Diploma thesis outline:

Java Combinatorial Optimization Package (JCOP)

Ondřej Skalička

5th year, FEL ČVUT, Karlovo nám. 13, 121 35 Praha 2

March 21, 2010

1 Introduction

1.1 Motivation

Why to create JCOP (missing a tool for easy comparison of different algorithms which allows simple problem-algorithm communication; education at FEL.CVUT)

1.2 Purpose

What should and should not be JCOP used for (benchmarking - comparing different algs)

1.3 State of the Art

Other similar projects (JCOOL, OAT, FakeGame...)

2 Combinatorial Problems

Where are they used, what for, theoretical importance , P/NP/NP-C, common problems (SAT, TSP, Knapsack...)

2.1 Algorithms

DFS/BFS (+other graph search based), local non-graph based (SA, Tabu), global search (genetics and variations)

3 Analysis

3.1 Use cases

What can different users (developer/student, teacher, ...) do

3.2 Functional/Nonfunctional/Other requirements

Exceptations what JCOP should be able to do, platform (why java) etc. (+mention tutorials)

4 Implementation

4.1 Technologies

Java, SVN, SourceForge, Enterprise Architect

4.2 JCOP

Main parts of platform, what is responsible for what

4.3 Implementation details

Core of thesis, details (text, source code examples, model/diagram screenshots) about all parts of JCOP, why, where, how about them

4.4 Adding new elements

How to add new algorithms/problems/conditions/renders/solvers

4.5 Tests

Unit tests (TestNG)

5 Experimental results

5.1 Expected/Real results

Expectations such as "genetics performs poorly on SAT" etc, benchmark these known result and confirm that JCOP works ok

6 Conclusion

6.1 Future work

Possible extensions (GUI, adding new elements as in 4.4, distributed execution..)

6.2 references

6.3 appendixes

Much like appendixes online, lists of all implemented problems/solver/algorithms, used utils/libraries etc.