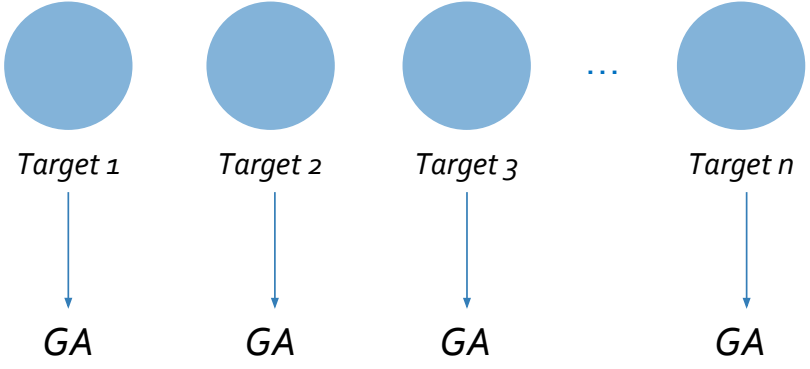
SQL Test Case Generation Using Multi-Objective Optimization

20184228 Seah Kim
20184400 Jeonggwan Lee
20186473 Liu Lingjun
20186505 Nick Heppert

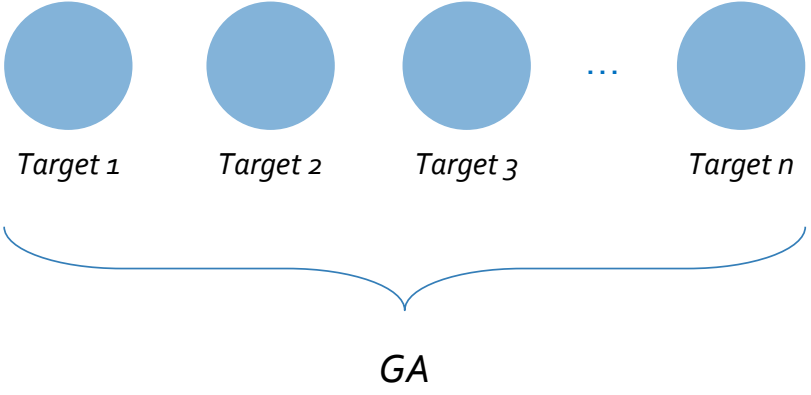
Multi-Objective Optimization

The order of each coverage target being executed is not optimized



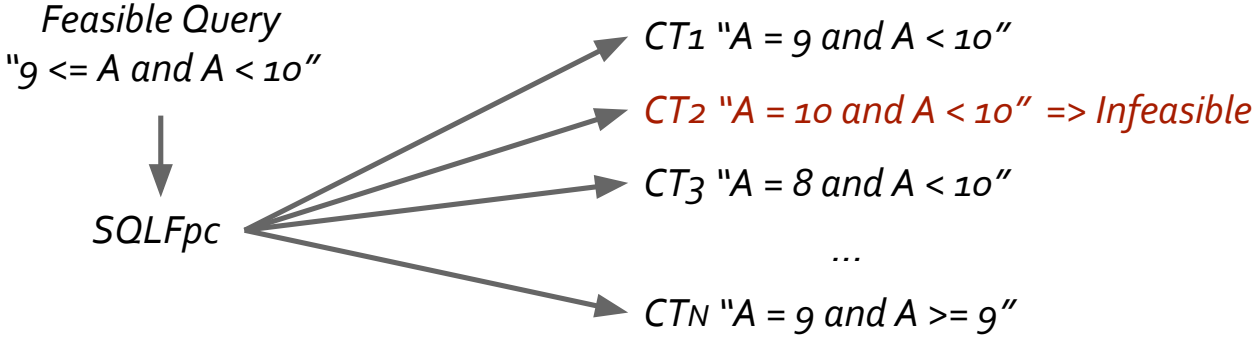
Multi-Objective Optimization

The order of each coverage target being executed is not optimized



Infeasible Coverage Targets in MOOP

Inefficient allocation of the budget might happen, such as infeasible coverage targets.



Issues we solved

- SQLfpc-website was not available from Korea

  → VPN to Germany

- Poor README on the Github page/No-out-of-the-box support

  → need to touch gradle.build-system

  → lose a lot of time compiling the program

Issues we solved

- Evaluation data was serialized with different version of the code

  → Deserialization issue

  → Re-serialization of whole data set

Plan

- EvoSQL baseline code analysis *(Done)*
- NSGA-II for MOOP *(In progress)*
- Infeasible targets
  - Early stop or detection of infeasible targets
  - by theoretically analyzing tendencies of infeasible targets
- Evaluate speed and coverage compared to EvoSQL

Issues we might face

- Existing code base clustered/some parts unmaintained
  → hard to add code
  → They have copy-paste-artifacts from online tutorials
- Lack of computational resources : could be hard to achieve the evaluation level of the paper
  → 10 (evaluations) * ~2000 (queries) * ~30 (min)
  = ~600,000 mins = ~10,000 hours = ~416 days
- Unsure if we can detect infeasible targets