The Structure Problem in Search-Based Software Engineering

Lu Zhang
Peking University
Shonan Meeting
October 2014

Agenda

- Introduction
- Example
- What is the Obstacle
- Some Thought

Introduction

- Power of Search Based Software Engineering
  - Widely applicable
- Obstacle Imposed by Structures
  - Solutions to many software engineering problem are highly structural
  - Existing search algorithms typically deal with very simple structures

Example: Test Case Prioritization

- Input: A set of test cases, whose coverage information is known

- Output: A permutation of the test cases to maximize a certain goal (e.g., APFD)

GA-Based Test Case Prioritization

- Each permutation is represented as a string
  - Two permutations for a set of five test cases
    - T1, T2, T3, T4, T5
    - T5, T4, T3, T2, T1
- How to define the mutation operator and the crossover operator?
  - Generate incorrect candidates and then correct them
  - Define problem specific operators

ILP-Based Test Case Prioritization

- Background
  - A set of (integer) variables
  - A set of linear equations and inequalities
  - A linear objective to maximize or minimize
- Variables for test cases

$$x_{ij} = \begin{cases} 1, & \text{if the } j\text{-th test case in } T' \text{ is } t_i \ (1 \leq i, j \leq n) \\ 0, & \text{otherwise} \end{cases}$$

- Constraints

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (1 \leq j \leq n)$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad (1 \leq i \leq n)$$

What is the Obstacle

- It's the structure
- A typical search framework requires a candidate solution to be in a very simple form
  - GA: A string
  - ILP: A set of variable
- We need to find a way to encode the structure into the simple form
  - Incur extra computational resources
  - Problem specific and ad hoc
  - Insufficiency to explore different structures

Some Thought

- Back to the example
  - A simple greedy strategy
    - Each time choose the test case that can cover the maximum number of yet uncovered statements
  - Empirical evidence
    - The greedy strategy typically achieves better effectiveness than GA and almost the same effectiveness as ILP
    - The greedy strategy is much faster
- What do we learn?
  - The greedy strategy does not explore the space of candidate solutions

Some Thought (cont.)

- What is in the nature?
  - Many species have complex structures
  - Each being grows from one cell (with encoded genes) to an individual with a complex structure
  - Structures are not explicitly encoded

Some Thought (cont.)

- A two dimensional model
  - Two types of spaces
    - The space formed with candidate solutions
    - The space formed with different structures of each solution
  - Maybe different search strategies

Questions and Discussions