REFERENCES

[1] A. Mahmood and E. J. McCluskey, "Concurrent error detection using watchdog processor—A survey," *IEEE Trans. Comput.*, vol. C-37, no. 2, pp. 160-174, Feb. 1988.

[2] M. Schuette and J. Shen, "Processor control flow monitoring using signatured instruction streams," *IEEE Trans. Comput.*, vol. C-36, no. 3, pp. 264-276, Mar. 1987.

[3] N. R. Saxena and E. J. McCluskey, "Extended precision checksums," in *Dig. Papers 17th Annu Int. Symp. Fault-Tolerant Comput. (FTCS)*, July 1987, pp. 142-147.

[4] T. Sridhar and S. Thatte, "Concurrent checking of program flow in VLSI processors," in *Dig. Papers 1982 IEEE Test Conf.*, Nov. 1982, pp. 191-199.

[5] M. Namjoo, "Techniques for concurrent test of VLSI processor operation," in *Dig. Papers 1982 IEEE Test Conf.*, June 1982, pp. 461-468.

[6] K. Wilken and J. Shen, "Continuous signature monitoring: Efficient concurrent-detection of processor control errors," in *Dig. Papers 1988 IEEE Test Conf.*, Sept. 1988, pp. 914-925.

[7] HP 9000/930 and HP 9000/840 Computers, *Precision Architecture and Instruction Reference Manual*, HP, Nov. 1986.

[8] G. Kane, *R2000 RISC Architecture*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[9] J. Shen and S. Tomas, "A roving monitoring processor for detection of control flow errors in multiple processor systems," *Microprocessing and Microprogramming*, vol. 20, no. 4&5, pp. 249-269, May 1987.

[10] J. C. Huck, "Comparative analysis of computer architectures," Ph.D. dissertation, Stanford Univ., Mar. 1983.

[11] D. P. Siewiorek and R. S. Swarz, *The Theory and Practice of Reliable System Design*. Bedford, MA: Digital, 1982.

[12] U. Gunneflo, J. Karlsson, and J. Torin, "Evaluation of error detection schemes using fault injection by heavy-ion radiation," in *Proc. Nineteenth Int. Symp. Fault-Tolerant Comput.*, June 1989, pp. 340-347.

Optimal Diagnosis Procedures for k-out-of-n Structures

MING-FENG CHANG, WEIPING SHI, AND W. KENT FUCHS

Abstract—**This paper investigates diagnosis strategies for repairable VLSI and WSI structures based on integrated diagnosis and repair. Knowledge of the repair strategy, the probability of each unit being good, and the expected test time of each unit is used by the diagnosis algorithm to select units for testing. The general problem is described followed by an examination of a specific case. For k-out-of-n structures, we give a complete proof for the optimal diagnosis procedure proposed by Ben-Dov. A compact representation of the optimal diagnosis procedure is described, which requires $O(n^2)$ space and can be generated in $O(n^2)$ time. Simulation results are provided to show the improvement in diagnosis time over on-line repair and off-line repair.**

Index Terms—**Diagnosis, k-out-of-n, repair, VLSI, wafer probe testing, WSI.**

I. INTRODUCTION

Repairable VLSI or WSI structures may contain spare modules (units) for purposes of tolerating manufacturing defects and enhancing yield. In order to tolerate faulty units, it may be necessary to diagnose the system and apply reconfiguration strategies for repair. In the traditional approach, diagnosis and repair have been treated as separate stages: first diagnosis is performed and then the locations of the faulty units are passed to the reconfiguration algorithm [Fig. 1(a)]. An example of this approach is the diagnosis and repair algorithms for repairable memory arrays by Chang, Fuchs, and Patel [1].

Manuscript received June 21, 1989; revised November 30, 1989. This work was supported by the Semiconductor Research Corporation under Contract 88-DP-109. A brief version of this paper was presented at the 1989 IEEE International Conference on Computer Aided Design (ICCAD-89).

The authors are with the Center for Reliable and High-Performance Computing, Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801.

IEEE Log Number 8933876.

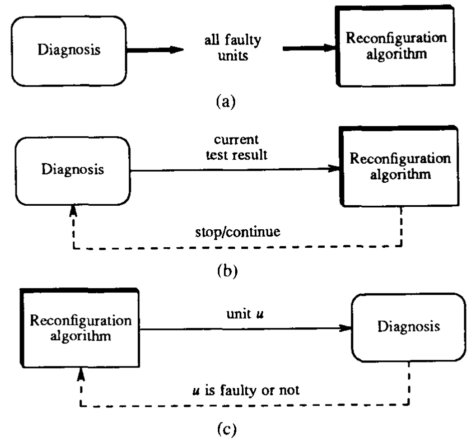

Fig. 1. Different diagnosis and reconfiguration methods. (a) Traditional off-line repair. (b) On-line repair. (c) Optimal diagnosis and repair.

However, in some applications the reconfiguration algorithm may not need complete information regarding the location of defective units to determine a repair solution. Haddad and Dahbura [2] proposed an on-line repair method which can detect unrepairable random-access memory chips during testing and hence eliminate unnecessary tests. A similar approach was proposed by Huang and Lombardi in their development of a memory repair algorithm which is executed in an on-line fashion with the diagnosis algorithm [3]. The on-line repair approach operates on partial diagnosis information, and therefore can potentially terminate the diagnosis procedure early and provide a repair solution earlier than an off-line repair [Fig. 1(b)].

In what we will call *optimal diagnosis and repair*, the reconfiguration algorithm not only works on-line with the diagnosis algorithm, but also tells the diagnosis algorithm which unit to test next [Fig. 1(c)]. The reconfiguration and diagnosis algorithms terminate when they determine a repair solution for the structure regardless of the status of the remaining units, or they determine the structure has to be discarded regardless of the status of the remaining units. In addition to exploiting the repair strategy, the diagnosis algorithm can also exploit knowledge about the expected yield of each unit and the expected test time of each unit in the structure.

Defect densities as well as distributions may vary across the wafer [4], and the expected access and test time of specific units may vary depending on the location as well as the type of unit under test. Even for identical units, the expected test time may differ because of the difference in yield. A fault-free unit has to be tested by all the test vectors, while the testing of a faulty unit is aborted when a fault is detected. Thus, the expected test time of a unit with higher yield is usually longer than that of a unit with lower yield.

In this paper, we consider structures whose units are tested individually and sequentially, such as in wafer probe testing [5]. An optimal diagnosis procedure gives the sequence in which units are tested, and minimizes the expected time for determining whether the structure is repairable or unrepairable. The specification of the structure includes the following information: 1) the structure of the system (location and type of each unit, the reconfigurable design),

2) the probability of each unit being faulty or fault-free, and 3) the expected access and test time for each unit. We believe this paper introduces a new and challenging topic of research concerning integrated diagnosis and repair.

The repairable structure examined in detail in this paper is the k-out-of-n structure. Such structures consist of n units, and the structure is functional if at least k units are fault-free. Each unit u_i in the structure has yield p_i and expected test time t_i. The optimal diagnosis problem for k-out-of-n structures was partially solved by Halpern [6] with the assumption that $t_i = t_j$ for all i and j, and also by Salloum and Breuer [7] with the assumption that $t_i/p_i \neq t_j/p_j$ and $t_i/(1-p_i) \neq t_j/(1-p_j)$ for all i and j. Ben-Dov [8] gave a solution without these restrictions, but his proof (Lemma 2) is incomplete. Other work on this problem includes a branch-and-bound algorithm by Ben-Dov [9], and approximation algorithms by Jedrzejowicz [10].

In this paper, we give a complete proof for the optimal diagnosis procedure proposed by Ben-Dov [8]. We also describe a compact representation of the diagnosis procedure—the compact representation requires only $O(n^2)$ space and can be generated in $O(n^2)$ time. The compact representation is basically the efficient storage of the decision tree, which can be used to decide the diagnosis procedure in $O(1)$ time in real time. Finally, we present simulation results showing the improvement in diagnosis time over on-line and off-line repair approaches.

In Section II, we briefly describe the problem and notation. Section III gives the optimal diagnosis procedure and proof for k-out-of-n structures. Section IV describes the compact representation. Simulation results are presented in Section V.

II. k-out-of-n Structures

A k-out-of-n structure consists of a set of n units $N = \{u_1, u_2, \cdots, u_n\}$. Each unit is either faulty or fault-free. The system itself is functional if at least k units are fault-free. For each unit u_i, let p_i be the *a priori* probability that u_i is fault-free, and t_i be the expected time to test u_i. For simplicity, let $q_i = 1 - p_i$. A simple 3-out-of-5 structure is shown in Table I. For each unit, its probability of being fault-free (yield) and its expected test time are shown in the table.

A diagnosis procedure can be represented by a binary decision tree. Each nonleaf node specifies a unit to be tested. If the unit is faulty, then the left subtree is taken, otherwise the right subtree is taken. Each leaf node represents the result of the testing, either success or failure. An example diagnosis procedure for the 3-out-of-5 structure is shown in Fig. 2. Clearly, the number of possible diagnosis procedures is exponential in n. The objective is to efficiently find the optimal diagnosis procedure which has the minimum expected test time.

A diagnosis procedure for a k-out-of-n structure, $P(N, k)$, can be described recursively as

$$P(N,k) = \begin{cases} (\text{Test}(u_i), P(N-\{u_i\},k), P(N-\{u_i\},k-1)) & \text{if } 0 < k \le n \\ \text{success} & \text{if } k = 0 \\ \text{failure} & \text{if } k > n \end{cases}$$

where u_i is the unit to be tested first, and if u_i is faulty, then procedure $P(N - \{u_i\}, k)$ is used to test the remaining units, else procedure $P(N - \{u_j\}, k - 1)$ is used to test the remaining units. The diagnosis procedure terminates if k fault-free units have been found or if $n - k + 1$ faulty units have been found. We use $P^*(N, k)$ to represent an optimal diagnosis procedure.

Relabel the units of the system so that

$$\frac{t_1}{p_1} \le \frac{t_2}{p_2} \le \cdots \le \frac{t_n}{p_n}$$

and let π be a permutation so that

$$\frac{t_{\pi(1)}}{q_{\pi(1)}} \le \frac{t_{\pi(2)}}{q_{\pi(2)}} \le \cdots \le \frac{t_{\pi(n)}}{q_{\pi(n)}}.$$

The optimal diagnosis problem for the highly restricted case where $k = 1$ or $k = n$ was first solved by Butterworth [11]. We briefly repeat his result here for the sake of completeness. In this special case, since all the nonleaf vertices are on a single path, a diagnosis procedure can be simply specified by a linear list of units in N.

Theorem 2.1: $P^*(N, 1) = (u_1, u_2, \cdots, u_n)$ and $P^*(N, n) = (u_{\pi(1)}, u_{\pi(2)}, \cdots, u_{\pi(n)})$.

III. The Optimal Diagnosis Procedure

Now we are ready to attack the general problem. Let $U_i = \{u_j | 1 \le j \le i\}$ and $V_i = \{u_{\pi(j)} | 1 \le j \le i\}$. Intuitively, U_k is the set of units which should be tested first if we want to locate k fault-free units, and V_{n-k+1} is the set of units which should be tested first if we want to locate $n-k+1$ faulty units. The set $U_k \cap V_{n-k+1} \neq \emptyset$, otherwise there would be at least $n + 1$ units in N which is a contradiction.

Theorem 3.1: The following testing procedure $P(N, k)$ is optimal:

$$P(N,k) = \begin{cases} (\text{Test}(u), P(N-\{u\},k), P(N-\{u\},k-1)) & \text{if } 0 < k \le n \\ \text{success} & \text{if } k = 0 \\ \text{failure} & \text{if } k > n \end{cases}$$

where u is any unit in $U_k \cap V_{n-k+1}$.

In order to prove the theorem, we need the following definitions and lemmas.

We associate each unit $u_i \in N$ with an indicator binary variable x_i. If u_i is faulty, $x_i = 0$. Otherwise, $x_i = 1$. We call the vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ an instance, which is a faulty/fault-free combination of units in N. The probability that a specific instance $\boldsymbol{x}$ appears is $\prod_{i=1}^{n} p_i^{x_i} q_i^{1-x_i}$. Throughout the paper, the bold capital letters are used to represent sets of instances. Let $\boldsymbol{S}_i = \{\boldsymbol{x} | x_i = 1$ *and* $\sum_{j=1}^{i} x_j = k\}$, and $\boldsymbol{F}_i = \{\boldsymbol{x} | x_{\pi(i)} = 0$ *and* $\sum_{j=\pi(1)}^{\pi(i)} (1 - x_j) = n - k + 1\}$. It is clear that $\boldsymbol{S}$'s and $\boldsymbol{F}$'s form a partition of the set of all instances. Notice for $i < k$, $\boldsymbol{S}_i = \emptyset$ and for $i < n - k + 1$, $\boldsymbol{F}_i = \emptyset$.

Lemma 3.1: For any $\boldsymbol{x} \in \boldsymbol{S}_i$, where $k \le i \le n$, the set of units tested by $P(N, k)$ is U_i.

Proof: By induction on the size of N. When $|N| = 1$, it is trivial that $P(\{u_1\}, 1) = \text{Test}(u_1)$, and $U_1 \cap V_1 = \{u_1\}$.

Assume the lemma is true for $|N| < n$, we will prove it is also true for $|N| = n$. For any instance $\boldsymbol{x} \in \boldsymbol{S}_i$, let u_j be the first unit tested by $P(N, k)$. By the definition of $P(N, k)$, $u_j \in U_k \subseteq U_i$. The remainder of the proof is to show that both $P(N - \{u_j\}, k)$ and $P(N - \{u_j\}, k - 1)$ will test $U_i - \{u_j\}$.

If u_j is faulty, then the kth fault-free unit in ascending subscript order, which is u_i, is now at position $i - 1$ in $N - \{u_j\}$. Since $|N - \{u_j\}| < n$, from the inductive hypothesis, $P(N - \{u_j\}, k)$ tests the remaining first $i - 1$ units, that is, $U_i - \{u_j\}$.

If u_j is fault-free, then u_i becomes the $(k - 1)$th fault-free unit in $N - \{u_j\}$. By inductive hypothesis, $P(N - \{u_j\}, k - 1)$ will test the remaining first $i - 1$ units, that is, $U_i - \{u_j\}$. □

TABLE I
EXAMPLE 3-OUT-OF-5 STRUCTURE

unit	u_1	u_2	u_3	u_4	u_5
yield	0.7	0.9	0.8	0.5	0.6
expected test time	1.0	1.7	1.6	1.2	1.5

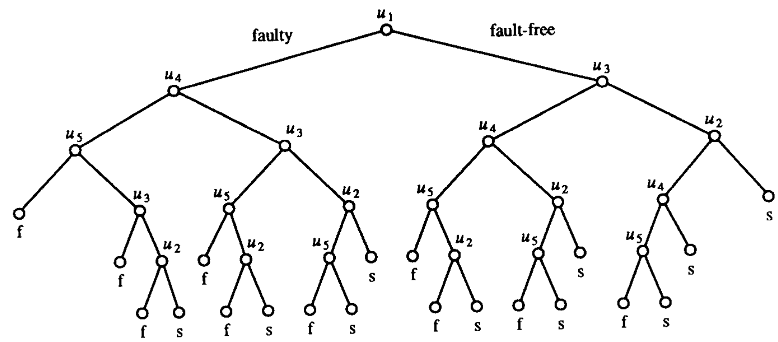

Fig. 2. An example diagnosis procedure for a 3-out-of-5 structure.

By a similar argument, we can prove the following.

Lemma 3.2: For any $x \in F_i$, where $n-k+1 \leq i \leq n$, the set of units tested by $P(N, k)$ is V_i.

It is surprising to notice that for any given instance, no matter how we choose $u_i \in U_k \cap V_{n+k-1}$ when $|U_k \cap V_{n-k+1}| > 1$, $P(N, k)$ always tests the same set of units. We write $E(P(N, k))$ as the expected test time when applying $P(N, k)$ to instance x, over all instances x.

Proof of Theorem 3.1. By induction on the number of units of N. When $|N| = 1$, it is trivial that $P(\{u_1\}, 1) = \text{Test}(u_1)$ is the optimal diagnosis procedure, since it is the only procedure.

Assume $P(N, k)$ is optimal for all N of size less than n, we will show that $P(N, k)$ is optimal for $|N| = n$. Let $P_i^*(N, k)$ be the diagnosis procedure which tests u_i first and the remainder diagnosis procedure is optimal. Then

$$\begin{aligned} P_i^*(N, k) &= (\text{Test}(u_i), P^*(N - \{u_i\}, k), P^*(N - \{u_i\}, k-1)) \\ &= (\text{Test}(u_i), P(N - \{u_i\}, k), P(N - \{u_i\}, k-1)) \end{aligned}$$

because $|N - \{u_i\}| < n$. Since $E(P^*(N, k)) = \min_{i=1}^{n} E(P_i^*(N, k))$, we will prove $P(N, k)$ is an optimal diagnosis procedure by showing that

$$E(P(N, k)) = E(P_i^*(N, k)) \qquad \text{for } u_i \in U_k \cap V_{n-k+1}$$

$$E(P(N, k)) \leq E(P_i^*(N, k)) \qquad \text{for } u_i \notin U_k \cap V_{n-k+1}.$$

If $u_i \in U_k \cap V_{n-k+1}$, by Lemma 3.1 and Lemma 3.2, $P_i^*(N, k)$ and $P(N, k)$ will test the same set of units for any instance.

If $u_i \notin U_k \cap V_{n-k+1}$, there are three possible cases:

1) $u_i \notin U_k$, $u_i \in V_{n-k+1}$
2) $u_i \in U_k$, $u_i \notin V_{n-k+1}$
3) $u_i \notin U_k$, $u_i \notin V_{n-k+1}$.

We now prove that $E(P(N, k)) \leq E(P_i^*(N, k))$ for case 1). Since $u_i \in V_{n-k+1}$, for any instance in F_j where $j > n-k$, $P(N, k)$ and $P_i^*(N, k)$ test the same set of units which is V_j. The fact that u_i is not in U_k implies $i > k$. For any instance in S_j, where $j > i$, $P(N, k)$ and $P_i^*(N, k)$ test the same set of units, which is U_j. Only for instances in $\bigcup_{j=k}^{i} S_j$, may $P(N, k)$ and $P_i^*(N, k)$ test different sets of units. Therefore, only units in U_i can be tested. Consider whether each unit $u_j \in U_i$ is tested or not by $P(N, k)$ and $P_i^*(N, k)$, respectively.

1.a) If $1 \leq j < k$, u_j is always tested by both procedures.

1.b) If $j = i$, u_j is tested by $P_i^*(N, k)$, but is tested by $P(N, k)$ only for $x \in S_i$.

1.c) $k \leq j < i$. If u_i is faulty, then u_j is tested by $P_i^*(N, k)$ if and only if u_j is tested by $P(N, k)$. If u_i is fault-free, $P(N, k)$ needs to locate k good units in U_i, while $P_i^*(N, k)$ needs to locate $k-1$ good units in U_{i-1}. Let $G(l, g)$ represent the set of instances which have exactly g good units in U_l. For all the instances in $\bigcup_{m=k}^{j-1} G(j-1, m)$, u_j is tested by neither diagnosis procedure. For all the instances in $\bigcup_{m=0}^{k-2} G(j-1, m)$, u_j is tested by both procedures. Only for the instances in $G(j-1, k-1)$, is u_j tested by $P(N, k)$ but not by $P_i^*(N, k)$.

Therefore, the difference in the total expected test time is as follows, where $\text{Prob}(G(j-1, k-1))$ denotes the probability an instance is in $G(j-1, k-1)$, and $\text{Prob}(S_j)$ denotes the probability an instance is in S_j.

$$\begin{aligned} &E(P(N, k)) - E(P_i^*(N, k)) \\ &\quad = \sum_{j=k}^{i-1} p_i \,\text{Prob}(G(j-1, k-1)) t_j - \sum_{j=k}^{i-1} \text{Prob}(S_j) t_i \\ &\quad = \sum_{j=k}^{i-1} p_i \,\text{Prob}(G(j-1, k-1)) t_j \\ &\qquad - \sum_{j=k}^{i-1} p_j \,\text{Prob}(G(j-1, k-1)) t_i \\ &\quad = \sum_{j=k}^{i-1} \text{Prob}(G(j-1, k-1))(p_i t_j - p_j t_i) \\ &\quad \leq 0 \end{aligned}$$

because $t_j/p_j \leq t_i/p_i$.

Case 2) is the dual of case 1). In case 2), $u_i \notin V_{n-k+1}$ implies $i = \pi(m)$ for some $m > n-k+1$. Therefore, only for instance $x \in \bigcup_{j=n-k+1}^{m} F_j$, may the test time of $P(N, k)$ and $P_i^*(N, k)$ be different. By a similar argument as above, we can show that $E(P(N, k)) \leq E(P_i^*(N, k))$.

In case 3), the difference in expected test time is the sum of the differences in cases 1) and 2). □

IV. COMPACT REPRESENTATION

In this section, we describe how to precompute and save the optimal diagnosis procedure, so that whenever the diagnosis algorithm needs to make a decision, it simply checks the stored procedure in

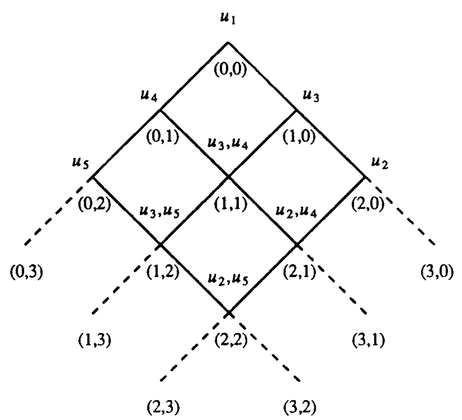

Fig. 3. A diagnosis procedure specified by the block-walking representation.

$O(1)$ time. Since the total number of vertices in the binary decision tree is $\Omega(2^n)$, any method which explicitly generates the binary decision tree requires exponential time and memory space. However, we will show that the binary decision tree can be implicitly represented by the *block-walking representation* which is of size $O(n^2)$ and can be computed in time $O(n^2)$. The block walking representation has been used in combinatorics to illustrate the number of shortest paths in which a walker can walk from one point to another point on a rectilinear grid [12].

Definition 4.1: For any vertex v in a binary decision tree, define its tested unit set $\mathrm{TU}(v)$ to be the set of units tested along the path from the root to v, including v.

Definition 4.2: For any vertex v in a binary decision tree, define its test state $\mathrm{TS}(v)$ to be an ordered pair (i, j), where i and j are the number of fault-free and faulty units tested along the path from the root to v, excluding v.

It is obvious that if two vertices v_1 and v_2 in the tree have the same tested unit set and test state, then the subtrees rooted by v_1 and v_2 can be made identical (except that the units tested at the root may be different). Therefore, we can merge v_1 and v_2, and let them share the same diagnosis procedure thereafter. If all the vertices with the same test state (i, j) can be merged, then let $\mathrm{TU}(i, j)$ denote the tested unit set $\mathrm{TU}(v)$ for v such that $\mathrm{TS}(v) = (i, j)$. If all the vertices with the same test state can be merged, then the testing procedure can be simplified to a block-walking representation.

Definition 4.3: Given a k-out-of-n structure $N = \{u_1, u_2, \cdots, u_n\}$, the block walking representation of its diagnosis procedure is a 5-tuple $(G, S, F, \delta_s, \delta_f)$ where G is the set of intermediate states, S is the set of success states, F is the set of failure states, and δ_s and δ_f indicate which unit to test if the last test has succeeded or failed, respectively. Formally, we have

$$G = \{(i, j) | 0 \le i \le k-1, 0 \le j \le n-k\},$$

$$S = \{(k, j) | 0 \le j \le n-k\},$$

$$F = \{(i, n-k+1) | 0 \le i \le k-1\}, \text{ and}$$

$$\delta_s, \delta_f : G \to N.$$

For simplicity, we write $\delta_s((i, j))$ as $\delta_s(i, j)$, and $\delta_f((i, j))$ as $\delta_f(i, j)$. To illustrate the meaning of the functions δ_s and δ_f, consider an example $\delta_s(i, j) = u_l$ and $\delta_f(i, j) = u_m$. This means that at state (i, j), we will test unit u_l if the last test succeeded (passed) and we will test unit u_m if the last test failed. The test before state (0, 0) is assumed to have succeeded. Fig. 3 is a block-walking representation of the diagnosis procedure shown in Fig. 2.

Each grid point (i, j) represents a test state (i, j) of the diagnosis procedure. The diagnosis starts from the grid point (0, 0) and traverses downward. At each point a unit is tested and the diagnosis moves to the next point depending on the outcome of the test.

It should be pointed out that since the optimal diagnosis procedure is not necessarily unique, there may be some optimal diagnosis procedures which cannot be represented by the block-walking representation. However, the following theorem shows that it is always possible to describe one optimal procedure using the block-walking representation, which is indeed what we need.

Theorem 4.1: For any $X \subseteq N$, define $\mathrm{SS}(X)$ to be the unit in X with the smallest subscript. If we choose $u = \mathrm{SS}(U_k \cap V_{n-k+1})$ in $P(N, k)$, then the diagnosis procedure can be represented by a block-walking representation.

Proof: This procedure is optimal by Theorem 3.1. The remainder of the proof is to show that for any vertices v_1 and v_2 in the binary decision tree, $\mathrm{TU}(v_1) = \mathrm{TU}(v_2)$ if $\mathrm{TS}(v_1) = \mathrm{TS}(v_2)$. Clearly, v_1 and v_2 must be on the same level of the decision tree. The proof is by induction on m, the length of the path from the root to v_1 and v_2.

The base case in which $m = 0$ is trivial, since only the root has the test state (0, 0). Also, for the states (i, j) with $i = 0$ or $j = 0$, there exists only one vertex for each state.

Now assume the theorem is true for all vertices in the tree above level m. In other words, assume all vertices with test state (i, j), where $i + j < m$, have the same tested unit set. Replace the top $m - 1$ levels of the tree by a block-walking representation. Consider an arbitrary test state (i, j) with $i + j = m$ and $i, j > 0$. Clearly, there are only two states $(i - 1, j)$ and $(i, j - 1)$ which can possibly have an edge to state (i, j). By induction hypothesis, states $(i - 1, j)$ and $(i, j - 1)$ are two points in the block-walking representation. Therefore, there are at most two vertices v_1 and v_2 with test state (i, j), where v_1 is a son of point $(i - 1, j)$ and v_2 is a son of point $(i, j - 1)$. Therefore,

$$\mathrm{TU}(v_1) = \mathrm{TU}(i-1, j-1) \cup \{\delta_f(i-1, j), \delta_s(v_1)\}$$

$$\mathrm{TU}(v_2) = \mathrm{TU}(i-1, j-1) \cup \{\delta_s(i, j-1), \delta_f(v_2)\}.$$

Hence, $\mathrm{TU}(v_1) = \mathrm{TU}(v_2)$ if $\{\delta_f(i-1, j), \delta_s(v_1)\} = \{\delta_s(i, j-1), \delta_f(v_2)\}$. From the following observation,

$$\delta_f(i-1, j) = \mathrm{SS}(U_{k+j} \cap V_{n-k+i} - \mathrm{TU}(i-1, j-1))$$

$$\delta_s(v_1) = \mathrm{SS}(U_{k+j} \cap V_{n-k+i+1} - \mathrm{TU}(i-1, j-1) - \{\delta_f(i-1, j)\})$$

$$\delta_s(i, j-1) = \mathrm{SS}(U_{k+j-1} \cap V_{n-k+i+1} - \mathrm{TU}(i-1, j-1))$$

$$\delta_f(v_2) = \mathrm{SS}(U_{k+j} \cap V_{n-k+i+1} - \mathrm{TU}(i-1, j-1) - \{\delta_s(i, j-1)\}).$$

We conclude $\{\delta_f(i-1, j), \delta_s(v_1)\} = \{\delta_s(i, j-1), \delta_f(v_2)\}$, because both sets consist of two units which have the smallest two subscripts in $U_{k+j} \cap V_{n-k+i+1} - \mathrm{TU}(i-1, j-1)$. □

The equivalent block-walking representation for the diagnosis procedure defined in Theorem 4.1 can be generated by the following algorithm.

Algorithm: Block-Walking-Representation:

Input: A k-out-of-n structure with the yield $p_1, p_2, \cdots, p_n$ and test time $t_1, t_2, \cdots, t_n$.

Output: A block walking representation of an optimal diagnosis procedure $(G, S, F, \delta_s, \delta_f)$.

Method: Since G, S, and F can be constructed easily, we only show how to compute δ_s and δ_f.

1: tmp := $U_k \cap V_{n-k+1}$;
  $\delta_s(0, 0)$:= SS(tmp);
2: **for** i := 1 **to** $k - 1$
  begin
    tmp := tmp $\cup \{u_{\pi(n-k+1+i)}\} - \{\delta_s(0, i-1)\}$;
    $\delta_s(0, i)$:= SS(tmp)
  end;
3: **for** i := 0 **to** $k - 1$
  begin
    tmp := $V_{n-k+1+i} - \{\delta_s(0, j) | 0 \le j \le i\}$;

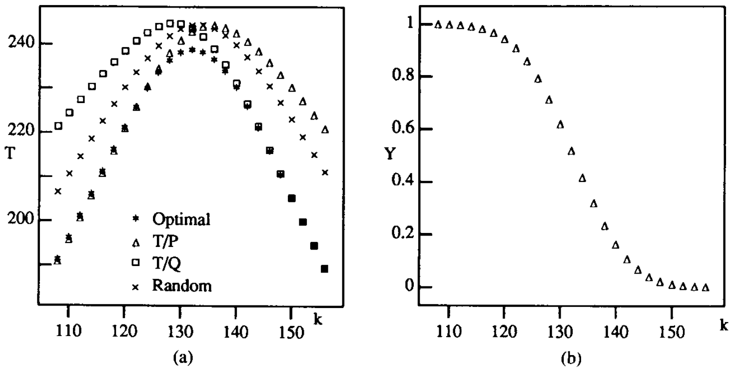

Fig. 4. k-out-of-256 structures, $t_i = 1$, p_i: (0.3, 0.7). (a) The expected diagnosis time. (b) The yield for different k.

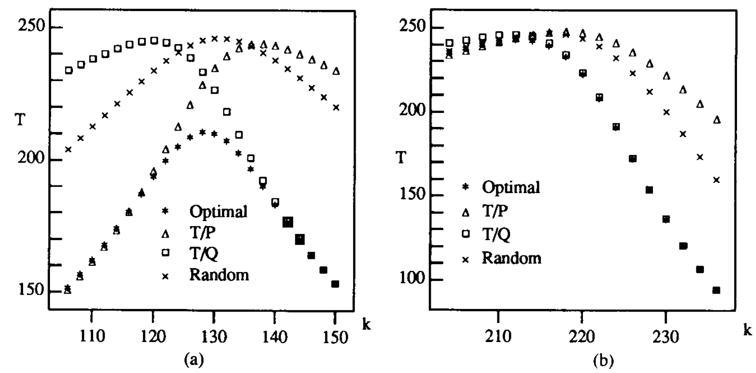

Fig. 5. The expected diagnosis time of two k-out-of-256 structures. (a) $t_i = 1$, p_i: (0.01, 0.99). (b) $t_i = 1$, $p_i = (0.71, 0.99)$.

```
        Sort tmp into ascending subscript order;
        for j := 1 to n - k
            δ_f(i, j) := the jth unit of tmp
    end;
4:  for i := 1 to n - k
        for j := 1 to k - 1
        if (δ_f(i, j) = δ_f(i - 1, j))
            then δ_s(i, j) := δ_f(i, j - 1)
        else
            δ_s(i, j) := δ_f(i, j)
End of Algorithm.
```

In step 3, the sorting can be done in $O(n)$ time by a radix sort on the subscript. Therefore, each step can be finished in $O(n^2)$ time and the total time complexity of the algorithm is $O(n^2)$. After a block-walking representation is constructed, the expected diagnosis time of the system can be obtained in a bottom-up fashion by iteratively computing the test time at each grid point; this requires $O(n^2)$ time.

Application of the algorithm to the example in Table I results in the optimal diagnosis procedure shown by the compact representation in Fig. 3.

V. Simulations

Simulations were performed to determine the reduction in the test time by using the optimal diagnosis procedure as compared to other approaches. Three other diagnosis procedures were chosen for comparison:

T/P procedure: the diagnosis procedure always tests the unit with minimum t_i/p_i.

T/Q procedure: the diagnosis procedure always tests the unit with minimum t_i/q_i.

Random procedure: the test sequence is chosen at random.

In Fig. 4, the test time for each unit was set to 1 and the fault-free probability was chosen randomly in a uniform distribution on (0.3, 0.7). Since the T/P procedure and optimal testing procedure will test the same set of units for a functional instance of the structure, when the yield of the entire structure is high, their performances are very close. On the other hand, when the yield of the structure is low, the performance of the T/Q procedure is near optimal.

In Fig. 5(a), the test time for each unit was set to 1 and the fault-free probability was chosen as a uniform distribution on (0.01, 0.99). For this case in which the variance of p_i is large, the improvement of the optimal procedure over the random procedure is more than 17% as shown in the example in Fig. 5(a).

When k is close to n, the performance of the T/P procedure is close to that of the optimal diagnosis procedure [Fig. 5(b)], because $n - k + 1$ is small (only a small number of faulty units need be located) and it is more cost-effective to initially test units which are more likely to be faulty. Symmetrically, when k is small, the performance of the T/P procedure is near optimal. The expected diagnosis time for two examples with small n, 16, are shown in Fig. 6. The results are similar to the examples with large n.

Let the expected test time for faulty units be T_f and the test time for fault-free units be T_s. Two examples with $T_f = 0.5$ and $T_f = 0.25$, respectively, are shown in Fig. 7. Since the test time for a faulty unit is less than that of a fault-free unit, the performance of the T/Q procedure is close to that of the optimal procedure as T_f decreases.

VI. Conclusion

This paper has described the problem of wafer probe-based optimal diagnosis and repair for structures in which the yield of indi-

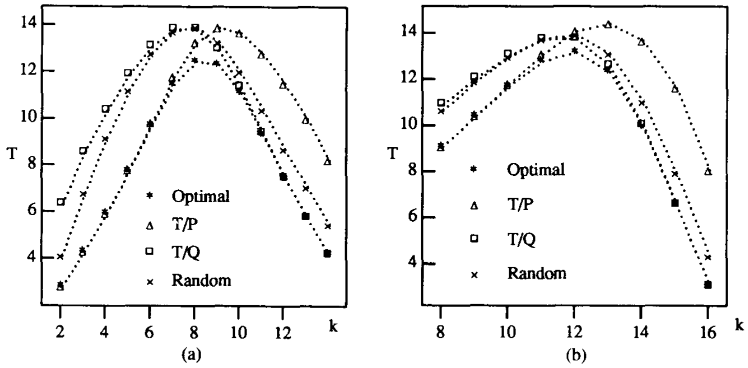

Fig. 6. The expected diagnosis time of two k-out-of-16 structures. (a) A k-out-of-16 system ($p_i: 0.3-0.7$). (b) A k-out-of-16 system ($p_i: 0.71-0.99$).

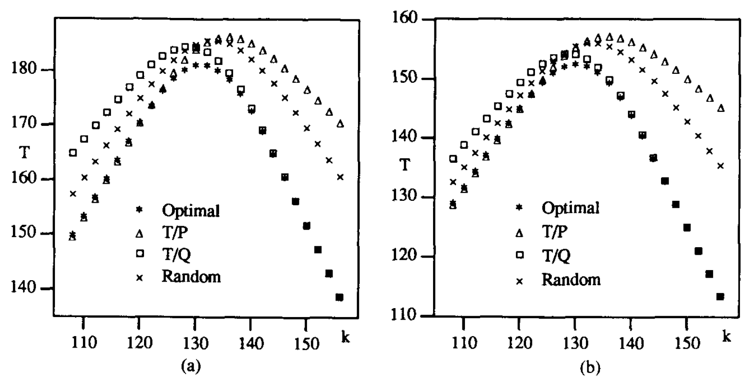

Fig. 7. k-out-of 256 structures with different T_f, $p_i: (0.3, 0.7)$. (a) $T_s = 1$, $T_f = 0.5$. (b) $T_s = 1$, $T_f = 0.25$.

vidual units and the expected time for testing each unit is known. A proof of the optimality of a diagnosis algorithm was given for k-out-of-n structures along with a method of compactly representing an optimal solution. Simulation results illustrate the reduction in diagnosis time achieved by exploiting knowledge of the repair strategy as well as knowledge of the expected yield and test time of each unit in the structure.

References

[1] M.-F. Chang, W. K. Fuchs, and J. H. Patel, "Diagnosis and repair of memory with coupling faults," *IEEE Trans. Comput.*, vol. 38, pp. 493-500, Apr. 1989.
[2] R. W. Haddad and A. T. Dahbura, "Increased throughput for the testing and repair of RAMs with redundancy," in *Proc. IEEE Int. Conf. Comput.-Aided Design*, 1987, pp. 230-233.
[3] W. K. Huang and F. Lombardi, "Minimizing the cost of repairing WSI memories," in *Proc. IEEE Int. Conf. Wafer Scale Integration*, Jan. 1989, pp. 183-192.
[4] A. V. Ferris-Prabhu, L. D. Smith, H. A. Bonges, and J. K. Paulsen, "Radial yield variations in semiconductor wafers," *IEEE Circuits Devices*, pp. 42-47, Mar. 1989.
[5] A. B. Leslie and A. F. Matta, "Wafer-level testing with a membrane probe," *IEEE Design Test Comput.*, vol. 6, pp. 10-17, Feb. 1989.
[6] J. Halpern, "Fault testing for a k-out-of-n system," *Oper. Res.*, vol. 22, pp. 1267-1271, Nov.-Dec. 1974.
[7] S. Salloum and M. A. Breuer, "An optimum testing algorithm for some symmetric coherent systems," *J. Math. Anal. Appl.*, vol. 101, pp. 170-194, 1984.
[8] Y. Ben-Dov, "Optimal testing procedures for special structures of coherent systems," *Management Sci.*, vol. 27, pp. 1410-1420, Dec. 1981.
[9] ——, "A branch and bound algorithm for minimizing the expected cost of testing coherent systems," *Euro. J. Oper. Res.*, vol. 7, pp. 284-289, 1981.
[10] P. Jedrzejowicz, "Minimizing the average cost of testing coherent systems: Complexity and approximation algorithms," *IEEE Trans. Reliability*, vol. R-32, pp. 66-70, Apr. 1983.
[11] R. Butterworth, "Some reliability fault-testing models," *Oper. Res.*, vol. 20, pp. 335-343, Mar.-Apr. 1972.
[12] A. Tucker, *Applied Combinatorics*. New York: Wiley, 1984, p. 447.

The Full-Use-of-Suitable-Spares (FUSS) Approach to Hardware Reconfiguration for Fault-Tolerant Processor Arrays

MENGLY CHEAN AND JOSE A. B. FORTES

Abstract—**A general approach to hardware reconfiguration is proposed for VLSI/WSI fault-tolerant processor arrays. The technique, called FUSS (full use of suitable spares), uses an indicator vector, the**

Manuscript received July 2, 1989; revised November 30, 1989. This work was supported in part by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization and was administered through the Office of Naval Research under Contracts 00014-85-k-0588 and 00014-88-k-0723.

M. Chean is with Bellaire Research Center, Shell Development Company, Houston, TX 77025.

J. A. B. Fortes is with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907.

IEEE Log Number 8933877.

0018-9340/90/0400-0564$01.00