T-79.4201 **Autumn 2006**
Search Problems and Algorithms
Tutorial 12
Problems

1. One corollary of the NFL theorem is that the expected value of any performance measure $\Phi(d^y_m)$ is independent of the optimisation algorithm a used, when the underlying objective function f is chosen uniformly at random from the space $\mathcal{Y}^{\mathcal{X}}$. To illustrate this result, compute explicitly the expected maximum value (i.e. $E[\max\{d^y(1), \ldots, d^y(m)\}]$) encountered in:

   (a) a local search of length $m = 2$ in the space of binary strings of length 2 ($\mathcal{X} = \{0,1\}^2$), when the range of the objective functions is $\mathcal{Y} = \{0,1\}$;

   (b) a local search of length $m = 3$ in the space of binary strings of length 3 ($\mathcal{X} = \{0,1\}^3$), when the range of the objective functions is $\mathcal{Y} = \{0,1,2\}$.

   (You do not need to verify that the expected maxima really are algorithm independent.)

2. Consider the following k-*Set Splitting* problem: Given a collection $\mathcal{C}$ of k-element subsets of a finite set S, is there a subset $S' \subseteq S$ such that no $C \in \mathcal{C}$ is contained in either S' or $S - S'$ (i.e., S' "splits" all the sets in $\mathcal{C}$ in two pieces). The problem is NP-complete for $k \geq 3$. Make an educated guess concerning the location of "hard instances" for this problem.

3. Consider the problem for which you programmed a local search method in your first programming assignment. Can you identify a parameter β in the problem analogous to the clauses-to-variables ratio α of the Satisfiability problem? At which values of β would you guess that your problem would be most difficult to solve? [*Highly optional:* Make some relevant computer experiments using your existing local-search code, e.g.: (a) plot the time evolution of the problem's objective function for different types of input instances (if there is a lot of variance in the time series, take averages over several runs with different random number sequences); (b) try to experimentally determine the region of "hard instances" for the problem.]