Algorithm 1 The ENRICH Approach.

Inputs. Sys: A simulator
$(R_1, \ldots, R_n)$: The initial search space
BUDGET: Max number of iterations
Outputs. NR: New ranges for non-robustness

1: **function** NR=ENRICH(Sys, $(R_1, \ldots, R_n)$, BUDGET)
2: i=0; TS= []; NR = $(R_1, \ldots, R_n)$; TSAll = []; ▷ Variables Initialization
3: **do**
4: TS=GENTESTS(Sys, NR) ▷ Test input generation
5: TSAll = TS ∪ TSAll; ▷ Combine new and old tests
6: RT =BUILDRT(TSAll); ▷ Build regression tree
7: NR=REDUCTION(RT); ▷ Search space reduction
8: i++; ▷ Increases the counter
9: **while** (i¡BUDGET)
10: **return** NR;

Algorithm 2 Test Generation.

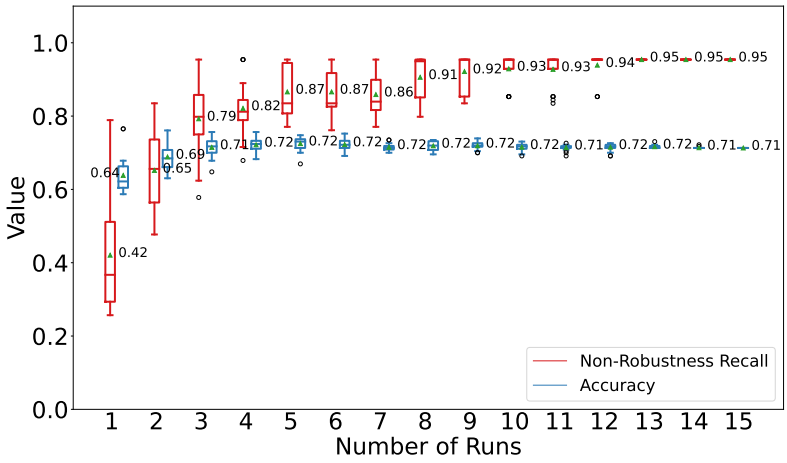
Inputs. Sys: A simulator
$(R_1, \ldots, R_n)$: The input search space
Outputs. TS: New Test Suite

1: **function** TS=GENTESTS(Sys, $(R_1, \ldots, R_n)$)
2: **for** i=0; i¡TestSuiteSize; i++
3: tc=ART(Sys, $(R_1, \ldots, R_n)$); ▷ Adaptive Random Testing
4: v=ROBUSTNESSMEASURE($Sys(tc)$); ▷ Execution of a Test Case
5: TS=TS∪{$\langle tc, v \rangle$} ▷ Add the Test to the Test Suite
6: **end for**
7: **return** TS

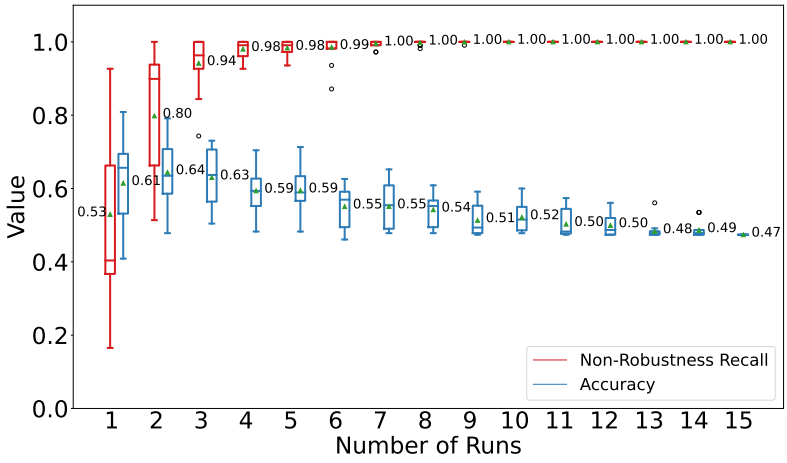
Algorithm 3 Reduction.

Inputs. RT: Regression Tree
Outputs. $(R'_1, \ldots, R'_n)$: New Ranges for Non-robust Behaviour

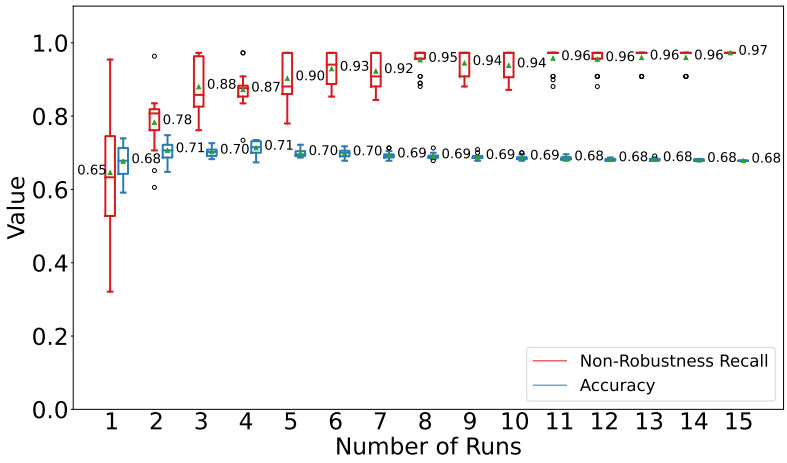
1: **function** $(R'_1, \ldots, R'_n)$=REDUCTION(RT)
2: $(P_1, \ldots, P_m)$=EXTRACTPATHS(RT);
3: (P_i, P_k)=EXTRACTNONROBUSTPATHS($(P_1, \ldots, P_m)$);
4: $(R'_1, \ldots, R'_n)$=EXTRACTRANGES(P_i, P_k);
5: **return** $(R'_1, \ldots, R'_n)$



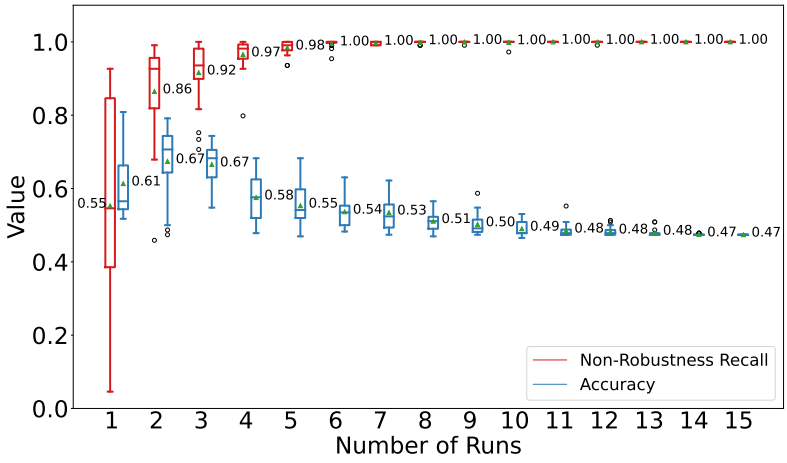
(a) ENRICH $\varepsilon = 25\%$



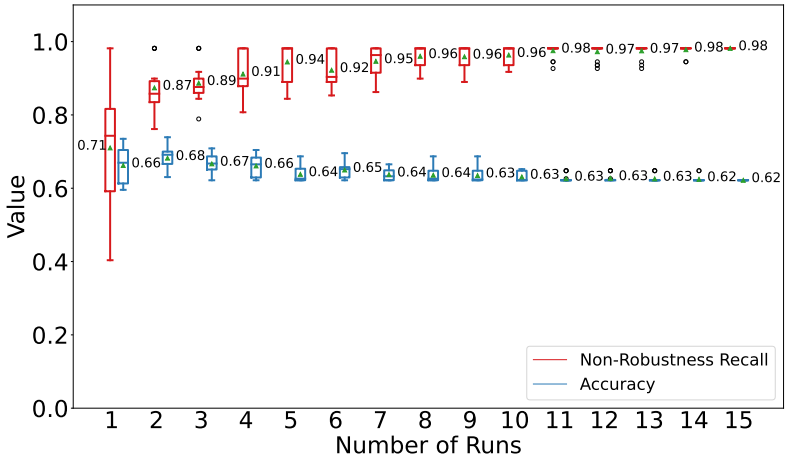
(b) BASELINE $\varepsilon = 25\%$



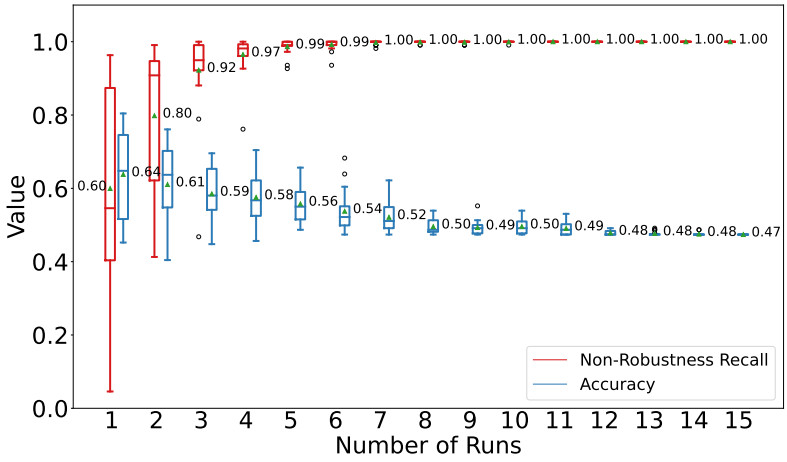
(c) ENRICH $\varepsilon = 30\%$



(d) BASELINE $\varepsilon = 30\%$



(e) ENRICH $\varepsilon = 35\%$



(f) BASELINE $\varepsilon = 35\%$

Figure 1: Comparison of Accuracy and Non Robustness Recall of ENRICH and BASELINE for ε between 25% and 40% over different combination of runs.