CME 305: Discrete Mathematics and Algorithms

Instructor: Reza Zadeh (rezab@stanford.edu)

Problem Session #2 – 02/15/17

1. Prove that the size of a maximal matching is at least $\frac{1}{2}$ of the size of maximum matching.

2. Given a network $G(V, E, s, t)$, describe a algorithm to determine whether G has a unique minimum $s - t$ cut.

3. An $n \times n$ grid is an undirected graph consisting of n rows and n columns of vertices, as shown below. We denote the vertex in the ith row and the jth column by (i, j). All vertices in a grid have exactly four neighbors, except for the boundary vertices, which are the points (i, j) for which $i = 1$, $i = n$, $j = 1$, or $j = n$.

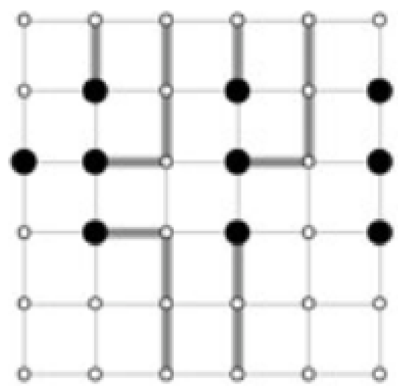

Figure 1: Grid for the escape problem. Starting points are black, and other grid vertices are white.

   Given $m \leq n^2$ starting points $(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$ in the grid, the escape problem is to determine whether or not there are m edge-disjoint paths from the starting points to any m different points on the boundary. For example, the grid in Figure 1 has an escape.

   Use maximum flow to give an efficient algorithm to solve the escape problem.

4. You've periodically helped the medical consulting firm Doctors Without Weekends on various hospital scheduling issues, and they've just come to you with a new problem. For each of the next n days, the hospital has determined the number of doctors they want on hand; thus, on day i, they have a requirement that *exactly* p_i doctors be present.

   There are k doctors, and each is asked to provide a list of days on which he or she is willing to work. Thus doctor j provides a set L_j of days on which he or she is willing to work.

The system produced by the consulting firm should take these lists and try to return to each doctor j a list L'_j with the following properties.

(A) L'_j is a subset of L_j, so that doctor j only works on days he or she finds acceptable.

(B) If we consider the whole set of lists $L'_1, \ldots, L'_k$ it causes exacty p_i doctors to be present on day i, for $i = 1, 2, \ldots, n$.

(a) Describe a polynomial-time algorithm that implements this system. Specifically, give a polynomial-time algorithm that takes the numbers $p_1, p_2, \ldots, p_n$, and the lists $L_1, \ldots, L_k$, and does one of the following two things.

- Return lists $L'_1, L'_2, \ldots, L'_k$ satisfying properties (A) and (B); or
- Report (correctly) that there is no set of lists $L'_1, L'_2, \ldots, L'_k$ that satisfies both properties (A) and (B).

(b) The hospital finds that the doctors tend to submit lists that are much too restrictive, and so it often happens that the system reports (correctly, but unfortunately) that no accecptable set of lists $L'_1, L'_2, \ldots, L'_k$ exists.

Thus the hopspital relaxes the requirements. They add a new parameter $c > 0$ and the system now should try to return to each doctor j a list L'_j with the following properties.

(A*) L'_j contains at most c days that do not appear on the list L_j.

(B) (*Same as before*) If we consider the whole set of lists $L'_1, \ldots, L'_k$ it causes exacty p_i doctors to be present on day i, for $i = 1, 2, \ldots, n$.

Describe a polynomial-time algorithm that implements this revised system. It should take numbers $p_1, p_2, \ldots, p_n$, the lists $L_1, L_2, \ldots, L_k$, and the parameter $c > 0$, and do one of the following things.

- Return lists $L'_1, L'_2, \ldots, L'_k$ satisfying properties (A*) and (B); or
- Report (correctly) that there is no set of lists $L'_1, L'_2, \ldots, L'_k$ that satisfies both properties (A*) and (B).

5. **Kleinberg and Tardos 11.9**

   Consider the following maximization version of the 3-Dimensional Matching Problem. Given disjoint sets X, Y, and Z, and given a set $T \subseteq X \times Y \times Z$ of ordered triples, a subset $M \subseteq T$ is a *3-dimensional matching* if each element of $X \cup Y \cup Z$ is contained in at most one of these triples. The *Maximum 3-Dimensional Matching Problem* is to find a 3-dimensional matching M of maximum size.

   Give a polynomial-time algorithm that finds a 3-dimensional matching of size at least 1/3 times the maximum possible size.

6. MAXCLIQUE is the problem of finding the maximum size clique in an undirected graph $G = (V, E)$. Show that this problem is NP-complete.

7. Recall that an instance of 3-SAT consists of formula in conjunctive normal form (CNF) where each clause has 3 literals each,

$$(X_1 \vee \bar{X}_3 \vee X_4) \wedge (\bar{X}_1 \vee X_2 \vee X_5) \ldots$$

In *Not-All-Equal* 4SAT (NAE4SAT), there are 4 literals per clause, and an instance is satisfiable iff there exists an assignment of values to variables such that there is at least one T and one F literal in each clause.

Show that NAE4SAT is NP-complete by a reduction from 3SAT.