

The Ideal Test Length is Infinity

Alex Groce,* Josie Holmes*

Abstract—How long should a test case be? In previous work, it has been assumed that the solution to this problem, though seldom applied in practice, is to gather empirical evidence as to a length that maximizes faults discovered or coverage obtained for a test budget, and then to generate tests of that optimal length. In this work, we show that, when it is possible to produce tests on-the-fly, without a fixed maximum length, it is (under assumptions that should hold in most cases) best to simply make each test “infinite” — only terminating a test when it fails, or when memory or resource constraints make further extension of the test impractical. This is true both for fundamental reasons tied to the purpose of testing (a test that terminates without failing has, in some sense, no value), and due to the extremely high overhead of test setup and initialization. We show that transforming infrastructure to generate tests of unbounded length on the fly, rather than produce tests of a fixed size can produce a large improvement in the effectiveness of testing for complex, real-world test generators, and that the merits of unbounded test length are considerable even for testing Python libraries, where startup costs are relatively small.

I. INTRODUCTION

The problem of choosing an optimal length for generated test cases is a long-standing concern in automated testing [1].

II. RELATED WORK

III. FORMAL ANALYSIS

IV. CASE STUDIES

V. CONCLUSIONS AND FUTURE WORK

REFERENCES

- [1] J. H. Andrews, A. Groce, M. Weston, and R.-G. Xu, “Random test run length and effectiveness,” in *Automated Software Engineering*, 2008, pp. 19–28.