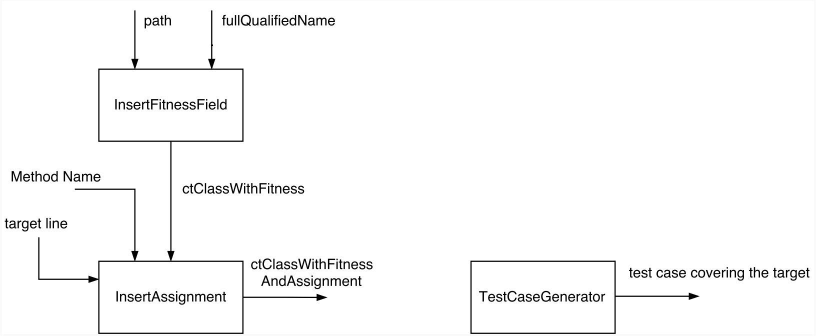
Test Generation

Benjamin DANGLOT

12th, May 2017

Instrumentation and Test Generation



Hill Climbing Algorithm

3 Algorithms:

- HillClimbing Best: Take the best solution over the neighbors (6 potentials)
- HillClimbing First: Take the first solution that improve the fitness
- Random: generate new random solution and keep it if it better

Neighbors:

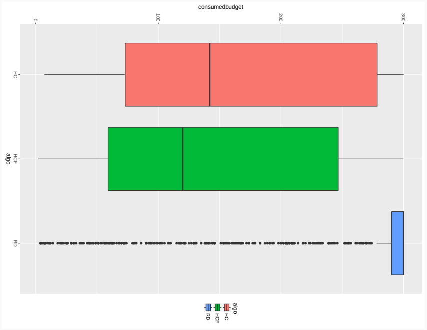
- Random neighbor: replace one value by a random one.
- Inc1Neighbors: $+1$ on one value of the vector solution.

Empirical results: Answer to the RQs

HillClimbing Best Selection:	768 success
HillClimbing First Selection:	806 success
Random:	254 success

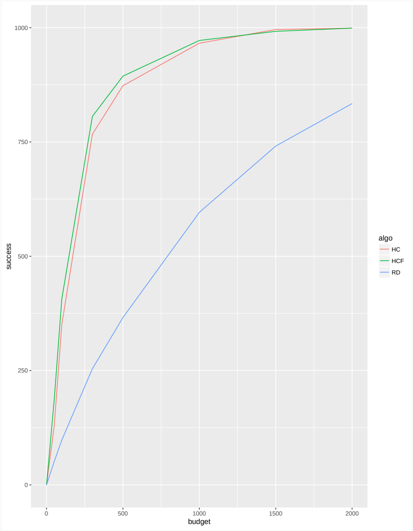
RQ_1: Yes, the Hill Climbing algorithm is more effective than a Random Algorithm.

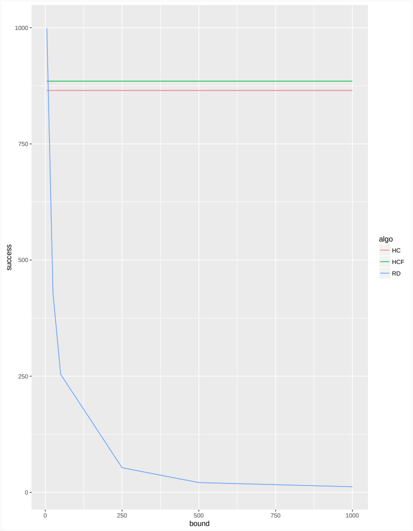
Empirical results: Budgets



RQ$_2$: Yes, the Hill Climbing algorithm is more effecient than a Random Algorithm.

Empirical results: impact of Budgets and Bounds





Statistical Analysis

Fisher test result on the number of success:

- p-value $< 2.2e - 16$
- alternative hypothesis: true odds ratio is not equal to 1
- 95 percent confidence interval: 7.798722 11.856348

Wilcoxon test on the consumed budged:

- W = 234970, p-value ¡ 2.2e-16
- alternative hypothesis: true location shift is not equal to 0

Possible Improvement

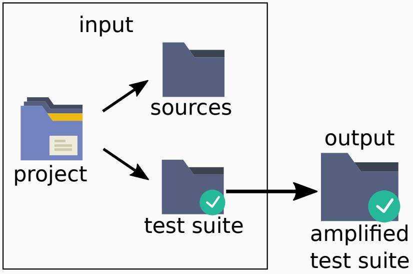
Run the Test Generator over multiple target:

- Add a loop over all the target (by the developer himself)
- Instrument the targeted class
- Add the compilation by Spoon
- Build a CustomClassLoader with the previous compiled file
- Run the HillClimbing to generate test case for the current target by reflection
- Generate the test cases with Spoon

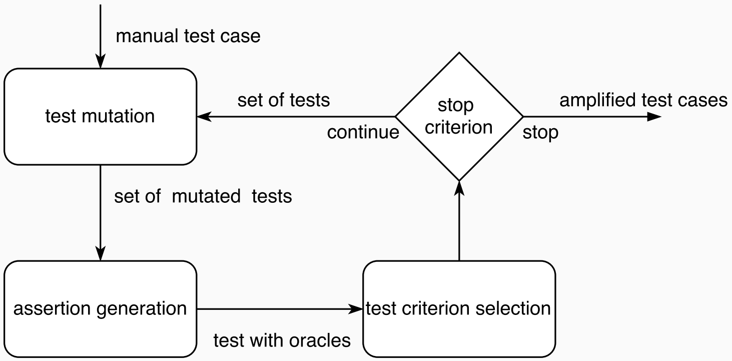
About Me

From Lille, France. Phd Student at Inria.

Within H2020 STAMP: **S**oftware **T**esting **AMP**lification.



DSpot


manual test case
test mutation
set of tests
stop criterion
continue
stop
amplified test cases
set of mutated tests
assertion generation
test with oracles
test criterion selection