Department of Mathematics and Computer Science
University of Southern Denmark, Odense

September 2, 2009
Marco Chiarandini

DM811 - Heuristics for Combinatorial Optimization

Assignment Sheet 2, Fall 2009

Due date: September 9. You are welcome to post all the answers in your Lecture Journal.

Running Assignment

1. The following are possible alternatives to consider in the design of a local search for GCP. They lead to the definition of the solution representation (and hence to the candidate solutions), the neighborhood structure and the evaluation function.

k in local search	assignment of colors to V	type of coloring
k-fixed	complete	proper
k-fixed	partial	proper
k-fixed	complete	unproper
k-fixed	partial	unproper
k-variable	complete	proper
k-variable	partial	proper
k-variable	complete	unproper
k-variable	partial	unproper

   - Determine which combination read in the rows of this table leads to a promising local search approach.
   - Implement your favorite.
   - How do you select the neighboring solution to move to? Does your local search have a natural termination or do you have to halt it?

2. A preprocessing rule is a trivial polynomial time procedure that simplifies a given instance of a problem by removing parts that are trivially solved. Often preprocessing rules can be applied repeatedly, as new simplifications may become available after a previous preprocessing. Think about possible preprocessing rules for the graph coloring problem.

3. Publish the results of your solvers in the wiki journal. Divide the results clearly into two sections:

   - Your best construction heuristic
   - Your best local search

   In each section report the results of 10 trials on each of the following instances:

   `queen11_11, queen12_12, queen13_13, queen14_14, queen15_15, queen16_16, DSJC1000.1, DSJC1000.5, DSJC1000.9, DSJC500.1, DSJC500.5, DSJC500.9` (available from the section Resources of the Assignments), using the following format:

```
CPRN instance_name solution seconds
```

Exercise 1

Definition 1 Travelling Salesman Problem
Input: *A graph $G = (V, E)$ and a cost function $\omega : V \times V \mapsto \mathbf{R}$.*
Task: *Find an Hamiltonian cycle of minimum cost.*

Design at least one construction heuristic and at least one local search (solution representation + neighborhood + evaluation function).

Exercise 2

Definition 2 Set Covering
Input: *Collection C of subsets of a finite set S and a weight function $\omega : C \mapsto \mathbf{R}$.*
Task: *Find a set cover for S, i.e., a subset $C' \subseteq C$ such that every element in S belongs to at least one member of C' and the sum of the costs associated with the subsets in C' is minimal.*

Is this problem NP-hard? If the problem is NP-hard, design at least one construction heuristic and at least one local search (solution representation + neighborhood + evaluation function).

Exercise 3

Definition 3 Single Machine Total Weighted Tardiness Problem
Input: *A set J of jobs $\{1, \dots, n\}$ to be processed on a single machine and for each job $j \in J$ a processing time p_j, a weight w_j and a due date d_j.*

Task: *Find a schedule that minimizes the total weighted tardiness $\sum_{j=1}^{n} w_j \cdot T_j$, where $T_j = \{C_j - d_j, 0\}$ (C_j completion time of job j).*

Is this problem NP-hard? If the problem is NP-hard, design at least one construction heuristic and at least one local search (solution representation + neighborhood + evaluation function).