Department of Electronics Engineering
National Chiao Tung University
Algorithms, Fall 2017

December 27, 2017
Iris Hui-Ru Jiang

Sample Solutions to Homework #4

1. (25)

   a. This is false. Consider a graph with nodes s, v, w, t, edges $(s, v), (v, w), (w, t)$, capacities of 2 on (s, v) and (w, t), and a capacity of 1 on (v, w). Then the maximum flow has value 1, and does not saturate the edge out of s.

   b. This is false. Consider a graph with nodes s, v_1, v_2, v_3, w, t edges (s, v_i) and (v_i, w) for each i, and an edge (w, t). There is a capacity of 4 on edge (w, t), and a capacity of 1 on all other edges. Then setting $A = \{s\}$ and $B = V - A$ gives a minimum cut, with capacity 3. But if we add one to every edge then this cut has capacity 6, more than the capacity of 5 on the cut with $B = \{t\}$ and $A = V - B$.

2. (25)

   The flow network is shown in Figure 1. The maximum flow algorithm gives the max number of apartments sold.

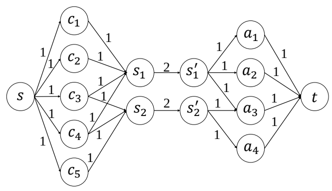

Figure 1: Flow network for problem 2.

3. (25)

   a. The decision problem VC_D can be formulated as follows: Given a graph $G(V, E)$, is there a vertex cover of size at most k?

   b. Given an instance ϕ of 3SAT with m clauses $C_1, \ldots, C_m$ and l literals $l_1, \ldots, l_l$, an graph construction of the reduction is shown in the given figure. The graph G is composed of two parts, the literal part (the upper nodes consisting of the l literals and their negations) and the clause part (the lower nodes forming m triangles). In the literal part, each literal and its negation are connected through an edge. In the clause part, edges exist among each triangle. Between the literal part and the clause part, nodes are connected if they define the same literal or the negation of a literal.

   For the reduction, we are going to take an instance of 3SAT (a boolean formula) and reduce it to a vertex cover instance that has a cover of size at most k if and only if the 3SAT formula has a satisfying assignment.

   If there is a satisfying assignment of the boolean 3SAT formula, a vertex cover of G can be found as follows: in the literal part, we add the True literals into the vertex cover, which is of size l. By choosing the l nodes, edges whose two end nodes are both in the literal part are covered. In addition, since ϕ is satisfied, we must have chosen at least one of the literals in each clause to be True, which indicates that in each triangle, the outgoing edge (the one going outside of the triangle) of at least one node is covered. Thus, we need only two vertices from each triangle to cover the rest of the edges. This yields $2m$ more vertices, and the generated vertex cover is of size $k = l + 2m$.

Conversely, assume there's a vertex cover of G of size $k = l + 2m$. We know that at least l vertices cover the edges in the literal parts because the edge between each literal and its negation cannot be covered in any other way. The other $2m$ vertices must cover the triangles since each triangle requires two vertices to be covered. We take the literal nodes (in the literal part) in the vertex cover to be our assignment of the formula. This assignment must satisfy the formula since every clause is satisfied by our graph construction

Finally, since this reduction constructs a graph with exactly $2l + 3m$ nodes and can be done in polynomial time, we have proved that VC_D is NP-complete.

4. (25)

   a. You can backup the array after every n insertions.

   b. Let $\phi_i = i \mod n$. Then when $i \mod n = 0$, $a_i = n + 0 - (n-1) = 1$. When $i \mod n \neq 0$, $a_i = 1 + (i \mod n) - ((i-1)) \mod n = 2$.