# Resource Allocation Algorithms

Lecture 11

Lecture by Dr. Tami Tamir
Typeset by Steven Karas

2018-05-31
Last edited 20:55:54 2018-05-31

**Disclaimer** These lecture notes are based on the lecture for the course Resource Allocation Algorithms, taught by Dr. Tami Tamir at IDC Herzliyah in the spring semester of 2017/2018. Sections may be based on the lecture slides written by Dr. Tami Tamir.

## 1 Recap

In the previous section, we covered job scheduling games as a special case of congestion games. We also defined coalitions as sets of players who can make a collective move that improves all their respective outcomes. A Strong Nash Equilibrium extends the weaker concept such that there are no collectives with positive utility moves. In a job scheduling game, there is always a strong NE.

We also discussed network congestion games, and necessary conditions for a NE to exist. If the players are weighted, there may not be a NE. If players can split their flow across multiple paths, then the price of anarchy grows by 2.5.

Finally, we began to cover mechanism design, which is a set of tools for designing fair auctions. In a set system $E, F$ where $E$ is the set of resources, each represented by a player, and $F$ is the set of all feasible subsets of resources. The goal of such an auction is to purchase a single feasible subset from $F$. Denote $c_e$ as the cost of actually using $e \in E$. Denote $b_e$ as the bid offered by the player for resource $e$. The protocol selects a subset of resources from $F$. Resources that were not selected get 0, and for each resource $e \in f$ where $f \in F$ is the selected feasible solution gets paid $p_e(b)$.

## 2 Auctions

### 2.1 Vickrey-Clarke-Groves Mechanisms

The traditional approach for designing fair mechanisms to find the socially optimal solution. Selects the minimal cost feasible subset, but doesn't necessarily pay the same as the bids.

The payoff for each resource $e$ is the maximum that $e$ could bid, if they had perfect information. This is equivalent to the optimal cost without using $e$, less the optimal cost if $e$ were free.

$$p(e, c) = \mathrm{OPT}(c(e) \to \infty) - \mathrm{OPT}(c(e) \to 0)$$

**Analysis of VCG for paths** Slides 21 to 24 have examples that show why the mechanism gives an incentive for truthfulness.

**Analysis of VCG for spanning trees** Slides 26 to 28 have examples that show why the mechanism gives an incentive for truthfulness.

A proof sketch for why such a mechanism is truthful is to do case-wise analysis of the incentives for if the resource is in the selected set, and if it benefits from either increasing or decreasing its bid.

### 2.2 Characterization of truthfulness

A path selection mechanism is truthful iff it has thresholds. In other words, given perfect information, there exists some $T_e$ such that if $b_e < T_e$ then $e$ wins (and gets paid $T_e$), and if $b_e > T_e$ then $e$ loses.

**Reasonable solutions** If there are only a handful of options, then the implied VCG price may be unreasonable. It's obvious that in scenarios with a monopoly, it's impossible to give a good bound on the solution price. Define the frugality ratio as a measure of how much competition there is in a given set system $E, F$ with mechanism $M$:

$$\text{FR}(M, (E, F)) = sup_c \frac{M\text{'s payments}}{\text{OPT}'}$$

Where $\text{OPT}'$ is the price of the next best disjoint solution.

### 2.2.1 Spanning trees

Slide 41 has a fully worked example that shows the frugality ratio for a $K_4$ graph.

### 2.2.2 Path auctions

$C_{>4}$ graphs are particularly bad for frugality. Consider the case where all edges cost 0, except one that costs 1. A VCG will pay $n-1$.

**Lower bound on frugality for paths** [1]

Let $G$ be a graph with two disjoint paths from $s$ to $t$: $p$ and $p'$ of lengths $k$ and $k'$. Define $k \cdot k'$ as the price vector for the graph. $c(i, j)$ is the price vector in which $i \in p$ has price $\frac{1}{\sqrt{k}}$ and $j \in p'$ has price $\frac{1}{\sqrt{k'}}$ and all others have price 0. For every such vector, the deterministic mechanism $M$ decides if it wants to purchase $p$ or $p'$.

Define a full bipartite graph with on one side all the edges in $p$ as vertexes, and on the other all the edges in $p'$. An edge from $i$ to $j$ exists if $M$ purchased $p'$ in $c(i, j)$, and from $j$ to $i$ otherwise. Denote the edges from $p$ to $p'$ as forward edges, and those from $p'$ to $p$ as backward edges. There are $kk'$ edges and w.o.l.g. at most $\frac{kk'}{2}$ forward edges. Note that this is the outgoing degree of the component $p$. Therefore, there is an edge $i \in p$ with an outgoing degree of at least $\frac{k'}{2}$.

Define the price vector $c(i, 0)$ such that $i \in p$ has a price $\frac{1}{\sqrt{k}}$ and all the others as 0. We claim that any mechanism will select $p'$ for $c(i, 0)$, and will pay for every edge with an incoming edge from $i$ in the bipartite graph at least $\frac{1}{\sqrt{k'}}$.

A mechanism must be monotonous, if a resource $e$ bids $b_e$, it must be selected if it bids $b'_e < b_e$. Moreover, the payment $M$ offers must be based on some threshold, which is the highest payment that still selects a given resource. We know that for every $j$ in the outgoing neighbors of $i$, this threshold is at least $\frac{1}{\sqrt{k'}}$, as this is the bid that won in the past (from $c(i, j)$).

From the claim, the payments of $M$ are at least $\frac{k'}{2} \cdot \frac{1}{\sqrt{k'}} = \frac{\sqrt{k'}}{2}$. Now consider the frugality ratio:

$$FR = \frac{\frac{\sqrt{k'}}{2}}{\frac{1}{\sqrt{k'}}} = \frac{\sqrt{k'}\sqrt{k'}}{2}$$

For $k = k' = \frac{n}{2}$, $FR = \Theta(\frac{n}{4})$.

**Reducing frugality** This is an active area of research. If we can change the graph, this can potentially improve the frugality. However, note that we can only remove resources, not add them. While counter-intuitive, removing a single edge can reduce the payment by a factor of $O(m)$.

An example of this can be found on slides 37, 38.

## 2.3 Digital Goods Auctions

In traditional auctions, the number of goods is limited. However, for digital goods, products are available in unlimited supply. The formulation we will use is where each player has a private utility value $u_i$. In a single round sealed bid auction, each player suggests his bid $b_i$. The outcome is a subset of bids that are fulfilled, where each player gets a price $p_i \leq b_i$. Denote the set of all bids as $B$.

The most obvious algorithm is to have each player pay his bid. However, this is unfair that different players pay different prices. We will only consider fixed price auctions, where we set a fixed price and all players who bid more than the fixed price pay that price.

[1] This proof was published in 2001, and is considered relatively new research

Given a set of bids $B = b_1 \geq b_2 \geq \cdots \geq b_n$, determine the price that maximizes revenue. Denote this price $OPT(B) = b_i \in B$ that maximizes $i \cdot b_i$. This is also called the threshold price.

The profit $F(B)$ is:

$$\max_i (i \cdot b_i)$$

It's clear that the fixed price profit is less than the simple profit. The ratio is tightly bound though: $F(B) \geq \frac{P(B)}{H_n}$, where $H_n = \sum_{i=1}^n \frac{1}{i}$. The proof is arithmetic:

$$P(B) = \sum_i b_i = \sum_i \frac{ib_i}{i} \leq \sum_i \frac{F(B)}{i} = F(B)H_n$$

### 2.3.1 Truthfulness

An auction is truthful if the dominant strategy for a bidder $i$ is $u_i$. A deterministic optimal threshold auction mechanism is to remove each bid $b_i$, and determine the best fixed price. If $b_i$ is at least this price, they are selected and pay that fixed price. However, this approach is not competitive. Consider an instance with two bid values $h$ and 1, with $r$ bids of $h$, and $(h-1)(r-1)$ bids of 1. A fully worked example can be found on slides 56-58. Generally, such a mechanism will provide profit $r$, while $P(B) = hr + (h-1)(r-1) \approx 2hr$, and $F(B) = hr$. It has been proven that all deterministic auctions, although they do work well in practice.

### 2.3.2 Random Sampling Optimal Price (RSOP)

1. Partition the bids into two sets $B'$ and $B''$ randomly (but fairly).
2. Compute $f' = \text{OPT}(B')$ and $f'' = \text{OPT}(B'')$, the optimal fixed sale prices for each partition.
3. Offer $f'$ to all bidders in $B''$ and $f''$ to all bidders in $B'$.

There is a proof[2] that an RSOP mechanism has a competitive ratio between 4 and 15, compared to $F^2(B)$.

# 3 Additional Algorithmic Game Theory Topics

This section will not appear on the exam.

## 3.1 Envy-free mechanisms

Mechanisms where stability is defined as when no player would want to swap with another. Cake cutting is one such problem.

Define valuation functions $v_i$ for each player $i$ in $[0, 1]$. An envy-free partition is one in which for every pair of players $i, j$, it holds that $v_i(X_i) \geq v_i(X_j)$.

**Assigning chores** Assign chores such that the makespan is minimal, but the assignment is fair.

## 3.2 Stackelberg Games

Asymmetric games in which players can control multiple resources.

## 3.3 Voting

How to conduct votes, possible rankings of winners. Arrow's impossibility theorem.

## 3.4 Stable matchings

The stable marriage problem. Applications for organ donation.

[2] Not proven here, but the paper will be uploaded later to the course site

# 4 Exam

Over the next two weeks, a list of slides we skipped will be published to help us focus on relevant material to study for the exam.

We will meet next in two weeks, where we will solve some practice problems. Example problems will also be uploaded.

# References

[1] Michael L Pinedo. *Scheduling: theory, algorithms, and systems.* Springer, 2016.