Test Vector Chains for Increased Targeted and Untargeted Fault Coverage

Irith Pomeranz
Purdue University
Sudhakar M. Reddy
University of Iowa

Test Generation Concepts

- Single-bit changes to generate new test vectors from existing ones
    - [Snethen, DAC-77], [Girard, VTS-97], [Tsai, ITC-97], …
- n-detection test generation where each target fault is detected by n different tests for untargeted fault coverage
    - [Ma, ITC-95], [Reddy, TCAD-97], [Grimaila, VTS-99], [Benware, ITC-03], …

Test Vector Chains

- A new concept for obtaining new test vectors from existing ones through sequences of single-bit changes.
- A test vector chain is defined based on two test vectors t_1 and t_2 included in an initial test set T_0.
- Single-bit changes are made to gradually modify t_1 into t_2.

Example of Test Vector Chain

t_1	0	0	0	0	1	1	1	1
	1	0	0	0	1	1	1	1
	1	0	1	0	1	1	1	1
	1	0	1	0	1	0	1	1
t_2	1	0	1	0	1	0	0	1

Motivation

- Suppose that t_1 (t_2) detects a subset of faults F_1 (F_2).
- The new test vectors that are close to t_1 (t_2) are likely to detect the faults in F_1 (F_2).
- These vectors will increase the numbers of detections for the faults in F_1 (F_2).

Motivation

- During the transition from detecting F_1 to detecting F_2, test vectors that detect faults outside of F_1 and F_2 may be generated.
- These vectors will detect or increase the numbers of detections for faults outside of F_1 and F_2.
- When the number of detections of a fault increases, the likelihood of detecting untargeted faults associated with it increases.

Definition of Test Vector Chain

- $t_i(j)$ is the value of input j under t_i.
- $diff(t_1,t_2)$ includes -1 and every input j such that $t_1(j) \neq t_2(j)$.
- For every input j in $diff(t_1,t_2)$, $t^j_{1,2}$ is such that for $k \leq j$ $t^j_{1,2}(k)=t_2(k)$ and for $k>j$ $t^j_{1,2}(k)=t_1(k)$.
- For every j in $diff(t_1,t_2)$, $t^j_{1,2}$ is included in the test vector chain $C(t_1,t_2)$.

Test Generation Procedure Based on Test Vector Chains

- Given test set T_0. Set of faults G.
- Simulate G under T_0 with fault dropping.
- For every pair of test vectors t_1,t_2 in T_0:
  - Find the test vector chain C(t_1,t_2).
  - Simulate G under C(t_1,t_2) with fault dropping. If any fault is detected, add t_1,t_2 to a set P.

Test Generation Procedure Based on Test Vector Chains

- Remove redundant test vector chains from P by forward-looking reverse order simulation.
- Reinitialize G. Simulate G under T_0 with fault dropping. Set $T_{chains}=T_0$.

Test Generation Procedure Based on Test Vector Chains

- For every t_1,t_2 in P:
  - Find the test vector chain C(t_1,t_2). For every t in C(t_1,t_2):
    - Simulate G under t with fault dropping. If any fault is detected, add t to T_{chains}.

Experimental Results

- T_0 is a deterministic compact one-detection stuck-at test set [Kajihara, TCAD-95].
- Set of target faults G:
- 10 copies of each stuck-at fault to generate 10-detection test sets.
- For a circuit with L lines, cL randomly selected four-way bridging faults.
  - c=10 if $|T_{chains}| \leq 10|T_0|$, otherwise a value of c such that $|T_{chains}| \approx 10|T_0|$.

Test Set Quality

- Measured by the fault coverage with respect to a set BR100 of 100L randomly selected four-way bridging faults.
- BR100 represents a large set of untargeted faults.
- For comparison, an n-detection test set T_{ndet} of the same size as T_{chains}.

Results for G(10-detections)

	T_0		T_{chains}			T_{ndet}	
circuit	vec	BR100	vec	n	BR100	n	BR100
s820	94	69.57	1294	9	75.10	13	75.01
s953	76	76.72	860	10	80.39	11	81.04
s1196	138	78.88	1616	10	84.55	13	83.86
s1423	26	83.15	810	10	93.45	30	95.57
s5378	100	89.71	2391	10	94.17	24	97.36
s15850	97	86.38	4416	10	93.30	45	97.60

Results for G(4-way bridg)

	T_0		T_{chains}		T_{ndet}
circuit	vec	BR100	vec	BR100	BR100
s820	94	69.57	358	74.97	73.71
s953	76	76.72	314	81.19	80.55
s1196	138	78.88	529	85.31	82.84
s1423	26	83.15	255	93.83	95.19
s5378	100	89.71	981	95.66	97.03
s15850	97	86.38	1328	95.72	97.42