambda(j,l)(1-\psi_{jl})}\psi_{jl}^n\right)e^{-\lambda(j,l)}\lambda^n(j,l)/n! \\
&= \frac{1}{1-\phi_i}\left(\sum_{n=0}^{\infty} n e^{-\lambda(j,l)}\lambda^n(j,l)/n! - \sum_{n=0}^{\infty} n\phi_i e^{-\lambda(j,l)\psi_{jl}}(\lambda(j,l)\psi_{jl})^n/n!\right) \\
&= \frac{1}{1-\phi_i}\left(\lambda(j,l)-\phi_i\lambda(j,l)\psi_{jl}\right) \\
&= \frac{\lambda(j,l)(1-\phi_i\psi_{jl})}{1-\phi_i}. \qquad (44)
\end{aligned}
$$

Combining (43) and (44), we obtain the third term of (40),

$$
\begin{aligned}
\sum_{i:l\notin L_i} \boldsymbol{E}\left\{N_{ijl} \mid \boldsymbol{\xi}\right\} &= \sum_{i:l\notin L_i}\left\{\xi_i\lambda(j,l)\psi_{jl} + \frac{1-\xi_i}{1-\phi_i}\lambda(j,l)(1-\phi_i\psi_{jl})\right\} \\
&= \lambda(j,l)\sum_{i:l\notin L_i}\left(\frac{\xi_i-\phi_i}{1-\phi_i}\psi_{jl}+\frac{1-\xi_i}{1-\phi_i}\right). \qquad (45)
\end{aligned}
$$

Finally, using (42) and (45) in (40) and adding the classified type j defects, we obtain the expression for the refined frequency estimator,

$$
\begin{aligned}
\hat{\lambda}(j,l) &= I^{-1}\left\{d_{jl} + \sum_{i:l\in L_i}\left(\frac{1-\xi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{v_{jk}}{v_k} + \frac{\xi_i-\phi_i}{1-\phi_i}\sum_{k\in U_{il}}\frac{w_{jk}}{w_k}\right)\right. \\
&\qquad \left. +\lambda(j,l)\sum_{i:l\notin L_i}\left(\frac{1-\xi_i}{1-\phi_i}+\frac{\xi_i-\phi_i}{1-\phi_i}\psi_{jl}\right)\right\}.
\end{aligned}
$$

Such a refinement of $\lambda(j,l,m)$ for all defect type, layers, and lots completes the E-step.

5.3 Prediction on good and on failed chips. Proof of Lemma 1.

Since $\xi_i \sim$Bernoulli(ϕ_i), we have $\boldsymbol{E}\{\xi \mid \phi\} = \phi$. Unconditionally, ξ_i are i.i.d. random variables with the *compound* distribution,

$$
\boldsymbol{P}\{\xi = 1\} = \int \phi dF(\phi), \quad \boldsymbol{P}\{\xi = 0\} = \int (1-\phi)dF(\phi).
$$

By the strong law of large numbers,

$$\hat{y}_g = \frac{\sum_{i=1}^{I} \phi_i \xi_i}{\sum_{i=1}^{I} \xi_i} \to \frac{\boldsymbol{E}(\phi\xi)}{\boldsymbol{E}(\xi)} = \frac{\boldsymbol{E}(\phi;\ \xi = 1)}{\boldsymbol{P}\left\{\xi = 1\right\}} = \boldsymbol{E}\left\{\phi \mid \xi = 1\right\}. \tag{46}$$

By the Bayes formula,

$$dF(\phi \mid \xi = 1) = \frac{\boldsymbol{P}\left\{\xi = 1 \mid \phi\right\} dF(\phi)}{\int \boldsymbol{P}\left\{\xi = 1 \mid \phi\right\} dF(\phi)} = \frac{\phi\, dF(\phi)}{\boldsymbol{E}_F(\phi)}. \tag{47}$$

Objectively, we deal with a Bayesian model, where $F(\phi)$ is a *prior distribution* of ϕ_i. Then, taking expectation over the *posterior* distribution of ϕ, and combining it with (46), we obtain

$$\lim_{I\to\infty} \hat{y}_g = \boldsymbol{E}\left\{\phi \mid \xi = 1\right\} = \int \phi \frac{\phi}{\boldsymbol{E}_F(\phi)} dF(\phi) = \frac{\boldsymbol{E}_F(\phi^2)}{\boldsymbol{E}_F(\phi)}.$$

The posterior distribution of ϕ_i given a bad chip, $\xi = 0$, is considered similarly. In this case, we have

$$F(\phi \mid \xi = 0) = \frac{\boldsymbol{P}\left\{\xi = 0 \mid \phi\right\}\ F(\phi)}{\int \boldsymbol{P}\left\{\xi = 0 \mid \phi\right\} dF(\phi)} = \frac{(1-\phi)F(\phi)}{1 - \boldsymbol{E}_F(\phi)},$$

so that

$$\lim_{I\to\infty} \hat{y}_b = \boldsymbol{E}\left\{\phi \mid \xi = 0\right\} = \int \phi \frac{(1-\phi)}{1 - \boldsymbol{E}_F(\phi)} dF(\phi) = \frac{\boldsymbol{E}_F(\phi) - \boldsymbol{E}_F(\phi^2)}{1 - \boldsymbol{E}_F(\phi)}.$$

References

[1] M. Baron, C. K. Lakshminarayan and Z. Chen. Markov random fields in pattern recognition for semiconductor manufacturing. *Technometrics*, 43:66–72, 2001.

[2] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. B*, 39(1):1–38, 1977.

[3] L. Devroye and L. Györfi. *Nonparametric density estimation. The L_1 view.* Wiley, New York, 1985.

[4] A. V. Ferris-Prabhu. Modeling the critical area in yield forecast. *IEEE J. Solid-State Circuits*, SC-20:878–880, 1985.

[5] M. H. Hansen, V. N. Nair and D. J. Friedman. Circuit fabrication processes for spatially clustered defects. *J. Amer. Statist. Assoc.*, 39:241–253, 1997.

[6] R. S. Hemmert. Poisson process and integrated circuit yield prediction. *Solid-State Electronics*, 24:511–515, 1981.

[7] M. B. Ketchen. Point defect yield model for wafer scale integration. *IEEE Circuits and Devices*, 1:24–34, 1985.

[8] M. D. Longtin, L. M. Wein and R. E. Welsch. Sequential screening in semiconuctor manufacturing, I: Exploiting spatial dependence. *Operations Research*, 44:173–195, 1996.

[9] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* Wiley, New York, 1997.

[10] F. J. Meyer and D. K. Pradhan. Modeling defect spatial distribution. *IEEE Trans. Computers*, 38:538–546, 1989.

[11] L. S. Milor. Yield modeling based on in-line scanner defect sizing and a circuit's critical area. *IEEE Trans. on Semiconductor Manufacturing*, 12 (1):26–35, 1999.

[12] P. Mullenix, J. Zalnoski, and A. J. Kasten. Limited yield estimation for visual defect sources. *IEEE Trans. on Semiconductor Manufacturing*, 10 (1):17–23, 1997.

[13] R. Ott, H. Ollendorf, H. Lammering, T. Hladschik, and W. Haencsh. An effective method to estimate defect limited yield impact on memory devices. *Proc. IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pages 87–91, 1999.

[14] E. Papadopoulou and D. T. Lee. Critical area computation via Voronoi diagrams. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 18 (4):463–474, 1999.

[15] O. D. Patterson and M. H. Hansen. The impact of tolerance on kill ratio estimation for memory. *IEEE Trans. on Semiconductor Manufacturing*, 15 (4):404–410, 2002.

[16] S. L. Riley. Limitations to estimating yield based on in-line defect measurements. In *Proc. 1999 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pages 46–54, 1999.

[17] J. Shier. A statistical model for integrated-circuit yield with clustered flaws. *IEEE Trans. Electron Devices*, 35:524–525, 1988.

[18] C. H. Stapper. On yield, fault distributions, and clustering of particles. *IBM J. Res. Develop.*, 30:326–338, 1986.

[19] C. H. Stapper. Large area fault clusters and fault tolerance in VLSI circuits: A review. *IBM J. Res. Develop.*, 33:162–173, 1989.

[20] C. H. Stapper, F. M. Armstrong and K. Saji. Integrated circuit yield statistics. *Proc. IEEE*, 71:453–470, 1983.

[21] W. Taam and M. Hamada. Detecting spatial effects from factorial experiments: an application from integrated-curcuit manufacturing. *Technometrics*, 35:149–160, 1993.

[22] A. Venkataraman and I. Koren. Determination of yield bounds prior to routing. In *Proc. 1999 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pages 4–13, 1999.

[23] I. A. Wagner and I. Koren. An interactive VLSI CAD tool for yield estimation. *IEEE Trans. on Semiconductor Manufacturing*, 8:130–138, 1995.

[24] R. M. Warner. Applying a composite model to the IC yield problem. *IEEE J. Solid-State Circuits*, SC-9:86–95, 1974.

[25] M. Watanabe and K. Yamaguchi. *The EM Algorithm and Related Statistical Models*. Marcel Dekker, New York, 2003.

[26] S. S. Wilks. The large-sample distribution of the likelihood ratio for testing composite hypotheses. *Ann. Math. Statist.*, 9:60–62, 1938.