1 Algorithmic Mechanism Design

Key Ideas

1. Single Parameter Domain: each agent has a private input $w_i \in \mathbb{R}$, and a summarization function $q_i : A \rightarrow \mathbb{R}_{\geq 0}$ (known to the mechanism designer). For an allocation $a$, agent $i$’s valuation function is $v_i(w_i, a) = w_i \cdot q_i(a)$. Examples include single item (possibly multi-unit) auction. A mechanism $(x, t)$ is strategy-proof (SP) for a single parameter domain if and only if, for all $i$ and $w_i^{-i}$

- (monotonicity) A choice rule $x$ is monotone non-decreasing, if, for all $i \in N$, all $\hat{w}_i^{-i}$, $q_i(x(w'_i, \hat{w}_i^{-i})) \geq q_i(x(w_i, \hat{w}_i^{-i}))$ for all $w'_i > w_i$.
- (payment identity) payment rule $t$ satisfies $t_i(\hat{w}_i, \hat{w}_i^{-i}) = \hat{w}_i \cdot q_i(x(\hat{w}_i, \hat{w}_i^{-i})) - \int_{z=0}^{\hat{w}_i} q_i(x(z, \hat{w}_i^{-i})) dz$ (1)

2. Knapsack: we have some number $m \geq 1$ of identical, indivisible items to sell to a set of $n$ bidders. Each bidder $i$ demands a public quantity $k_i$ of items, but her private value $w_i$ on her bid is private; so, we have a report $\hat{w} = (\hat{w}_1, \ldots, \hat{w}_n)$ of items. The knapsack auction is defined by the following allocation and payment rules:

- The allocation rule is as follows: first let $W_g$ be the value of the greedy allocation $N_{\text{approx}}$: remember that this ranks bidders in decreasing order of bang-for-buck, $\hat{w}_i/k_i$, and allocates in decreasing order, skipping a bid if it exceeds capacity. Let $W_h = \max_i \hat{w}_i$ be the value of the largest bid (assuming all $k_i$ are less than capacity). If $W_g \geq W_h$ items are allocated according to $N_{\text{approx}}$, else the highest bidder wins.
- The payment rule: simply the critical value (from single-parameter domains). In other words, 0 if a bidder is unallocated, else the minimum amount she must bid to be allocated.

3. Single-minded CA: there are bidders $N = \{1, \ldots, n\}$, a set $G$ of distinct and indivisible items, bidders report pairs $(\hat{T}_i, \hat{w}_i)$ (with true valuations $(T_i, w_i)$), with $T_i, \hat{T}_i \subseteq G$, and $w_i, \hat{w}_i \geq 0$. If bidder $i$ is allocated set $S$, then $v_i(S)$ is 0 if $T_i$ is not contained in $S$, else $w_i$. Suppose that $\sigma : 2^G \times \mathbb{R} \rightarrow \mathbb{R}$ is a monotone scoring function (so that $\sigma(w'_i, T'_i) \geq \sigma(w_i, T_i)$, if either $w'_i \geq w_i$ or $T'_i \subseteq T_i$ (or both)). The single-minded CA is defined by:

- Allocation rule: sort bids in order of decreasing score; accept greedily, skipping a bid if one or more items in the set $\hat{T}_i$ has been allocated.
• Payment: bidder \( i \) is allocated by its critical value.\(^1\)

4. Min-makespan scheduling: \( G \) is a set of tasks, \(|G| = m\), and \( c_{ij} > 0 \) is the time agent \( i \in N \) takes to complete task \( j \in G \). Let \( z \in \mathcal{Z} \) denote a feasible assignment of tasks. The design objective is to minimize \( \text{makespan}(z) \) over all assignments \( z \), defined by

\[
\text{makespan}(z) = \max_{i \in N} \left[ \sum_{j \in z_i} c_{ij} \right].
\]

We focus on the special case of a single-parameter setting: each agent has a type \( w_i = -r_i \), the negative unit processing time. \( v_i(w_i, z) = -r_i \cdot \left( \sum_{j \in z_i} t_j \right) = -r_i \cdot q_i(z) \), where \( q_i(z) \) is the total amount of work provided to agent \( i \). For the single-parameter problem, there is a deterministic SP optimal min-makespan mechanism (pset!).

5. Agent-independent pricing function: for every \( v_{-i} \) there does not exist two different reports \( v_i, v'_i \) of agent \( i \) such that the same outcome is selected but \( t_i(v_i, v_{-i}) \neq t_i(v'_i, v_{-i}) \). In other words, for all \( v_{-i} \), the payment is independent of \( v_i \) as long as the same outcome is chosen. This together with “agent optimizing” (agents get what they want when facing these prices) are necessary and sufficient for SP.

1.1 Exercises

1. Consider 5 items and 4 bidders, with the value quantity pair \((6,1),(5,2),(12,5),(7,3)\) for bidders 1,2,3,4.
   (a) What is the optimal allocation?
   (b) What is the VCG payment?
   (c) What is the VCG-based greedy (per item value sort) allocation and payment?
   (d) Show a useful deviation of if the allocation is VCG-based greedy.

2. Give three potential score functions for the greedy single-minded CA algorithm.

3. Three tasks A,B,C and two agents. Costs of agent 1 for the three tasks are 5,6,12, which costs of agent 2 are 8,2,13.
   (a) What is the VCG allocation and payment?
   (b) What is the min-makespan allocation?
   (c) Can the strategy-proof single-parameter minmakespan mechanism be used to solve this problem?

\(^1\)Technically this domain is not single-parameter, but it is shown in the reading that the knapsack auction is SP with this payment rule.
2 Online Advertising Auctions

Key Ideas

1. Several types of ads were discussed in class:
   - **sponsored search**: advertisers bid on search-terms ahead of time (with *standing bids*). Each time a search term is entered, a position auction is run with eligible advertisers. Payments are usually made per-click. Generally GSP.
   - **contextual ads**: generally keyword-based standing bids like sponsored search, only ads appear on external webpages that users visit, and are targeted based on the content of the webpage. Also behavioral targeting. Generally VCG.
   - **display ads**: targeted based on page position and demographic (e.g., front page espn.com), increasingly sold through SPSB auctions in real-time ad exchanges.

2. In the **Generalized Second-Price (GSP)** auction, advertisers submit one bid each for multiple slots.
   - **Assumptions**: Advertisers $i$ have uniform per-click values $v_i$, independent of position, number of clicks, etc. Each advertiser has an ad quality $Q_i$ (calculated by the auction designer). Each ad position $j$ has a position value $pos_j$ and each ad’s click-through rate is modeled by $CTR_{ij} = Q_i \cdot pos_j$. Effective values are given by $v_{ij} = CTR_{ij} \cdot v_i$.
   - **Allocation rule**: Advertisers are ranked into slots according to decreasing $b_i \cdot Q_i$, where $b_i$ is the $i^{th}$ advertiser’s bid per click.
   - **Payment rule**: Advertisers pay the least they would have needed to bid to keep their slots.

\[
PPC_{gsp,i} = \frac{Q_{i+1}b_{i+1}}{Q_i}
\]  

where we assume that indices are ordered by effective bid order.

- The GSP auction is not strategy-proof. However, among its Nash equilibrium, there exist envy-free (value-ordered) Nash equilibrium.

- **Envy-Free**: A bid profile is envy-free if no bidder prefers the position and price of any other bidder in the outcome of the action.
  - (GSP1) NE + Envy Free: bids in GSP are value-ordered, auction is efficient
  Among the envy-free Nash equilibrium, there exist balanced equilibrium, which recognize that bidders, for competitive reasons, may prefer to drive up the price of other bidders.

- **Balanced bidding**: each advertiser bids just high enough such that the next-highest advertiser bidding down to switch slots would not decrease the original advertiser’s utility. Bids in this equilibrium are value-ordered (generally by $Q_i \cdot v_i$). In the case of no quality scores, the BB equation is:

\[
pos_{i-1}(v_i - b_i) = pos_{i}(v_i - b_{i+1})
\]  

for each advertiser $i \geq 2$ and $b_1 \geq b_2$.

- Balanced bidding outcome is same as truthful outcome in VCG auction!
3. The **VCG position auction** ranks the advertiser according to decreasing $Q_i \cdot b_i$. The expected payment by advertiser $i$ is given as

$$t_{\text{vcg},i}(b) = \sum_{k=i+1}^{n} (\text{pos}_k - \text{pos}_{k+1})Q_kb_k$$

and the per click payment for advertiser $i$ is $t_{\text{vcg},i}(b)/(Q_i \cdot \text{pos}_i)$. The VCG position auction is strategyproof and allocatively efficient. It is increasingly used for contextual ads.

In fact, the balanced bidding outcome in the GSP auction is the same as the truthful outcome in the VCG auction. Search engines still generally use GSP though (why?)

**Exercises**

1. **GSP auctions**

   Consider the following position auction setting. Suppose our auction is a GSP auction with three positions and four bidding advertisers A, B, C, and D. The three positions have position effects $\text{pos}_1 = 0.3, \text{pos}_2 = 0.28, \text{pos}_3 = 0.1$. The bidders have per-click values (e.g. the customer value upon the click) $v_A = $100/click, $v_B = $50/click, $v_C = $18/click, and $v_D = $10/click.

   (a) Suppose that the advertisers create ads with qualities $Q_A = 0.5, Q_B = 0.7, Q_C = 0.4, Q_D = 0.8$. If advertisers bid truthfully (i.e. bid $b_i = v_i$), what will the outcome be (i.e. what are the position allocation and payment for each advertiser)?

   (b) From now on suppose that all advertisers have equal qualities $Q_i = 1$ for all $i \in \{A, \ldots, D\}$. Now what is the outcome of truthful bidding?

   (c) Is this truthful bidding a Nash equilibrium?

   (d) Suppose that we have bids $b_A = 20, b_B = 50, b_C = 18, b_D = 10$. Is this a Nash equilibrium? Is this outcome value-ordered (i.e. are the bids non-decreasing in the values)?

   (e) The bids not being value-ordered in Nash equilibrium implies that this bidding outcomes does not satisfy the balanced-bidding condition. Check that the balanced-bidding condition is violated for some ad $i$.

   (f) What is the balanced bidding outcome in this example? For this to be Nash, what is the key assumption we must make about the knowledge of the bidders?

   (g) Check that the balanced bidding outcome is Nash.

2. **VCG in Online Advertising**

   Consider the following position auction setting. Suppose our auction is a VCG auction with three positions and four bidding advertisers A, B, C, and D. The three positions have position effects $\text{pos}_1 = 0.3, \text{pos}_2 = 0.28, \text{pos}_3 = 0.1$. The bidders have per-click values (e.g. the customer value upon the click) $v_A = $100/click, $v_B = $50/click, $v_C = $18/click, and $v_D = $10/click.

   (a) Suppose that the advertisers create ads with qualities $Q_A = 0.5, Q_B = 0.7, Q_C = 0.4, Q_D = 0.8$. What are the position allocation and payment for each advertiser if advertisers bid truthfully?
(b) From now on suppose that all advertisers have equal qualities $Q_i$ for all $i \in \{A, B, C, d\}$. Now what is the outcome of truthful bidding?

(c) Is this VCG auction strategy-proof? Is there any useful deviation for advertisers? Check it for advertiser A in this example.

3 Combinatorial auctions

Key Ideas

1. A Combinatorial Auction (CA) is an auction where bidders $N = \{1, \ldots, n\}$ bid on bundles of distinct, indivisible goods $G = \{A, B, C, \ldots\}$, $|G| = m$. An efficient CA allocates items to maximize the sum of the value of bidders. An allocation $X = (X_1, \ldots, X_n)$ allocates $X_i \subset G$ to bidder $i \in N$, and is feasible if each item is allocated to at most one bidder. An efficient allocation solves

$$
\max_X \sum_{i=1}^{n} v_i(X_i)
$$

s.t. $X_i \cap X_j = \emptyset$, for all $i, j \in N, i \neq j$

2. Combinatorial Auctions are especially useful in cases where valuation functions for packages are not simply additive in the value of the items. Two examples of such valuation functions include subadditive and superadditive valuation functions. A superadditive function exhibits complements, and satisfies

$$
v_i(S \cup S') \geq v_i(S) + v_i(S')
$$

for all disjoint packages $S$ and $S'$, and a subadditive function exhibits substitutes, with

$$
v_i(S \cup S') \leq v_i(S) + v_i(S')
$$

for all disjoint packages $S$ and $S'$.

3. Some examples of actual CAs include allocating airport landing spots and allocating spectrum frequencies.

Imagine if airport landing spots were each auctioned off in sequence as single good auctions. An airline carrier might win gates A2, A5, B4, C7 (in different terminals) which is inefficient for business. The airline would much rather win a connected block of gates all in the same terminal. A combinatorial auction allows airlines to express their desires for contiguous gates when bidding.

4. There are several bidding languages in CAs: OR, XOR, OR*, OR-of-XOR, XOR-of-OR. Some desirable properties of bidding languages include being expressive and succinct. The OR* language is fully expressive and fairly succinct.

5. Solving the optimal allocation problem in a CA is much harder than for the single good auctions and can be formulated as an Integer Program (IP). It can be reduced from WEIGHTED-INDEPENDENT-SET and is NP-hard.
6. The Winner Determination Problem is tractable if the bids satisfy one of the following structural properties: cyclic structure, tree structure, pair bids, or hierarchical structure. Another way to make the winner determination problem tractable is to modify the bidding language.

7. VCG can be used to allocate and price bundles in CAs, but there are lots of problems: low revenue, vulnerability to loser collusion, false-name problems, etc.

3.1 Things you should be able to do

1. Given a combinatorial auction and agents’ valuations for bundles be able to figure out the allocation and VCG payments (assuming there are only a few goods).

2. Express an agent’s valuations using either the OR (if possible), XOR, or OR* language.

3.2 Questions

1. Consider a VCG mechanism applied to a combinatorial auction with two goods \{A, B\} and bids 
\((AB, $2), (A, $2), (B, $2)\) from three different bidders.

   (a) What is the outcome of the VCG mechanism in this example?

   (b) Use a variation on the example to show that the revenue of the VCG mechanism is not monotonic-increasing in the number of bidders.

   (c) Use a variation on the example to show why the VCG mechanism is susceptible to collusion by losers.

   (d) Use a variation on the example to show why the VCG mechanism is susceptible to manipulation by false-name bidders (or “sybil-attack”), where one bidder bids under multiple identities.

   (e) Do any of these problems occur in the single-item Vickrey auction? For each problem, explain why or why not.

2. Consider a weighted undirected graph \(G = (V, E)\) shown in Figure 1, where the number in each vertex is the weight of it. Then show how to construct OR bids in the corresponding CA and write down the integer programming.

3. Prove that if

\[ V(L) \leq \sum_{i \in N} p_i + \sum_{I \in L} \pi_i \]

is violated for some \(L\), then there is an allocation to the bidders in \(L\) where the value created could be divided between the seller and bidders in \(L\) to make them all strictly better off.

4. Show that the core of a CA is always non-empty, by setting bidders’ payments to their values and using the equation

\[ V(L) \leq \sum_{i \in N \setminus L} p_i + \sum_{i \in L} v_i(X_i). \]
4 Matching

1. Two Sided Matching Problems: We have two distinct sets of agents. Each agent wants to either be matched with an agent in the other set or remain unmatched. Agents have a strict preference over all possible candidates from the other side (including the null candidate, or not being matched). An important property of two sided matching is stability. A matching is stable if there is no blocking pair of agents who prefer each other to their assigned match.

2. Deferred Acceptance Algorithm: If every agent has a strict preference ordering over all partners, we cannot achieve stability and strategy-proof-ness for all agents in two-sided matching. This is the Gibbard-Satterthwaite Impossibility result. We can achieve stability along with strategy-proof for one side of the market with the deferred acceptance algorithm. Mechanism:

- Participants on one side of the market (e.g. students) proposes to the participants (e.g. professor) on the other side of the market.
- Professors decides whether or not to accept the proposed offer depending on her preference. If accepted, and the professor had held an offer, she rejects the student with the previously held offer.
- If students are rejected, then they move down and propose to their next favorite professor.
- Algorithm ends when there is a stage where no students are rejected.

Note that

(a) This algorithm always terminates with a stable matching
(b) It is only strategy proof for the side proposing. No matching mechanism for two-sided matching is both stable and strategy proof.
(c) A professor is in the set of achievable professors to a student if there exists some stable matching between the professor and that student. The student proposing DA matches each student their most preferred achievable professor, and matches each professor with their least preferred achievable student.

3. Assignment Problems: Assignment problems are one-sided matching problems, in that there is a single group of agents and a set of items, and agents have preferences on items but items do not have preferences on agents. Two variants of the assignment problem are the House Allocation Problem, in which one item is assigned to each agent, and the Housing Markets Problem, in which each agent initially owns one of the items, and an assignment represents a reallocation of the items. Agents have a strict preference ordering over all items.

4. Serial Dictatorship and Random Serial Dictatorship: Serial Dictatorship fixes an ordering of agents, and let them pick one-by-one what they prefer (out of the options that has not been picked). The mechanism is strategy proof and Pareto optimal. Random Serial Dictatorship mechanism, uniform randomly assigns the ordering. The mechanism is also strategy proof and Pareto optimal for the house allocation problem.

5. Top Trading Cycles Mechanism: An assignment in the housing markets problem is said to be in the core if there is no set of agents in the assignment that would do better by trading among themselves (i.e. if there is no blocking coalition). The Top Trading Cycle Mechanism selects an assignment that is Pareto-optimal, satisfies participation, strategy proof and in the (unique) core. Mechanism:
   - Let agents point to the agent that has their most preferred house that is still in the market.
   - Find all cycles, and allocate the house to the upstream agent. The agents that are allocated exit the market.
   - Repeat until no agents are left.

6. It is important to note that for any of the algorithms, the ‘proposal’ and ‘acceptance’ (DA), ‘cycle formation’ and ‘pointing’ (TTC), and ‘pick one-by-one’ (SD,RSD) are all done implicitly by the mechanism designer. The agents (students and professors, house traders) only submit a list of their preferences, and the MD, using the algorithm described, spits out the matching allocation.

7. Kidney Paired Donation: The Kidney Paired donation problem can be formulated as a problem of finding vertex disjoint cycles on a directed compatibility graph $G = (V,E)$. The vertices in $V$ are patient-donor pairs and there is a directed edge $e = (u,v)$ from a pair $u$ to a pair $v$ if the donor in pair $u$ is compatible with the patient in pair $v$. The cycles must be vertex disjoint since we assume that each patient is able to donate only one kidney and each patient needs one kidney. The length of each cycle is limited to $K$ for logistical reasons. When $K = 2$, we can formulate this problem as a maximum cardinality matching (which is polynomial time solvable due to Edmond’s algorithm). For $K = \infty$ this problem can be solved using linear programming. However, for $3 \leq K < \infty$ this problem is NP-hard.
5 Information Elicitation and Peer Prediction

5.1 Information Elicitation

The goal of effective information elicitation is to make it an equilibrium for participants to invest effort and report accurate information. The two main kinds of information elicitation tasks depend on whether the event being predicted can be verified (resolved) in future (e.g., will it rain tomorrow), or whether it’s something that isn’t easily verifiable (e.g., how good a restaurant is for kids).

1. Scoring rules attempt to incentivize agents to report their beliefs about some event/outcome that will be realized in the future.

A scoring rule \( s \) takes a reported belief, and defines a payment to the forecaster, contingent on the outcome. Given a set of possible outcomes \( O \), a scoring rule is a function \( s : \Delta(O) \times O \rightarrow \mathbb{R} \cup \{-\infty\} \)

2. An agent’s payment (which can be negative) is a function of her report and the realized outcome. The expected payment for a forecaster with belief \( p \) who reports belief \( q \) (possibly \( q \neq p \)) is \( S(q,p) = E_{o \sim p}[s(q,o)] = \sum_{k=0}^{m-1} p_k \cdot s(q,o_k) \), where \( o \sim p \) denotes that the outcome is distributed according to the probability \( p \). The expectation is taken with respect to the forecaster’s true belief, and depends on the report \( q \) and the realized outcome \( o_k \).

3. A scoring rule is strictly proper if it incentivizes an agent to report her true belief. A scoring is strictly proper if, for every belief \( p \), the expected payment is uniquely maximized through the truthful report.

4. The quadratic and logarithmic scoring rules are examples of strictly proper scoring rules.

5. The logarithmic scoring rule is \( s_{\log}(q,o_k) = \ln(q_k) \).

6. The quadratic scoring rule (Brier’s rule) is \( s_{\text{quad}}(q,o_k) = 2q_k - \sum_{k'=0}^{m-1} q_{k'}^2 \).

7. If \( s(q,o_k) \) is a strictly proper scoring rule on \( m \) outcomes, then any rule \( s'(q,o_k) = \alpha_k + \beta \cdot s(q,o_k) \) obtained via a positive affine transform with \( \alpha \in \mathbb{R}^m \) and \( \beta \in \mathbb{R}_{>0} \) is strictly proper.

8. Sometimes agents must exert effort to figure out their beliefs. By increasing the value of \( \beta \) in our scoring rule, we can make it worthwhile for agents to exert effort.

Exercises

**Biased Coin** Suppose we have a biased coin that comes up heads with an unknown probability \( p \). Based on our observations of past coin flips, we form the belief that \( p = \frac{\text{number of } H}{\text{number of flips}} \).

We have flipped the coin 9 times and have received 3 heads.

1. Under a logarithmic scoring rule, what should I report as my belief about the probability that the next flip is an H to maximize expected score?

2. What is my expected payoff from reporting truthfully?

3. What is my expected payoff from reporting 0.25?

4. What is my expected payoff from reporting 0.5?
5.2 Peer Prediction

1. In the peer prediction method, agents are asked to report their signal and payments are made based on joint reports of the signals. The agents report signals without knowing what the other agents’ reports are.

2. Two simple mechanisms are output agreement and 1/Prior. There are also mechanisms based on scoring rules.
   a. Output Agreement
      \[ t_i(r_1, r_2) = \mathbb{1}(r_1 = r_2) \text{ for } i \in \{1, 2\} \]
      for reports \( r_1, r_2 \) from agents 1 and 2, respectively
   b. 1/Prior
      \[ t_i(r_1, r_2) = \frac{1}{P(r_1)} \mathbb{1}(r_1 = r_2) \text{ for } i \in \{1, 2\} \]
      for reports \( r_1, r_2 \) from agents 1 and 2, respectively
   c. Scoring rules based mechanisms.
      - Logarithmic scoring rule: \( t_{\log}(q,o_k) = \ln(q_k) \)
        \[ t_1(r_1, r_2) = t_{\log}(P(X_2 = r_2 | X_1 = r_1), r_2) \]
      - Quadratic scoring rule: \( t_{\text{quad}}(q,o_k) = 2q_k - \sum_{k'=0}^{m-1} q_{k'}^2 \)
        \[ t_1(r_1, r_2) = t_{\text{quad}}(P(X_2 = r_2 | X_1 = r_1), r_2) \]

3. Strictly proper Peer Prediction mechanism. Let the strategy profile is \((\sigma_1, \sigma_2)\), the expected payment to agent \(i\) is
   \[ S_i(\sigma_1, \sigma_2) = E_X [t_i(\sigma_1(X_1), \sigma_2(X_2))] \]
   A peer prediction mechanism is strictly proper iff
   \[ S_i(\mathbb{1}, \mathbb{1}) > S_i(\sigma_1, \mathbb{1}), \text{ for all } \sigma_i \neq \mathbb{1}, \text{ for } i \in \{1, 2\} \]
   It is equivalent to
   \[ E_{j \in p(j)} [t_1(j, \ell)] > E_{j' \in p(j)} [t_1(j', \ell)] \]
   where \(p(j)\) is the conditional signal distribution on one agent’s expected payment for reporting her true signal \(j\), here it is \(P(X_2 | X_1 = j)\).

   a. OA mechanism: Self-dominant signal distribution
      \[ P(X_2 = j | X_1 = j) > P(X_2 = j' | X_1 = j) \]
   b. 1/Prior mechanism: Self-predicting signal distribution
      \[ P(X_2 = j | X_1 = j) > P(X_2 = j | X_1 = j') \]

5. One issue with running peer prediction mechanisms in practice is that there can be additional, uninformative equilibria. Participants in a peer-prediction mechanism may coordinate and receive high payments without revealing any useful information. A good solution is to use the multi-task bonus/penalty mechanism (and its generalizations).
Exercises

Peer Prediction  Suppose we have true, hidden states of the world $H = \{0, 1\}$ and possible signals $S_i = \{0, 1\}, \forall i \in \{1, 2\}$, and that $P(H = 0) = 0.4$. Further suppose that $P(X_i = 0|H = 0) = 0.6$ and $P(X_i = 1|H = 1) = 0.8$.

1. Calculate the joint distribution on signals.

2. Calculate the payments in truthful equilibrium under Output Agreement. Is OA strictly proper here? Justify your answer.

3. Are there additional, uninformative equilibria under Output Agreement? If so, what are they?

4. Now think about the scoring-rule based peer-prediction mechanism. Calculate payments under the logarithmic scoring rule for each pair of reports. Then, give transformed payments such that strict properness still holds but payments are between 0 and 1.

5. Calculate payments under the 1/Prior peer prediction mechanism. Is this mechanism strictly proper? Justify your answer.

6 Prediction Markets

6.1 Review

1. Prediction markets are systems for aggregating the beliefs of many people about one or more future events by allowing agents to place bets on the future outcomes of those events. One example of such a market is the Iowa Electronic Market (one of few operating in the U.S.). The challenge in designing prediction markets is to create incentives such that those with new information or strongly held beliefs will choose to participate in the market.

2. In the prediction market, agents can buy contracts on the outcome of an event. Ideally, if an agent believes the probability of an event to be $p$, then he or she would buy contracts on the event at a price less than $p$ and short the contracts on the event at a price greater than $p$. Types of contracts include winner-take-all and index contracts.

3. A winner-take-all contract pays off if the underlying event occurs, and the contract owner gets nothing if the event does not occur. Then the current market price corresponds to the market’s belief regarding the probability of the event happening. If the contract pays off $10 when the event happens, and the contract is trading at $3$, then the market believes that there is a 30% chance that the event will occur, assuming risk neutrality.

4. Two important prediction market designs are the continuous double auction (CDA) and the automated market maker (AMM). In the CDA, an order book maintains outstanding bids
and asks. Orders are matched and carried out whenever there is a pair of a bid and ask such that the bid price is lower than the ask price (the trade is executed at the price of whichever of the bid or ask was submitted first). In the AMM, the prediction market operator acts as a market maker and is always willing to trade. The main advantage of this design is greater liquidity.

5. For AMM, we want a mechanism with the following properties:
   - Normalization. The sum of the probabilities of all outcomes is 1.
   - No round-trip arbitrage. For example, you wouldn’t be able to buy all outcomes for less than 1.
   - Strictly positive prices
   - Responsiveness. Price goes up when you buy, price goes down when you sell.
   - High liquidity
   - Myopic incentives. If price is 0.2 and your true probability is 0.8, you are willing to buy from 0.2 to 0.8 in one transaction.
   - Bounded loss

The log market scoring rule (LMSR) is a market maker with these properties.

6. Prediction markets advantages over classical methods such as asking experts and opinion polls include high predictive accuracy, self-selection, incentive alignment, agents putting money behind their bets, running in real-time, and self-organizing. What are the limitations?

6.2 Exercise

6.2.1 Continuous Double Auctions

We will analyze a market with a Continuous Double Auction mechanism and one contract traded (XYZ). Assume that there are no outstanding orders at the beginning of this question and that prices are set with preference given to earlier orders.

1. A trader, $A_1$, puts in a buy order at $0.50$, and another trader, $A_2$, puts in a sell order at $0.60$. What happens?

2. Another trader, $A_3$, puts in a buy at $0.65$. Now what happens? If a trade occurs, at what price?

3. Another trader, $A_4$, puts in a sell order at $0.40$. Now what happens?

6.2.2 Logarithmic Market Scoring Rule

Consider the following market making mechanism that is being used to predict the outcome of the 2016 election. There is a first guess probability that Hillary Clinton will win with $p=50\%$. Whenever a trader comes along and disagrees with the market’s probability $p_{\text{last}}$, he may change the market probability to $p_{\text{new}}$. He is paid according to the logarithmic scoring rule on $p_{\text{new}}$, but he must pay out the logarithmic scoring rule on $p_{\text{last}}$.

What is an agent’s expected payoff if he moves the market from $p_{\text{last}}$ to his best guess probability $p_{\text{new}}$ and is scored using the logarithmic scoring rule?
6.2.3 Automated Market Maker with Market Scoring Rule

We analyze a market with an automated market maker using a logarithmic market scoring rule. Suppose that there are two outcomes, \( o_0 \) and \( o_1 \), possible, and that the automated market maker uses the cost function

\[
C(x_0, x_1) = 2 \ln(e^{x_0} + e^{x_1}).
\]

The initial state is \((0, 0)\).

1. A trader, \( A_0 \), puts in a buy order for 1 contract of \( o_0 \) occurring. What does he pay?
2. A trader, \( A_1 \), puts in a buy order for 2 contracts of \( o_0 \) occurring. What does he pay?
3. Compute the profits or losses of \( A_0 \) and \( A_1 \) if outcome \( o_0 \) occurs and if outcome \( o_1 \) occurs.
4. Suppose the market is currently trading at \( p = 0.40 \) for \( o_0 \). A new participant who believes that \( o_0 \) will occur with prob 0.8 comes along. Assuming she is risk neutral and not budget-constrained, how should she determine what trade to perform?

7 Reputation Systems

7.1 Review

7.1.1 Key Points

1. **Modeling reputation systems through game theory**: a group of \( n + 1 \) agents with \( n \) odd, where in each round agents are paired up randomly to play the Prisoners’ Dilemma game. How do the strategies for two-player PD translate to this setting? Now assume there is a public counter documenting how many times a player has defected. How can this reputation system affect the equilibria?

2. **Whitewashing attacks** consist of an agent exiting the system when she has a bad reputation and creates a new identity to start anew. This can be combated by raising the bar to entry, for instance requiring new users to verify accounts with a photo ID.

3. **eBay’s old reputation system** had a flaw that made submitted feedback viewable immediately, even if the other party had yet to leave their own feedback. In this system, sellers could manipulate buyers to leave positive feedback, making the reputation feedback uninformative. Potential fixes investigated by eBay included simultaneous revealing of feedback and detailed seller ratings, the latter of which eBay decided to implement.

4. **eBay’s new reputation system** Included one-directional feedback: the buyer can provide feedback on several attributes of the interaction with the seller, and this information is delayed and pooled so that the seller doesn’t know which buyer provided the specific feedback. In a later change, sellers are prevented from giving negative feedback (they can only give positive feedback or no feedback).
7.1.2 Things you should be able to do

1. Understand moral hazard and adverse selection problems in different settings.

2. Reason about the strategic aspects of the reputation game.

3. Discuss eBay’s old and new reputation system and explain the pros and cons of different reputation mechanisms.

4. Discuss the different design decisions (directionality, type of feedback requested, etc.) of various reputations systems such as those of Amazon, Airbnb, and Yelp

7.2 Exercises

1. Whitewashing
   Consider the repeated Prisoner’s Dilemma game where agents have a published reputation of how many times they have played D. Once an agent plays D twice, everyone else begins defecting against him. Assume that agents are allowed to whitewash their identity and re-appear as a new agent. Let $f$ denote the initiation fee paid in round 0 and every subsequent re-entry into the market in the repeated Prisoner’s Dilemma game.
   What is the sequence of payoffs you’d get with

   (a) cooperating forever?

   (b) defecting forever?

   (c) defecting twice, whitewashing, and then repeating?

2. Comment on eBay’s old reputation system. What were some pros and cons of this system?

3. Discuss the pros and cons of a simultaneous-feedback system (i.e. the one eBay considered implementing).

4. Discuss the pros and cons of the detailed seller rating system (i.e. the one eBay implemented).