The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual
Chapter 7, Problem 7.25

This problem is clearly in NP (we can guess the subset of tests and check, for each pair of classes, that there exists one test in the subset that yields different answers for the two classes).

The problem is very closely related to set covering: we can think of the task of the test set as distinguishing between all pairs of classes and of each test as "covering" (distinguishing among) a collection of pairs; the problem is then to find the smallest collection of tests that, taken together, "cover" all pairs of classes. Thus we prove that the problem is NP-complete by transforming *X3C* to it.

The basic idea is to make every set element a class and every 3-set a test. However, a test collection corresponding to an exact cover would only manage to distinguish between groups of classes (3 classes per group); thus we need to add tests to distinguish among classes within a group—yet these additional tests should not help with any other breakdown into groups, because we want any solution to the test problem to include the equivalent of an exact cover by 3-sets. The solution is to create *a pair of* classes for each element of S and to use our 3-sets as tests that distinguish between the two classes in pair, but not between across pairs—for which purpose we shall have to add more tests.

Given a set $S = \{x_1, \ldots, x_{3n}\}$ and a collection of 3-sets, $\mathcal{C} = \{S_1, \ldots, S_k\}$, we construct an instance of our problem as follows. For each element x_i of S we create two classes y_i and z_i in our problem. For each 3-set $S_j = \{x_{j_1}, x_{j_2}, x_{j_3}\}$ in $\mathcal{C}$, we create a test $T_j = \{y_{j_1}, y_{j_2}, y_{j_3}\}$. We add $3n - 1$ tests, one for each i, $1 \le i < 3n$, of the form $\{y_i, z_i\}$. Finally, we set the bound to $4n - 1$. Clearly, this transformation can be carried out in polynomial time.

Note that the tests that correspond to 3-sets cannot distinguish among any of the z_is; each such test distinguishes 3 of the y_is from their matching z_is, and also discriminates among the y_is themselves. Note also that the additional $3n$ tests cannot distinguish between a y_i and its matching z_i, but that taken together they can narrow the identification down to a single $\{y_i, z_i\}$ pair. Finally, note that we will need all $3n - 1$ of the additional tests in order to be sure to distinguish among all of the z_is—omitting even one will leave at least one pair of these indistinguishable, since no help can be provided by the tests corresponding to 3-sets. Thus any solution to the problem will include all $3n - 1$ additional tests and thus also can include at most n of the tests corresponding to 3-sets. Since the $3n - 1$ additional tests narrow the identification down to a single $\{y_i, z_i\}$ pair, we only need tests to distinguish between a y_i and its corresponding z_i. Each test corresponding to a 3-set can provide that information for exactly 3 pairs; we need that information for all $3n$ pairs, so we can get it if and only if the tests selected carry information about n disjoint 3-sets, that is, if and only if the instance of *X3C* has a solution.