

Lecture 5: Lavi Swamy Reduction

Lecturer: *Matt Weinberg*Scribe: *Dingli Yu*

These notes may contain errors, and are not intended for use as a primary source. I have included references at the end of the notes for those who wish detail surrounding skipped steps. Presentation taken from Chapter 12 of the AGT book [NRTV07].

1 Logistics

In the last lecture, we use configuration LP to solve combinatorial auctions given the valuation functions are gross substitute. We also give a constant approximation of auctions with XOS valuations by proposing a rounding scheme of the configuration LP.

In this lecture, we will cover another rounding algorithm for configuration LP, most of which was covered in COS 521 last semester. Nevertheless, we will cover more given the background we have learned.

2 The configuration LP

Let's first recall the configuration LP.

The configuration LP has exponentially many variables, $x_i(S)$ for each bidder i and set S , denoting the fraction of set S that bidder i receives. The linear function to maximize is $\sum_{i,S} v_i(S) \cdot x_i(S)$. The constraints are:

- For all i , $\sum_S x_i(S) \leq 1$ (dual variable u_i).
- For all j , $\sum_i, \sum_{S \ni j} x_i(S) \leq 1$ (dual variable p_j).
- $x_i(S) \geq 0$ for all i .

It should be clear that this is a fractional relaxation of our problem. Unfortunately, the LP has exponentially many variables, so we can't solve it directly. Fortunately, there are only polynomially many constraints, so we should consider looking at the dual. The dual aims to minimize $\sum_i u_i + \sum_j p_j$, such that:

- For all i, S , $u_i \geq v_i(S) - \sum_{j \in S} p_j$.
- For all i , $u_i \geq 0$.
- For all j , $p_j \geq 0$.

Recall that the dual can be solved with polynomially many demand queries, so we can find an optimal fractional solution with polynomially many demand queries.

3 Rounding for Arbitrary valuations

First, we'll show how to round a fractional solution to the configuration LP with arbitrary monotone valuations. In the second half of class, we'll show how to leverage this to get a truthful mechanism.

- Given as input $v_i(S)$ for all $(i, S) \in X$ (think of X as the set of (i, S) for which $x_i(S)$ is non-zero).
- Sort the elements of X in decreasing order of $v_i(S)/\sqrt{|S|}$.
- Initialize $A = \emptyset$ (the set of awarded items). Visit the elements of X in sorted order. When processing (i, S) , award bidder i set S if and only if $S \cap A = \emptyset$ and bidder i has not previously received a set. If S is awarded to i , update $A := A \cup S$.

We now wish to claim that the above algorithm guarantees a $\sqrt{2m}$ approximation to whatever fractional solution is initially given. Also note we do not make any further queries and assumptions on valuation functions in the algorithm.

THEOREM 1

Given as input $\{x_i(\cdot)\}_{i \in [n]}$, the above greedy algorithm above guarantees welfare at least $\sum_{i,S} x_i(S)v_i(S)/\sqrt{2m}$. Moreover, if there are k non-zero values of $x_i(S)$, the algorithm runs in time $\text{poly}(k)$.

PROOF: To prove this, observe that every pair in the list is either awarded (because the bidder had not received a set, and the items in that set were not previously awarded), or removed due to a conflict with a higher-priority pair. So our proof proceeds by proving that for each pair included, all of the lower-priority pairs with which it conflicts together do not contribute that much more welfare than this pair itself.

To see this, let some pair (i, S) be awarded, and let Y_i denote the pairs of lower (or equal, including (i, S) itself, assuming (i, S) is removed by itself) priority that conflict with (i, S) . Then we must have:

$$\sum_{(j,T) \in Y_i} x_j(T) \cdot v_j(T) = \sum_{(j,T) \in Y_i} x_j(T) \cdot v_j(T) / \sqrt{|T|} \cdot \sqrt{|T|} \leq \frac{v_i(S)}{\sqrt{|S|}} \cdot \sum_{(j,T) \in Y_i} x_j(T) \cdot \sqrt{|T|}.$$

The last inequality is simply observing that by definition, (j, T) is lower priority in the ordering than (i, S) . We now will use the Cauchy-Schwarz inequality on the two vectors $\vec{a} = \langle \sqrt{x_j(T)} \rangle_{(j,T) \in Y_i}$ and $\vec{b} = \langle \sqrt{x_j(T) \cdot |T|} \rangle_{(j,T) \in Y_i}$. Observe that the last sum above is exactly $\vec{a} \cdot \vec{b}$. Therefore, it is upper bounded by $|\vec{a}|_2 \cdot |\vec{b}|_2$, yielding:

$$\begin{aligned} \sum_{(j,T) \in Y_i} x_j(T) \cdot v_j(T) &\leq \frac{v_i(S)}{\sqrt{|S|}} \cdot \sum_{(j,T) \in Y_i} x_j(T) \cdot \sqrt{|T|} \\ &\leq \frac{v_i(S)}{\sqrt{|S|}} \cdot \sqrt{\left(\sum_{(j,T) \in Y_i} x_j(T) \right) \cdot \left(\sum_{(j,T) \in Y_i} x_j(T) \cdot |T| \right)}. \end{aligned}$$

Now we simply need to upper bound both terms that appear under the square root. Observe in the left sum that every $(j, T) \in Y_i$ either has $j = i$, or $T \cap S \neq \emptyset$. Therefore, this sum is upper bounded by $\sum_U x_i(U) + \sum_{\ell \in S} \sum_j \sum_{U \ni \ell} x_j(U)$. Further observe that the first sum (the sum over sets fractionally awarded to bidder i) is at most 1 by the constraints of the configuration LP. Moreover, for each $\ell \in S$, the inner sum (over all bidders and all sets awarded that contain ℓ) is also at most one by the constraints of the configuration LP. Therefore, this sum in total is at most $1 + |S|$.

For the right sum, this could be rewritten to sum first over all items, then over all sets (with each item contributing one each time it is counted) as:

$$\sum_{(j,T) \in Y_i} x_j(T) \cdot |T| = \sum_{\ell} \sum_{(j,T) \in Y_i, T \ni \ell} x_j(T).$$

The rewritten sum is now clearly upper bounded by $\sum_{\ell} \sum_j \sum_{U \ni \ell} x_j(U)$, where again the inner sums are upper bounded by 1, so the entire sum is upper bounded by m . Now, we conclude that:

$$\sum_{(j,T) \in Y_i} x_j(T) \cdot v_j(T) \leq \frac{v_i(S)}{\sqrt{|S|}} \cdot \sum_{(j,T) \in Y_i} x_j(T) \cdot \sqrt{|T|} \leq \frac{v_i(S)}{\sqrt{|S|}} \cdot \sqrt{|S| + 1} \cdot \sqrt{m} \leq v_i(S) \cdot \sqrt{2} \cdot \sqrt{m}.$$

So now the proof is complete: for all (j, T) , there is *some* (i, S) such that (i, S) blocks (j, T) (possibly (i, S) itself). So summing over all (i, S) that are accepted over all (j, T) that they block indeed sums over the entire fractional solution, which is at most

$$\sqrt{2m} \sum_{(i,S) \text{ awarded}} v_i(S).$$

□

To see that this entire procedure runs in polynomially many demand queries, observe that when solving the configuration LP, we may wlog assume that we get a solution which satisfies $n2^m$ of the constraints with equality (because there are $n2^m$ variables). Observe that only $n + m$ of the constraints are non-trivial, and the rest are of the form $x_i(S) \geq 0$. Therefore, there exists a solution to the configuration LP (which can be found via random perturbation of the objective) which has only $k = n + m$ non-zero $x_i(S)$ s.

4 The Lavi-Swamy Reduction: A truthful mechanism

Now, we show how to leverage the above rounding procedure to get a truthful mechanism. Note that it is unknown how to get a truthful mechanism from any of the rounding procedures discussed prior to this lecture, so try to keep an eye out for the special property of this specific one.

DEFINITION 1 *We say that a rounding algorithm \mathcal{A} verifies an integrality gap of $c \leq 1$ for the configuration LP if it takes as input v and outputs a (deterministic) allocation such that $\sum_i v_i(\mathcal{A}_i(v)) \geq c \cdot \text{CONFIGOPT}(v)$, where $\text{CONFIGOPT}(v)$ is the fractional optimum of the configuration LP. Note that we are insisting that the algorithm work for arbitrary valuations, even if we only care about our solution working for a restricted class of valuations.*

PROPOSITION 1

[[LS05]] Let \mathcal{A} verify an integrality gap of c for the configuration LP. Then for any fractional solution x to the configuration LP with k non-zero coordinates, one can decompose $c \cdot x$ into a distribution over integral allocations in $\text{poly}(n, m, k)$ black-box calls to \mathcal{A} (and $\text{poly}(n, m, k)$ overhead).

PROOF: Consider the following LP, which tries to write $c \cdot x$ as a convex combination of (perhaps exponentially many) integral allocations. Let \mathcal{I} denote the set of integral allocations.

$$\begin{aligned}
& \text{minimize } \sum_{y \in \mathcal{I}} \lambda_y \\
& \text{subject to } \sum_{y \in \mathcal{I}, y_i = S} \lambda_y = c \cdot x_i(S) \quad \forall (i, S) \text{ such that } x_i(S) > 0, \text{ dual variables } w_i(S) \\
& \quad \sum_{y \in \mathcal{I}} \lambda_y \geq 1 \quad \text{Dual variable } z \\
& \quad \lambda_y \geq 0
\end{aligned}$$

Here, the constraint $\sum_{y \in \mathcal{I}} \lambda_y \geq 1$ might seem redundant, but it will be helpful to keep the dual clean). Quickly observe that we don't need to enforce the constraint $\sum_{y \in \mathcal{I}, y_i = S} \lambda_y = 0$ when $x_i(S) = 0$, we can simply ignore any integral allocations that award set S to bidder i when $x_i(S) = 0$. Formally, one should instead update \mathcal{I} to be the set of integral allocations which never award bidder i a set S for which $x_i(S) = 0$. Finally, also observe that if we find a solution to this LP where $\sum_{y \in \mathcal{I}} \lambda_y = 1$, this is exactly a convex combination over integral allocations. So our goal is to find such a solution, via the dual:

$$\begin{aligned}
& \text{maximize } z + \sum_{(i, S), x_i(S) > 0} c x_i(S) \cdot w_i(S) \\
& \text{subject to } z + \sum_{(i, S), x_i(S) > 0, y_i = S} w_i(S) \leq 1 \quad \forall y \in \mathcal{I} \\
& \quad z \geq 0
\end{aligned}$$

We now want to claim that we can solve the dual in the desired runtime, given black-box access to \mathcal{A} , and that the value of the dual must be 1. First, observe that $z = 1, w_{i,S} = 0$ for all (i, S) is a valid dual solution and has value 1. So the dual has value at least one (this is why it was helpful to write the superfluous constraint in the primal, although we could have drawn the same conclusions without it and a little extra reasoning). Also observe that the dual has only $k + 1$ variables, so if we can get a separation oracle, we can solve it in time $\text{poly}(k)$.

Next we want to show that the dual has no feasible solutions with value > 1 . Assume for contradiction that (w, z) was such a solution. Then consider running algorithm \mathcal{A} on input valuations with $v_i(S) = w_i(S)$. Then this produces some integral allocation y with

$\sum_{i,S} X_i^y(S)w_i(S) \geq c \sum_{i,S} x_i(S)w_i(S) > 1 - z$. The last inequality follows from hypothesis that we started with a feasible solution of value > 1 . But now we can rearrange this into a violated constraint in the dual, and therefore the solution was in fact not feasible.

Note that the above indeed proves that the dual has value equal to 1, but also defines a poly-time separation oracle for any potential solution with value > 1 . (This may seem unnecessary since we just proved that the optimum is $z = 1$ and all $w_i(S) = 0$, but in order to recover the optimal primal from the optimal dual, we'll need to also solve perturbations of the dual, for which we'll actually need this separation oracle, as the trivial solution won't be the exact optimum). \square

This is the key proposition, let's now see how to use this to get a maximal in distributional range mechanism.

1. Let \mathcal{A} denote the algorithm that decomposes a point $\frac{1}{\sqrt{2m}} \cdot x$, where x is feasible for the configuration LP, into a distribution over integral allocations. This is based on the greedy algorithm, plugged through Proposition 1.
2. Let S denote the set of all distributions over integral allocations that \mathcal{A} might ever output on input x , where x is feasible for the configuration LP. Let \mathcal{B} be the algorithm that on input v , solves the configuration LP, then runs \mathcal{A} to get a distribution over integral allocations. **Importantly, note that \mathcal{B} finds the welfare maximizing distribution in S .**
3. Use the VCG payments with maximal-in-distributional-range algorithm \mathcal{B} . This is a truthful mechanism. Observe that it guarantees a $\frac{1}{\sqrt{2m}}$ -approximation.

The really important step is the bold sentence in step two. To see that, we claim any possible output of \mathcal{A} on any input x cannot be strictly better than $1/\sqrt{2m}$ of the configuration LP (otherwise, the configuration LP outputs a worse solution than x). Note that we could certainly run a different rounding algorithm and guarantee our same approximation ratio without the strong property provided by Proposition 1. The problem is that such a mechanism would not be maximal-in-distributional-range, so there wouldn't necessarily exist payments to make the procedure truthful. Observe also that in order to truly match the *distribution* of the fractional allocation x , we really needed an algorithm that verified the integrality gap of the underlying polytope, not just an algorithm that achieved an approximation guarantee.

References

- [LS05] Ron Lavi and Chaitanya Swamy. Truthful and near-optimal mechanism design via linear programming. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2005.
- [NRTV07] Noam Nisan, Tim Roughgarden, E. Tardos, and V. V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.