

# AOI 데이터를 이용한 효과적인 Defect Size Distribution 구축방법: 반도체와 LCD생산 응용

하정훈<sup>\*</sup>

홍익대학교 정보컴퓨터공학부

## Effective Construction Method of Defect Size Distribution Using AOI Data: Application for Semiconductor and LCD Manufacturing

Chunghun Ha

School of Information & Computer Engineering, Hongik University, Seoul, Korea

Defect size distribution is a probability density function for the defects that occur on wafers or glasses during semiconductor/LCD fabrication. It is one of the most important information to estimate manufacturing yield using well-known statistical estimation methods. The defects are detected by automatic optical inspection (AOI) facilities. However, the data that is provided from AOI is not accurate due to resolution of AOI and its defect detection mechanism. It causes distortion of defect size distribution and results in wrong estimation of the manufacturing yield. In this paper, I suggest a size conversion method and a maximum likelihood estimator to overcome the vague defect size information of AOI. The methods are verified by the Monte Carlo simulation that is constructed as similar as real situation.

**Keyword:** defect size distribution, automatic optical inspection, semiconductor and LCD manufacturing, yield estimation, maximum likelihood estimator, Monte Carlo simulation

### 1. 서론

liquid crystal display(LCD) 또는 반도체 공정에서 수율의 감소를 유발하는 주원인은 제조공정 중에 발생하는 defect이다. 이러한 defect는 정상동작 중에 성장하여 불량률을 야기하기도 한다(Shindo *et al.* 1998, Barnett and Singh 2003). Defect는 제조공정상 발생하는 설계와 다른 모든 결함을 의미하는 데, 특히 defect 중에서 제품의 불량(failure)을 야기하는 것을 fault라 한다(Stapper and Rosner 1995). Defect 및 fault는 그 크기, 위치, 및 특성에 따라 global 또는 spot defect, hard 또는 soft defect, inter-layer 또는 intra-layer defect, functional 또는 parametric fault, temporary 또는 permanent fault 등으로 구분할 수 있다(Kuo *et al.* 1998).

Defect는 그 발현 특성에 따라 non-random defect와 random defect로 구분된다. non-random defect는 설비이상, 환경변화, 재료변경, 시스템이상 등에 의해 발생하는 defect로서 주로 라인수율(wafer 또는 glass에 기준한 수율)에 영향을 미친다. 이러한 defect는 정밀한 분석을 통하여 불량률의 원인을 파악하고 공정 내에서 각종 품질관리 기법(Total Productive Maintenance, Statistical Process Control 등)을 통하여 개선이 가능하다고 알려져 있다. 개선을 통하여 생산이 안정화되면, 이러한 defect에 기인한 불량률도 감소하고 따라서 수율은 증가하게 된다. 이러한 현상을 일반적으로 수율학습(yield learning)이라고 한다. 이에 반해, random defect는 그 발생 원인이 명확하지 않고 제어 불가능하며 예측이 어려워 장기간의 분석을 필요로 하는 defect로서 주로

이 논문은 2006년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2006-331-D00671).

<sup>\*</sup>연락처 : 하정훈 교수, 121-791 서울 마포구 상수동 72-1 홍익대학교 정보컴퓨터공학부 정보산업공학전공, Fax : 02-336-1130,

E-mail : chunghun.ha@hongik.ac.kr

2007년 07월 접수, 2회 수정 후 2007년 12월 게재확정.

die수율(wafer 또는 glass 내의 개별 제품에 대한 수율)에 영향을 미친다. 수율의 극대화를 위해서는 수율의 향상이 필수적이므로(Ha 2007), random defect를 극복 또는 회피할 수 있는 방안이 매우 중요하다.

Random defect를 극복하는 가장 일반적인 방법은 메모리 반도체에서 적용하고 있는 redundancy 설계로서 fault가 발생하였을 때, fault가 발생한 memory segments를 미리 설계한 redundancy로 대체하는 것이다(Yoo *et al.* 1996, Li *et al.* 2005). 이 방법은 구현이 용이하고 수율 향상에 큰 역할을 하나 여분의 회로, 즉 redundancy에 대한 비용부담과 fault가 발생 시 redundancy를 대체하는 기술적 문제가 발생한다. defect에 대응하는 다른 방법은 defect-robust 설계로서 이는 주요 불량률에 야기하는 defect의 특성을 계층적으로 분석하여 defect에 영향을 적게 받는 레이아웃(회로의 물리적 구현상태)을 설계하는 것이다. 어느 방법을 적용하더라도 defect에 대한 특성을 분석하고 합리적 모델링을 수립하는 것은 매우 중요하다고 할 수 있다.

Defect를 분석하고 모델링하는데 가장 널리 사용되는 방법은 Stapper(1973, 1984, 1993)가 제시한 defect에 기초한 통계적 수율예측방법이다. 이 방법은 우선 제조공정에서 발생하는 defect에 대한 분포(defect size distribution: DSD)를 구한다. 발생한 defect 중 일부만 fault가 되므로 단순한 defect와 fault가 되는 defect를 구분해야 하는 데, 이 때 필요한 것이 critical area이다. Critical area는 특정 크기의 defect가 발생하였을 때 fault가 되는 점들의 집합으로서 반도체 또는 LCD의 레이아웃에 의해 결정된다. DSD와 critical area가 결정되면 단위면적 당 평균 fault의 개수를 통계적으로 산출이 가능하고, die수율은 Poisson 분포 또는 compounded Poisson 분포를 적용하여 쉽게 계산할 수 있다. Stapper의 통계적 수율예측 기법은 이미 많은 연구자들(Kikuda *et al.* 1991, Koren and Koren 1998, Ha 2004)에 의해 그 실효성이 검증된 모델링 방법이다.

Defect에 기초한 통계적 수율예측 방법을 사용하기 위한 가장 기본적인 데이터 중의 하나는 위에서 언급한 DSD이다. DSD는 실제 제조 공정에서 발생한 defect의 데이터를 이용하여 도출한다. 제조 공정에서 발생하는 defect를 검출하는 공정 및 설비를 automatic optical inspection(AOI)라고 하는 데, 이는 광학을 이용하여 반복되는 패턴의 비교를 통하여 defect를 검출하는 방법을 사용한다. AOI 설비는 자체적으로 분해능(resolution)을 가지고 있으며, 공정흐름 상 AOI 설비의 위치에 따라 다른 분해능을 갖는다. AOI의 제공하는 defect size 정보는 분해능으로 인해 이산적인 양의 정수값을 가지며, 실제 defect의 size인 연속된 정보를 제공하지 않으므로 부정확한 defect size를 제공한다. 이로 인하여 낮은 분해능을 가진 AOI의 정보로부터 도출한 DSD는 왜곡된 형태와 크기를 갖게 되고 부정확한 수율예측을 유발한다.

본 논문의 목적은 AOI에서 제공하는 defect의 불연속적인 크기 데이터를 이용하여 효과적으로 정확한 DSD를 도출하는 방법을 연구하는 것이다. 이를 통하여 정확한 수율을 예측할 수

있는 기반을 공고히 함으로써, 향후 LCD/반도체 제조 공정에서 수율 향상에 이바지 할 수 있을 것으로 예상하고 있다. 기존의 연구는 대부분 DSD의 결과를 이용한 수율 예측 및 이에 대한 검증(Stapper and Rosner 1995, Kikuda *et al.* 1991, Koren and Koren 1998, Ha 2004), 또는 특정한 형태의 defect에 대한 DSD의 검증 및 개선방안(Milor 1999)에 대한 연구에 집중하고 있을 뿐, AOI의 분해능으로 야기되는 문제점을 분석 및 검증하고 해결 방안을 제시하는 것은 본 논문이 처음이다. 또한, 본 논문에서는 AOI로부터 획득한 defect의 크기 데이터를 토대로 효과적으로 DSD를 도출할 수 있는 Maximum Likelihood Estimator(MLE)를 유도함으로써 정확한 DSD를 도출할 수 있는 방법을 제시한다. 본 논문에서 제시한 방법은 실제 상황과 유사한 Monte Carlo 시뮬레이션을 통하여 분석하고 검증하였다.

본 논문의 구성은 다음과 같다. 우선, 제 2장에서는 defect를 이용한 die수율의 도출방법을 논함으로서 DSD가 수율에 미치는 영향에 대해 살펴볼 것이다. 제 3장에서는 AOI의 검출원리 및 AOI의 분해능으로 야기되는 문제점에 대하여 논할 것이다. 제 4장에서는 AOI를 이용한 DSD의 도출에 관한 문제를 Monte Carlo 시뮬레이션을 통하여 검증하고 AOI로부터 획득한 이산적인 defect의 크기 데이터를 연속적인 defect size로 변환하는 방법을 제시한다. 제 5장에서는 MLE를 이용한 DSD의 도출방법을 논하고 MLE의 적합성을 검증한다. 마지막으로 제 6장에서는 본 논문의 결론을 제시한다.

## 2. Defect를 이용한 die수율의 도출 방법 고찰

모든 defect는 직경(일반적으로 size로 불림)이  $x$ 인 원으로 가정한다. 실제 모든 defect는 정확한 원은 아니지만(Milor 1999), 타원 또는 특정한 모양으로 간주할 경우 해석이 매우 난해하고, random defect는 특정한 방향성을 띠지 않으므로 원이라고 가정한다. 우선 앞으로 빈번하게 사용하게 될 기호를 정의하면 다음과 같다.

- $X$  : defect의 크기를 나타내는 확률변수(random variable). defect를 원이라고 가정할 경우 직경을 의미.
- $D$  : defect의 밀도(density)에 대한 확률변수. 단위 면적 당 defect의 개수.
- $f_X(x)$  : 확률변수  $X$ 의 확률밀도함수(probability density function). DSD라고 불림.
- $f_D(y)$  : 확률변수  $D$ 의 확률밀도함수.
- $A_c(x)$  : defect 크기  $x$ 에 따른 critical area.
- $\lambda$  : 단위면적당 평균 fault의 개수

### 2.1 DSD

LCD 또는 반도체 공정에서 random defect는 size에 대하여 특

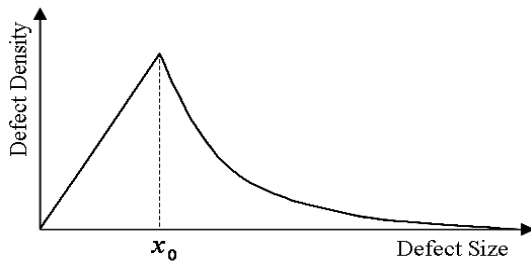


Figure 1. Defect Size Distribution

정한 분포를 가지고 있다고 알려져 있다. 그 분포를 defect size distribution 또는 간략히 DSD라고 부르며, 다음과 같은 확률밀도함수와 <Figure 1>의 형태를 갖는다(Stapper and Rosner 1995).

$$f_X(x|x_0, \omega) = \begin{cases} \frac{2(\omega-1)}{\omega+1} \frac{x}{x_0^2}, & 0 \leq x \leq x_0 \\ \frac{2(\omega-1)}{\omega+1} \frac{x_0^{\omega-1}}{x^\omega}, & x_0 \leq x \leq \infty \end{cases} \quad (1)$$

여기서  $\omega$ 는  $x_0$  이후의 DSD에 대한 함수의 차수를 나타내는 계수로서 보통 2.5~3.7정도를 갖는다고 알려져 있다.  $x_0$ 는 최대 밀도를 갖는 defect의 크기로서 defect의 크기와 검출기의 분해능에 영향을 받는 파라미터이다. 일반적인 DSD의 형태는 위와 같으나, 이는 확정적인 것은 아니며, 공정과 defect 검출장비에 따라 DSD와 파라미터  $x_0, \omega$ 는 다른 값을 갖는다. DSD는 defect에 기초한 확률적 수율분석의 시작점이므로 정확한 DSD를 찾는 것은 매우 중요하다.

## 2.2 평균 defect 밀도

DSD는 크기에 따른 분포만을 나타내므로 defect가 어느 정도의 밀도로 있는지 알기 위해서는 평균 defect 밀도를 계산하여야 한다. defect 분석에 관심이 있는 영역(예를 들면 pixel 또는 wafer)을  $A$ 라 하고  $s$ 를  $A$ 상에 있는 한 점이라고 하자.  $N(ds)$ 를  $s$ 의 주변의 작은 영역에서의 defect의 개수라고 정의하고 한 점  $s$ 에 두 개 이상의 defect가 동시에 존재할 수 없다고 가정하면,  $s$ 에서의 defect 밀도는 다음과 같이 정의할 수 있다(Hansen and Thyregod 1996, Hwang *et al.* 2007).

$$D = \lim_{|ds| \rightarrow 0} \frac{P\{N(ds) = 1\}}{|ds|}$$

여기서  $|ds|$ 는  $ds$ 의 면적을 의미한다. 따라서 평균 defect 밀도는 다음과 같이 정의할 수 있다.

$$D_0 = E[D] = \int_0^\infty y f_D(y) dy$$

단위면적에서 size  $x$ 에 대한 평균 defect 밀도  $D_0(x)$ 는 평균

defect 밀도  $D_0$ 와 DSD의 곱(즉,  $D_0(x) = D_0 f_X(x)$ )이므로 다음의 등식이 성립한다.

$$D_0 = \int_0^\infty D_0 f_X(x) dx = \int_0^\infty D_0(x) dx, \quad x \geq 0$$

## 2.3 Critical Area

모든 defect가 fault가 되는 것은 아니다. fault는 defect의 크기와 defect의 위치에 따라 결정되는데, 이는 critical area를 통하여 구할 수 있다. Critical area는 크기  $x$ 의 defect가 특정 위치에 존재할 때, 설계 layout에 의해 fault가 되는 defect들의 중심의 면적이다. 다시 말하면, 크기가  $x$ 인 defect의 중심이 critical area 안에 떨어지면 fault가 되고 거기서 벗어나면 fault가 되지 않는다. Critical area는 반도체의 설계도면으로부터 면적을 이용한 계산 또는 Monte Carlo 시뮬레이션을 통하여 계산이 가능하다(Allan and Walton 1997, Kuo *et al.* 1998). 평균 critical area  $A_c$ 는 다음과 같이 표현할 수 있다.

$$A_c = \int_0^\infty A_c(x) f_X(x) dx$$

## 2.4 평균 fault의 개수

세 개의 중요한 데이터, 즉, DSD, 평균 defect density, critical area가 확보되면, 다음 식으로부터 단위면적당 평균 fault 개수  $\lambda$ 를 구할 수 있다.

$$\begin{aligned} \lambda &= \int_0^\infty A_c(x) D_0(x) dx = D_0 \int_0^\infty A_c(x) f_X(x) dx \\ &= D_0 A_c \end{aligned}$$

## 2.5 die 수율의 계산

최종 목표인 die 수율은 평균 fault의 개수인  $\lambda$ 를 각종 수율모델에 적용하여 계산이 가능하다. 수율모델은 Poisson 모델, Murphy 모델, Seeds 모델, negative binomial 모델(Cunningham 1990) 등이 있으며, 이 중에서 Poisson 모델은 defect의 개수가 적고 random한 경우 또는 die의 크기가 작을 경우 많이 적용하고, gamma 분포를 Poisson 모델에 결합한 Negative Binomial 모델은 defect가 균등하게 퍼져 있지 않고 일정한 지역에 집중되어 있을 경우(clustering이 있을 경우) 또는 die의 크기가 클 경우에 많이 적용된다.

Poisson 수율 모델을 고려하면, 확률변수  $F$ 가 고려하는 면적(die)에 존재하는 fault의 개수라고 하고 단위면적당 평균 fault 개수가  $\lambda$ 일 때  $k$ 개의 fault가 발생할 확률은 다음과 같다.

$$\Pr(F=k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}, \quad k=0, 1, 2, \dots \quad (2)$$

die 수율은 fault가 없을 때의 확률이므로

$$Y_{die} = \Pr(F=0|\lambda) = e^{-\lambda} = e^{-A_c D_0} \quad (3)$$

가 된다.

Poisson 모델은 die의 크기가 작은 경우 random fault가 wafer 상에 균일하게 분포하여 비교적 정확한 수율 계산이 가능하다. 그러나 die의 크기가 커질수록 random fault는 wafer상에서 특정한 die에 집중되는 것처럼 보인다. 이것을 clustering이라고 하며, 이때의 수율모델은 Poisson 모델, Murphy 모델, Seeds 모델, negative binomial 모델 등을 사용하게 된다. 이러한 compound 수율 모델은 평균 defect의 개수가 상수가 아니고 확률변수라고 가정한다. 즉, 식 (3)에서  $A_c$ 는 상수이므로  $D_0$ 가 상수가 아닌 확률변수  $D$ 가 된다. 따라서 die 수율은 다음과 같이 표현할 수 있다.

$$Y_{die} = \int_0^{\infty} e^{-A_c y} f_D(y) dy$$

여러 종류의 compound 수율 모델은 각기 다른 종류의  $f_D(y)$ 를 사용한다. 대표적으로 많이 사용하는 negative binomial 모델은 그 파라미터에 따라 다양한 형태로 표현이 가능한 감마확률분포  $G(\alpha, \beta)$ 를 사용한다.  $G(\alpha, \beta)$ 의 확률밀도함수는

$$f_D(y) = \frac{1}{\Gamma(\alpha)\beta^\alpha} y^{\alpha-1} e^{-y/\beta}, \quad y \geq 0$$

이고, 이때의 die 수율은

$$Y_{die} = \int_0^{\infty} e^{-A_c y} \frac{1}{\Gamma(\alpha)\beta^\alpha} y^{\alpha-1} e^{-y/\beta} dy = \left(1 + \frac{A_c D_0}{\alpha}\right)^{-\alpha}$$

이다. 여기서 파라미터  $\alpha$ 는 clustering factor라고 한다.  $G(\alpha, \beta)$ 의 평균과 분산은

$$\bar{\lambda} = \alpha\beta = D_0 A_c, \quad Var(\lambda) = \sigma^2 = \alpha\beta^2$$

이므로,

$$\alpha = \frac{\bar{\lambda}^2}{\sigma^2 - \bar{\lambda}}$$

와 같이 추정할 수 있다. 그러나 이 값은 음수를 가질 수 있으므로 Cunningham(1990)은 다음의 값을 제시하였다.

$$\alpha = ave \left[ \left( \frac{\bar{\lambda}}{\sigma} \right)^2 \right] \frac{1}{1 - ave(\bar{\lambda})/\sigma^2}$$

clustering factor  $\alpha$ 는 수율 계산에 있어서 매우 중요한 역할을 한다.  $\alpha$ 의 변화에 따라  $G(\alpha, \beta)$ 의 형태가 바뀌고 이는 곧 die 수율에 영향을 미친다. 하지만, 아직까지는  $\alpha$ 의 계산에 있어서 실증적으로 효과적이라고 입증된 것은 없는 상태이고, 때로는 die 수율로부터 역산해서 사용하기도 한다.

지금까지의 die 수율계산방법은 단일 layer에서만 적용되며, 반도체나 LCD와 같이 여러 개의 layer로 구성된 경우에는 각 layer의 수율을 구한 후 layer별 수율을 곱하면 전체수율을 계산할 수 있다(Zhou *et al.* 2002).

### 3. AOI의 검출원리 및 문제점

Defect를 이용한 수율 모델을 적용하기 위해서 가장 기본적이며 필수적인 데이터는 DSD이다. 제조공정에서 발생하는 defect는 AOI 설비를 통하여 검출하고 데이터를 저장하게 된다. AOI는 광학적 검사장비이므로 설비에 따른 분해능을 가지고 있다. 분해능이 높으면 검출하는 defect에 대한 정밀도는 높아지나 검사 시간이 증가하고, 분해능이 낮으면 정밀도는 낮아지나 검사 시간이 감소한다. 일반적으로 제조 공정에서는 공정 플로우상에 있는 AOI 검사(inline AOI)는 낮은 분해능을, 불량 검출을 위해 공정 플로우 밖에서 하는 검사(stand-alone)는 높은 분해능을 적용한다. AOI의 분해능은 도출된 DSD의 특성에 많은 영향을 주는 데, 다음에 설명하는 검출원리(LCD를 기준으로 설명함)와 매우 관련이 깊다. LCD패널의 화면은 <Figure 2>와 같이 pixel이라 불리는 단위 패턴의 반복적인 결합이다. Pixel은 일반적으로 동일한 설계를 적용하므로 두 개의 pixel간 image는 동일하여야 한다. 두 개의 pixel은 AOI의 고분해능 CCD 카메라

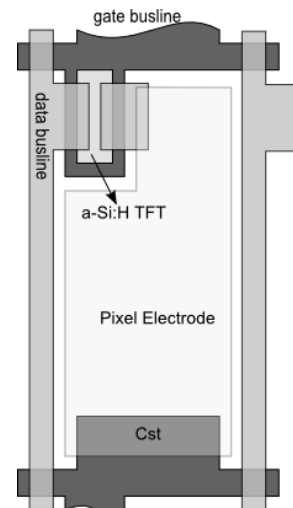


Figure 2. pixel layout

를 이용하여 최소비교단위(이는 AOI의 분해능에 따라 다르며 가로 및 세로의 길이는 일반적으로 같다. <Figure 3>에서 box 1개)를 기준으로 pixel간 비교가 이루어진다. 만약에 최소비교단위의 gray레벨이 차이가 발생하면 이것을 defect로 간주하고 위치 및 정보를 데이터로 저장한다.

AOI는 각 좌표 및 gray level외에 defect의 크기와 관련된 세 개의 의미 있는 데이터  $SIZE_X$ ,  $SIZE_Y$ , 및  $SIZE_A$ 를 제공한다.  $SIZE_X$ 는 defect가 가로축으로 걸쳐진 최소비교단위의 최대개수이고,  $SIZE_Y$ 는 defect가 세로축으로 걸쳐진 최소비교단위의 최대개수,  $SIZE_A$ 는 defect가 걸쳐진 최소비교단위의 총 개수를 의미한다 (<Figure 3> 참조). DSD를 도출하기 위해서는 defect의 size에 따른 발생 확률을 구해야 하고, defect size는 세 개의 데이터  $SIZE_X$ ,  $SIZE_Y$ ,  $SIZE_A$ 로부터 환산하여야 한다.

그러나 AOI의 데이터로부터 defect의 size를 도출하는 것은 용이하지 않다. 그 이유는 AOI의 분해능 때문인데, AOI의 분해능이 충분히 높다면(AOI 분해능이 작은 값을 가짐) 문제가 발생하지 않으나, AOI의 분해능이 낮으면(AOI 분해능이 큰 값을 가짐) defect size에 대한 많은 정보가 유실된다. 예를 들어 <Figure 3>에서 보면, (a)와 (b)는 defect의 크기는 같고 분해능의 차이가 2배이다. 분해능이 높은 (a)에서는  $SIZE_X$ 와  $SIZE_Y$ 가 각각 6으로서 defect의 면적은  $\frac{\pi}{4}(SIZE_X)(SIZE_Y) \approx 28.26$ 으로  $SIZE_A$ 의 크기인 29와 유사하다. 반면에 분해능이 낮은 (b)의 경우 유추한  $SIZE_A$ 의 크기는 약 7로서 측정값인 9와 큰 차이를 보인다. LCD 공정에서 사용하는 inline AOI의 분해능은 일반적으로 4~7 $\mu m$  정도이고,  $SIZE_X$ ,  $SIZE_Y$ ,  $SIZE_A$ 는 불연속적인 자연수로 표현된다. 따라서 LCD에서 가장 중요한 size 영역인 2~15  $\mu m$ (LCD는 배선폭이 5~10 $\mu m$  정도이므로) 구간의 size 데이터가 부정확하게 된다. AOI의 분해능으로 발생하는 또 다른 문제점은 실제 defect의 크기가 다르더라도 검출 시에 동일한 크기의 데이터를 가질 수 있다는 것이다. 예를 들면 <Figure 3>의 (b)와 (c)는 동일한 분해능을 가진 다른 크기의 defect이나 검출된 크기 데이터는 동일하다. AOI의 분해능에 따라 발생하는 마지

막 문제점은 defect의 위치에 따라 AOI의 검출 데이터가 변한다는 것이다. 예를 들어 AOI의 분해능이 7 $\mu m$ 이고 defect의 size가 6  $\mu m$ 라고 가정하자. defect의 중심의 위치가 최소비교단위의 중앙에 위치하면  $SIZE_X$ ,  $SIZE_Y$ ,  $SIZE_A$ 는 모두 1로서 defect의 실제 size는 약 7 $\mu m$ 로 추정된다. 반면에 defect의 중심의 위치가 4개의 인접한 최소 비교단위의 중심에 위치한다면,  $SIZE_X$ 는 2,  $SIZE_Y$ 는 2,  $SIZE_A$ 는 4로서 defect의 size는 약 14 $\mu m$ 로 인식이 된다. 이렇게 불확실한 정보로는 정확한 DSD를 도출하기 어려우며, 이에 따라 정확한 수율 예측도 불가능하게 된다.

#### 4. AOI 검출 데이터의 size 변환 방법

AOI의 검출방법과 분해능에 의해 defect size가 왜곡되는 것은 필연적인 것으로 이에 대한 해결책은 AOI의 분해능을 증가시키거나 defect image를 통한 직접측정 밖에 해결방법은 없다. 하지만 이러한 방법들은 새로운 설비의 구입 또는 공정 사이클 타임의 증가가 필수적이므로 현 공정 내에서 바로 적용하기는 어렵다. Milor(1999)는 AOI 검출 데이터가 실제 defect와 일치하지 않는 것을 실험을 통하여 보여 주었으며,  $SIZE_X$ ,  $SIZE_Y$ ,  $SIZE_A$ 의 데이터를 조합하여 실제 defect의 size에 근사한 값을 도출하였다. 그러나 Milor는 실험과 trial and error 방법을 통한 방법을 제시하였을 뿐 AOI의 분해능으로 인한 근본 원인과 합리적인 방법은 제시하지 못하였다. 본 논문에서는 주어진 AOI의 이산적인 데이터를 통하여 연속적인 defect size를 도출하는 방법을 제시하고 Monte Carlo 시뮬레이션을 통하여 이를 검증할 것이다.

##### 4.1 Monte Carlo 시뮬레이션

Monte Carlo 시뮬레이션은 실제 defect 발생환경과 유사하게 구성하였다. 우선 AOI의 비교대상인 pixel을 구성한다. pixel은 최소비교단위를 가로와 세로에 각각 100개 이상을 인접하여

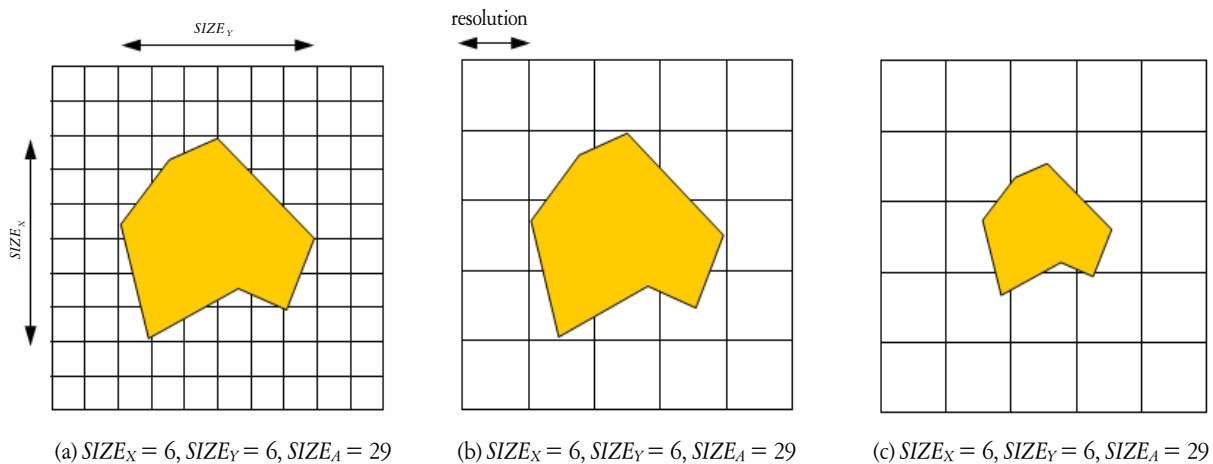


Figure 3. AOI Defect Size Data According To Various Condition

생성한다. 다음에 size가  $x$ 인 원형 defect를 발생시키는 데, 난수를 이용하여 defect 중심의 위치를 pixel의 내부에 random하게 발생하도록 한다. defect의 위치와 size, pixel의 최소비교단위의 위치와 크기에 따라  $SIZE_x$ ,  $SIZE_y$ ,  $SIZE_A$ 를 계산한다. 위의 실험을  $N$ 번 실행하여 발생하는 데이터를 이용한다. Monte Carlo 시뮬레이션은 MATLAB R2006c를 이용하여 실행하였다.

4.2 분해능에 따른 분포 검증

AOI의 분해능에 따른 문제점을 확인하기 위하여 defect의 size를  $10\mu m$ 으로 고정하고 AOI의 분해능을  $2\mu m$ 와  $7\mu m$ 로 각각  $N=500$ 에 대해 실험을 하였다. 각 경우에 대한  $SIZE_x$ ,  $SIZE_y$ ,  $SIZE_A$ 의 히스토그램은 Figure 4(AOI resolution =  $2\mu m$ )와 <Figure 5>(AOI resolution =  $7\mu m$ )에 제시하였다. 히스토그램에서 매우 낮은 빈도수를 가지고 발생하는 값들은 pixel의 좌외곽부에 defect가 발생하여 원래의 defect의 크기보다 작게 감지한 경우이다.

<Figure 4>와 <Figure 5>를 비교해 보면, 제 3장에서 분석한 바와 동일하게 AOI의 분해능이 낮을수록  $SIZE_x$ ,  $SIZE_y$ ,  $SIZE_A$ 의

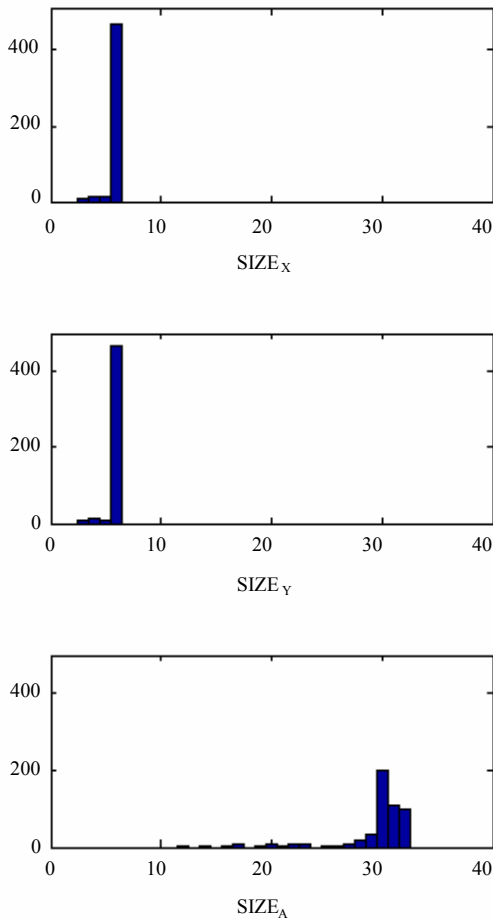


Figure 4. Histogram with  $x = 10\mu m$  and AOI resolution =  $2\mu m$

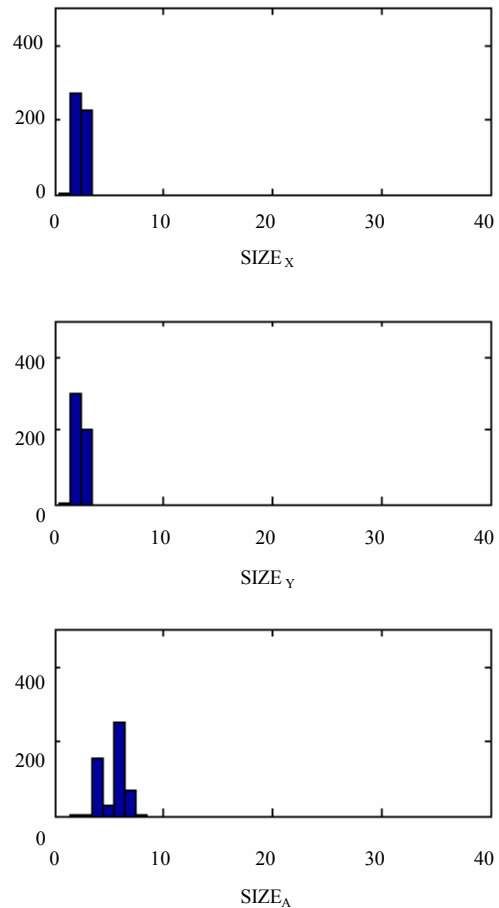


Figure 5. Histogram with  $x = 10\mu m$  and AOI resolution =  $7\mu m$

산포가 커짐을 확인할 수 있다. 여기서 주목해야 할 점은 Figure 5의  $SIZE_A$ 이다.  $SIZE_A$ 는  $SIZE_x$ 나  $SIZE_y$ 보다 중요한 정보이다. 이는 면적에 대한 정보이므로  $SIZE_x$ 나  $SIZE_y$ 보다 분해능의 영향이 상대적으로 적으며, defect가 원형이 아닌 경우에도 defect의 size를 효과적으로 도출할 수 있는 중요한 정보이다. 실제로 defect의 size는  $SIZE_A$ 를 이용하는 것이 오차가 적으며,  $SIZE_x$  및  $SIZE_y$ 는 보완 또는 검증을 위한 데이터 역할을 한다.

<Figure 5>의  $SIZE_A$ 의 히스토그램에서 특이한 점은  $SIZE_A = 5$  근방에서 defect size에 따른 밀도가 연속적이지 못하고 급격히 감소하는 현상을 보이고 있다는 것이다. 이는 defect가 무작위로 발생하였을 때  $SIZE_A$ 가 4일 확률이나 6일 확률보다 5일 확률이 매우 낮다는 것을 의미한다. 다시 말하면, 실제 defect를 AOI를 통하여 감지하고 히스토그램을 작성하였을 때, <Figure 6>의 경우와 같이 왜곡된 분포형태를 가질 수 있다. 이로 인하여 특정한 size의 defect의 발생이 매우 간헐적으로 발생할 수 있다고 판단하고 잘못된 의사결정(예를 들면 확률밀도가 낮은 size로 layout을 결정함)을 할 수 있는 가능성이 존재한다. 이러한 왜곡현상을 회피하기 위해서는 히스토그램 작성 시 적절한 Bin 크기를 선정해야 한다. 최적의 Bin 크기를 결정하는 방법은 표준편차를 이용하는 Scott의 방법과 IQR(interquartile range)을

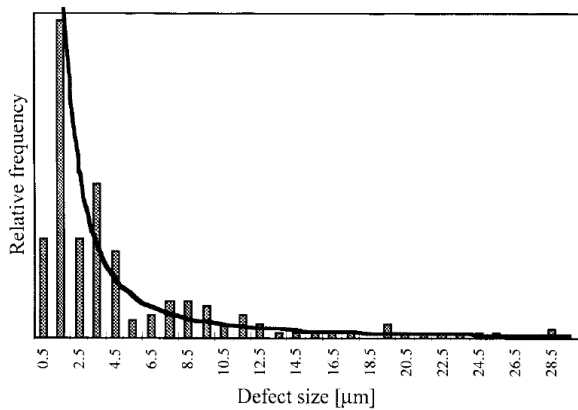


Figure 6. Defect Size Histogram (Nunari et al. 1998)

이용하는 Freedman and Diaconis 등의 방법(Izenman 1991)이 있으나, 본 논문에서는 논의하지 않는다.

### 4.3 연속적인 defect size 도출방법

SIZE<sub>x</sub>, SIZE<sub>y</sub>, 그리고 SIZE<sub>A</sub>의 정보는 모두 이산적인 데이터이고, defect size는 연속적인 데이터이다. 따라서 DSD를 구성하기 위해서는 SIZE 정보를 defect size로 변환하기 위한 방법이 필요하다. 가장 간단한 방법은 Milor(1999)가 제시한 방법으로 다음과 같이 계산한다.

$$x_{XY} = k_{AOI} \sqrt{(SIZE_X)(SIZE_Y)}$$

$$x_A = k_{AOI} \sqrt{\frac{4(SIZE_A)}{\pi}}$$

여기서  $k_{AOI}$ 는 AOI의 분해능을 의미한다. Milor(1999)의 논문에서는 AOI의 분해능을 고려하지 않고  $x_{XY}$ 와  $x_A$ 를 직접 계산하였으나 본 논문에서는 수식의 일관성을 유지하기 위해  $k_{AOI}$ 를 추가하였다. Milor는  $x_{XY}$ 와  $x_A$  모두 실제 defect size와 차이가 있다고 주장하며, image를 통한 실측결과 최적의 defect size로서  $x_{XY}$ 와  $x_A$ 의 평균을 제시하였다. Milor가 제시한 방법의 단점은 offset이 존재하는 것이다. 예를 들어 defect size가  $x \ll k_{AOI} = 5\mu\text{m}$ 인 경우를 고려해 보자. 이 경우  $SIZE_X = SIZE_Y = SIZE_A = 1$ 로 예측할 수 있고, Milor가 제시한 방법으로 계산된 defect size는  $x_{XY} = k_{AOI} = 5\mu\text{m}$ 와  $x_A = (1.128)k_{AOI} = 5.64\mu\text{m}$ 로서 실제 defect size와 많은 차이를 보이게 된다.

본 논문에서 제시하는 방법은 offset을 제거한 defect size 계산식으로  $SIZE_A = 1$  일 때, defect size는 0이 된다.

$$x_{HA} = k_{AOI} \sqrt{\frac{4(SIZE_A)}{\pi}} - k_{AOI} \sqrt{\frac{4}{\pi}}$$

$$= k_{AOI} \sqrt{\frac{4}{\pi}} (\sqrt{SIZE_A} - 1) \tag{4}$$

여기서 defect가 존재한다면  $SIZE_A > 0$ 이므로  $x_{HA} \geq 0$ 을 항상 만족하며,  $x \ll k_{AOI}$ 인 경우에도 도출한 defect size의 오차가 적다.

본 논문에서 제시한 식 (4)의 size 환산방법의 효율성을 입증하기 위하여 Monte Carlo 시뮬레이션을 시행하였다. 시뮬레이션은 AOI의 분해능은  $5\mu\text{m}$ 로 가정한 후, (0, 30)의 interval에서  $2\mu\text{m}$ 의 간격으로 defect를 500개씩 발생시키고, 이 때 도출한  $SIZE_X$ ,  $SIZE_Y$ ,  $SIZE_A$ 의 평균값을 구하였다. <Figure 7>은  $SIZE_A$ 의 평균값에 대한 defect size  $x$ 의 계산값을 나타낸다. <Figure 7>을 보면, 본 논문에서 제시한 환산방법( $x_{HA}$ )은 인위적으로 발생시킨 defect size, 즉 generated  $x$ 와 거의 일치하나, Milor(1999)가 제시한 방법( $x_{XY}$ ,  $x_A$ ,  $AVE(x_{XY}, x_A)$ )은 일정한 offset을 가지고 있다. 이 차이는 AOI의 분해능을 높일수록 작아진다.

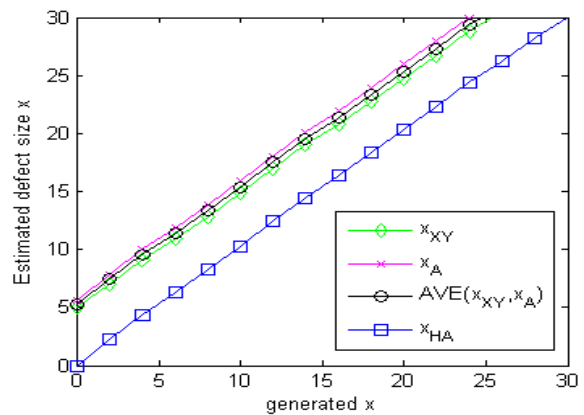


Figure 7. Estimated Defect Size According to SIZEA

### 4.4 MLE(Maximum Likelihood Estimator)를 이용한 DSD의 도출방법

DSD는 defect의 size에 따른 분포로서 식 (1)에서와 같이 두 개의 모수(parameter)  $x_0$ 와  $w$ 를 갖는 확률밀도함수이다. 히스토그램은 DSD의 형태를 파악하는 것은 가능하지만 Bin의 크기에 따른 편차가 크기 때문에 확률밀도함수인 DSD를 정확하게 도출하는 것은 매우 어렵다. 확률밀도함수의 형태를 알고 있을 때, 모수를 추정하는 가장 보편적인 방법은 최우추정법(MLE)이다. DSD에 MLE를 적용할 때, 두 개의 모수 중  $x_0$ 는 히스토그램에서 최대 빈도를 갖는 값으로 추정이 가능하므로  $x_0$ 를 상수로 가정하면 단순화하여  $w$ 의 MLE를 구할 수 있다.

$w$ 의 MLE를 도출하기 위한 Likelihood function은 다음과 같이 표현할 수 있다.

$$L(w) = \prod_{i=1}^{\tilde{N}} \frac{2(w-1)}{w+1} \frac{x_i}{x_0^2} \prod_{j=1}^{\tilde{N}} \frac{2(w-1)}{w+1} \frac{x_0^{w-1}}{x_j^w}$$

$$= \left[ \frac{2(w-1)}{w+1} \right]^{\tilde{N}} (x_0^{-2})^{\tilde{N}} (x_0^{w-1})^{\tilde{N}} \prod_{i=1}^{\tilde{N}} x_i \prod_{j=1}^{\tilde{N}} x_j^{-w}$$

여기서  $N$ 은 총 샘플의 개수이고,  $\hat{N}$ 은  $x \leq x_0$ 인 샘플  $x_i$ 의 개수,  $\tilde{N} = N - \hat{N}$ 은  $x > x_0$ 인 샘플  $x_j$ 의 개수이다. 여기서 양변에  $\log$ 를 취하면, 다음의 등식이 성립한다.

$$\begin{aligned} \log L(w) = & N[\log 2 + \log(w-1) - \log(w+1)] \\ & - 2\hat{N}\log x_0 + \tilde{N}(w-1)\log x_0 \\ & + \sum_{i=1}^{\hat{N}} \log x_i - w \sum_{j=1}^{\tilde{N}} \log x_j \end{aligned}$$

MLE를 구하기 위해 양변을  $w$ 에 대하여 미분하면,

$$\frac{d \log L(w)}{dw} = N \left[ \frac{1}{w-1} - \frac{1}{w+1} \right] + \tilde{N} \log x_0 - \sum_{j=1}^{\tilde{N}} \log x_j$$

이 구해지고, 이 값을 0으로 하는  $w$ 에 대한 MLE  $\hat{w}$ 는

$$\hat{w} = \sqrt{\frac{2}{\frac{1}{N} \left( \sum_{j=1}^{\tilde{N}} \log x_j - \tilde{N} \log x_0 \right)} + 1} \quad (5)$$

이 된다.

#### 4.5 MLE $\hat{w}$ 의 적합성 검증

MLE  $\hat{w}$ 가 효과적이지 판단하기 위하여 다음과 같이 Monte Carlo 시뮬레이션을 실행하였다. 우선,  $x_0$ 와  $w$ 에 임의의 상수를 지정한 후, DSD를 생성한다. 다음에 생성된 DSD를 따르는  $N$ 개의 defect를 발생시킨다. 발생한  $N$ 개의 defect로부터 각각의  $SIZE_A$ 를 계산하고 이를 식 (4)을 이용하여  $x_{HA}$ 를 구한다.  $x_{HA}$ 의 히스토그램으로부터  $x_0$ 의 추정치  $\hat{x}_0$ 를 히스토그램의 최빈수로 결정한다. 다음에 식 (5)를 이용하여  $\hat{w}$ 를 계산한다. 실험에 사용한 주요 파라미터는 다음과 같다.

- a.  $N = 10,000$ ,
- b.  $x_0 = 7$ ,
- c.  $w \in \{2, 3, 4\}$ ,
- d. AOI resolution  $\in \{1, 2, 3, 4, 5\}$ .

<Figure 8>은  $w = 3$  및 AOI resolution = 2로 고정된 상태에서 추정된 DSD를 그래프로 표현한 것이다. <Figure 8>에서  $DSD_{real}$ 은 DSD의 확률밀도함수, 즉 식 (1)을 이용하여 계산한 DSD를 의미하고,  $DSD_{estimated-dia}$ 는  $DSD_{real}$ 에 따라 발생시킨  $N$ 개의 실수 형태의 defect의 size에 대한 데이터와  $x_0 = 7$ 를 가지고 식 (5)의 MLE를 이용하여  $\hat{w}$ 를 추정하여 구한 DSD이고,  $DSD_{estimated-SIZE_A}$ 는  $DSD_{real}$ 에 따라 발생시킨  $N$ 개의 defect에 대해  $SIZE_A$ 를 구하

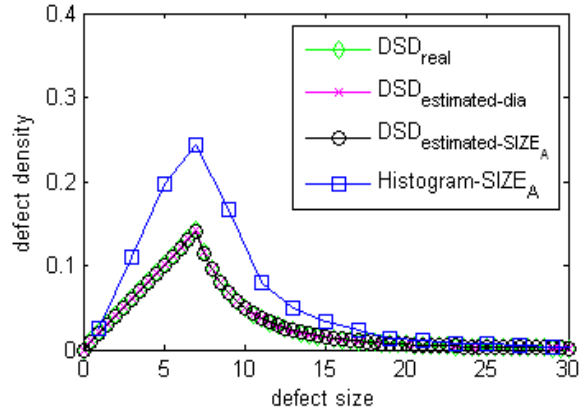


Figure 8. Comparison of DSDs with  $w=3$  and AOI resolution = 2

고,  $SIZE_A$ 를 이용하여  $\hat{x}_0$ 과  $\hat{w}$ 를 동시에 추정하여 도출한 DSD이며, Histogram- $SIZE_A$ 는  $N$ 개의 defect의  $SIZE_A$ 에 대해 식 (4)를 이용하여 size를 변환한 defect의 히스토그램을 선으로 표현한 그래프이다. Histogram- $SIZE_A$ 를 구할 때 Bin의 크기는 AOI resolution과 동일한 값을 사용하였고, 밀도는  $N$ 으로 나누어 주었다. 본 논문에서 Bin의 크기를 AOI resolution과 일치시킨 이유는 AOI에서는 defect의 크기 정보로서 실수가 아닌 정수값만을 제공하므로 Bin 크기의 최적화 기법을 적용하기 어렵기 때문이다. 만약 Bin의 크기를 줄이면, 중간에 data가 존재하지 않는 Bin이 필연적으로 발생하고, 반대로 Bin의 크기를 키우면 Histogram 자체를 그리기 어렵다. 현실적으로 이것을 극복할 수 있는 방법은 분해능이 높은 stand-alone AOI data를 이용하여 Histogram을 그려  $x_0$  값을 도출하고 이것을 inline AOI의 data에 적용시키는 방법을 사용할 수 있다.

<Figure 8>에서 보듯이 본 논문에서 제시한 방법을 이용한 DSD의 도출이 실제 DSD와 거의 일치함을 알 수 있다. 히스토그램은 DSD와 형태는 유사하나 Bin 크기에 따라 그 밀도가 크게 차이가 난다.

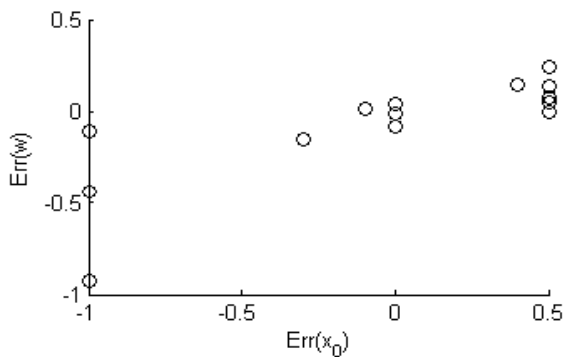
정량적으로  $\hat{w}$ 의 적합성을 비교하기 위하여  $w$  및 AOI resolution을 값을 변화시켜  $\hat{x}_0$ 과  $\hat{w}$ 을 도출하였다. 실험은 각 조건에 따라 10회씩 반복 실시하였으며, 실험결과는 <Table 1>에 정리하였다.

실험결과 <Table 1>을 보면 몇 가지 주목할 점이 있다. 첫째,  $\hat{x}_0$ 는 AOI resolution이  $1\mu m$ 인 경우를 제외하고 10번의 반복 실험 결과가 동일하였다. 이는 AOI 분해능이 큰 경우에는 히스토그램의 Bin 크기 또한 커지므로 변동성이 줄어들어 발생한 결과로 판단된다. 둘째,  $\hat{w}$ 은 변수인  $w$ 와 AOI 분해능에 관계없이 DSD의 모수인  $w$ 를 잘 추정하고 있으므로 DSD의 함수가 식 (1)과 같이 주어진 경우 MLE는 신뢰할 만한 추정치로 판단된다. 셋째,  $\hat{w}$ 의 정확도는 실제  $x_0$ 와  $x_0$ 의 추정치인  $\hat{x}_0$ 의 차이와 관련이 있다.  $x_0$ 와  $w$ 의 오차를 각각  $Err(x_0) = \hat{x}_0 - x_0$  및  $Err(w) = \hat{w} - w$ 로 정의하고 이 값들에 대한 산포도를 그리면 <Figure



Table 1.  $\hat{x}_0$  and  $\hat{w}$  According to  $w$  and AOI resolution

$w$	AOI res.	$\hat{x}_0$		$\hat{w}$	
		mean	std.	mean	std.
2	1	7.4	0.3162	2.1472	0.0449
	2	7.0	0	2.0458	0.0167
	3	7.5	0	2.0808	0.0174
	4	6.0	0	1.8927	0.0169
	5	7.5	0	2.0526	0.0151
3	1	6.9	0.5164	3.0184	0.1900
	2	7.0	0	2.9885	0.0290
	3	7.5	0	3.1350	0.0458
	4	6.0	0	2.5605	0.0110
	5	7.5	0	3.0537	0.0363
4	1	6.7	0.4216	3.8543	0.3818
	2	7.0	0	3.9210	0.0576
	3	7.5	0	4.2437	0.0534
	4	6.0	0	3.0754	0.0292
	5	7.5	0	4.0007	0.0510

Figure 9. Scatter Diagram of  $Err(x_0)$  and  $Err(w)$ 

9>와 같고 이들의 상관계수는 0.7953으로 강한 양의 상관관계를 갖고 있음을 알 수 있다. 따라서 정확한  $\hat{w}$ 를 도출하기 위해서는 정확한  $\hat{x}_0$ 의 도출이 필수적임을 알 수 있다.

## 5. 결론

본 논문에서는 AOI의 크기 데이터로부터 정확한 defect size를 유도하고 이를 이용하여 정확한 DSD를 도출하는 방법을 제시하였다. 본 논문에서 제시한 방법은 DSD를 효과적으로 도출함으로써 defect를 이용한 통계적 수율예측의 정확도를 높일 수 있을 것으로 예상된다. 하지만 여전히 남아있는 문제는 inline AOI의 분해능이다. defect에 대한 실시간 정보는 inline AOI를 통하여 구할 수 있으나, 사이클 타임의 문제로 inline AOI의 분해능을 높이는 것은 어렵다. 낮은 분해능은 제 4장에서 언급하

였듯이 부정확한  $\hat{x}_0$ 를 초래하고 이는 다시 부정확한  $\hat{w}$ 로 귀착된다. 이를 극복하는 방법은 상대적으로 분해능이 높은 stand-alone AOI를 적극적으로 활용하는 것이다. 다른 문제는 DSD의 확률밀도함수에 대한 검증이다. 많은 연구자들이 DSD를 식 (1)과 같은 형태로 모델링하고 이에 대한 언급을 하였으나, 통계적 검증을 통한 검증은 이루어 지지 않았다. 더구나 식 (1)은 LCD에서와 같이 defect의 size가 큰 경우에 어떠한 검증 결과도 존재하지 않는다. 향후 연구에서는 이에 대한 추가적 논의가 필요하다고 생각한다.

## 참고문헌

- Allan, G. A. and Walton, A. J. (1997), Efficient critical area estimation for arbitrary defect shapes, 1997 *IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, 20-28.
- Barnett, T. S. and Singh, A. D. (2003), Extracting integrated-circuit yield-models to estimate early-life reliability, *IEEE Transactions on Reliability*, 52(3), 296-300.
- Cunningham, J. A. (1990), The use of evaluation of yield models in integrated circuit manufacturing, *IEEE Transactions on Semiconductor Engineering*, 3(2), 60-71.
- Ha, C. (2004), *Reliability-yield allocation for semiconductor integrated circuits: modeling and optimization*, Ph.D. dissertation, Texas A&M University.
- Ha, C. (2007), Relationship between yield and cost considering repair and rework for LCD manufacturing system, *Journal of the Korean Institute of Industrial Engineers*, accepted.
- Hansen, C. K. and Thyregod, P. (1996), Modeling and estimation of wafer yields and defect densities from micro electronics test structure data, *Quality and Reliability Engineering International*, 12, 9-17.
- Hwang, J. Y., Kuo, Way, and Ha, C. (2007), Modeling of integrated circuit yield using a spatial nonhomogeneous Poisson process, *Naval Research Logistics*, to be submitted.
- Izenman, A. J. (1991), Recent developments in nonparametric density estimation, *Journal of the American Statistical Association*, 86(413), 205-224.
- Kikuda, S., Miyamoto, H., Mori, S., Niuro, M., and Yamada, M. (1991), Optimized redundancy selection based on failure-related yield model for 64-Mb DRAM and 126 beyond, *IEEE Journal of Solid-State Circuits*, 26(11), 1550-1555.
- Koren, I. and Koren, Z. (1998), Defect tolerance in VLSI circuits : Techniques and yield analysis, *Proceedings of the IEEE*, 86(9), 1819-1837.
- Li, J. F., Yeh, J. C., Huang, R. F., and Wu, C. W. (2005), A built-in self-repair design for rams with 2-d redundancy, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 13(6), 742-745.
- Milor, L. S. (1999), Yield modeling based on in-line scanner defect sizing and a circuit's critical area, *IEEE Transactions on Semiconductor Manufacturing*, 12(1), 26-35.
- Shindo, W., Nurani, R. K., and Strojwas, A. J. (1998), Effects of defect propagation/growth on inline defect-based yield prediction, *IEEE Transactions on Semiconductor Manufacturing*, 11(4), 546-551.
- Stapper, C. H. (1973), Defect density distribution for LSI yield calculation, *IEEE Transactions on Electron Devices*, ED-20, 655-657.
- Stapper, C. H. (1984), Modeling of defects in integrated circuit photolithographic patterns, *IBM Journal of Research and Development*, 28(4), 461-475.
- Stapper, C. H. (1993), Improved yield models for fault-tolerant memory chips, *IEEE Transactions on Computers*, 42(7), 872-881.

Stapper, C. H. and Rosner, R. J. (1995), Integrated circuit yield management and yield analysis: development and implementation, *IEEE Transactions on Semiconductor Manufacturing*, 8(2), 95-102.

Way Kuo, Chien, W. K. and Kim, T. (1998), *Reliability, Yield, and Stress Burn-in: A Unified Approach for Microelectronics Systems Manufacturing and Software Development*, Norwell, Kluwer Academic Publishers.

Yoo, J. H., Kim, K. C., Lee, K. C., and Kyung, K. H. (1996), A 32-Bank 1Gb self-strobing synchronous DRAM with 1 Gbyte/s bandwidth, *IEEE Journal of Solid-State Circuits*, 31(11), 1635-1644.

Zhou, C., Ross, R., Vickery, C., Metteer, B., Gross, S., and Verret, D. (2002), Yield prediction using critical area analysis with inline defect data, *IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, 82-86.



### 하정훈

연세대학교 전자공학과 학사  
 Texas A&M University 산업공학 석사  
 Texas A&M University 산업공학 박사  
 현재: 홍익대학교 정보컴퓨터공학부 조교수  
 관심분야: 최적화, 생산경영, SCM