**Multi-Objective Particle Swarm Optimization-based**

**Test Data Generation for Path Coverage Testing**

**Abstract**

# 1. Introduction

Xem xét một program under test như sau:

int getDayNum(int year, int month)

{

int maxDay=0;

if( month>=1 && month<=12) //bch1: branch 1

{

if(month==2) //bch2: branch 2

{

if(year%400==0||(year%4==0&&year%100==0))

//bch3: branch 3

maxDay=29;

else //bch4: branch 4

maxDay=28;

}

else if(month==4||month==6||month==9||month==11)

//bch5: branch 5

maxDay=30;

else //bch6: branch 6

maxDay=31;

}

else //bch7: branch 7

maxDay=-1;

return maxDay;

}

Với program under test này, Mao [1] đã sử dụng giải thuật PSO với duy nhất 1 hàm fitness function là sự kết hợp của branch functions for branch predicates và branch weight. Phương pháp này đã chứng minh được rằng giải thuật PSO có hiệu quả hơn giải thuật GA trong việc sinh test data, tuy nhiên vẫn có nhược điểm là việc tính toán các branch weight cho một program under test vẫn hoàn toàn là công việc manual.

Cách tiếp cận của chúng tôi là, với mỗi branch predicate, thì tạo ra một fitness function. Sau đó chúng tôi sẽ sử dụng multi-objective particle swarm optimization để tìm nghiệm thỏa mãn các fitness function này.

# 2. Background

## 2.1. Fitness function

Table 1. The branch functions for several kinds of branch predicates

|  |  |  |
| --- | --- | --- |
| No | Predicate | Branch distance function *f*(bch*i*) |
| 1 | Boolean |  |
| 2 | ¬a |  |
| 3 | a = b |  |
| 4 | a ≠ b |  |
| 5 | a < b |  |
| 6 | a ≤ b |  |
| 7 | a > b |  |
| 8 | a ≥ b |  |
| 9 | a and b |  |
| 10 | a or b |  |

# 3. Related work

# 4. Proprosed approach

# 5. Experimental result

# 6. Conclusion

**References**

1. C. Mao: Generating Test Data for Software Structural Testing Based on Particle Swarm Optimization. Arabian Journal for Science and Engineering, vol 39, issue 6, pp 4593–4607 (June 2014).