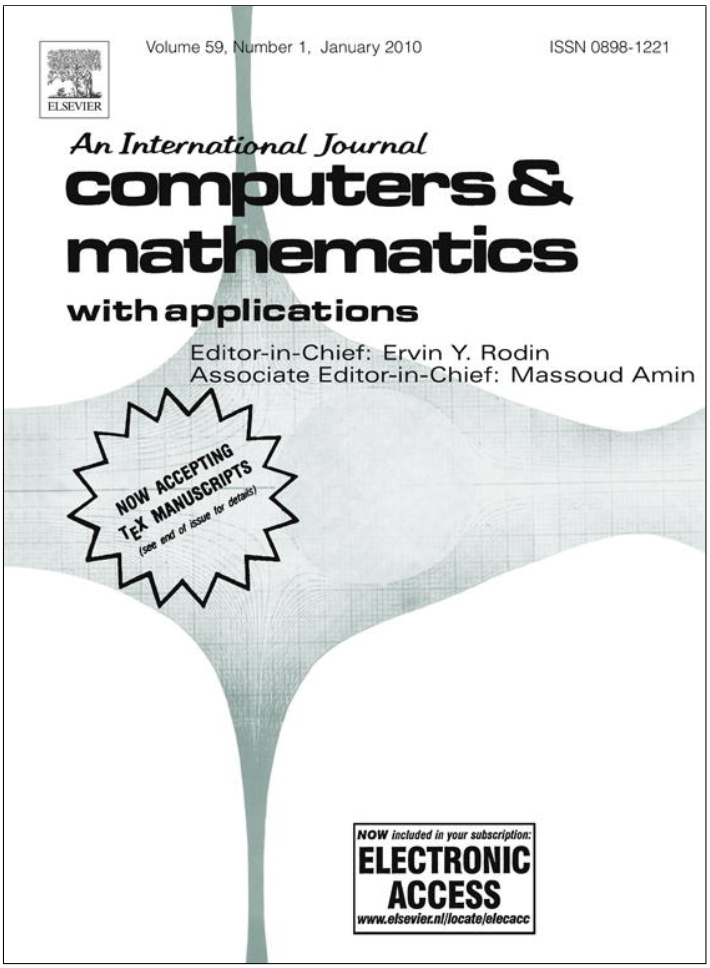

Computers and Mathematics with Applications 59 (2010) 98–107

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Yield analysis of partial defect tolerant bit-plane array

Vladimir Ćirić [a,*], Aleksandar Cvetković [b], Ivan Milentijević [a]

[a] Computer Science Department, Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, P.O.Box 73, Niš, Serbia

[b] Faculty of Sciences and Mathematics, University of Niš, Višegradska 33, P.O.Box 224, Niš, Serbia

ARTICLE INFO

Article history:
Received 25 February 2009
Received in revised form 28 July 2009
Accepted 26 August 2009

Keywords:
Partial defect tolerance
Error tolerance
Effective yield
Systolic arrays

ABSTRACT

Silicon complexity places long-stand paradigms at risk. Key concerns include increasing process variations, defect rates, infant mortality rates, and susceptibility to internal and external noises. These trends are likely to decrease functional yield. Fabrication of die with 100% working transistors and interconnections becomes prohibitively expensive. This paper examines the size and the position of the candidate part of the architecture for defect tolerance application, for the given topology and defect probability where yield can be improved in comparison to error tolerant design. In order to achieve the mentioned goal, we modified the existing mathematical description of yield by involving error tolerant concept introducing a function $\Gamma(\alpha)$ that models the topology of architecture. The evaluation is demonstrated on the bit-plane semi-systolic array, as a relatively complex array topology. The method that we hereby present for the chosen topology is described and proved in formal mathematical way, and it easily covers simpler topologies. It will be shown that partial involvement of defect tolerant design can significantly improve effective yield for defect rates which are common in nanotechnology.

1. Introduction

As device size continues to shrink, it approaches the scale of individual atoms and molecules. With atom spacing in a silicon lattice around 0.5 nm, 65 nm drawn features are a little more than 100 atoms wide. Key features, such as gate lengths, are effectively half or a third this size. Continued geometric scaling will take us to the realm where feature sizes are measured in single-digit atoms sometime in the next couple of decades (see [1]). Key concerns include increasing process variations, defect rates, infant mortality rates, and susceptibility to internal and external noises (see [2], [3]). These trends are likely to decrease functional yield. As VLSI scaling continues along its traditional path, we will soon be in a situation where chips will have billions of devices and thousands of defects (see [4,5]). Fabrication of die with 100% working transistors and interconnections becomes prohibitively expensive (see [6]). While scaling approaches the physical limits of devices and fabrication technology, designers will increasingly have to consider qualitative changes.

Fault tolerance (FT) is the ability of a system to continue correct operation of its tasks after hardware or software faults occur (see [1]). Correct operation typically implies that no errors occur at any system output. Other FT definitions replace the word correct with satisfactory or reliable. Defect tolerance (DT) refers to any circuit implementation that provides higher yield than an implementation that is not defect tolerant, for a given level of defects and process variations. Enhancements in this category include redundancy (often in the form of spares) as well as defect avoidance, in the form of layout and circuit design techniques that reduce circuits' sensitivity to fabrication defects and process variations (see [1,6,7]).

* Corresponding author at: Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, P.O.Box 73, Niš, Serbia. Tel.: +38 118529603; fax: +38 118588399.

E-mail addresses: vladimir.ciric@elfak.ni.ac.yu (V. Ćirić), aleksandarcvetkovic@pmf.ni.ac.yu (A. Cvetković), milentijevic@elfak.ni.ac.yu (I. Milentijević).

0898-1221/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2009.08.068

Silicon complexity places long-stand paradigms at risk (see [6]). The concept of building useful computational systems with parts that might be initially defective, experience externally induced transient errors, or eventually develop a permanent lifetime fault is not new. Researchers addressed these problems as far back as the 1940s with the work of von Neumann, Gödel, and Klein, continuing with the emergence of fault tolerant computing in the 1960s and, more recently, the field of defect tolerance (see [8]). In many applications, however, certain types of errors at the entire systems outputs might be acceptable, provided their severities are below the given thresholds (see [6,9,10]). Such systems are called Error Tolerant (ET) systems. Multimedia applications are one example of ET systems. In multimedia, designers take advantage of the signal processing ability of people to convert the original source of signals to lower quality packets of information, since this usually provides acceptable performance to the end user, reduced bandwidth and hardware costs. An interesting question is: if some signal processing device has a minor hardware defect, will it still produce results that are good enough for the end user? If so, they could also be sold rather than be discarded (see [9]). Relaxing the requirement of 100% correctness for devices and interconnections may dramatically reduce costs of manufacturing, verification, and testing (see [2]).

Defined by output error thresholds, the most significant part of ET systems can be further designed to be DT, resulting in Partial Defect Tolerant (PDT) systems, in contrast to Full Fault Tolerant (FFT) systems, where DT (or FT) design is applied to the system as a whole (see [10,11]). As stated in [10], PDT design is preferred in comparison to FFT design. However, the introduced silicon overhead for PDT system implementation may not improve fabrication cost per die when defect probability is small. The reduced number of rejected defective chips can be so small that the introduced silicon overhead for PDT implementation is not justified. The goal of this paper is to examine the size and the position of defect tolerant part in the array, for the given topology and defect probability, where yield can be improved in comparison to error tolerant design. In order to achieve the mentioned goal, we modified mathematical description of yield from [1] by involving error tolerant concept introducing a function $\Gamma(\alpha)$ that depends on array topology. The evaluation will be demonstrated on the bit-plane semi-systolic array, as a relatively complex array topology. The method that we hereby present for the chosen topology is described and proved in formal mathematical way, and it easily covers simpler topologies. In the paper, dependency between the yield that can be achieved using PDT array and defect probability is mathematically formulated and compared with ET design. The size and position of candidate array partition for defect tolerance application will be given as a function of acceptable error magnitude, and used in yield evaluation. It will be shown that PDT design can significantly improve effective yield for defect rates, which are common in nanotechnology.

The paper is organized as follows: Section 2 gives a brief architectural and design overview of the PDT bit-plane array, Section 3 is devoted to the yield analysis for partial defect tolerant design, Section 4 discusses error significance of the bit-pane topology, in Section 5 we give the evaluation of the partial defect tolerance at the example of the bit-plane array, while in Section 6 the concluding remarks are given.

2. Partial defect tolerance

With the aim to clarify the yield analysis, we give a brief overview of the PDT and illustrate the basic concepts using the example of bit-plane FIR filtering array.

Output words $\{y_i\}$ of an FIR filter are computed as

$$y_i = c_0 x_i + c_1 x_{i-1} + \cdots + c_{k-1} x_{i-k+1}, \tag{1}$$

where $c_0, c_1, \ldots, c_{k-1}$ are coefficients while $\{x_i\}$ are input words. Computation (1) can be realized in different manners. When high performances are required systolic arrays are frequently used. Semi-systolic arrays share with systolic arrays not only the desirable simplicity and regularity properties, but also pipelining and multiprocessing schemes of operation.

Bit-plane (BP) FIR filter is a semi-systolic architecture with bit-plane operations (see [12,13]). It provides regular connections with extensive pipelining and high computational throughput (see [12–14]). A functional block diagram of a BP array is shown in Fig. 1. The following notation is adopted: m — coefficient word length; k_C — number of coefficients $(c_0, c_1, \ldots, c_{k_C-1})$; n — input word length; c_i^j — bit of coefficient c_i (with weight 2^j); $c_i \equiv c_i^{m-1} c_i^{m-2} \cdots c_i^0$, where $c_i^0, c_i^1, \ldots, c_i^{m-1}$ are the bits of coefficient c_i with weights $2^0, 2^1, \ldots, 2^{m-1}$, respectively; $c^j \equiv c_{k-1}^j c_{k-2}^j \ldots c_0^j$, where $c_0^j, c_1^j, \ldots, c_{k-1}^j$, are the bits with weight 2^j of coefficients $c_0, c_1, \ldots, c_{k-1}$, respectively; l_0 — the number of basic cells within one row of a BP array; y_i^j — the bit of output word y_i with weight 2^j.

There are m bit-plane elements that form the array shown in Fig. 1. Each BP (Fig. 1) is formed as a set of k_C rows. A row performs the basic multiply-accumulate operation between the intermediate result from the previous row and the product of the input word and one coefficient bit. Delayed for one clock cycle per row, the output word is available after $k_C \cdot m$ clock cycles (see [12,13]).

Fig. 2 shows the bit-plane array from Fig. 1 redrawn in such a manner that all connections between cells become regular (see [10]). Regular connections enable simplification of error significance maps development for the array (see [10]). Regularity is achieved by introducing a set of fictive nodes, represented by circular nodes in Fig. 2. In order to clarify the PDT architecture design, we will define areas of interest for PDT design and illustrate them on the array from Fig. 2.

Denote the number of architecture outputs L ($L = l_0 + m$ in the case of bit-plane array).

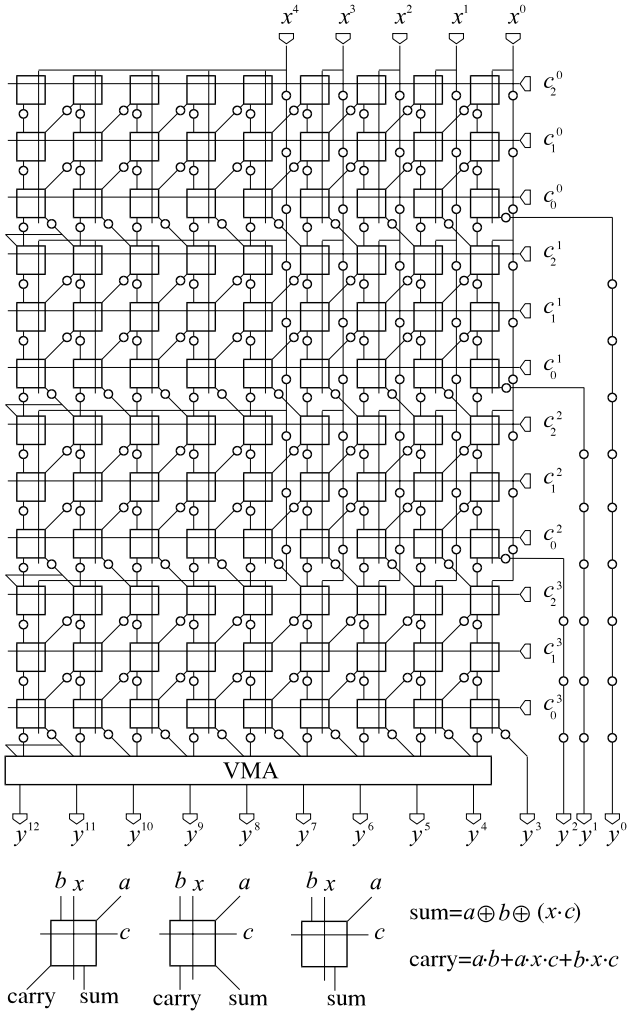

Fig. 1. The BP array for $k_C = 3$ and $m = 4$.

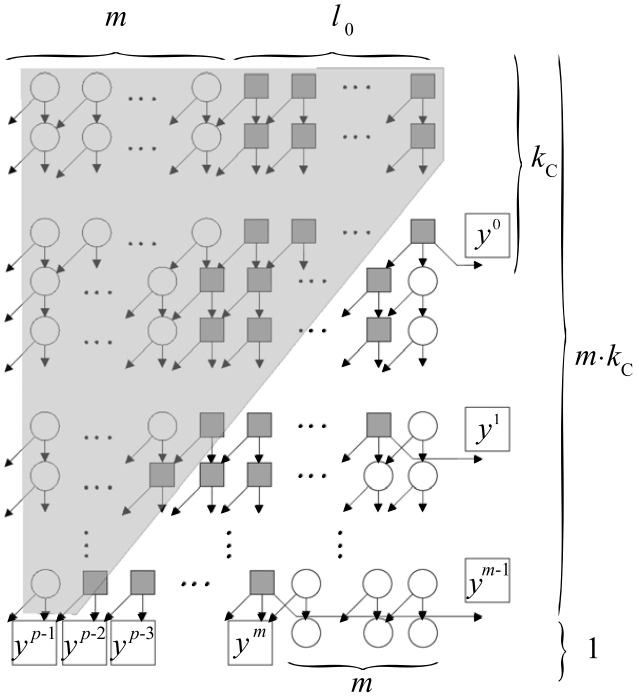

Fig. 2. BP array with regular connections.

Definition 1 (*Error Tolerance*). The error tolerance is a design concept which allows the architecture to have defects that can produce an error Δ below the given threshold ψ at the system output

$$\Delta \leq \psi = 2^{L-\alpha}.$$

In the previous α is the number of the most significant output bits not prone to errors.

Fabricated dies with errors, such as $\Delta \leq \psi$, can rather be used than discarded. However, for given ψ, the errors where $\Delta > \psi$ cannot be tolerated.

Definition 1 represents Euclidean distance between correct and erroneous output signal, and shows the architecture tolerance to errors. However, Euclidean distance, in the case of the BPA from Fig. 2, can be misleading.

Euclidean distance is a suitable metric for the abstraction of signals as error sources. For example, if the magnitude of the correct output signal is $011\cdots111$, and the error within a signal is small, eq. $000\cdots001$, the output result will be $100\cdots000$. However, this seems to be an error in all bits of the output result. Observing the particular cell from Fig. 2 as an error source, the cell has bounded influence on the output result. The boundaries of influence are defined by topology of interconnections, thus, the metric that can abstract the error as a function of position is Hamming distance, rather than Euclidean, which is defined as

$$\Delta_H = \sum_{i=L-\alpha}^{L-1} (y^i_{corr} \oplus y^i_{err}), \tag{2}$$

where y^i_{corr} and y^i_{err} are the bits with weight 2^i from correct, and erroneous result, respectively.

Definition 2 (*Minor Defect*). We assume that architecture has a minor defects, and that it produces the acceptable output result if, for given α, it contains errors for which

$$\Delta_H = 0.$$

For a given architecture the parts of the system that are not prone to errors are defined by the specific application (see [6]).

Definition 3 (*Error Significance Set*). The set of cells, or more general, the set of subsystems of the architecture, that can induce an error in the output y^η, is termed the error significance set of the output y^η and is denoted $\mathbf{M}_\eta$ (see [10]).

As given in Definition 2, $\alpha(\alpha \leq L)$ is the number of the most significant outputs of ET system not prone to errors ($y^{L-1}, y^{L-2}, \ldots, y^{L-\alpha}$ in Fig. 2).

Definition 4 (*PDT System*). Partial defect tolerant system is an ET system where subsystems, which can induce error ($\Delta_H > 0$), are defect or fault tolerant (see [10]).

Definition 5 (*Architecture Partitioning*). The Non-Tolerant Area (NTA) of the architecture is the function P_{DT} from the set $\{0, 1, \ldots, L-1\}$ into the system subsets, so that

$$P_{DT}(\alpha) = \bigcup_{\eta=L-\alpha}^{L-1} \mathbf{M}_\eta, \quad \alpha \in \{0, 1, \ldots, L-1\}. \tag{3}$$

The rest of the system is called Error Tolerant Area (ETA), and is denoted by ($\overline{P_{DT}}$). The shaded area in Fig. 2 shows the NTA of the bit-plane array for $\alpha = 2$, while the non-shaded area represents the ETA.

In accordance with Definitions 4 and 5, ET system becomes PDT system by making NTA defect tolerant.

3. Error tolerance vs. partial defect tolerance

Let C be the fabrication cost of one defective or non-defective die, and let p be a probability of having a defective subsystem. We call a die "usable die" if it is non-defective, or has minor defects, in respect to Definition 2, which can be tolerated by the application. In order to compare the error tolerant and partial defect tolerant design methods, we define the price per usable die.

Definition 6 (*Price Per Usable Die*). For a fabricated die without defects or with a minor defect, in respect to the Definition 2, the price per usable die is a function $U(p, \alpha)$, from the set $\{(p, \alpha) \mid p \in [0, 1] \wedge \alpha \in \{0, 1, \ldots, L-1\}\}$ to positive real numbers, so that

$$U(p, \alpha) = u \cdot C,$$

where u ($u \geq 1$) is the factor that depends on defect probability p as well as the system geometry.

If the probability of having defect is $p = 0$, the price for fabrication of usable die is equal to the price of production of one die, i.e., for $p = 0$ we have $u = 1$ regardless of geometry. However, if the probability of having defect is $p \neq 0$, the price of fabrication of one usable die is greater than C. We call the factor u from Definition 6 fabrication yield.

Definition 7 (*Yield*). We define fabrication yield Y of the design as

$$Y(p, \alpha) = \frac{C}{U(p, \alpha)}.$$

According to Definition 6, the yield is defined as a comparison parameter scaled into the interval $Y \in [0, 1]$, where $Y = 1$ means that there are no defective dies, and $Y = 0$ indicates that there is no usable die. Value $Y = 1/2$ tells that 2 dies should be produced in order to have 1 usable die.

Let $\Gamma(\alpha)$ be the probability that the given subsystem belongs to NTA, where α is the number of the most significant outputs required to be error-free. Straightforward calculation gives the following simple lemma.

Lemma 1. *If T is a total number of subsystems, and p is probability of having a defective subsystem, the cost for fabrication of one non-defective die is*

$$U^{ET}(p, \alpha) = \frac{C}{(1-p)^{\Gamma(\alpha)\cdot T}}. \tag{4}$$

Lemma 2. *Yield of ET system is:*

$$Y_{ET}(p, \alpha) = \frac{C}{U^{ET}(p, \alpha)} = (1-p)^{\Gamma(\alpha)\cdot T}. \tag{5}$$

Proof. From (4) and Definition 7, Eq. (5) is obtained directly. □

In order to make the NTA more prone to defects, PDT design requires replacement of system's submodules with DT modules in the NTA area, thus new system cost C' differs from the cost of ET system, variable C in (4). Furthermore, depending on the chosen DT method, the probability of having a non-defective cell is more complex than $(1 - p)$, as given in (4), and it is denoted as R.

Lemma 3. *The cost of a usable die designed using PDT architecture paradigm is*

$$U^{PDT}(p, \alpha) = \frac{C'}{R^{\Gamma(\alpha)\cdot T}}. \tag{6}$$

Lemma 4. *The yield of PDT system is*

$$Y_{PDT}(p, \alpha) = \frac{C}{U^{PDT}(p, \alpha)} = \frac{C}{C'} \cdot R^{\Gamma(\alpha)\cdot T}. \tag{7}$$

Proof. Eq. (7) is derived using (6) and Definition 7. □

We illustrate the previous concepts using the following example. Let the chosen defect tolerant method in the NTA system area be Spare Components with 2 spares (SC3), see [1,7]. Then, instead of having one cell in NTA, there are three cells, i.e., the cell and two spares. The overall cost of the system is

$$C' = \Gamma(\alpha) \cdot 3 \cdot C + (1 - \Gamma(\alpha)) \cdot C = (1 + 2\Gamma(\alpha))C, \tag{8}$$

while the probability of having non-defective SC3 DT cell is equal to the probability that there is at least one (of two) defect-free spares (see [6,7]), if the "original" cell is defective

$$R = (1-p)^3 + 3(1-p)^2 p + 3(1-p)p^2. \tag{9}$$

Substituting (8) and (9) into (7), we get the yield of PDT system. The yields of ET system (5) and SC3 case of PDT system (7) are shown in Fig. 3. It is assumed that $T = 168$, as in Fig. 1, and $\Gamma(\alpha) = 0.5$.

Fig. 3 shows that both yields decrease when the probability of having defective cell increases. For the case from Fig. 3, when $p = 0$ yield of PDT system is $Y_{PDT} = 0.5$. This comes from the fact that we involved the additional spare components in Eq. (8), so the price per die, for given parameters, get doubled. At the same time, the yield of ET system is $Y_{ET} = 1$, meaning that there is no additional overhead, and for the given probability the price per usable die equals the price of producing one die, as defined in Definition 7. However, there is p from which the yield of the PDT system becomes more preferred than ET.

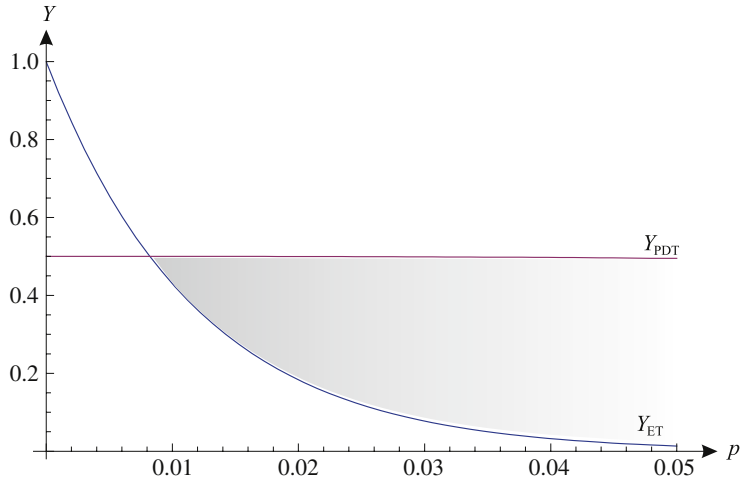

Fig. 3. Comparison of fabrication costs of one non-defective die for ET and PDT system, assuming that $\Gamma(\alpha) = 0.5$, and price per die (defective or non-defective) is $C = 1$. The total number of cells $T = 168$ is the same as in the example from Fig. 1 (see [10]).

Table 1
U^{ET} and U^{PDT} for p varying from $p = 0$ to $p = 0.010$.

p	0	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009
U^{ET}	1.	1.087	1.183	1.287	1.400	1.523	1.657	1.804	1.963	2.137
U^{PDT}	2.	2.000	2.002	2.004	2.008	2.012	2.018	2.024	2.032	2.041

That probability depends on the given system's geometry, which shapes the function $\Gamma(\alpha)$ and the degree of error tolerance of the application α.

As an illustration, Table 1 gives the costs per usable die for yields shown in Fig. 3. If the probability of having defective subsystem is 0.003, in the case of ET, 1.287 dies should be produced to obtain one non-defective die. For the same probability, 2.004 dies should be produced in the case of PDT, making the ET design preferred. If the probability p is greater than 0.008, PDT becomes preferred. The shaded area in Fig. 3 shows where PDT is preferred.

4. Error significance of the bit-plane array

In order to define the intersection of Eqs. (5) and (7), and to calculate the probability starting from which the PDT improves yield for bit-plane array, the function $\Gamma(\alpha)$ has to be obtained.

Theorem 5. *The probability that the given subsystem belongs to the NTA of the bit-plane array is given by*

$$\Gamma(\alpha) = \frac{\aleph(P_{DT}(\alpha))}{mk_CL} = \begin{cases} 0, & \alpha = 0 \\ \dfrac{(m \cdot k_C + 3)}{2L}, & \alpha = 1 \\ \dfrac{k_C^2m^2 + k_Cm - 2 + \alpha(2k_Cm + 3) - \alpha^2}{2 \cdot m \cdot k_C \cdot L}, & \text{other} \end{cases} \tag{10}$$

where $\aleph(A)$ represents the cardinal number or cardinality of a set A (see [15, p. 28]).

Proof. From (3), function $\Gamma(\alpha)$ can be obtained as follows

$$\Gamma(\alpha) = \frac{\aleph(P_{DT}(\alpha))}{T}, \tag{11}$$

where T is total number of array cells ($T = m \cdot k_C \cdot L$). The sets $\mathbf{M}_\eta$ in (3) have common elements (Fig. 2). Henceforth, in order to calculate the cardinality of (3), from the cardinality of the individual sets, that fact has to be considered. The function $\aleph(P_{DT}(\alpha))$ can be obtained from (3) as follows

$$\begin{aligned} \aleph(P_{DT}(\alpha)) &= \aleph(\mathbf{M}_{L-1}) + \aleph(\mathbf{M}_{L-2} \setminus \mathbf{M}_{L-1}) + \aleph\left(\mathbf{M}_{L-3} \setminus \left(\mathbf{M}_{L-1} \bigcup \mathbf{M}_{L-2}\right)\right) + \aleph\left(\mathbf{M}_{L-\alpha} \setminus \bigcup_{i=L-1}^{L-\alpha-1} \mathbf{M}_i\right) \\ &= \sum_{j=L-1}^{L-\alpha} \aleph\left(\mathbf{M}_j \setminus \left(\bigcup_{i=L-1}^{j-1} \mathbf{M}_i\right)\right), \end{aligned} \tag{12}$$

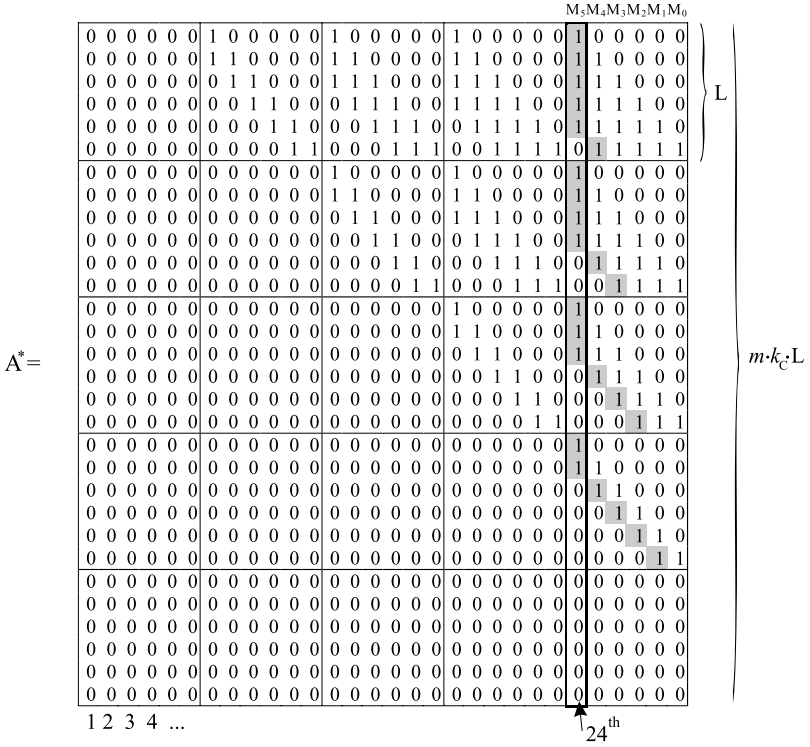

Fig. 4. Transitive closure $\mathbf{A}^*$ for bit-plane array with $k = 2$, $m_C = 2$, $l_0 = 4$.

where $A \setminus B$ denotes the set with elements which belong to the set A, and does not belong to the set B. In other words, $\aleph(P_{DT}(\alpha))$ equals the number of array cells that can induce error in the most significant output bit y^{L-1} plus the number of the array cells with influence to the output bit y^{L-2} without cells that are already taken into consideration.

Cardinality of individual sets $\mathbf{M}_\eta$ can be obtained from transitive closure of a directed graph that represents the architecture from Fig. 2. Using results from [10], transitive closure of the bit-plane array from Fig. 2 is given by

$$\mathbf{A}^* = \begin{bmatrix} 0 & G_C & G_C^2 & \dots & G_C^{m\cdot k_C} \\ 0 & 0 & G_C & \dots & G_C^{m\cdot k_C - 1} \\ & & & \vdots & \\ 0 & 0 & 0 & \dots & G_C \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \tag{13}$$

where elements $G_C = \{g_{i,j}^C\}$ are

$$(g_{i,j}^C)^d = \begin{cases} 1, & j + d \geq i \geq j \\ 0, & \text{other.} \end{cases} \tag{14}$$

Let dimensions of the bit-plane array be $k_C = 2$, $m = 2$, $l_0 = 4$ and $L = 6$. There are $L \cdot (k_C + m + 1) = 30$ nodes within the array, including output nodes. Hence, the transitive closure is a 30×30 matrix (example taken from [10]). The transitive closure for such a small-size array is given in Fig. 4. The 24th column (shaded column in Fig. 4) corresponds to the error significance set $\mathbf{M}_5$ of output bit y^5 (see [10]), which implies that the number of elements equal to 1 within the column is equal to the cardinal number $\aleph(\mathbf{M}_5)$. According to (12) we have $\aleph(P_{DT}(1)) = \aleph(\mathbf{M}_5)$, which can be obtained from (13) and Fig. 4 as

$$\aleph(P_{DT}(1)) = \sum_{i=2}^{m\cdot k_C+1} i = \frac{m \cdot k_C \cdot (m \cdot k_C + 3)}{2}. \tag{15}$$

In addition to the array cells included in (15), there are $m \cdot k_C$ cells that should be included when calculating $\aleph(P_{DT}(2))$, (column $\mathbf{M}_4$ in Fig. 4)

$$\begin{aligned} &\aleph(P_{DT}(2)) = \aleph(P_{DT}(1)) + m \cdot k_C \\ &\aleph(P_{DT}(3)) = \aleph(P_{DT}(2)) + m \cdot k_C - 1 \\ &\dots \\ &\aleph(P_{DT}(L)) = \aleph(P_{DT}(L-1)) + m \cdot k_C - (L-2). \end{aligned} \tag{16}$$

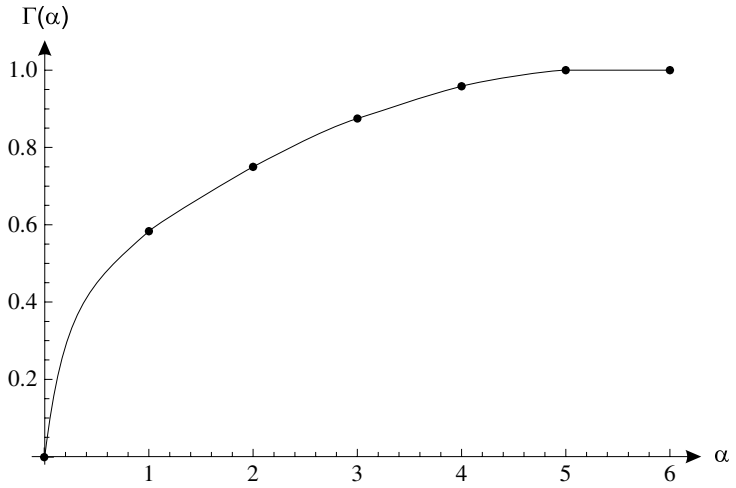

Fig. 5. $\Gamma(\alpha)$ of the bit-plane FIR filtering array for $m = 2$, $k_C = 2$, $L = 6$.

From (15) and (16), for $\alpha > 1$, we have

$$\begin{aligned}\aleph(P_{DT}(\alpha)) &= \sum_{i=2}^{m\cdot k_C+1} i + (\alpha-1)\cdot m\cdot k_C - \sum_{i=1}^{\alpha-2} i \\ &= \frac{m\cdot k_C\cdot(m\cdot k_C+3)}{2} + (\alpha-1)\cdot m\cdot k_C - \frac{(\alpha-1)^2-\alpha+1}{2} \\ &= \frac{k_C^2m^2+k_Cm-2+\alpha(2k_Cm+3)-\alpha^2}{2}. \end{aligned} \tag{17}$$

Eqs. (15) and (17) give

$$\aleph(P_{DT}(\alpha)) = \begin{cases} 0, & \alpha = 0 \\ \dfrac{m\cdot k_C\cdot(m\cdot k_C+3)}{2}, & \alpha = 1 \\ \dfrac{k_C^2m^2+k_Cm-2+\alpha(2k_Cm+3)-\alpha^2}{2}, & \text{otherwise} \end{cases} \tag{18}$$

Finally, substitution of (18) into (3) proves the theorem. □

Let us illustrate the previous theorem. From (18), numerical values for the cardinality of $P_{DT}(\alpha)$, $\alpha = 0, 1, 2, \ldots, 6$, with $m = 2$, $k_C = 2$, $L = 6$, for example given in Fig. 4, are

$$n_i \in \aleph(P_{DT}(\alpha)),\ \alpha = 0, 1, \ldots, 6 \Rightarrow n_i = \{0, 14, 18, 21, 23, 24, 24\}. \tag{19}$$

Fig. 5 shows increment of $\Gamma(\alpha)$ of bit-plane array NTA with decrease of application error threshold (increase of α) for $m = 2$, $k_C = 2$, $L = 6$ (example from Fig. 4). Note that values given in Fig. 5 are discrete.

5. Yield of the partial defect tolerant bit-plane array

Probability p starting from which PDT has greater yield than ET, for the bit-plane array, can be obtained as abscise of the point of intersection in Fig. 3. Denote $\gamma = \Gamma(\alpha)$, then from (5) and (7)–(9):

$$(1-p)^{\gamma T} = \frac{C}{C'}R^{\gamma T}. \tag{20}$$

Theorem 6. *Probability p starting from which PDT has greater yield than ET, for the bit-plane array, is*

$$p = \frac{1}{2}\left(-1+\sqrt{-3+4(1+2\Gamma(\alpha))^{\frac{1}{\Gamma(\alpha)T}}}\right) \tag{21}$$

Proof. We have to solve (20) for p. The equation is nonlinear. Therefore, in order to linearize the equation we denote

$$A = \frac{\log(1+2\gamma)}{\gamma\cdot T} = \log[(1-p)^2+3(1-p)p+3p^2]. \tag{22}$$

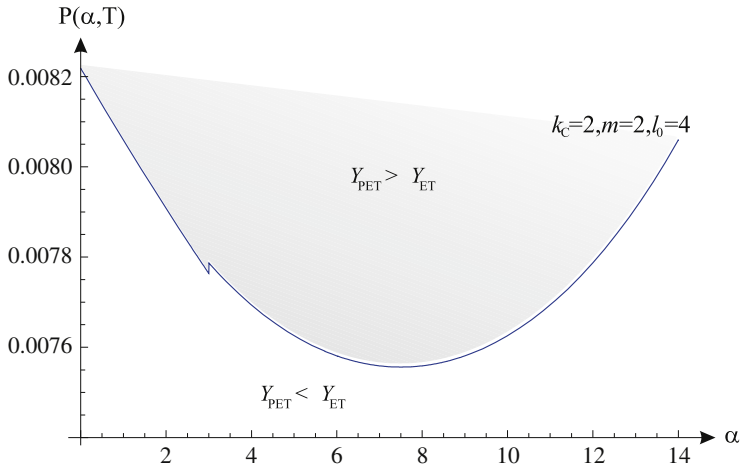

Fig. 6. $P(\alpha, T)$ where $k = 3$, $m = 4$, $L = 14$ and $T = 168$, for different values of parameter α.

Let $B = e^A$, then

$$B = e^A = (1-p)^2 + 3(1-p)p + 3p^2 = p^2 + p + 1. \tag{23}$$

Solving (23), we get two solutions

$$p_{1,2} = \frac{1}{2}(-1 \pm \sqrt{-3+4B}). \tag{24}$$

Since $A \geq 0$, we have $B \geq 1$. Hence, solution with the minus sign is discarded because it gives negative probability.

Let $P(\alpha, T)$ be the function from $\{(\alpha, T) \mid \alpha = 0, \ldots, L, T \in \mathbb{N}\}$ to the probability p of the point of intersection from Fig. 3. From (22)–(24) the following form of function $P(\alpha, T)$ is obtained

$$P(\alpha, T) = \frac{1}{2}(-1 + \sqrt{-3 + 4(1 + 2\Gamma(\alpha))^{\frac{1}{\Gamma(\alpha)T}}}), \tag{25}$$

which proves the statement. □

For the sake of illustration, the dependency between ET application's susceptibility to errors (α) and probability where PDT has preferred yield over ET is obtained for the example in Fig. 1 ($T = 168$) from (25), and is shown in Fig. 6. The values at $P(\alpha, T)$ axis are zoomed to emphasize the dependency, and are shown starting from value $p = 0.0076$ till $p = 0.0082$. It can be noticed that variations of α have very slight influence at the point of intersection of yields in Fig. 3. For example, from Fig. 1, with $T = 168$, the PDT has better yield in comparison to ET for probabilities of having defective cell greater than approximately $p = 0.008$, as shown in Fig. 6.

Fig. 7 shows $P(\alpha, T)$ for different number of cells, for constant γ ($\gamma = 0.5$). It can be noticed that probability of having a defective cell for which PDT yield is preferred in comparison with ET yield, exponentially decreases with the increase of total number of cells T.

From Fig. 7, it can be concluded that the application of defect tolerance on the architecture partition can be well exploited in nanotechnology. In [1, p. 833], it is indicated that "for today's large chips with $T > 10^9$ devices, the defect rate p must be below 10^{-10} to expect 90 percent or greater chip yield". Therefore, in the case of BP topology, if the probability p increases only 10 times and reaches 10^{-9}, or even more, which is common in nanotechnology, the yield falls significantly below the yield of the PDT system. In such cases, the PDT becomes the preferred design method. Furthermore, if probability $p = 10^{-8}$, and $T = 10^9$, the yield of ET system, according to (5) is $Y_{ET} = 0.0067$, while the yield of PDT system, according to (7), is $Y_{PDT} = 0.5$.

6. Concluding remarks

In this paper the analysis of yield in the case of partial application of defect tolerance was presented. The size and the position of defect tolerant part in the array, for the given topology and defect probability, where yield can be improved in comparison to error tolerant design, were examined. We modified mathematical description of yield from [1] by involving error tolerant concept introducing a function $\Gamma(\alpha)$, which depends on array topology. The evaluation was demonstrated on the bit-plane semi-systolic array, as a relatively complex array topology. The method shown for the chosen topology is described and proved in a formal mathematical way, and it easily covers simpler topologies. It was shown that the partial defect tolerant design can significantly improve effective yield for defect rates common in nanotechnology.

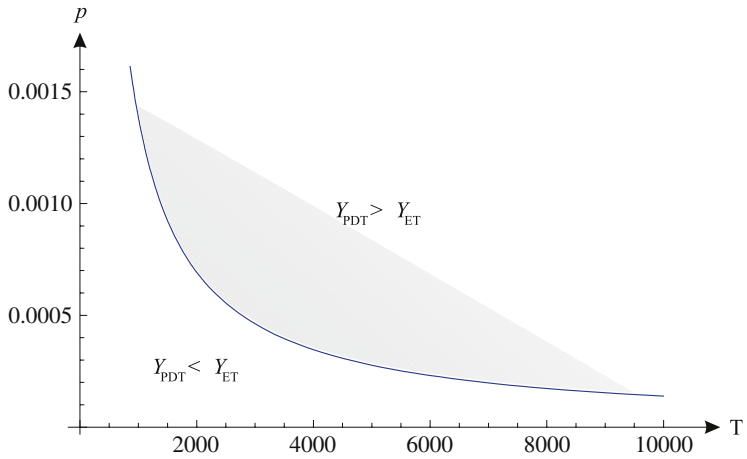

Fig. 7. $P(\alpha, T)$ for different values of parameter T, and $\gamma = c^{ta} = 0.5$.

References

[1] A. DeHon, Defect and fault tolerance, in: S. Hauck, A. DeHon (Eds.), Reconfigurable Computing: The Theory and Practice of FPGA-Based Computing, Morgan Kaufmann, 2008.
[2] International technology roadmap for semiconductors, recomendations, 2001. http://public.itrs.net/.
[3] S. Zhang, M. Choi, N. Park, Defect characterization and yield analysis of array-based nanoarchitecture, in: 4th IEEE Conference on Nanotechnology, 2004, pp. 50–52.
[4] M. Breuer, Intelligible test techniques to supporterror-tolerance, in: Proceedings on the 13th Asian Test Symposium, ATS 2004, IEEE Computer Society, 2004. 0-7695-2235-1/04.
[5] T.-Yu Hsieh, K.-Jong Lee, M. Breuer, Reduction of detected acceptable faults for yield improvement via error-tolerance, in: Proceedings of the conference on Design, automation and test in Europe, Nice, France, 2007, pp. 1599–1604.
[6] M. Breuer, S. Gupta, T. Mark, Defect and error tolerance in the pressence of massive numbers of defects, IEEE Transactions on Design & Test of Computers 21 (2004) 216–227.
[7] B.W. Johnson, Fault Tolerance: The Electrical Engineering Handbook, CRC Press, 1993.
[8] J. von Neumann, Probabilistic logic and synthesis of reliable organisms from unreliable components, in: C.E. Shannon, J. McCarthy (Eds.), Automata Studies, Princeton University Press, 1956, pp. 43–98.
[9] M. Breuer, Multimedia applications and imprecise computation, in: Proceedings on the 8th Euromicro conference on Digital System Design, Euromicro, Porto, Portugal, September 2005. 0-7695-2433-8/05.
[10] V. Ciric, J. Kolokotronis, I. Milentijevic, Partial error-tolerance for bit-plane FIR filter architecture, International Journal of Electronics and Communication (AEU) (ISSN: 1434-8411) 63 (5) (2009) 398–405.
[11] J. Kolokotronis, V. Ciric, I. Milentijevic, Error signifcance map for bit-plane FIR fltering array, in: Proc. 26th International Conference on Microelectronics, MIEL 2008, vol. 2, Ni, Serbia, May 2008, pp. 429–432.
[12] V. Ciric, I. Milentijevic, Configurable folded array for FIR filtering, Journal of Systems Architecture, An Internation Journal 54 (1–2) (2008) 177–196.
[13] T. Noll, Semi-systolic maximum rate transversal filters with programmable coefficients, in: Workshop of Systolic Architectures, Oxford, 1986, pp. 103–112.
[14] H.T. Kung, C.E. Leiserson, Systolic arrays (for VLSI), Carnegie Mellon University, 1978, Tech. report, CS-79-103, Pittsburgh, PA.
[15] F. Hausdorff, Set Theory, AMS Bookstore, 2005, translated by J. R. Aumann.