Estimating Fault Detection Effectiveness

Rick Kuhn[1], Raghu Kacker[1], Yu Lei[2]
{kuhn, raghu.kacker}@nist.gov, ylei@uta.edu
[1]NIST, [2]U. of Texas Arlington

A *t*-way covering array can detect *t*-way faults, however they generally include other combinations beyond *t*-way as well. For example, a particular test set of all 5-way combinations is shown capable of detecting all seeded faults in a test program, despite the fact that it contains up to 9-way faults. This poster gives an overview of methods for estimating fault detection effectiveness of a test set based on combinatorial coverage for a class of software.

Detection effectiveness depends on the distribution of *t*-way faults, which is not known. However based on past experience one could say for example the fraction of 1-way faults is F_1 = 60 %, 2-way faults F_2 = 25 % F_3 = 10 % and F_4 = 5 %. Such information could be used in determining the required strength *t*. It is shown that the fault detection effectiveness of a test set may be affected significantly by the *t*-way fault distribution, overall, simple coverage at each level of *t*, number of values per variable, and minimum *t*-way coverage. Using these results, we develop practical guidance for testers.

A Simple Model: Assume deterministic software that computes the same output for a given set of input parameters and values. Faults are also deterministic in that we assume a failure-triggering combination of input values will always produce a failure if it is present in the input. Under these assumptions, two factors in fault detection effectiveness are the fault distribution within the SUT, and combinatorial coverage of the tests. If faults are detected if and only if a failure inducing combination appears in a test, then the probability of detection can be estimated within a certain range using the *t*-way coverage of tests and an approximate distribution of *t*-way faults. A first approximation of the fault detection effectiveness can be developed by considering the proportion of combinatorial coverage at different levels of *t*, up to some reasonable value of *t*, and empirical judgment of the proportion of faults at each level of *t*. For example, if it is expected that 50% of failures for a system are 1-way and 50% 2-way, then the proportion of faults that can be expected to be found in testing is $.5 * S_1 + .5 * S_2$, where S_1 and S_2 are the proportion of 1-way and 2-way combinations covered, respectively. If a test set is developed that tests all values of all parameters (assuming a small set of discrete values) and 75% of 2-way combinations, then the fault detection effectiveness for this test set would be 0.5*1.0 + 0.5*0.75 = 87.5%. Under this simple model, we can estimate the detection effectiveness $D = \sum_{1 \le t \le k} F_t \times S_t$, where k = maximum interaction strength in failures, F_t = proportion of faults that are *t*-way, and S_t = *t*-way coverage.

Input Model Considerations: The model above is inadequate in many applications, because effectiveness may also depend on input variable values. We consider an additional factor M_t = minimum proportion of *t*-way coverage. Any set of *t* variables with *v* values each has v^t possible settings. If *c* represents the number out of v^t settings covered in a test set for a particular set of *t* variables, M_t is the minimum value c/v^t, among all sets of *t* variables in the test set.

Suppose we have two test sets, T_1 and T_2, that both provide 100% 1-way coverage and 80% 2-way coverage, with coverage statistics T_1: $S_1 = 0.80$, $M_1 = 0.80$ and T_2: $S_2 = 0.80$, $M_2 = 0.5$. Is their fault detection effectiveness the same, or could one provide better fault detection? The answer depends on the input model. Suppose the code contains the following segment:

```
if (x <= && y <= 0){faulty code}
else               {good code}
```

and the input model partitions values for `x` and `y`:

```
x = {-9999, -1, 0, 1, 9999}
y= {-9999, -1, 0, 1, 9999}
```

Then for the 25 pairs of input values for x and y, 9 will induce the fault. Therefore if at least 17/25 = 68% of input combinations are covered in a test set, at least one will induce the fault. T_1 ensures this, because all 2-way combinations are covered to at least 80% (M_1 = .80). Although T_2 has the same level of overall simple coverage, S_2 = .80, its minimum coverage, M_2, is only 50%, so pair {*x*, *y*} may have less than 17/25 coverage.

The poster provides an overview of these and other aspects of combinatorial coverage for estimating fault detection effectiveness. We show the rate at which (*t* + *k*)-way coverage increases with *t* and number of variables *n*, and decreases with values *v*, and how these results can be used to guide development of tests achieving an approximate level of fault detection.