Multiplets, Models, and the Search for the Meaning: Improving Per-Test Fault Diagnosis

Abdul-Rahman Elshafei – ID 260130

Road Map

- Introduction
- SLAT & iSTAT
- Multiplet Scoring
- Matching Passing Tests
- Matching Complex Failures
- Multiplet Ranking
- Experimental Results
- Analyzing Multiplets
- Results of Multiplets Classification

Introduction

Challenges with Diagnostics Today:

- not easy to know what kind of defect is present
- presence of multiple defects may interfere with each other to modify expected fault behaviors
- intermittent faults are difficult to reproduce
- large circuit size may make application of diagnosis algorithms impractical

Introduction

Disadvantages of single stuck-at-fault models:

- multiple faults
- complex faults

Complex faults:

- faults in which the fault behavior involves several circuit nodes, multiple erroneous logic values, pattern dependent, intermittent or unpredictable

Introduction

Problems of current multiple stuck-at-faults model diagnostic algorithms:

- forsake inter-test dependence and instead consider each test independently
- cannot identify patter-dependent or intermittent faults
- diagnoses are made for each test pattern independently
- problem of constructing a plausible defect scenario to explain the observed behavior

Introduction

Disadvantage of per-test diagnosis (STAT):

1. can be fooled by aliasing, when the fault effects from multiple or complex faults mimic the response from a single stuck at fault

2. they have large candidate sets that difficult to understand and use

Proposed Algorithm (iSTAT):

- Improve candidate matching by introducing scoring and ranking techniques.
- Improves process of per test fault diagnosis by including more information to score candidates and pairing down the candidate list to a manageable number
- Product of per test diagnosis is improved by suggesting a way of interpreting the candidates to infer the most likely defect type

Multiplets

- ***multiplets***: it is a collection of all candidate faults are arranged into sets the cover all the matched tests.

Test Number	Exactly-Matching Faults
1	A
2	B
3	C, D, E

- Multiplets:
  1. (A, B, C)
  2. (A, B, D)
  3. (A, B, E)

SLAT (Single Location At a Time)

- SLAT is a pre-test fault diagnosis algorithm based on concept of multiplets sets which explains or covers all of failing test patterns

- SLAT Procedure:

1. finds failing tests, then identifies and collects faults that match them
2. simple recursive algorithm is used to traverse all covering sets smaller than a pre-set maximum size
3. reports only minimal-sized multiplets in its final diagnosis

SLAT (Single Location At a Time)

Problem with SLAT:

- not enough evidence to point to particular fault in cases such as outputs with a lot of fan-in or a defect in an area with many equivalent faults

iSTAT vs SLAT:

1. iSTAT improves per-test diagnosis by considering the weight of evidence pointing to individual faults and to quantify that evidence into multiplet scoring

2. the scoring mechanism is used to rank multiplets to narrow the resulting candidate set

3. it uses the results from both passing tests and complex failing tests to improve the scoring of candidate fault sets

How Scoring is Done

- mechanism of scoring is based on the Dempster-Shafer method of evidentiary reasoning
- the Dempster-Shafer is a generalization of the Bayes Rule of Conditioning

1. **Degree of Belief:** probability assigned to a proposition relative to the strength of evidence presented
2. **Degree of doubt:** represented by p(Φ)
3. **Belief Function**

How Scoring is Done

- each failing test that is matched exactly by one or more fault candidates results in a belief function
- each candidate is assigned an equal portion of the belief from the test result
- iSTAT uses a degree of doubt $p(\Phi) = 0.01$

Multiplet Scoring Example 1

Test Number	Matching Faults
1	A

$$p_1(A) = 0.99$$

Multiplet Scoring Example 2

Test Number	Matching Faults
1	A
2	A, D

$$p(A) = p_2(A)p_1(A) + p_2(A)p_1(\Phi) + p_2(\Phi)p_1(A)$$
$$= 0.5049$$
$$p(D) = p_2(D)p_1(D) + p_2(D)p_1(\Phi) + p_2(\Phi)p_1(D)$$
$$= 0.00495$$
$$p(AD) = p_2(D)p_1(A) = 0.49005$$
$$p(\Phi) = p_2(\Phi)p_1(\Phi) = 0.0001$$

Multiplet Scoring Ex 3

Test Number	Matching Faults
1	A
2	A, D
3	B

$$p(AB) = (0.99)(0.5049) = 0.499851$$
$$p(A) = (0.01)(0.5049) = 0.005049$$
$$p(ABD) = (0.99)(0.49005) = 0.4851495$$
$$p(AD) = (0.01)(0.49005) = 0.0049005$$
$$p(BD) = (0.99)(0.00495) = 0.0049005$$
$$p(D) = (0.01)(0.00495) = 0.0000495$$
$$p(B) = (0.99)(0.0001) = 0.000099$$
$$p(\Phi) = (0.0001)(0.01) = 0.000001$$

Multiplet Scoring Ex4

Test Number	Matching Faults
1	A
2	A, D
3	B
4	C, D

$$p(ABD) = 0.492$$

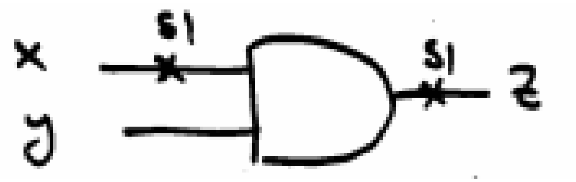
$$p(ABC) = 0.247$$

- by assigning a probability score to each candidate set it provides much more guidance in selecting candidates from what can be large diagnosis

Passing Tests

- ignoring passing tests results in having larger candidate size because it will include all fault candidates whose fault signatures are supersets of the observed behavior.

- Example:



Matching Passing Tests

- only multiplets are considered, not individual faults
- candidates that predict a passing test will share in belief assigned based on that test
- if some of the component faults of a multiplet predict failures for a passing test:
  1. none of these faults were activated
  2. fault was sensitized but none of the its failures propagated to observed outputs
- both conditions happen due to interaction between multiple faults

Matching Passing Tests

- each passing test, a multiplet is initially assigned a belief value:
  - 0 $\longrightarrow$ all faults predict failure
  - 1 $\longrightarrow$ all faults predict a pass
- initial score is divided by the total score over all multiplets, thus total belief =1
- iSTAT uses degree of doubt of 0.5

Matching Complex Failures

- iSTAT combines all the predicted failing outputs and then ignores misprediction of the observed failing outputs
- the degree of belief for each matching multiplet is 1 divided by the number of matching multiplets
- degree of doubt is 0.1, therefore the belief assigned to individual matching multiplets is normalized by multiplying by 0.9

Multiplet Ranking

- iSTAT considers a wider range of defect scenarios than can SLAT and many other per test algorithms
- Example:

Test Number	Matching Faults
1	A, B
2	A, C
3	A, C
4	A, B

$$p(A) = 0.5002$$
$$p(BC) = 0.4998$$

- SLAT will only consider p(A)

Multiplet Ranking Example 2

Test Number	Matching Faults
1	A, B
2	A, C
3	A, C
4	A, B
5	B, C

$$p(A) = 0.3335$$
$$p(BC) = 0.6665$$

Experimental Results

- authors used simulated defects in an industrial circuit
- defects were created by modifying the circuit netlist and simulating the test vectors to obtain faulty behavior.

Experimental Results

Defect No.	Simulated Defect	Faults in SLAT and Top-Ranked iSTAT Multiplets	SLAT Multiplets	Top-Ranked iSTAT Multiplets	Success?
1	Single stuck-at fault	1	7	4	Y
2	2 independent stuck-at faults	2	21	8	Y
3	2 independent stuck-at faults	2	1	1	Y
4	2 interfering stuck-at faults	2	9	4	Y
5	3 interfering stuck-at faults	3	2	1	Y
6	4 stuck-at faults, 3 interfering	4	2	1	Y
7	Two-line wired-OR bridge	2	2	1	Y
8	Two-line wired-AND bridge	2	2	1	Y
9	Two-line wired-AND bridge	2	1	1	Y
10	Two-line wired-XNOR bridge	3	13	7	Y
11	Two-line dominance bridge	1	3	1	P
12	Two-line dominance bridge	1	2	1	P
13	Net fault (3 branch stuck-at faults)	4	90	1	Y
14	Net fault (3 branch stuck-at faults)	3	4	1	Y
15	Gate replacement (OR to AND)	1	1	1	Y
16	Gate replacement (OR to NOR)	2	11	7	Y
17	Gate replacement (MUX to NAND)	2	3	2	Y
18	Gate output inversion	1	3	1	Y
19	Multiple logic errors on one gate	1	1	1	Y
20	Multiple logic errors on one gate	2	27	10	Y

Analyzing Multiplets

Analyzing Multiplets

- purpose is to analyze each multiplet in a diagnosis to determine whether the component faults are in some way related to one another or if they are simply a collection of random faults

- interpret multiplets by correlating them with common fault models such as:
  1. transition fault models
  2. bridging fault models
  3. stuck at fault models

- then calculating for every multiplet a correlation score for each model

Plausibility Metrics

- ***plausibility***: upper probability limit that a multiplet represents an instance of a particular fault model

- for each multplet, iSTAT computes a plausibility score for each fault model:

Condition	Plausibility score
complete agreement of faults to defect assumptions	1.0
No agreement	0.0

A. Single or Intermittent stuck at faults

Condition	Plausibility
Multiplet is of size 1	1.0
otherwise	0.0

B. Node/Transition Fault

- ***node fault***: a multiplet that consists of opposite polarity on the same node

Condition	Plausibility
multiplet size = 2 & faults belong to same node	1.0
otherwise	0.0

C. Net Fault

Condition	Plausibility
multiplet size ≥ 2 and all faults are on the same net	1.0
multiplet size > 3	% of faults are in the same net
multiplet size = 1	0.0

D. Gate Fault

- When all of the faults in a multiplet involve common gate or standard cell

Condition	Plausibility
multiplet size ≥ 2 & all faults are on the same gate ports	1.0
multiplet size > 3	% of faults on the same gate
multiplet size = 1	0.0

E. Two-Line Bridging fault

Condition	Plausibility
multiplet size 2,3 or 4 & all faults are on two nodes	% of common tests for faults of opposite polarity + % of common tests for faults of same polarity that pass
otherwise	0.0

F. Path Fault

Condition	Plausibility
multiplet is size ≥ 2 & all faults exist on a path from an output to an input	1.0
multiplet size ≥ 3	% of faults on the same path
• multiplet size = 1 • all faults are on the same node	0.0

Simulation Results

Defect No.	Simulated Defect	Single Stuckat	Node Fault	Net Fault	Gate Fault	2-Line Bridge	Path Fault
1	Single stuck-at fault	**1.0**	0.0	0.0	0.0	0.0	0.0
2	2 independent stuck-at faults	0.0	0.0	0.0	0.0	0.0	0.0
3	2 independent stuck-at faults	0.0	0.0	0.0	0.0	**1.0**	0.0
4	2 interfering stuck-at faults	0.0	0.0	0.0	0.0	0.0	0.0
5	3 interfering stuck-at faults	0.0	0.0	0.0	0.0	0.0	**0.67**
6	4 stuck-at faults, 3 interfering	0.0	0.0	0.0	0.0	0.0	**0.75**
7	Two-line wired-OR bridge	0.0	0.0	0.0	0.0	**1.0**	0.0
8	Two-line wired-AND bridge	0.0	0.0	0.0	0.0	**1.0**	0.0
9	Two-line wired-AND bridge	0.0	0.0	0.0	0.0	**1.0**	0.0
10	Two-line wired-XNOR bridge	0.0	0.0	0.0	0.0	**1.0**	0.0
11	Two-line dominance bridge	**1.0**	0.0	0.0	0.0	0.0	0.0
12	Two-line dominance bridge	**1.0**	0.0	0.0	0.0	0.0	0.0
13	Net fault (3 branch stuck-at faults)	0.0	0.0	**1.0**	0.0	0.0	0.0
14	Net fault (3 branch stuck-at faults)	0.0	0.0	**1.0**	0.0	0.0	0.0
15	Gate replacement (OR to AND)	**1.0**	0.0	0.0	0.0	0.0	0.0
16	Gate replacement (OR with NOR)	0.0	**1.0**	0.0	**1.0**	0.0	0.0
17	Gate replacement (MUX - NAND)	0.0	0.0	0.0	**1.0**	0.0	0.0
18	Gate output inversion	**1.0**	0.0	0.0	0.0	0.0	0.0
19	Multiple logic errors on one gate	**1.0**	0.0	0.0	0.0	0.0	0.0
20	Multiple logic errors on one gate	0.0	0.0	0.0	**1.0**	0.0	0.0

Questions ?