Measuring Combinatorial Coverage of System State-space for IV&V

Rick Kuhn and Raghu Kacker
National Institute of Standards & Technology
Gaithersburg, MD 20899
kuhn@nist.gov, raghu.kacker@nist.gov

One important aspect of verification and validation (V&V) is assurance that critical points in the system state-space have been covered in testing. Subject matter experts are generally required to design tests that are expected to evaluate significant, but possibly rare, conditions. The full system state-space, consisting of all valid system configurations, is generally impossible to cover to any significant level, because the number of configurations is too large. However, a good deal of empirical research suggests that the number of factors interacting in system failures is relatively small. For example, a system failure may occur when a particular switch is on and some value for speed is exceeded, a 2-way interaction. In empirical studies, failures reported involved a maximum of 4-way to 6-way interactions across a variety of application domains. Thus while testing can explore only a tiny portion of the full state-space, it may be possible to produce tests that cover nearly all *t*-way interactions (of discretized values) for small levels of *t*. For example, a large test set that is well designed may cover most 2-way and 3-way combinations of discretized values of system variables.

Combinatorial testing is an approach based on covering all *t*-way combinations for some specified level of *t*, but this form of testing may not be practical because of established test practices, legal or contractual test requirements, or use of legacy test sets. An alternative to creating a combinatorial test set from scratch is to investigate the combinatorial coverage properties of an existing test set, possibly supplementing it with additional tests to ensure thorough coverage of system variable combinations. Determining the level of input or configuration state-space coverage can help in understanding the degree of risk that remains after testing. If a high level of coverage of state-space variable combinations has been achieved, then presumably the risk is small, but if coverage is much lower, then the risk may be substantial. This talk describes some measures of combinatorial coverage that can be helpful in estimating this risk, which we have applied to tests for spacecraft software but have general utility for other combinatorial coverage problems. This method will be illustrated through a prior application to NASA spacecraft software [1], preliminary results on more recent NASA software, and non-NASA software. An example below introduces the method in more depth and shows some previous results.

Example. For the four tests in Table 1, there are C(4, 2) = 6 possible 2-variable combinations and $C(4,2)2^2$ = 24 possible variable-value configurations (00,01,10,11 possible for each pair). Of these, 19 variable-value configurations are covered and the only ones missing are *ab*=11, *ac*=11, *ad*=10, *bc*=01, *bc*=10, as shown in Table 2. But only two, *bd* and *cd*, are covered with all four value pairs. So for simple *t*-way coverage (number of *t*-way combinations fully covered), we have only 33% (2/6) coverage, but 79% (19/24) of possible 2-way combinations covered. For a better understanding of this test set, we can compute the configuration coverage for each of the six variable combinations, shown in Table 2. For this test set, one of the combinations (*bc*) is covered at the 50% level, three (*ab*, *ac*, *ad*) are covered at the 75% level, and two (*bd*, *cd*) are covered at the 100% level. And, as noted above, for the whole set of tests, 79% of variable-value configurations are covered. All 2-way combinations have at least 50% configuration coverage. These concepts have been implemented in an analysis tool, applied to over 7,000 tests for a NASA spacecraft in Fig. 1. The graph shows coverage for 2-way to 4-way combinations. For example, the red line at coverage ≥ 75% (*y* axis) extends nearly to 100% of combinations (*x* axis), showing that this test set provides relatively good combinatorial coverage of 2-way combinations.

Test	a	b	c	d
1	0	0	0	0
2	0	1	1	0
3	1	0	0	1
4	0	1	1	1

Table 1. Example test w/ 4 binary variables *a..d*

Vars	Configurations	Coverage
a b	00, 01, 10	.75
a c	00, 01, 10	.75
a d	00, 01, 11	.75
b c	00, 11	.50
b d	00, 01, 10, 11	1.0
c d	00, 01, 10, 11	1.0

Table 2. Coverage of 2-way combinations of *a..d*

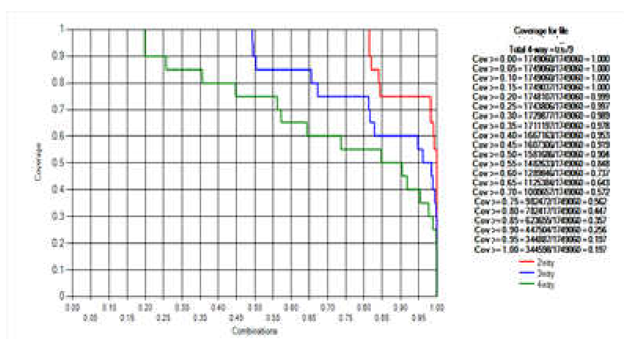

Figure 1. Coverage for 7,489 tests

[1] J.R. Maximoff, M.D. Trela, D.R. Kuhn, R. Kacker, "A Method for Analyzing System State-space Coverage within a t-Wise Testing Framework", *IEEE International Systems Conference 2010*, Apr. 4-11, 2010, San Diego.