IMTC 2004 - Instrumentation and Measurement
Technology Conference
Como, Italy, 18–20 May 2004

Yield Evaluation Methods of SRAM Arrays: a Comparative Study

M. Ottavi[1], L. Schiano[2], X. Wang[2], Y-B. Kim[2], F. J. Meyer[3], F. Lombardi[2]

[1]Università di Roma “Tor Vergata”,
Rome, ITALY
Email: ottavi@ing.uniroma2.it

[2] ECE Department,
Northeastern University,
Boston MA 02115, USA
E–mail: {lschiano, xiawang, ybk, lombardi}@ece.neu.edu

[2]Wichita State University,
Wichita KS 67260, USA
Email: fred.meyer@wichita.edu

Abstract – *With the increasing needs for memory testing and repair, yield evaluation is an essential decision-making factor to define redundancy allocation and testing strategies. In particular, yield evaluation can resolve the many issues revolving around cost-effective BIST solutions and purely ATE based techniques to guarantee higher test transparency. In this document, two different yield calculation methodologies for SRAM arrays are presented. General yield calculation formulas for VLSI chips are initially presented. The regular repetitive structure of a RAM array is considered because it shows major yield improvements with the introduction of redundancy. Two repair yield evaluation formulas for one dimensional redundant array are introduced and compared; the first one is based on Markov modeling, the second one is based on an approximation.*

Keywords – *Manufacture,SRAM, Yield, Markov chains.*

I. INTRODUCTION

Large productions of chips require monitoring to preserve the quality of the shipped product. The estimate of the yield (i.e. the percentage of working chips in a production batch) is an essential feature to monitor manufacturing as well as to select proper repair strategies if required.

This is valid for memory. Based on the expected yield, different strategies can be chosen for spare allocation and test process.

For spare allocation, an expected high yield could justify a small amount of spare rows/columns. To maintain the defect level of the shipped product at acceptable values, more redundancy may be needed at design level.

Furthermore, the impact of a yield estimate is essential in the selection of test and repair processes. Memory testers are designed by enhancing parallel testing to exploit maximum throughput with low test time. SRAM ATEs can test up to 128 DUT simultaneously, with test speeds up to 500 MHZ.

However, embedded SRAM and DRAM testing implies additional higher costs, due to the increased test time and ATE performances (for “at speed” testing for example). To lower test costs, and to allow tester designers to meet different memories requirements, DFT/BIST and BISR solutions have been proposed [4] [5] [2] [3]. With the increase in memory size, efficient solutions are required to maintain acceptable production yield [6]. Yield evaluation can also be used for selecting the most appropriate testing strategy. When the expected yield is high, BIST or BISR solutions may be preferred, otherwise more expensive ATE solutions may be chosen. Generally, the former technique leads to low test transparency, i.e. the fraction of defects not detected by test is higher. However, the test/repair cost reduction can justify the use of BISR if an higher yield is expected.

An expected low yield would require the use of purely ATE based test and repair techniques. Albeit more expensive, they can guarantee a lower defect level for the shipped product.

In this paper two different approaches for yield calculation of SRAM arrays are presented. The first approximated approach (Method A) is the basis of the CAYA (Compiler-based Array Yield Analysis) tool, described in [7]. Method B is based on Markov modeling and used as reference. Both the proposed approaches characterize the yield of repairable memory arrays with spare rows only. This condition is typical of manufactured

0-7803-8248-X/04/$17.00 ©2004 IEEE

TABLE I
FUNCTIONAL FAULTS FOR SRAM ARRAYS

Functional fault	**Description**
chip kill (ck)	unrepairable fault
single cell (sc)	only one faulty cell
horizontal pair (hp)	two adjacent faulty cells horizontally
vertical pair (vp)	two adjacent faulty cells vertically
single word line (row)	one faulty word line
single data column (col)	one faulty data column (bit lines, DIOs, AIO)

products. The effectiveness and accuracy of the two proposed yield calculation methods are then compared with respect to industrial design data of defect. The paper is organized as follows. Section 2 presents global yield modeling issues. In Section 3 the two methods are described. In section 4 both these methods are compared. Finally in section 5 conclusions are drawn.

II. GLOBAL YIELD MODEL

The evaluation of the effects of manufacturing defects D on a chip is related to the evaluation of the defect distribution and the critical area A in which they occur. The expected value of the AD product provides λ_0 which represents the mean value of the probability distribution of the number of defects (or faults) on the chip. From the layout of a circuit, the critical area is calculated by critical area extraction tools which are based on shape expansion or monte-carlo simulation. The defect density is obtained from the manufacturing line. The number of faults present in a SRAM can be calculated as follows. Let Λ be a (5×1) matrix of the average number for each type of functional faults; let A be a $(5 \times n)$ critical area matrix, An entry $A_{i,j}$ denotes the critical area of functional fault type j in critical area i; let D be a $(n \times 1)$ defect density matrix for n defined structural level faults. Using the defect density matrix (obtained from the manufacturing line), the number of faults (of different types) is given as

$$\Lambda^T = (\lambda_{ck}, \lambda_{sc}, \lambda_{hp}, \lambda_{vp}, \lambda_{row}, \lambda_{col}) \tag{1}$$

where ck, sc, hp, vp, row, col are the functional defect types as described in Table I, and

$$\Lambda = AD \tag{2}$$

Let λ_0 be the sum of the average numbers of faults of each type in the SRAM. The critical area method is used to model the effect of defect size and density on the chip. Short or open defects are then modeled into functional defect densities. From the analysis performed on an industrial embedded SRAM, for λ_0 as average faults per chip, the values of Table II have been computed.

TABLE II
FAULT TYPES PERCENTAGES

Chip Kill Fault	$\lambda_{ck} = 0.05 \cdot \lambda_0$
Single Cell Fault	$\lambda_{sc} = 0.45 \cdot \lambda_0$
Horizontal Pair Fault	$\lambda_{hp} = 0.1 \cdot \lambda_0$
Vertical Pair Fault	$\lambda_{vp} = 0.1 \cdot \lambda_0$
Single Row Fault	$\lambda_{row} = 0.15 \cdot \lambda_0$
Single Column Fault	$\lambda_{col} = 0.15 \cdot \lambda_0$

The evaluation of the effects of D on a chip is related to the evaluation of defect size distribution and the critical area A in which they occur. The expected value of the AD product provides λ_0 that is the mean number of faults expected on a chip. Namely, the value of λ_0 represents the mean value of the probability distribution of the faults present on the chip. The probability of having k faults on a chip can be approximated with the discrete Poisson distribution of a random variable X=k [8] [9].

$$P\{X = k\} = \frac{e^{-\lambda_0}\lambda_0^k}{k!} \tag{3}$$

Where k is the number of faults on the chip It can be demonstrated that mean and variance of (3) are both equal to λ_0. If the chip has no redundancy the yield is obviously given by equation (3) computed for $k = 0$

$$Y = P\{X = 0\} = e^{-\lambda_0} \tag{4}$$

Namely, yield is defined as the probability of having 0 faults on a chip.

However, it has been noticed from the experimental data that the mean value of the failures distribution and the variance do not match λ as a Poisson distribution requires [9]. This has been explained as the effect of a non constant distribution of defect density λ also known as clustering effect. A mixed Poisson distribution is therefore applied using a gamma distribution as mixing function. The result is modeled with a Polya-Eggenberger distribution

$$P\{X = k\} = \frac{\Gamma(k+\alpha)(\frac{\lambda_0}{\alpha})^k}{k!\,\Gamma(\alpha)(1+(\frac{\lambda_0}{\alpha}))^{\alpha+k}} \tag{5}$$

The mean and the variance of this distribution function are $E(X = k) = \lambda_0$ and $Var(X) = \lambda_0 \cdot (1 + \frac{\lambda_0}{\alpha})$ respectively. Therefore, yield for a non redundant chip is, from equation (5) calculated in $k = 0$:

$$Y = P\{X = 0\} = \left(1 + \frac{\lambda_0}{\alpha}\right)^{-\alpha} \tag{6}$$

called negative binomial yield model. In (6) α represents the clustering effect of the mean defect density λ_0 on different chips, a usual value of α adopted in industry is around 2.

III. YIELD ESTIMATION MODELS OF REPAIRABLE ARRAYS

In this section the description of the two yield estimation models is reported. In Literature many ways to evaluate the yield of a redundant chip have been proposed. In particular, the yield of a redundant RAM array has been widely studied [9] [8] [11]. In this section two different yield estimation models for repairable arrays are provided. The models we will compare in the following are based on two different approaches: the first one [7] (Method A) is focused on reducing the computational complexity and makes some approximations while the second one (Method B) is focused on having more accurate results.

A. Method A

In [9] the concept of fault pattern FP has been introduced. Given the average λ_0 failures occurring on a chip, they can be split into the functional effects they cause, $i.e.$ the value λ_0 can be seen as the sum of the possible λ_i failures of F possible fault types (with i = 1,2,..., F).

$$\lambda_0 = \sum_{i=1,\ldots,F} \lambda_i \tag{7}$$

A fault pattern is defined as a vector $FP = (i_1, i_2 \ldots, i_F)$ where i_k is the number of type-k faults. Using the FP the yield of a repairable chip is therefore the probability of having all the fixable FP, *i.e.* if the FP are assumed to be disjoint:

$$Y = \sum_{all\, fixable\, FP_i} Pr\{FP_i\} \tag{8}$$

In [11] an estimation of the yield is given by means of the analysis of a finite state Markov chain representing all the possible chip repair states. The analysis of the yield is made assuming to calculate all the possible fault patterns therefore is an estimation of the yield obtainable with an exhaustive repair algorithm. The approach of method A is composed of two steps, the first step evaluates the Yield as the sum of the probabilities of all repairable fault patterns on the chip with Poisson distribution, the second step introduces the clustering effect by performing the inversion of the result obtained by step one and using the λ_r into a negative binomial distribution function. Let λ be the sum of the average numbers of faults of each type in the eSRAM. So,

$$\lambda = \lambda_{sc} + \lambda_{pv} + \lambda_{hp} + \lambda_{nb} + \lambda_{co} + \lambda_{ck} \tag{9}$$

Let Y_r be the yield of the eSRAM after repairing the memory using the provided redundancy; repair effectively translates into a process by which hopefully all faults can be corrected. Let the number of faults left unrepaired be λ_r and Y_p be the so-called perfect yield which is the probability that there is no fault left unrepaired (i.e. the number of faults to be repaired is λ). Hence,

$$Y_p = P(0) = (1 + \lambda/\alpha)^{-\alpha} \tag{10}$$

Stapper provided a yield model for calculating the yield of a redundant memory after repair [1]; Stapper's model enumerates the probability of successfully repairing all combinations of fault types using the provided redundancy. Let C_F denote the possible combinations of faults. The proposed model also enumerates the probability of all combinations of faults of different types that can be repaired by the provided redundancy (C_F denotes such possible combinations). Assume a Poisson distribution for the faults (no clustering); for k faults of type i,

$$P_i(k) = \frac{e^{-\lambda_i} \lambda_i^{-k}}{k!} \tag{11}$$

So,

$$Y_r = P_{ck}(0) \sum_{C_F} P_{sc}(i) P_{pv}(j) P_{hp}(k) P_{nb}(l) P_{col}(m) \tag{12}$$

In equation 12, Y_r is the repairable yield for (i+j+k+l+m) faults of different types. From Y_r λ_r is obtained as with inversion of $Y_r = e^{-\lambda_r}$ and defined as the number of faults left unrepaired Finally the yield considering the clustering effect is obtained as:

$$Y_r = (1 + \lambda_r/\alpha)^{-\alpha} \tag{13}$$

B. Method B

A repair process is composed of a series of states in which the chip passes depending on the number of faults that must be repaired. This case is like a M out of N reliability problem in which M elements out of N must work typically modeled with Markov modeling. Therefore a repair process can also be modeled by means of a Continuous Time Markov Chain (CTMC) where states are M = number of spare rows + one failure state and one good state 1 .

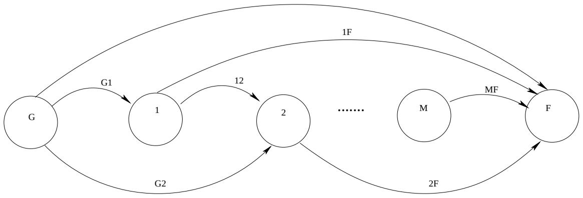

Fig. 1. Markov chain for spare rows case

The rates on the edges are as reported in table III where:

$$\lambda_{pv}(i, i+1) = n_c n_r \lambda_{pv} \left[1 - \left(\frac{i}{n_r}\frac{i-1}{n_r-1} + \frac{n_r-i}{n_r}\frac{n_r-i-1}{n_r-1}\right)\right]$$
$$\lambda_{pv}(i, i+2) = n_c n_r \lambda_{pv} \left(\frac{n_r - i}{n_r} \cdot \frac{n_r - i - 1}{n_r - 1}\right) \tag{14}$$

$\lambda_{G,F}$	$n_c\lambda_{col} + \lambda_{ck}$
$\lambda_{G,1}$	$n_c n_r \lambda_{sc} + n_r \lambda_{ro} + n_c n_r \lambda_{hp}$
$\lambda_{G,2}$	$n_c \lambda_{pv}$
$\lambda_{1,F}$	$n_c\lambda_{col} + \lambda_{ck}$
$\lambda_{1,2}$	$n_c(n_r - 1)\lambda_{sc} + (n_r - 1)\lambda_{ro} + n_c(n_r - 1)\lambda_{hp} + \lambda_{pv}(1,2)$
$\lambda_{1,3}$	$\lambda_{pv}(1,3)$
$\lambda_{2,F}$	$n_c\lambda_{col} + \lambda_{ck}$
$\lambda_{2,3}$	$n_c(n_r - 2)\lambda_{sc} + (n_r - 2)\lambda_{ro} + n_c(n_r - 2)\lambda_{hp} + \lambda_{pv}(2,3)$
$\lambda_{2,4}$	$\lambda_{pv}(2,4)$
$\lambda_{i,F}$	$n_c\lambda_{col} + \lambda_{ck}$
$\lambda_{i,i+1}$	$n_c(n_r - i)\lambda_{sc} + (n_r - i)\lambda_{ro} + n_c(n_r - i)\lambda_{hp} + \lambda_{pv}(i,i+1)$
$\lambda_{i,i+2}$	$\lambda_{pv}(i,i+2)$
$\lambda_{M,F}$	$n_c(n_r - s_r)\lambda_{sc} + (n_r - s_r)\lambda_{ro} + n_c\lambda_{col} + \lambda_{ck} + (n_r - s_r)\lambda_{ro} + n_c(n_r - s_r)\lambda_{hp} + \lambda_{pv}(s_r, s_r + 1)$

TABLE III

DEFECT INTENSITIES

The following assumptions have been made:

1. all the transition are slow $i.e.$ the transitions occur with exponential p.d.f. and the rates(λ) are constant
2. the faults are assumed independent and therefore the rates are summed on each edge

Using these rates the solution of the CTMC is computed as the solution of the following set of differential equations:

$$P'(\lambda) = P(\lambda)A \qquad (15)$$

where $P(\lambda)$ is a vector whose elements are the probabilities of being in sate (G,1,2,...M,F) and A is called generating matrix and its elements are the rates of the transitions

$$\mathbf{A} = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \cdots \\ \lambda_{12} & \lambda_{22} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

The numerical solution of equation 15 is made by considering that its solution can be written as:

$$P(\lambda) = P(0)\, e^{(A\lambda)}$$

considering $\Delta\lambda \ll 1$ the above equation can be written as:

$$P(\Delta\lambda) = P(0)(I - A\Delta\lambda)$$

and therefore in general for any given λ

$$P(\lambda + \Delta\lambda) = P(\lambda)(I - A\Delta\lambda)$$

that can be solved numerically. It has to be noticed that this approach requires a matrix multiplication for each step. Once the Yield is calculated as the probability of not being in the fail state the obtained result is then numerically integrated with the gamma distribution function in order to obtain the actual yield including clustering effect. Hence, the performed operation is:

$$Y_t(\lambda_0,\alpha) = \int_0^\infty Y(\lambda)\cdot g(\lambda,\lambda_0,\alpha)\, d\lambda \simeq$$

$$= \sum_{n=0}^{n=large} Y(n\Delta\lambda)\cdot g(n\Delta\lambda,\lambda_0,\alpha)\,\Delta\lambda \qquad (16)$$

where $g(\lambda,\lambda_0,\alpha)$ is the gamma probability distribution function. The gamma probability distribution function is defined as [13]

$$g(x) = \gamma x^{b-1} e^{-cx} U(x) \qquad (17)$$

where $\gamma = \frac{c^b}{\Gamma(b)}$, $U(x)$ is the step function, b and c are positive numbers, and $\Gamma()$ is the gamma function defined as :

$$\Gamma(b+1) = \int_0^\infty y^b e^{-y} dy$$

with $b > -1$. $\Gamma()$ is also called generalized factorial because $\Gamma(b+1) = b\Gamma(b)$ and if b is an integer $\Gamma(b+1) = b!$ being $\Gamma(1) = 1$. Using this definition and assuming for the parameters x,,c b the following values: $x = \lambda$ $c = \frac{\alpha}{\lambda_0}$ and finally $b = \alpha$ the following formulation of the gamma distribution function found in literature [9] is then obtained:

$$g(\lambda,\lambda_0,\alpha) = \frac{\alpha^\alpha}{\lambda_0^\alpha \cdot \Gamma(\alpha)} \lambda^{\alpha-1} e^{-\frac{\alpha}{\lambda_0}\lambda}$$

IV. COMPARISON OF THE YIELD ESTIMATION METHODS

In this section is provided a comparison between the yield estimation obtained with the two methods described above.

- Method A: Markov model (step 1) + numerical gamma function integration (step 2)
- Method B: Fault pattern (step 1) + inversion approximated solution (step 2)

Both methods are composed of two steps, the first one calculates the yield assuming to have a Poisson distributed failure probability and the second step introduces suitable modifications to the values obtained in the first step in order to take into account the clustering effect. The above methods have been compared in both steps by calculating the obtained yield on different size RAM arrays. In particular the sizes chosen have been 0.5 Mbit, 1 Mbit, 2 Mbit. For each of these sizes the cases of 0, 1, 2, 4 and 8 spare rows have been considered.

from the above reported results we can see that the first steps of the considered methods give close values. Also, the plot reported in fig 2 shows that the yield values obtained by two different methods overlap for every value of λ.

TABLE IV

ARRAY SIZE 0.5M (1024 ROWS 512 COLUMNS),λ 0.1 STEP 1

	Method A	Method B	Difference (B-A)
no spare	9.0438208e-01	9.0483742e-01	4.5534e-04
one row	9.6833027e-01	9.6817604e-01	-1.5423e-04
two rows	9.7949935e-01	9.7944126e-01	-5.8090e-05
four rows	9.8017626e-01	9.8019469e-01	1.8430e-05
eight rows	9.8017904e-01	9.8019867e-01	1.9630e-05

TABLE V

ARRAY SIZE 1M (1024 ROWS 1024 COLUMNS),λ 0.2 STEP 1

	Method A	Method B	Difference (B-A)
no spare	8.1790694e-01	8.1873075e-01	8.2381e-04
one row	9.3357813e-01	9.3335306e-01	-2.2507e-04
two rows	9.5786854e-01	9.5775123e-01	-1.1731e-04
four rows	9.6072255e-01	9.6075544e-01	3.2890e-05
eight rows	9.6075096e-01	9.6078944e-01	3.8480e-05

The results of the outputs of step 2 of both methods are compared in tables (VII VIII IX) . These tables show that computed yields are very close using both methods for the considered λ values. In all considered cases the results obtained with method B are slightly higher than those obtained with method A. However, if we consider the plot of the yield comparing both approaches, we can see that for higher values of lambda the approach B always underestimates the yield as shown in figure 3.

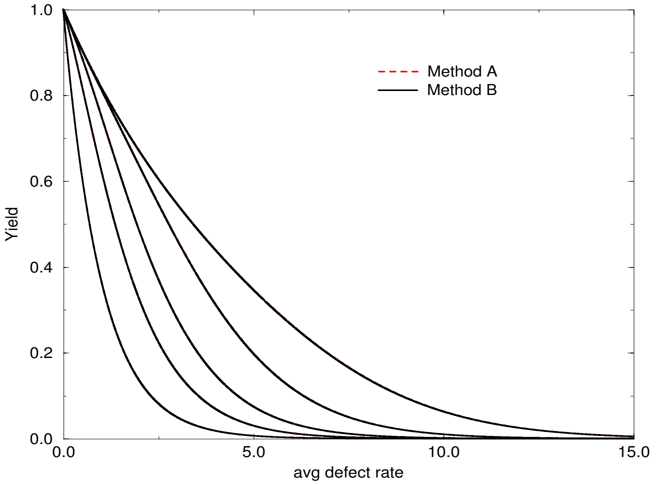

Fig. 2. Step 1: yield comparison varying λ

TABLE VII

ARRAY SIZE 0.5M (1024 ROWS 512 COLUMNS),λ 0.1 STEP 2

	Method A	Method B	Difference (B-A)
no spare	9.0702483e-01	9.0702948e-01	4.6500e-06
one row	9.6750202e-01	9.6842654e-01	9.2452e-04
two rows	9.7916222e-01	9.7954620e-01	3.8398e-04
four rows	9.8028259e-01	9.8029211e-01	9.5200e-06
eight rows	9.8029552e-01	9.8029605e-01	5.3000e-07

V. CONCLUSIONS

In this paper two methods for calculating the yield of repairable RAM arrays have been compared. The analyzed methods have been chosen to be representative of two different approaches. Method A is the basis of the CAYA (Compiler-based Array Yield Analysis) tool described in [7]. It is simple and its computation is fast but it is based on approximations. Method B is much more complex and the computation time is higher but the results are accurate. Both methods have been divided into two steps; the intermediate and final results have been compared. Results are shown to be very close for values of low lambda and of memory size coming from manufacturing lines. For higher lambda values Method A underestimates the yield even if the results of the first steps are still close. Therefore the first steps of both computations are mostly equivalent. To obtain accurate results for higher values of lambda, the second step of Method B should be used.

TABLE VI

ARRAY SIZE 2M (2048 ROWS 1024 COLUMNS),λ 0.4 STEP 1

	Method A	Method B	Difference (B-A)
no spare	6.6897176e-01	6.7032005e-01	0.0013
one row	8.5818863e-01	8.5800966e-01	-1.7897e-04
two rows	9.1130299e-01	9.1109901e-01	-2.0398e-04
four rows	9.2276958e-01	9.2281806e-01	4.8480e-05
eight rows	9.2304236e-01	9.2311629e-01	7.3930e-05

REFERENCES

[1] C. H. Stapper, A. N. Mclaren and M. Dreckmann, “Yield model for productivity optimization of VLSI memory chips with redundancy and partially good product” IBM J. Res. Develop., vol. 24, no. 3, pp. 309-409, 1980

[2] B.Cowan, O.Farnsworth, P. JaKobsen, S.Oakland, M.R. Oulette, D.L.Wheater, ”On-Chip Reapair and an ATE Independent Fusing Methodology”, in *IEEE Intern. Test Conf.*, pp.178-186, 2002.

[3] Y.Nagura, M.Mullins, A.Sauvageau, Y.Fujiwara, K.Furue, R.Ohmura, T.Komoike, T.Okitaka, T.Tanizaki, K.Dosaka, Y.Koda, T.Tada, ”Test cost reduction bi At-Speed BISR for Embedded DRAMs”, in *IEEE Intern. Test Conf.*, pp.182-187, 2001.

[4] I. Kim, Y. Zorian, G. Komoriya, H. Pham, F. P. Higgins, J. L. Lewandowski, “Built In Self Repair for Embedded High Density SRAM”, in *IEEE Intern. Test Conf.*, pp.1112-1118, 1998.

TABLE VIII

ARRAY SIZE 1M (1024 ROWS 1024 COLUMNS),λ 0.2 STEP 2

	Method A	Method B	Difference (B-A)
no spare	8.2643868e-01	8.2644628e-01	7.6000e-06
one row	9.3164165e-01	9.3443883e-01	0.0028
two rows	9.5670597e-01	9.5819119e-01	0.0015
four rows	9.6105559e-01	9.6113543e-01	7.9840e-05
eight rows	9.6116827e-01	9.6116878e-01	5.1000e-07

TABLE IX

ARRAY SIZE 2M (2048 ROWS 1024 COLUMNS),λ 0.4 STEP 2

	Method A	Method B	Difference (B-A)
no spare	6.9443285e-01	6.9444444e-01	1.1590e-05
one row	8.5649589e-01	8.6281062e-01	0.0063
two rows	9.0800183e-01	9.1301623e-01	0.0050
four rows	9.2361375e-01	9.2426897e-01	6.5522e-04
eight rows	9.2455311e-01	9.2455615e-01	3.0400e-06

[5] S. Tsuchida, ”Test and Repair of Non-Volatile Commodity and Embedded Memories”, in *IEEE Intern. Test Conf.*, p.1223, 2002.

[6] “International roadmap for semiconductors”, Technical report, SIA, http://public.itrs.net/, 2001.

[7] X.Wang, M. Ottavi, F. Lombardi, “Yield Analysis of Compiler-based Arrays of Embedded SRAMs” in *Transactions of 2003 IEEE Defect and Fault Tolerance in VLSI System*, Cambridge (MA), Nov. 2003.

[8] I. Koren and Z. Koren “Defect Tolerant VLSI Circuits: Techniques and Yield Analysis” , Proceedings of the IEEE, Col 86, pp. 1817-1836, Sept. 1998.

[9] C. H. Stapper, A. N. Mclaren and M. Dreckmann, “’Yield model for productivity optimization of VLSI memory chips with redundancy and partially good product” IBM J. Res. Develop., vol. 24, no. 3, pp. 309-409, 1980

[10] C. H. Stapper “Modeling of Integrated Circuit defect sensitivities” IBM J. Res. Develop., vol.27, no. 6, pp. 549-557, 1983

[11] B. Ciciani, G. Iazeolla “A Markov Chain-Based Yield Formula for VLSI Fault-Tolerant Chips” IEEE Transactions on computer-aided design, vol. 10, no. 2, pp. 252-259, Feb. 1991

[12] G. Battaglini, B. Ciciani, “An improved analytical yield evaluation method for redundant RAM’s” Proc. International Workshop on Memory Technology, Design and Testing, pp. 117-123, 24-25 Aug. 1998

[13] A. Papoulis “Probability, Random Variables and Stochastic Processes” Mc-Graw Hill

[14] Butler, Ricky W.: ”The SURE Approach to Reliability Analysis.” IEEE Transactions on Reliability, vol. 41, no. 2, June 1992, pp. 210–218.

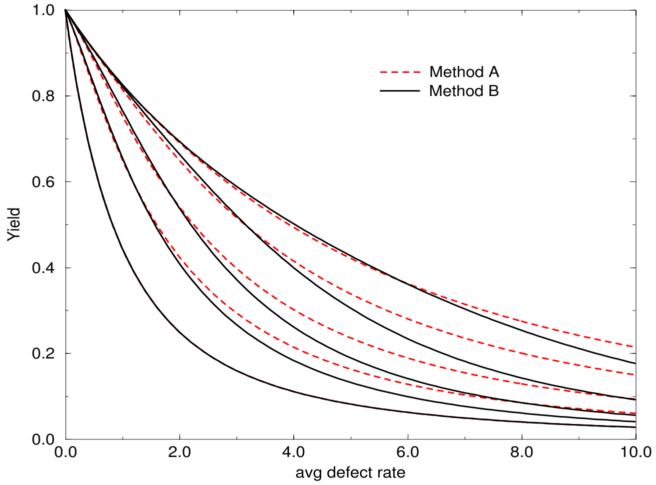

Fig. 3. Step 2: yield comparison varying λ