

软件质量保证 第 3 次作业

赵睿哲 1200012778

May 29, 2016

测试用例化简的目标是找到满足测试需求集 R 的测试用例集 T 的最小代表子集 T' 。该过程可以用整数线性规划 (Integer Linear Programming, 简称 ILP) 中的特例: 0-1 整数线性规划来形式化描述。0-1 整数线性规划可以证明为 NP 完全的¹。

本文首先介绍使用 0-1 整数线性规划描述的测试用例化简过程, 进而给出该方法的时间复杂度 (时间开销)。

1 线性规划模型

给定一个测试用例集 T , 其中包含 n 个测试用例 $\{t_1, t_2, \dots, t_n\}$ 。给定向量 $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$, 其中 $x_i = 0/1$ 代表测试用例 i 是否保留在化简过的测试用例集中。由于本题要求测试需求为语句覆盖, 因此假设程序 P 中存在 m 条可达语句, 则测试需求集 $R = \{r_1, r_2, \dots, r_m\}$ 中, 任意需求 r_j 的含义为: 第 j 条语句需要被覆盖。

测试用例化简的目标是令保留的所有测试用例个数之和最小, 因此目标函数为最小化:

$$Z = \sum_{i=1}^n x_i \quad (1)$$

令矩阵 \mathbf{A} 表示测试用例对语句的覆盖情况, 其中 $a_{ij} = 0/1$ 代表对于测试用例 i 是否能覆盖语句 j 。应满足的约束条件为所有可达语句都被覆盖, 因此:

$$\mathbf{Ax} \geq \mathbf{1} \quad (2)$$

约束条件 2 的含义为任意一条可达语句都要被覆盖至少一次。

2 时间开销分析

使用最简单的求解算法, 即遍历所有的 \mathbf{x} 的情况, 分别判断是否满足所有的约束条件, 最终找到最小的测试用例集 T' 。则时间复杂度应该为:

$$T(m, n) = O(2^n mn) \quad (3)$$

其中 2^n 为所有的 \mathbf{x} 的组合情况, mn 为计算约束条件 2 的时间复杂度。

¹实际上, 0-1 整数线性规划问题是 Karp 的 21 个 NP 完全问题 (https://en.wikipedia.org/wiki/Karp's_21_NP-complete_problems) 之一