CSC304: Algorithmic Game Theory and Mechanism Design Fall 2016

Allan Borodin (instructor)
Tyrone Strangway and Young Wu (TAs)

November 14, 2016

Lecture 17

- Announcements
  - I have posted the first 3 questions for the last assignment. I have added a fourth part to question 3.
  - The tests have been graded; average around 70%.
  - Usual policy as to requests for regrading.
  - There is a memorial service for Professor Kelly Gotlieb which I am attending tomorrow so that I cannot hold the usual office hours. I remind eveyone that I am available at other times on request or by dropping by (if I am free).
- Todays agenda
  - Finish discussion of (one-sided) matching markets.

Review of our matching markets discussion from lecture 16

Last lecture we defined the matching markets problem for one-sided markets with unit demand buyers and passive sellers/items.

A key concept was that of the demand set (of items) for a buyer and the demand graph $D(\mathbf{p})$.

The demand set and demand graph

Given a vector of prices $\mathbf{p}$, the demand set of items for agent i is $\{j : v_{i,j} - p_j \geq v_{i,k} - p_k \text{ for all } k\}$. The demand graph $D(\mathbf{p})$ is the unweighted bipartite graph where there is an edge between buyer i and item j if and only if j is in the demand set of i.

Recalling basic graph definitions

We recall that in a graph $G = (V, E)$, a matching is a set of edges $E' \subseteq E$ such that no vertex appears in more than one edge in E'. For a bipartite graph with $V = X \cup Y$ and $E \subseteq X \times Y$, a matching E' satisifes that every vertex in X and Y has degree at most 1 in $G' = (V, E')$.

A perfect matching in a bipartite such that every vertex is in the matching and this of course requires $|X| = |Y|$. A perfect matching is of course a maximum matching but not necessarily conversely.

We will show how to efficiently find a maximum matching in a unweighted bipartite graph and we will then be able to know if that matching is perfect or not.

Moreover, if the maximum matching is not perfect, the algorithm will exhibit a *constricted set* that is preventing a perfect matching. (In our application of matching markets, we will have found a *constricted set* of buyers which will determine a corresponding set of items whose prices need to be raised.

Constricted sets

Let $G = (V, E)$ be a $n \times n$ bipartite graph. Let $V = X \cup Y$ and for any subset $S \subseteq X$, let $N(S) = \{y \in Y : (x, y) \in E \text{ for some } x \in Y\}$. That is, $N(S) \subseteq Y$ is the *neighbourhood* of S.

Constricted sets

$S \subseteq X$ is a constricted set if $|N(S)| < |S|$

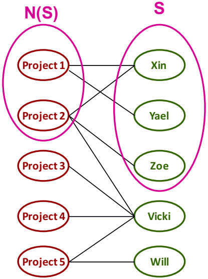


Figure: Example of a constricted set

An ascending auction algorithm for a matching market

We will do things a little different than the KP text. One simplification is that we are assuming integral valuations, so when we need to raise prices we just increases prices by 1. Our auction can raise more than one price in an iteration. But mainly the auctions follow the same idea and are based on the same basic result:

Hall's Marriage (aka Matching) Theorem

An $n \times m$ (with say $m \leq n$) bipartite graph $G = (V, E)$ with $V = X \times Y$ has a matching of size m if and only if for all $X' \subseteq X, |N(X')| \geq |X'|$

Hall's Theorem is stated and proved in Theorem 3.2.2 of the KP text. We will also sketch an constructive proof (i.e. an algorithm that either finds a maximum matching or finds a constricted set). An immediate consequence is that an $n \times n$ bipartite graph has a perfect matching if and only if there are no constricted sets.

An ascending auction template for a matching market continued

We let $\{v_{i,j}\}$ be the value of buyer i for item j. We let X be the set of items and Y the set of buyers.

An ascending auction for a matching market

Set the price vector $\mathbf{p} = (0, 0, \ldots, 0)$.
Let $D(\mathbf{p})$ be the demand graph.
Repeat until D has a perfect matching
Find a constricted set $S \subseteq X$ and raise the prices of all items in $N(S)$ by one unit. % There can be many constricted sets. For analysis, uniformly reduce prices so that the minimum price is 0.
Create a new demand graph for the updated prices
End Repeat

Theorem: The ascending auction terminates. A perfect matching in $D(\mathbf{p})$ is an envy-free allocation since every buyer is getting an item in their demand set.

Social welfare and termination in the ascending auction

We will show termination by what is called a *potential argument*. And in doing so, will show that the algorithm results in an allocation that is socially optimal.

Given the current prices $(p_1, \ldots, p_n)$ and the valuation profile of the buyers, define

- The potential of a buyer i is the utility $v_{i,j} - p_j$ for any item j in buyer is demand set.
- The potential of an item/seller j is its price p_j.
- The potential P_t of the auction (at any iteration t in the algorithm) is the total sum of all buyer and seller potentials.

Social welfare and termination in the ascending auction

We will show termination by what is called a *potential argument*. And in doing so, will show that the algorithm results in an allocation that is socially optimal.

Given the current prices $(p_1, \ldots, p_n)$ and the valuation profile of the buyers, define

- The potential of a buyer i is the utility $v_{i,j} - p_j$ for any item j in buyer is demand set.
- The potential of an item/seller j is its price p_j.
- The potential P_t of the auction (at any iteration t in the algorithm) is the total sum of all buyer and seller potentials.

Note that initially, the potential P_0 equals $\sum_i \max_j(v_{i,j})$.

Social welfare and termination in the ascending auction

We will show termination by what is called a *potential argument*. And in doing so, will show that the algorithm results in an allocation that is socially optimal.

Given the current prices $(p_1, \ldots, p_n)$ and the valuation profile of the buyers, define

- The potential of a buyer i is the utility $v_{i,j} - p_j$ for any item j in buyer is demand set.
- The potential of an item/seller j is its price p_j.
- The potential P_t of the auction (at any iteration t in the algorithm) is the total sum of all buyer and seller potentials.

Note that initially, the potential P_0 equals $\sum_i \max_j(v_{i,j})$.

Claim: After each iteration the potential decreases by at least 1. Why?

Conclusion: The auction must terminate and a perfect matching in the resulting demand graph provides a socially optimal allocation. Why?

An example of the ascending auction

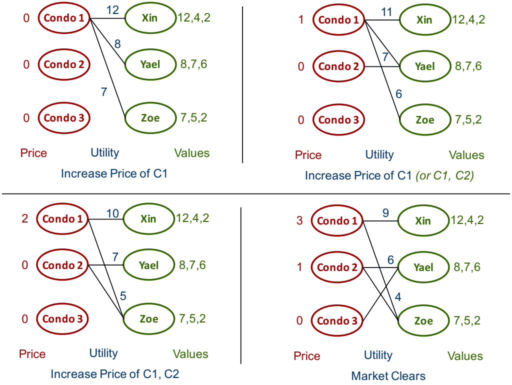


Figure: Example of the ascending auction

One not so subtle problem with the ascending auction

As stated, a buyer could be assigned to an item and receive negative utility. And we are assuming individual rationality since there is usually no reason a buyer will willingly accept an allocation resulting in negative utility. (Here we want ex-post IR since we are not in the Bayesian setting.)

One not so subtle problem with the ascending auction

As stated, a buyer could be assigned to an item and receive negative utility. And we are assuming individual rationality since there is usually no reason a buyer will willingly accept an allocation resulting in negative utility. (Here we want ex-post IR since we are not in the Bayesian setting.)

However, there is an easy way around this problem. Note that the demand graph $D(\mathbf{p})$ does not change if we uniformly raise or lower the price of each item by the same amount.

As stated in Easley and Kleinberg, at the end of each iteration we can reduce prices so that the lowest price is 0. Alternatively, at the end of the ascending auction we can eliminate any "sale" that resulted in negative utility. (By convention, we have been allowing sales having 0 utility.)

How to find a maximum matching or a constricted set

The Marriage theorem (also called the Matching Theorem) tells is that in a $n \times n$ bipartite graph, either there is a perfect matching or there is a constricted set. We will now sketch an algorithm that will find a perfect matching or find a constricted set; in doing so we are constructively proving the Hall Marriage Theorem.

I am going to follow the proof in the Easley and Kleinberg text "Networks, Crowds, and Markets". Their text can also be found online. Note that they refer to the demand graph as the *prefered seller graph (PSG)* but I think demand graph is more accepted.

The main concept in this proof is that of an an *augmenting path*.

Augmenting paths

Let M be a matching in a bipartite graph $G = (V, E)$ that is not of maximum size. Let $V = X \times Y$. Think of the Y nodes as representing buyers.

Augmenting Paths

An augmenting path is a simple path in G that starts at an unmatched node in Y and ends at a node in X. The path alternates between edges not in the matching and edges in the matchingr; i.e. the path starts with an edge not in the matching, then an edge in the matching, and then an edge not in the matching, ..., and ending in an edge not in the matching.

When do we have or not have an augmenting path?

Theorem: A matching is maximum if and only if there is no augmenting path.

If there is an augmenting path, then that path must contain exactly one more edge not in the matching than the number of edges in the matching. So we replace all the matched edges by unmatched edges increasing the

Augmenting path example from Chapter 10 of Easley-Kleinberg

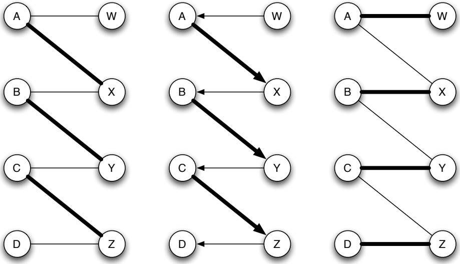
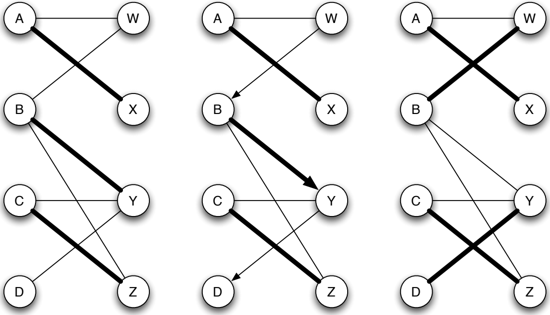


Figure: A simple example of an augmenting path leading to improved matching

Second augmenting path example from Chapter 10 of Easley-Kleinberg



(a) *A matching that is not of maximum size* (b) *An augmenting path* (c) *A larger (perfect) matching*

Figure: A somewhat more complicated example. Note than starting with the edge (W, A) does not yield an augmenting path whereas starting with edge (W, B) does lead to an augmenting path.

Finding a constricted set

Clearly, if there is an augmenting path, the matching is not a maximum matching. But what if we do not find an augmenting path?

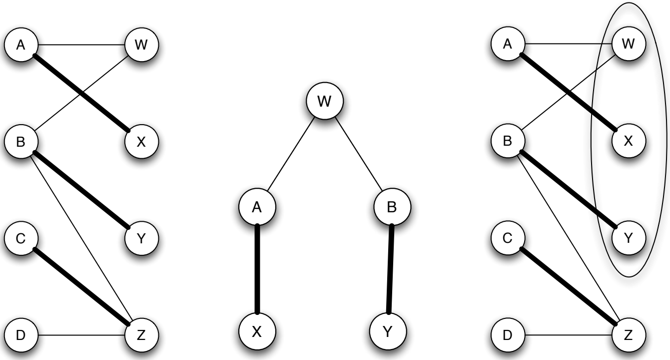


Figure: An example when there is no augmenting path thus yielding a constricted set

Using an “alternating breadth first search” to find an augmenting path or a constricted set

The last figure shows how a breadth first search (never repeating nodes already found) using alternating edges from a node W (not in the match) and ending with edges in the matching. This yielded a constricted set; namely the nodes at all the even levels (in this case, just the root W and the leaves A and B).

When we go from an odd numbered level (say level $2i-1$) for $i \geq 1$ to an even numbered level ($2i$), we have the same number of nodes at these two levels since we have matching between these two levels.

On the other hand if a node at an odd level cannot be continued (via an edge in the matching) then we have found an augmenting path.

Using an alternating breadth first search to find an augmenting path or a constricted set continued

Summarizing, from each unmatched node W in Y (e.g. the nodes representing buyers), we can run an "alternating breadth first search" starting at W. (If all nodes are matched, then there is nothing more to do.)

Clearly this breadth first search must terminate. If an augmenting path is not found then all the leaves are Y nodes. The constricted set consists of all nodes at even levels (including the root node W at level 0). (In our application, these are all nodes representing buyers.)

Using an alternating breadth first search to find an augmenting path or a constricted set continued

Summarizing, from each unmatched node W in Y (e.g. the nodes representing buyers), we can run an "alternating breadth first search" starting at W. (If all nodes are matched, then there is nothing more to do.)

Clearly this breadth first search must terminate. If an augmenting path is not found then all the leaves are Y nodes. The constricted set consists of all nodes at even levels (including the root node W at level 0). (In our application, these are all nodes representing buyers.)

The next figure presents a generic example of how a constricted set is found by an alternating breadth first search when the search failed to find an augmenting path.

Aside: For those familiar with the Ford Fulkerson method for the max flow problem, the maximum size bipartite problem reduces to max flow and augmenting paths correspond to paths in the residual graph.

A generic alternating breadth first search that failed to find an augmenting path

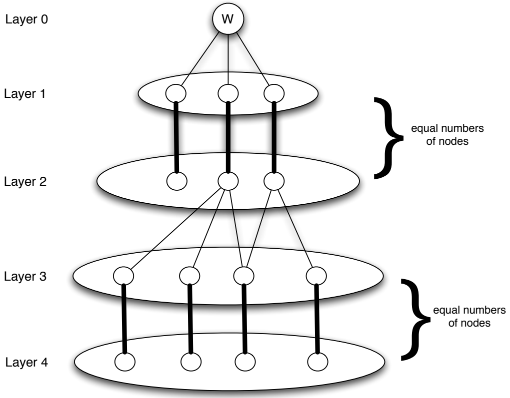


Figure: A generic example when there is no augmenting path thus yielding a constricted set

Returning to our matching markets application

We recall that in our application, we kept raising prices (for items corresponding to a constructed set of agents until we know that a perfect matching exists (and has been found by iteratively finding augmenting paths in the demand graph until no such augmenting path exists and a perfect matching has thus been found.

Returning to our matching markets application

We recall that in our application, we kept raising prices (for items corresponding to a constructed set of agents until we know that a perfect matching exists (and has been found by iteratively finding augmenting paths in the demand graph until no such augmenting path exists and a perfect matching has thus been found.

There need not be a unique perfect matching although any perfect matching gives buyers an item in their demand set.

Returning to our matching markets application

We recall that in our application, we kept raising prices (for items corresponding to a constructed set of agents until we know that a perfect matching exists (and has been found by iteratively finding augmenting paths in the demand graph until no such augmenting path exists and a perfect matching has thus been found.

There need not be a unique perfect matching although any perfect matching gives buyers an item in their demand set.

The prices that are derived are *market clearing* in that every item is sold (albeit perhaps at price 0) to a unique buyer . How much can we uniformly raise prices and still maintain the same demand graph.?

Returning to our matching markets application

We recall that in our application, we kept raising prices (for items corresponding to a constructed set of agents until we know that a perfect matching exists (and has been found by iteratively finding augmenting paths in the demand graph until no such augmenting path exists and a perfect matching has thus been found.

There need not be a unique perfect matching although any perfect matching gives buyers an item in their demand set.

The prices that are derived are *market clearing* in that every item is sold (albeit perhaps at price 0) to a unique buyer . How much can we uniformly raise prices and still maintain the same demand graph.?

Sellers can also set reserve prices $(r_1, \ldots, r_n)$ for their items. This reduces to the case of no reserve prices by changing the valuations to $w_{i,j} = \max(v_{i,j} - r_j, 0)$ and treating a negative utility $v_{i,j} < r_j$ for a match between agent i and item j as not letting that transaction take place. Alternatively, we can start with initial prices equal to reserve prices and again removing sales having negative utility for the buyer.

Some concluding comments about such market clearing prices

The matching provides a permutation π between buyers and items such that buyer i is matched with item $j = \pi(i)$ with j in buyer i's demand set. Hence the allocation is envy free.

Some concluding comments about such market clearing prices

The matching provides a permutation π between buyers and items such that buyer i is matched with item $j = \pi(i)$ with j in buyer i's demand set. Hence the allocation is envy free.

The resulting allocation is socially optimal. This follows as the allocation is maximizing $\sum_i [v(i, \pi(i)) - r_{\pi(i)}]$ which is equivalent to maximizing $\sum_i v(i, \pi(i))$

Some concluding comments about such market clearing prices

The matching provides a permutation π between buyers and items such that buyer i is matched with item $j = \pi(i)$ with j in buyer i's demand set. Hence the allocation is envy free.

The resulting allocation is socially optimal. This follows as the allocation is maximizing $\sum_i [v(i, \pi(i)) - r_{\pi(i)}]$ which is equivalent to maximizing $\sum_i v(i, \pi(i))$

If one chooses the minimum envy free price vector, the allocation is still optimal (with respect to social welfare), and the resulting prices are VCG prices.

Some concluding comments about such market clearing prices

The matching provides a permutation π between buyers and items such that buyer i is matched with item $j = \pi(i)$ with j in buyer i's demand set. Hence the allocation is envy free.

The resulting allocation is socially optimal. This follows as the allocation is maximizing $\sum_i [v(i, \pi(i)) - r_{\pi(i)}]$ which is equivalent to maximizing $\sum_i v(i, \pi(i))$

If one chooses the minimum envy free price vector, the allocation is still optimal (with respect to social welfare), and the resulting prices are VCG prices.

As stated in Lemmas 17.2.4, 17.2.5, and Theorem 17.2.6, the set of envy free price vectors forms a lattice in which the minimum envy free prices are the prices charged by the VCG mechanism.

Concluding remarks contnued

The observation about VCG being optimal for the sponsored search problem is actually a special case of the fact that VCG is optimal for the matching markets problem.

Concluding remarks contnued

The observation about VCG being optimal for the sponsored search problem is actually a special case of the fact that VCG is optimal for the matching markets problem.

But note that there are instances of CAs where VCG is *not* envy free.