Algorithmic Game Theory

Important Results

Nicholas Hayek

Taught by Prof. Adrien Vetta

CONTENTS

We assume familiarity with some rudimentary definitions (e.g. preference, utility). Unless otherwise stated, agents are denoted $i \in [n]$.

I Mechanism Design

PAYOFF MATRICES

DEF 1.1 A *normal form game* describes the payoffs for players A and B across all strategy combinations. We can represent this game with one or two matrices. Typically, A plays rows and B plays columns:

$$A = \begin{pmatrix} 0 & 5 \\ 1 & 4 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 1 \\ 5 & 4 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 0,0 & 5,1 \\ 1,5 & 4,4 \end{pmatrix}$$

DEF 1.2 We call (A, B) a *symmetric game* if $A = B^T$, or, equivalently, $B = A^T$. In this case, A and B have the same payoff matrices from their respective points of view.

CHOICE FUNCTIONS

DEF 1.3 Let $\mathcal{O}$ be a set of outcomes, with a feasible set $\mathcal{S} \subseteq \mathcal{O}$. Let $f : \mathcal{S} \to \mathcal{S}$ be the choice function of a player with the preference relation $\succ$. A player is *consistent*, or independent of irrelevant alternatives, if

$$\forall \mathcal{S}' \subseteq \mathcal{S} \; f(\mathcal{S}) \in \mathcal{S}' \implies f(\mathcal{S}) = f(\mathcal{S}')$$

In other words, if $\mathcal{S}' \subseteq \mathcal{S}$ contains the desired outcome within $\mathcal{S}$, then the "irrelevant alternatives" within $\mathcal{S}$ do not influence the agent's choice.

PROP 1.1 Assuming that its preference relation is associative, an agent is rational if and only if it is consistent.

A rational agent has infinite compute and optimizes for his best outcome.

PROP 1.2 Satisficing by iterating through an ordered list till a sufficient option is chosen is a consistent strategy.

Dictatorships

Let Π be the space of possible preference functions of an agent. Let there be n agents, and let $\mathcal{O}$ be a set of outcomes. Then $f : \Pi^n \to \mathcal{O}$ is called a *social choice function*, and $f : \Pi^n \to \Pi$ is called a *social welfare function*. (DEF 1.4, DEF 1.5)

Elections are, at their core, social choice functions.

DEF 1.6 A *Condorcet winner* is a candidate that wins in every pairwise vote with an alternative candidate.

This candidate need not exist

DEF 1.7 A *dictatorship* is a social choice or welfare function in which $f(\Pi^n) = o$ or $\succ_D$, respectively, where $o \succ_D \cdots$ and $D \in [n]$. In particular, this holds for all Π^n.

A social choice function, or a mechanism generally, that allows for an agent to increase their utility by lying, is called *strategically manipulable*. If the mechanism is not strategically manipulable, i.e. truth-telling is a dominant strategy, we call it *incentive compatible*. DEF 1.8 DEF 1.9

A social welfare function f in which $a \succ_i b \ \forall i \implies a \succ b$ under f is called *unanimous*. DEF 1.10

Theorem 1.1 Gibbard-Satterthwaite

Any incentive compatible social choice function is a dictatorship.

Theorem 1.2 Arrow's Impossibility Theorem

Any social welfare function that satisfies unanimity and consistency is a dictatorship.

We remark that the proof of Thm 1.1 is a special application of Thm 1.2.

VCG MECHANISM

A mechanism is *individually rational* if it is a dominant strategy to participate. DEF 1.11

This, as opposed to preference lists

p_i is also a function of $\mathcal{S} \in \mathcal{O}$, but we omit this for notation

Let agents have real-valued valuations $v_i(\mathcal{S})$ over outcomes, where $\mathcal{S} \in \mathcal{O}$. Let $b_i(\mathcal{S})$ be an agents' bid for the outcome $\mathcal{S}$. These bids will be paid to the mechanism. Let p_i be the price that the mechanism charges an agent. Therefore, the utility of an agent is $u_i(\mathcal{S}) = v_i(\mathcal{S}) - p_i$.

With the money flowing as such, we claim that we can create a incentive compatible social choice function that is *not* a dictatorship.

The *VCG mechanism* chooses prices and outcomes as follows: DEF 1.12

1. The outcome $\mathcal{S}^*(\mathbf{b})$ of the VCG mechanism, given bids, maximizes social welfare. It does so by choosing $\mathcal{S}^*(\mathbf{b}) = \text{argmax}_{\mathcal{S}} \sum_{i=1}^{n} b_i(\mathcal{S})$

2. Prices are chosen such that (equivelantly)

   (a) The utility of an agent is its marginal contribution to social welfare:

   $$u_i(\mathcal{S}^*(\mathbf{b})) = \sum_{j \in [n]} v_j(\mathcal{S}^*(\mathbf{b})) - \max_{\mathcal{S}} \sum_{j \neq i \in [n]} v_j(\mathcal{S})$$

   (b) The price of a bidder is the damage it causes to the others by participating:

   $$p_i = \max_{\mathcal{S}} \sum_{j \neq i \in [n]} v_i(\mathcal{S}) - \sum_{j \neq i \in [n]} v_i(\mathcal{S}^*(\mathbf{b}))$$

   (c) The price a bidder pays is the lowest bid the bidder could have made and still obtained the same outcome.

   $$p_i = \min \hat{b}_i : S^*(\mathbf{b}) = S^*(\mathbf{b} \setminus b_i(\mathcal{S}^*(\mathbf{b})) \cup \hat{b}_i)$$

PROP 1.3 The VCG mechanism is individually rational and incentive compatible.

This explains the implicit claim in (1): the socially optimal outcome is given by $\operatorname{argmax}_{\mathcal{S}} \sum_{i=1}^{n} u_i(\mathcal{S}) = \operatorname{argmax}_{\mathcal{S}} \sum_{i=1}^{n} v_i(\mathcal{S})$, noting

$$\sum_{i=1}^{n} u_i(\mathcal{S}) = \sum_{i=1}^{n} v_i(\mathcal{S}) - \sum_{i=1}^{n} p_i = \sum_{i=1}^{n} v_i(\mathcal{S})$$

where we let the mechanism be a participant, such that $\sum_{i=1}^{n} p_i \equiv 0$. We call the LHS the "social welfare" and the RHS "market efficiency" and will not distinguish between them. Finally, then, $\sum_{i=1}^{n} v_i(\mathcal{S}) = \sum_{i=1}^{n} b_i(\mathcal{S})$.

DEF 1.13

Eg. 1.1.1 Let $\mathcal{O} = \{\mathcal{S}_1, ..., \mathcal{S}_n\}$ consist of outcomes in which agent i receives an item. Hence, $v_i(S_j) = 0$ if $i \neq j$. A *second price auction* chooses $\mathcal{S}^*(\mathbf{b}) = \operatorname{argmax}_{\mathcal{S}} \sum_{i=1}^{n} b_i(\mathcal{S}) = \operatorname{argmax}_{i\in[n]} b_i$, i.e. the highest bid. It then charges a price p_i equal to the second highest bid, i.e. $p_i = \max_{j\neq i} b_j$.

This is a special case of VCG.

Eg. 1.1.2 Let $\mathcal{O}$ = {trade, no trade}. There are two agents, a buyer and a seller, who value the item at v_b and v_s, respectively. The VCG mechanism trades if and only if $v_b \geq v_s$. In the trade case, $p_s = -v_b$ and $p_b = v_s$. In the no trade case, $p_s = p_b = 0$.

Eg. 1.1.3 Let $\mathcal{O}$ = {bridge, no bridge}. Each agent has a non-negative value for the bridge. There is a cost X for the whole population if built. To maximize social welfare, we build the bridge if and only if $\sum_{i=1}^{n} v_i \geq X$.

II Equilibria

CORRELATED EQUILIBRIA

Recall that a Nash equilibria is a set of actions such that every agent's action is a mutual best response to all other agents' actions. Hence, if agent i has payoff function π_i, a Nash equilibria $\mathbf{a}$ is such that

$$\pi_i(\mathbf{a}) = \max_{\hat{a}_i} \pi(\hat{a}_i, \mathbf{a}_{-i}) \;\forall i \in [n]$$

In mixed strategy Nash equilibria, agents have probabilities $\mathbf{p}_i$ over actions. As above

$$\mathbb{E}[\pi_i(\mathbf{p}_i)] = \max_{\hat{\mathbf{p}}_i} \mathbb{E}[\pi_i(\hat{\mathbf{p}}_i, \mathbf{p}_{-i})] \;\forall i \in [n]$$

In a MSNE, every pure strategy in the support of the mixed strategy of an agent must be a pure best response to the mixed strategies of the other agents.

PROP 2.1 Every finite game has a mixed strategy Nash equilibrium.

This is a consequence of the fact that any continuous function mapping a compact convex set into itself contains a fixed point.

Suppose agents are given "signals" to play a strategy. This is delivered via a probability distribution over all possible combinations of actions.

Let $\mathcal{S} = \{\mathcal{S}_1, ..., \mathcal{S}_m\}$ be strategy profiles, i.e. $\mathcal{S}_i = \{S_1, ..., S_n\}$ describes a combination of strategies over all agents. Suppose $p(\mathcal{S}_i) \in [0,1]$ with $\sum_{i=1}^m p(\mathcal{S}_i) = 1$. Then $\mathcal{S}$ is called a *correlated equilibrium* if, for each agent, obeying any given signal is a dominant strategy, assuming all other agents obey it as well. DEF 2.1

In other words, if agent i is told to play S_i^*, then for any deviation $\hat{S}_i$:

$$\sum_{\mathcal{S}} \left(\frac{p(S_i^*, \mathcal{S}_{-i})}{\sum_{\mathcal{S}} p(S_i^*, \mathcal{S}_{-i})} \right) \pi_i(S_i^*, \mathcal{S}_{-i}) \geq \sum_{\mathcal{S}} \left(\frac{p(S_i^*, \mathcal{S}_{-i})}{\sum_{\mathcal{S}} p(S_i^*, \mathcal{S}_{-i})} \right) \pi_i(\hat{S}_i, \mathcal{S}_{-i})$$
$$\iff \sum_{\mathcal{S}} p(S_i^*, \mathcal{S}_{-i}) \pi_i(S_i^*, \mathcal{S}_{-i}) \geq \sum_{\mathcal{S}} p(S_i^*, \mathcal{S}_{-i}) \pi_i(\hat{S}_i, \mathcal{S}_{-i}) \; \forall i, S_i^*, \hat{S}_i$$

On the other hand, if we propose a distribution $\mathcal{S}$ to an agent, and it prefers it to any deviating pure strategy, we call $\mathcal{S}$ a *coarse correlated equilibrium*: DEF 2.2

$$\sum_{\mathcal{S}} p(\mathcal{S}) \pi_i(\mathcal{S}) \geq \sum_{\mathcal{S}} p(\mathcal{S}) \pi_i(\hat{S}_i, \mathcal{S}_{-i})$$

POTENTIAL GAMES AND SMOOTH GAMES

Consider a game in which we wish to minimize costs, c_i. We call this a *potential game* if there exists a global function Φ such that DEF 2.3

$$\Phi(\mathcal{S}) - \Phi(\hat{S}_i, \mathcal{S}_{-i}) = c_i(\mathcal{S}) - c_i(\hat{S}_i, \mathcal{S}_{-i}) \; \forall i, \hat{S}_i$$

In such a game, we can make repeated single-agent improvements to arrive at a PSNE. Nash equilibria aren't always great at maximizing social good, so we define the following two metrics to gauge this:

The *price of stability* (PoS) is defined as DEF 2.4

$$\frac{\text{social cost of best equilibrium}}{\text{social cost of optimal solution}} \quad \text{or} \quad \frac{\text{social welfare of optimal solution}}{\text{social welfare of best equilibrium}}$$

depending on if we wish to minimize costs (left) or maximize welfare (right). In both cases, a good (smaller) PoS indicates that there exists some efficient Nash.

Similarly, we have the *price of anarchy* (PoA): DEF 2.5

$$\frac{\text{social cost of worst equilibrium}}{\text{social cost of optimal solution}} \quad \text{or} \quad \frac{\text{social welfare of optimal solution}}{\text{social welfare of worst equilibrium}}$$

A good (small) PoA indicates that the system will always function efficiently.

The *selfish routing game* consists of agents on a directed graph, where edges are endowed with an affine latency function, $\ell_e(x_e)$, which is a function of the agents occupying that DEF 2.6

edge, x_e. Each agent i wishes to travel from a node s_i to t_i, and does so by selecting a path P_i. The cost $c_i(\mathcal{P})$ to an agent is the sum of latencies among edges in its path, i.e. $\sum_{e \in P_i} \ell_e(x_e)$. This is a potential game, with th potential function

We sometimes denote $x_e = x_e(\mathcal{P})$ when the path is ambiguous

$$\Phi(\mathcal{P}) = \sum_{e \in E} (\ell_e(1) + \ldots + \ell_e(x_e(\mathbb{P})))$$

PROP 2.2 The selfish routing game has PoS ≤ 2.

Proving price of stability bounds is easiest by manipulating a potential function, e.g.

$$c(\mathcal{S}^*) \geq \Phi(\mathcal{S}^*) \geq \Phi(\mathcal{S}^N) \geq \text{PoS}^{-1} c(\mathcal{S}^N)$$

where $\mathcal{S}^*$ is optimal and $\mathcal{S}^N$ is the PSNE generated by iterating over the potential function.

Arriving at bounds for the price of anarchy is more straight forward.

DEF 2.7 Let $\mathcal{S}^*$ be the optimal solution and let $\mathcal{S}$ be a PSNE. A minimization game is called *smooth game* if

$$\sum_i c_i(\mathcal{S}_i^*, \mathcal{S}_{-i}) \leq \lambda \cdot c(\mathcal{S}^*) + \mu \cdot c(\mathcal{S}) \quad \lambda > 0, \mu < 1$$

In this case, we can easily upper bound the price of anarchy.

PROP 2.3 In a smooth minimization game, PoA $\leq \frac{\lambda}{1-\mu}$

Similarly, a maximization game is called smooth if

$$\sum_i \pi_i(\mathcal{S}_i^*, \mathcal{S}_{-i}) \geq \lambda \cdot W(\mathcal{S}^*) + \mu \cdot W(\mathcal{S}) \quad \lambda > 0, \mu < 0$$

PROP 2.4 In a smooth maximization game, PoA $\leq \frac{1-\mu}{\lambda}$

PROP 2.5 The selfish routing game is smooth, with

$$\sum_i c_i(\mathcal{S}_i^*, \mathcal{S}_{-i}) \leq \frac{5}{3} c(\mathcal{S}^*) + \frac{1}{3} c(\mathcal{S}) \implies \text{PoA} \leq \frac{5/3}{1 - 1/3} = \frac{5}{2}$$

MINIMAX THEOREM

Suppose Alice (A, x) and Bob (B, y) play in a normal form game (Def 1.1) with mixed strategies. If Alice is risk-averse, she should assume that Bob is malicious, and hence play

$$\text{argmax}_x[\min_y x^T A y] = \text{argmax}_x[\min_c x^T A^{(c)}]$$

where we note that $\min_y x^T A y = \min_c x^T A^{(c)}$, since $(x^T A) y$ is minimized by choosing the lowest coordinate of $x^T A$.

DEF 2.8 Following this discussion, we call $\mathbf{x}_s = \text{argmax}_x[\min_c x^T A^{(c)}]$ the *safety strategy* for Alice.

DEF 2.9 Her *maximin value* is then $v_s^A = \max_x[\min_c x^T A^{(c)}] = \min_c \mathbf{x}_s^T A^{(c)}$.

Now, suppose that Alice is completely malicious. Assuming that Bob wishes to maximize his own payoff, she should play

$$\text{argmin}_x[\max_y x^T By] = \text{argmin}_x[\max_c x^T B^{(c)}]$$

Alice's *threat strategy* is then $\mathbf{x}_t = \text{argmin}_x[\max_c x^T B^{(c)}]$. The minimax'ed value above is Bob's payoff, so Bob's associated *minimax value* is $v_t^B = \min_x[\max_c x^T B^{(c)}] = \max_c \mathbf{x}_t^T B^{(c)}$. DEF 2.10 DEF 2.11

Similarly, when Bob plays safety and threat strategies, we have

$$\mathbf{y}_s = \text{argmax}_y[\min_x x^T By] = \text{argmax}_y[\min_r B_{(r)}y] \implies v_s^B = \max_y[\min_r B_{(r)}y] = \min_r B_{(r)}\mathbf{y}_s$$

$$\mathbf{y}_t = \text{argmin}_y[\max_x x^T Ay] = \text{argmin}_y[\max_r A_{(r)}y] \implies v_t^A = \min_y[\max_r A_{(r)}y] = \max_r A_{(r)}\mathbf{y}_t$$

A *primal linear program*, where $A, \mathbf{c}, \mathbf{b}$ are fixed, is given by DEF 2.12

$$\max \mathbf{c}^T\mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0$$

Any primal may be converted to a *dual linear program*, as follows DEF 2.13

$$\min \mathbf{b}^T\mathbf{x} : A^T\mathbf{x} \geq \mathbf{c}, \mathbf{x} \geq 0$$

If $\mathbf{x}$ is a primal solution and $\mathbf{y}$ is a dual solution, then $\mathbf{c}^T\mathbf{x} \leq \mathbf{b}^T\mathbf{y}$. This is called the Weak Duality Theorem, and its easily verified: **PROP 2.6**

PROOF.

$\mathbf{c}^T\mathbf{x} \leq (A^T\mathbf{y})^T\mathbf{x} = \mathbf{y}^T A\mathbf{x} \leq \mathbf{y}^T\mathbf{b} = \mathbf{b}^T\mathbf{y}$ □

Theorem 2.1 Strong Duality Theorem

If $\mathbf{x}$ is a primal solution and $\mathbf{y}$ is a dual solution, then $\mathbf{c}^T\mathbf{x} = \mathbf{b}^T\mathbf{y}$.

For zero-sum games, $\max_x[\min_y x^T Ay] = \min_y[\max_x x^T Ay] \implies v_s^A = v_t^A$. **PROP 2.7**

In a zero-sum game, i.e. $A = -B$, we also have

$$v_t^A = v_s^A = -v_t^B = -v_s^B$$

In particular,

$$\text{argmax}_x[\min_y x^T Ay] = \text{argmax}_x[\min_y -x^T By] = \text{argmax}_x[-\max_y x^T By] = \text{argmin}_x[\max_y x^T By]$$

Hence, Alice's safety strategy is the same as her threat strategy. It follows that the safety strategies of Bob and Alice form a Nash equilibrium.

In small 2-player zero-sum games, to compute threat and safety strategies, find when the inner max/minimizer, simplified to consider only columns or rows, has equivalent value. **PROP 2.8**

Theorem 2.2 Lemke-Howson Algorithm

Let (A, A^T) be a symmetric game (see Def 1.2). Let x be a mixed strategy. The a MSNE is given by $Ax \leq 1, x \geq 0$. Let $1^*, ..., n^*$ denote the constraints associated with $x_i \geq 0$. Let $1, ..., n$ denote the constraints associated with $A_{(i)}x \leq 1$.

The following finds a MSNE:

Start at constraints $(1^*, 2^*, 3^*)$. Loosen one constraint, usually 1^*. Identify the first new constraint that is met. Repeat this without backtracking until you see a distinct trio of numbers.

III Market Equilibria

HOUSE TRADING GAME

Suppose agents have one house each. They desire other agents' houses according to a preference list. For example, if agent 1 likes his house most, then perhaps

$$1 \succ_1 7 \succ_1 3 \succ_1 \cdots$$

We would like to initiate an inventive compatible set of trades.

DEF 3.1 Let H denote a subset of houses. Then the *trading graph* G_H is the directed graph of vertices in H, with $i \mapsto j$ if $j \succ_i \cdots$, i.e. j is i's first preference.

Theorem 3.1 Top Trading Cycle Algorithm

Repeat until $H = \emptyset$

Identify vertices C that are in directed cycles in G_H

$H \leftarrow H \setminus C$

Return trades according to the directed cycles that were identified.

This algorithm is both incentive compatible and runs in $O(n^2)$.

DEF 3.2 A *pareto improvement* is a reallocation (e.g. over the mechanism's allocation) such that at least one agent is strictly better-off, and all other agents are at least as well-off. We call DEF 3.3 an allocation *pareto optimal* if no pareto improvement exists.

DEF 3.4 Similarly, an allocation $\mathcal{O}$ is said to be in the *core* if no subset A of agents can find a pareto improvement amongst themselves. If such a subset does exist, we call it a *blocking* DEF 3.5 *coalition*.

PROP 3.1 The house-trading game has a unique core solution and the top trading cycle algorithm always finds it.

When used for kidney exchanges, the top trading cycle algorithm is in infeasible when cycles are too long, as surgeries must be simultaneous in order to avoid breaking the cycle.

DATING GAME

Similar to the house allocation game, consider two sets of agents (B and G), where agents in one set have a preference ordering on agents in the opposing set. We wish to establish a matching between sets B and G. Furthermore, we desire a *stable matching*: there is no pair $\{b, g\}$ such that each prefer eachother over their partner in the matching (we would call this a blocking pair). DEF 3.6

Theorem 3.2 Deferred Acceptance Algorithm

Repeat until all boys are matched

Select an unmatched boy b

Boy b proposes to his favorite girl that hasn't rejected him:

If g is unmatched then she tentatively matches b

Else if g is tentatively matched to b' then she tentatively matches with her favorite amongst b and b'.

The following are true with respect to Thm 3.2: PROP 3.2

1. The algorithm returns a stable matching, and does so in $O(n^2)$.
2. As the algorithm runs, the quality of matches strictly degrades for the boys.
3. As the algorithm runs, the quality of matches strictly improves for the girls.
4. Of all stable matchings, the algorithm returns the optimal one for the boys.
5. Of all stable matchings, the algorithm returns the worst one for the girls.
6. The algorithm is incentive compatible for the boys, but not the girls.

CORE PRICING

Let I be the set of agents. Let $v(S)$ be the value that a coalition $S \subseteq I$ could guarantee themselves. We allocate $v(I)$ to agents such that each coalition receives at least $v(S)$. If this allocation is feasible, then we call it a *core payment*. DEF 3.7

Typically we formulate a core solution as follows:

$$\min \sum_{i \in I} p_i : \sum_{i \in S} x_i \geq v(S) \ \forall S \subseteq I, x_i \geq 0 \ \forall i \in I$$

In the case of combinatorial auctions, let $I = [n]$ be the set of agents and the auctioneer. Let $\mathcal{S}^* = \{S_1^*, ..., S_n^*\}$ be the optimal allocation, and let $X \subseteq I$ contain the auctioneer. Let

$v(X)$ be welfare resulting from the optimal allocation among members in X. Then core payments dictate that

$$\sum_{i \in X}(v_i(S_i^*) - p_i) + \sum_{i \in I} p_i \geq v(X) \implies \sum_{i \in I \setminus X} p_i \geq v(X) - \sum_{i \in X} v_i(S_i^*)$$

FISHER MARKETS

DEF 3.8 In a *Walrasian market*, agents are given initial allocations of items $\mathbf{a}_i \in \mathbb{Z}^m$. Each agent has a value $u_i(\mathbf{x})$ for any bundle. Given prices on the goods $\mathbf{p}$, the agent can sell off their allocation for a gain of $\mathbf{p} \cdot \mathbf{a}_i$.

DEF 3.9 A *Walrasian equilibrium* is a price vector $\mathbf{p}$ and allocation of goods such that

1. Each buyer purchases its maximum value bundle.
2. There is no excess supply or demand, i.e.

$$\sum_{i=1}^{n} x_{ij} = \sum_{i=1}^{n} a_{ij} \; \forall j \in [m]$$

DEF 3.10 Such a price vector is called a *set of market clearing prices*.

Theorem 3.3 Fundamental Theorems of Welfare

1. A Walrasian equilibrium is Pareto optimal, assuming monotonic utility functions.
2. Any Pareto optimal solution can be supported as a market equilibrium, given a suitable reallocation of endowments.

DEF 3.11 In a *Fisher market*, there are multiple units q_j of good j, and an agent i is endowed with a budge of b_i dollars. Furthermore, agents have linear valuation functions, i.e. $v_i(\mathbf{x}) = \mathbf{v}_i \cdot \mathbf{x}$ for some $\mathbf{v}$.

Note that the Fisher market is a special case of the Walrasian market. For instance, we can set endowments so that $a_{ij} = \frac{b_i}{\sum_{j=1}^n b_k} q_j$. Hence, we know that market clearing prices exist.

We will consider a special case of Fisher market in which agents have desired goods. In particular, let Γ_i denote the set of target goods desired by agent i, and let an agent have value 1 for every item it obtains in its target set (and 0 otherwise).

PROP 3.3 Denote $q(\hat{J}) = \sum_{j \in \hat{J}} q_j$ and $b(I) = \sum_{i \in I} b_i$. Let $\Gamma(\hat{J}) = \{i : \exists j \in \hat{J} : j \in \Gamma_i\}$. We can think of $\Gamma(\hat{J})$ as the preimage of Γ_i. Let $\mathbf{p}$ be a uniform price vector, i.e. $\mathbf{p} = \langle p, ..., p \rangle$ for some $p \in \mathbb{R}$. If $\mathbf{p}$ is market clearing, then $p \cdot q(\hat{J}) \leq b(\Gamma(\hat{J})) \; \forall \hat{J} \subseteq J$.

PROOF. Suppose not. Then fix $\hat{J}$ with $pq(\hat{J}) > b(\Gamma(\hat{J}))$. Then the agents in $\Gamma(\hat{J})$ cannot afford all the goods in $\hat{J}$. No other agents have any value for goods in $\hat{J}$, so they will not spend any money in acquiring them. It follows that we have excess supply. $\square$

This will not guarantee us market clearing prices, and, in fact, this is unlikely to occur.

Conversely, we can show that, when $p \cdot q(\hat{J}) \leq b(\Gamma(\hat{J})) \; \forall \hat{J} \subseteq J$, $\mathbf{p}$ has no excess supply.

Let (G, A) be a bipartite digraph of agents and goods, with $g \mapsto a$ if $g \in \Gamma_a$. We put a capacity of ∞ on these edges. Let s and t be sources and sinks associated with G and A, respectively, with s directed toward all of G and all of A directed into t. Let these edges have capacity $p \cdot q_j$ and b_i, respectively, for $j \in G$ and $i \in A$.

By min cut-max flow theorem

PROP 3.4 If we find a flow on (G, A) of value $pq(J)$, then $\mathbf{p}$ has no excess supply.

Instead of clearing the whole market at a price p, we can clear "submarkets."

To do so, at step i, find the smallest price p_i such that there exists a list of goods J_i with $p_i q(J_i) = b(\Gamma(J_i))$. Then, remove J_i and $\Gamma(J_i)$ from our set of goods and buyers. As a result, p_i will be market clearing for the submarket consisting of J_i and $\Gamma(J_i)$.

As a result, we have $p_1 \leq p_2 \leq \cdots$, and $\{p_1, ..., p_\ell\}$ are market clearing for the global market.

MATCHING MARKETS

DEF 3.12 In a *matching market*, we have indivisible goods, with n buyers and n sellers, each of which sells one item to one buyer (e.g. houses). Each buyer has a distinct value assigned to each good. Naturally, buyers want to maximize their profits (value of items bought minus price paid) and sellers want to maximize their sale price.

In other words, buyer i wishes to buy from $\text{argmax}_j v_{ij} - p_j$. Denote by $H(\mathbf{p}) = (B, S)$ the bipartite graph consisting of $i \mapsto \text{argmax}_j v_{ij} - p_j$ edges. We call this the *best response graph*. **DEF 3.13**

A set of prices $\mathbf{p}$ are market-clearing if $H(\mathbf{p})$ contains a perfect matching (this matching dictates transactions). Recall that, to be market clearing, there must be no excess supply or demand, and each buyer receives its max-utility outcome. We will set $\mathbf{p}$ such that $H(\mathbf{p})$ contains a perfect matching using Hall's condition.

Hall's condition

PROP 3.5 A bipartite graph $(X, Y) : |X| = |Y|$ contains a perfect matching if $|X'| \leq |\Gamma(X')|$ for each $X' \subseteq X$, where $\Gamma(X')$ denote the collection of neighbors of X'. If there exists a set X' with $|X'| > |\Gamma(X')|$, we call X' an *unsupportable set*. **DEF 3.14**

Theorem 3.4 Market Matching Algorithm

$\mathbf{p} \leftarrow 0$

Repeat until $M(\mathbf{p})$ contains a perfect matching

Find an unsupportable set X in $M(\mathbf{p})$

Increases prices by 1 in $\Gamma(X)$

If $\mathbf{p}_i > 0 \; \forall i$ **then** decrease all prices by 1.

If $\mathbf{p}$ is market-clearing, then the matching outputted by Thm 3.4 maximizes social welfare.

COMBINATORIAL AUCTIONS

Again, we assume that goods are indivisible. This time, agents are allowed to purchase multiple goods, and have a valuation $v_i(S)$ for a set of goods S. Similar to Def 3.13, we
DEF 3.15 call $S_i^* = \operatorname{argmax}_S v_i(S) - \sum_{j \in S} p_i$ the *demand bundle* for agent i at prices $\mathbf{p}$.

In a combinatorial market, we say that an allocation $\{S_1^*, \ldots, S_n^*\}$ and price $\mathbf{p}$ is a Walrasian equilibrium if

1. Every agent receives their demand bundle.
2. There is no excess supply (all unsold items have price 0)
3. There is no excess demand (the demand bundles are disjoint)

Instead of searching directly for the market equilibrium, we can instead investigate the socially optimal allocation. This can be accomplished with a linear program:

$$\begin{aligned}
&\max \sum_{i=1}^{n} \sum_{S \subseteq [m]} v_i(S) x_{i,S} \\
\text{s.t.} \quad &\sum_{i=1}^{n} \sum_{S : j \in S} x_{i,S} \leq 1 \;\forall j \in [m] && \text{each item } j \text{ can be sold at most once} \\
&\sum_{S \subseteq [m]} x_{i,S} \leq 1 \;\forall i \in [n] && \text{each agent can acquire at most one bundle} \\
&x_{i,S} \geq 0 \;\forall i \in [n], S \subseteq [m]
\end{aligned}$$

Theorem 3.5 Fundamental Theorems of Welfare for Combinatorial Markets

1. If a Walrasian equilibrium exists, then it maximizes social welfare.
2. If there is an integral solution to the linear program optimizing social welfare, then there is a Walrasian equilibrium for the corresponding allocation.

As a result, we have that a Walrasian equilibrium exists if and only if there is a an integer solution to the linear program optimizing social welfare in a combinatorial market. **PROP 3.6**

In a linear program $\max c^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0$, an integral solution exists if and only if A is totally unimodular (i.e. all square submatrices have determinant -1, 0, or 1). **PROP 3.7**

One can show that the matching market problem is totally unimodular

If A or A^T's rows can be permuted such that every column contains only consecutive 1s, then A is unimodular. We may also perform determinant-preserving operations like $r_\ell - \alpha r_k$ before making this conclusion.

IV Modern Game Theory

BEST RESPONSE DYNAMICS

In symmetric routing games (and potential games generally), we used best-response dynamics to arrive at a PSNE. We will try to analyze best response dynamics broadly.

Then *best-response* of an agent, given a strategy profile $\mathcal{S} = \{S_1, ..., S_n\}$, is the strategy $\hat{S}_i$ such that $v(\hat{S}_i, \mathcal{S}_{-i})$ is maximized. In *best-response dynamics*, we update $\mathcal{S} \leftarrow \mathcal{S} \setminus S_i \cup \hat{S}_i$. **DEF 4.1** **DEF 4.2**

The *symmetric selfish routing game* is an extension of Def 2.6, wherein $s_1 = \cdots = s_n = s$ and $t_1 = \cdots t_n = t$, i.e. all agents wish to start and end at the same node. **DEF 4.3**

We can form a DAG describing improvements in our potential, which has sinks wherever a PSNE is reached. Unfortunately, paths to these sinks may be exponential in length.

To speed this up, suppose that an agent only performs their best-response if it can gain an ε-improvement, $c_i(\mathcal{S}) - c_i(\hat{S}_i, \mathcal{S}_{-i}) > \varepsilon c_i(\mathcal{S})$, for some fixed $\varepsilon \in \mathbb{R}$.

If no agent can make an ε-improvement, i.e.

$$(1-\varepsilon)c_i(\mathcal{S}) \leq c_i(\hat{S}, \mathcal{S}_{-i}) \; \forall i \in [n], \hat{S}_i$$

then we call $\mathcal{S}$ an *ε-PSNE*. **DEF 4.4**

If we run best-response dynamics with ε-improvements, always taking the agent who can obtain the largest cost decrease $c_i(\mathcal{S}) - c_i(\hat{S}_i, \mathcal{S}_{-i})$, the selfish routing game will converge to an ε-PSNE in at most **PROP 4.1**

Taking the agent who obtains the largest cost decrease is called max-gain

$$\frac{Bn}{\varepsilon} \ln(\Phi(\mathcal{P}))$$

steps, where $B > 1$ is a constant we impose on the latency functions, as follows:

We call this the bounded jump condition.

$$\ell_e(x+1) \leq B\ell_e(x)$$

Theorem 4.1 High Welfare Walks

Given a potential, smooth game (Def 2.7) with $\Phi(\mathcal{S}) \leq c_i(\mathcal{S})$ and a sequence $\mathcal{S}^1, S_2, \ldots$ of max-gain best-response dynamics,

$$c(\mathcal{S}^t) < \frac{2\lambda}{1-\mu} c(\mathcal{S}^*) : t \geq 1$$

for all but at most $\frac{2n}{1-\mu} \ln(\phi(\mathcal{S}^0))$ states in the sequence.

NO REGRET DYNAMICS

Consider an agent who may pick from n actions across each time step in T. Denote by a^t the action chosen. There is an unknown cost $c^t(a)$ for each action chosen at step $t \in T$. The agent wishes to minimize her total cost, i.e. $\sum_{t\in T} c^t(a^t)$.

DEF 4.5 In *online decision making*, an agent is revealed the cost of action a^t at timestep $t+1$. It uses a probability distribution $\mathbb{P}^t$ on actions. Her expected cost is thus $\sum_{t\in T} \sum_{a\in[n]} \mathbb{P}^t(a)c^t(a)$.

How does this agent do? Suppose the set of actions is $\{L, R\}$. Fix a decision making strategy $\{\mathbb{P}^t\}$. At time t, the adversary selects $c^t = \langle 1, 0\rangle$ if $\mathbb{P}^t(L) \geq \frac{1}{2}$, and $c^t = \langle 0, 1\rangle$ otherwise. Then the agents' expected cost is bounded by

$$\sum_{t\in T} \mathbb{P}^t(L)c^t(L) + \mathbb{P}^t(R)c^t(R) \geq \sum_{t\in T} \frac{1}{2} = \frac{T}{2}$$

DEF 4.6 In *offline decision making*, the agent has advanced knowledge of the best outcome at each time-step. For instance, in the setup above, the agent always has a cost-0 choice against the adversary.

Explicitly, an offline agent incurs cost $\sum_{t\in T} \min_{a\in[n]} c^t(a)$

DEF 4.7 Offline decision making provides for too strong a baseline, so we instead consider *fixed offline decision making*, where the candidate has advanced knowledge, as in Def 4.6, but must commit to one action across all timesteps.

Explicitly, a fixed offline agent incurs cost $\min_{a\in[n]} \sum_{t\in T} c^t(a)$.

DEF 4.8 Suppose an agent chooses actions $\{a^1, \ldots, a^T\}$. Then its *regret* is

$$\frac{1}{T}\left(\sum_{i=1}^{T} c^t(a^t) - \min_{a\in[n]} c^t(a)\right)$$

DEF 4.9 Conceptually, regret is the average cost increase over choosing the best fixed action in retrospect. We say that an agent has *no regret* if, for any $\varepsilon > 0$, there is some timestep T for which the expected regret is bounded by ε.

SPONSORED SEARCH AUCTIONS

DEF 4.10

In a a *sponsored search auction*, agents bid for slots with associated quantities. Each agent has a value-per-unit, uniform over all slots. Typically, we view slots as online advertising spots with associated click-thru-rates (CTR). Each agent then has a uniform value-per-click.

Denote an agent's value for a click as as v_i and a slot's CTR as α_i. Put a canonical ordering

$$v_1 \geq v_2 \geq \cdots \geq v_n$$
$$\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m$$

on n agents and m slots.

In this situation, the welfare of an agent is $v_k s_\ell$, where agent k is assigned slot ℓ.

PROP 4.2

An allocation of slots and agents maximizes social welfare if and only if agent i gets slot i. Suppose not. Then there is a bidder $k+1$ who is allocated a slot s_{k+1} that is better than the slot allocated to bidder k, s_k. If k and $k+1$ swap their allocations, we will see an increase in social welfare:

$$v_{k+1}s_{k+1} + v_k s_k \leq v_k s_{k+1} + v_k s_k \leq v_k s_{k+1} + v_k s_{k+1}$$

We can use a typical VCG characterization, e.g. utility = marginal contribution, or price = damage done to others

Using VCG prices (Def 1.12), the total price charged to bidder i for slot i is

$$p_i = \sum_{\ell=i+1}^{n} v_\ell(\alpha_{\ell-1} - \alpha_\ell)$$

DEF 4.11

Note that, in the VCG case, we first compute the flat-rate cost for a bidder. In GSP auctions, we compute the price-per-click for a biddser.

Alternatively, we can use a *generalized second price auction*, in which bidder i, still assigned to slot i, pays the price-per-click of bidder $i+1$'s bid. Comparing total costs with the VCG mechanism, we find that $p_i \leq b_{i+1}\alpha_i$, assuming truthful bidding for GSP.

INDEX OF DEFINITIONS