

Probabilistic Symbolic Execution^{*}

No Author Given

No Institute Given

Abstract. The abstract should briefly summarize the contents of the paper in 150–250 words

Keywords: Probabilistic Programming · Symbolic Execution · Logic · Verification.

1 Introduction

2 Overview

3 Algorithm

Sumit: Layout the Algorithm text and Algorithm in package Sumit: In 4 pages

Algorithm 1: Probabilistic Symbolic Execution Algorithm

```
1 set the current state to  $\vec{T}_{sc}$ ;  
2  $\vec{E} \leftarrow \text{GETSTATELIST}(\vec{T}_{sc})$ ;  
3 for  $\vec{s} \in \vec{E}$  do  
4   |  $\text{GETSOLVE}(\vec{s})$ 
```

4 Implementation

We build on top of KLEE [1], use Z3 [2] for solving the SMT formula for computing the $\mathbb{E}[v]$ values

5 Experimental Evalutaion

Sumit: What experiments and examples to work on? Sumit: In 4 pages

- (RQ1) Experiments
- (RQ2) Additional Examples

^{*} We submit an online anonymous docker image of our tool along with a ZIP file containing all experimental artifacts and scripts for regeneration.

6 Related Work

7 Conclusion

References

1. Cadar, C., Dunbar, D., Engler, D.: Klee: Unassisted and automatic generation of high-coverage tests for complex systems programs. In: Proceedings of the 8th USENIX Conference on Operating Systems Design and Implementation. p. 209–224. OSDI'08, USENIX Association, USA (2008)
2. de Moura, L., Bjørner, N.: Z3: An efficient smt solver. In: Ramakrishnan, C.R., Rehof, J. (eds.) Tools and Algorithms for the Construction and Analysis of Systems. pp. 337–340. Springer Berlin Heidelberg, Berlin, Heidelberg (2008)