Efficient Test Generation Guided by Field Coverage Criteria

Ariel Godio[1], Valeria Bengolea[2], Pablo Ponzio[2,3], Nazareno Aguirre[2,3], and
Marcelo F. Frias[1,3]

[1] Department of Software Engineering, CISoft-ITBA, Buenos Aires, Argentina
{agodio,mfrias}@itba.edu.ar
[2] Department of Computer Science, UNRC, Rio Cuarto, Argentina
{vbengolea,pponzio,naguirre}@dc.exa.unrc.edu.ar
[3] CONICET, Argentina

Abstract. Field-exhaustive testing is a testing criterion that requires suites to contain enough test inputs to cover *all* feasible values for fields within a certain input-size bound. While previous work shows that field-exhaustive suites can be automatically generated, the generation technique requires a formal specification of the inputs that can be subject to SAT-based analysis. Moreover, this constraint together with the restriction of producing *all* feasible values for input fields makes test generation costly, and field-exhaustive testing difficult to generalize to further testing domains.

In this paper, we deal with *field coverage* as testing criteria that measure the degree to which a program is tested by examining to what extent the values of inputs' fields are covered. We show that this notion generalizes field-exhaustive testing, withdrawing the need for a SAT-analyzable formal specification, and thus can be combined with any test generation technique to produce smaller test suites, reducing testing time. In particular, we consider field coverage: *(i)* in combination with test generation based on symbolic execution, to produce underapproximations of all testing sequences; *(ii)* as a relaxation of bounded-exhaustive testing, producing smaller suites using the Korat tool; and *(iii)* in combination with random testing, producing smaller test suites and even serving as a termination criterion for generation. As we show, in all these cases field coverage helps producing significantly smaller suites, thus contributing to testing time, while retaining the effectiveness of the corresponding original techniques, in terms of test suite quality.

Keywords: Field based testing · Symbolic execution · Transcoping.