CS 331 Assignment #4

Greg Plaxton

March 5, 2014

NOTE ADDED 3/9/14. In Bullet 2 preceding the statement of Exercise 4, I have clarified the original wording by expanding "any item v' in $V - v$" to "any item v' in $V - v$ such that (u, v') belongs to E". Similarly, in Bullet 3 I have clarified the original wording by expanding "u is unmatched in M and" to "u is unmatched in M, (u, v) belongs to E, and". END OF NOTE.

There are two parts to this assignment. The first part, described in Section 1.1 below, consists of four paper-and-pencil exercises. You are encouraged to attend the discussion sections and office hours to get hints on how to solve these exercises. You are also permitted to ask for clarifications on Piazza. You are not allowed to work with other students on these exercises. This part is due *at the start of class* on Monday, March 24. Please note that no extension will be given on this part of the assignment!

The second part, described in Section 2 below, consists of recommended exercises. There is nothing to be turned in for this part of the assignment, as these exercises will not be graded. You are encouraged to work on the recommended exercises in order to prepare for the tests.

1 Programming & Problem Solving

In the programming portions of Assignments 1 through 3, you have implemented successively faster algorithms for computing a suitable allocation of items to bids in an auction of the type introduced in Assignment 1. Apart from computing an allocation, an auction mechanism needs to determine a suitable selling price for each item. For example, consider a simple auction setting where a single item is for sale, and there is a single round of bidding in which each bidder has the opportunity to submit a sealed bid specifying an offered amount for the item. In this simple setting, the "obvious" mechanism allocates the item to the highest bidder at a price equal to the offer of the highest bidder; this is often called a "first price" auction. The more subtle "second price" mechanism allocates the item to the highest bidder at a price equal to the second highest offer. The main advantage of the second price auction is that it is "strategyproof"; informally, this means that a bidder in a second price auction can never go wrong by submitting an offer equal to the true value that he or she places on the item. It is easy to see that this property does not hold for first price auctions.

In this assignment, we will develop a framework for pricing the items in the auction of Assignment 1. This framework is applicable to a broader class of auctions called "unit-demand" auctions. Accordingly, in Section 1.1 below, we introduce the class of unit-demand auctions and develop our pricing framework in this broader context. In the next assignment, we will further refine this framework, and you will write a program to compute prices for the auction of Assignment 1. The pricing scheme that you will implement in the next assignment generalizes the second price mechanism discussed in the preceding paragraph. Furthermore, like the second price auction, the pricing mechanism that we develop for unit-demand auctions is strategyproof.

1.1 Paper-and-Pencil Exercises

In this assignment we consider a generalization of the auction addressed in Assignments 1 through 3. As before, we have a set of items in the auction. Instead of linear or single-items bids, we consider a more general class of "unit-demand" bids. A *unit-demand bid* specifies a separate offered amount for each of a number of items, subject to the constraint that the bid should win at most one item. For example, in an auction with 10 items numbered from 0 to 9, a possible unit-demand bid might make an offer of \$10 for item 3, an offer of \$25 for item 5, and an offer of \$17 for item 8. Note that a linear bid or a single-item bid can be modeled as a unit-demand bid.

There is a natural way to adapt the notion of a "configuration" introduced in Assignment 2 to the more general setting of unit-demands bids. Specifically, we now define a *configuration* as an edge-weighted bipartite graph $G = (U, V, E)$ where V corresponds to the set of items, U corresponds to the set of unit-demand bids (including dummy bids, as discussed in the next paragraph), and for each bid u in U and item v in V there is an edge (u, v) in E with weight z if bid u includes an offer of z for item v. In what follows, for any edge (u, v) in E, we denote the weight of edge (u, v) by $w(u, v)$.

As in Assignments 1 through 3, we assume that each item has a reserve price, and for each item v in V, we include a dummy bid in the set U that makes an offer only on item v, where the amount of this offer is equal to the reserve price of v. In the present assignment, we extend the definition of an item so that each item also has a *start price*, which is required to be at most the reserve price. For each item v in V, we include a second dummy bid in the set U that makes an offer only on item v, where the amount of this offer is equal to the start price of v.

Given such a unit-demand auction, we would like to determine which bid wins each item, and at what price. In terms of the corresponding configuration $G = (U, V, E)$, we seek a solution (M, p) where M is an MCM of G and p is a *price vector* that has a component p_v for each item v in V. Since M is an MCM, it is easy to see that $|M| = |V|$ (due to the presence of the dummy bids). For each item v in V, there is a unique edge (u, v) in M, and this edge signifies that bid u wins item v. As the name suggests, the price vector specifies the selling prices for the items.

Remark: For the purposes of this assignment, we will assume that all offered amounts, reserve prices, start prices, and prices are integers, and are allowed to be negative. To motivate the possibility of negative values, suppose Bob wants to sell an old rusted-out car that is sitting in his front yard. Bob might be willing to pay someone $100 to win this item, since the winner would be required to remove it, thereby improving the appearance of Bob's front yard. For this reason, it might make sense for Bob to set the start and reserve prices to negative values. Note that doing so does not preclude the possibility that the car could ultimately sell for a positive price (due to a bidding war between junk dealers, say). End of remark.

Recall that in the stable marriage problem, we defined a notion of a "stable" solution in order to capture certain desirable properties of a solution. We will take a similar approach to the problem of determining a "good" solution for a configuration $G = (U, V, E)$ associated with a unit-demand auction. Specifically, we say that a solution (M, p) is *stable* if the following conditions hold.

1. For every edge (u, v) in M, we have $p_v \leq w(u, v)$. Informally, this condition says that if we decide to sell item v to bid u, then the price of item v cannot exceed the offer of u for v.

2. For every edge (u, v) in M, and every edge (u, v') in E, we have $w(u, v') - p_{v'} \leq w(u, v) - p_v$. This condition may be interpreted as requiring that if bid u wins item v then — under price vector p — item v is at least as "profitable" for u as any other item. To explain this intuition further, for any edge (u_0, v_0) in E, we think of the weight $w(u_0, v_0)$ as representing the value that (the agent associated with) bid u_0 assigns to item v_0; hence the expression $w(u_0, v_0) - p_{v_0}$ represents the profit that bid u_0 realizes by winning item v_0 at price p_{v_0}.

3. For every bid u that is unmatched in M, and every edge (u, v) in E, we have $w(u, v) \leq p_v$. Informally, this condition says that if we decide not to sell any item to bid u, then every item v that u made an offer on needs to be priced at or above u's offer.

A price vector p is said to be *stable* for configuration G if there exists a stable solution (M, p) for G.

Exercise 1. Let $G = (U, V, E)$ be a configuration. Prove that if p is a stable price vector for G and v is an item in V, then p_v is at least the start price of v.

Exercise 2. Let $G = (U, V, E)$ be a configuration, let (M, p) and (M', p) be stable solutions for G, let (u, v) be an edge in M, and assume that $w(u, v) > p_v$. Then u is matched to some item v' in M', and $w(u, v) - p_v = w(u, v') - p_{v'}$.

Exercise 3. Let $G = (U, V, E)$ be a configuration. Prove that if (M, p) is a stable solution for G, then M is an MWMCM of G. Hint: Let M' be an MWMCM of G, and make use of the fact that the bipartite graph $G' = (U, V, M \oplus M')$ consists of a collection of disjoint paths and cycles, where each path and cycle in the collection is of even length.

Using a similar argument as in the proof of Exercise 3, one can prove the following lemma.

Lemma 1. *Let $G = (U, V, E)$ be a configuration, let p be a stable price vector for G and let M be an MWMCM of G. Then (M, p) is a stable solution for G.*

For any configuration $G = (U, V, E)$, any MWMCM M of G, and any price vector p for G (note: p need not be stable), let *digraph*(G, M, p) denote the edge-weighted digraph $G' = (V', E')$ that is defined as follows. The vertex set V' is equal to $U \cup V$. The edge set E' is determined as follows.

1. For any bid-item pair (u, v) in M, there is a directed edge from v to u in E' with weight zero.
2. For any bid-item pair (u, v) in M and any item v' in $V - v$ such that (u, v') belongs to E and $w(u, v) - p_v \leq w(u, v') - p_{v'}$, there is a directed edge from u to v' in E' with weight $w(u, v') - p_{v'} - w(u, v) + p_v$; observe that this weight is nonnegative.
3. For any bid u in U and any item v in V such that u is unmatched in M, (u, v) belongs to E, and $w(u, v) \geq p_v$, there is a directed edge from u to v in E' with weight $w(u, v) - p_v$; observe that this weight is nonnegative.

Exercise 4. Let G be a configuration, let M be an MWMCM of G, and let p be a price vector for G. Prove that *digraph*(G, M, p) does not contain a directed cycle of positive weight.

Using a similar argument as for Exercise 4, one can prove the following lemma.

Lemma 2. *Let G be a configuration, let M be an MWMCM of G, let p be a price vector for G, let (u, v) be a bid-item pair in M such that $w(u, v) = p_v$, and let u' be an unmatched bid in M. Prove that digraph(G, M, p) does not contain a directed path of positive weight from u' to u.*

2 Recommended Exercises

1. Problem 7.3, page 415.
2. Problem 7.5, page 416.
3. Let $G = (V, E)$ be a flow network, and let f and f' be two maximum flows in G. Let S (resp., S') be the set of vertices that are reachable from the source in the residual network G_f (resp., $G_{f'}$). Prove that $S = S'$.
4. Let $G = (V, E)$ be a flow network, and let (S, T) and (S', T') be two minimum-capacity cuts of G. Prove that $(S \cap S', T \cup T')$ and $(S \cup S', T \cap T')$ are also minimum-capacity cuts of G.
5. Problem 8.1, page 505.
6. Problem 8.3, page 505.