Lecture 9

Mechanism Design with Payment Constraints

Lecture 2 introduced the quasilinear utility model, where each agent acts to maximize her valuation of the chosen outcome, less the payment she makes. We placed no restrictions on payments other than the modest conditions that they are nonnegative and guarantee nonnegative utility to truthful bidders. This lecture is the first to consider mechanism design problems with *payment constraints*, in addition to the usual incentive and feasibility constraints.

Section 9.1 extends the quasilinear utility model to accommodate budget constraints. Section 9.2 studies multi-unit auctions where bidders have budgets, and proposes an elegant if non-DSIC solution: the uniform-price auction. The clinching auction, described in Section 9.3, is a more complex auction for the same problem that is DSIC. Section 9.4 considers mechanism design with no payments whatsoever, introduces the canonical house allocation problem, and studies the properties of the Top Trading Cycle algorithm.

9.1 Budget Constraints

In many applications, there are constraints on the payments charged by a mechanism. Exhibit A is *budget constraints*, which limit the amount of money that an agent can pay. Budgets are especially relevant in auctions where an agent might buy a large number of items. For example, in the sponsored search auctions (Section 2.6) used in practice, every bidder is asked for her bid-per-click and her daily budget. Per-item values and overall budgets model well how many people make decisions in auctions with lots of items, especially when the items are identical.

The simplest way to incorporate budgets into our utility model is to redefine the utility of agent i with budget B_i for outcome ω and payment p_i as

$$\begin{array}{rl} v_i(\omega) - p_i & \text{if } p_i \leq B_i; \\ -\infty & \text{if } p_i > B_i. \end{array}$$

A natural generalization, which we won't discuss, is to have a cost function that is increasing in the budget violation.

We need new auction formats to accommodate budget constraints. For example, consider the simple case of a single-item auction, where every bidder has a known budget of 1 and a private valuation. A second-price auction charges the winner the second-highest bid, which might well be more than her budget. More generally, no DSIC single-item auction with nonnegative payments maximizes the social welfare while respecting bidders' budgets (Problem 9.1).

9.2 The Uniform-Price Multi-Unit Auction

9.2.1 Multi-Unit Auctions

In a *multi-unit auction*, there are m identical items, and each bidder has a private valuation v_i for each item that she gets. Unlike the k-unit auctions of Example 3.2, we assume that each bidder wants as many units as possible. Thus bidder i obtains value $k \cdot v_i$ from k items. Such multi-unit auctions are single-parameter environments (Section 3.1). Finally, each bidder i has a budget B_i that we assume is *public*, meaning known to the seller in advance.[1]

9.2.2 The Uniform-Price Auction

The first multi-unit auction that we consider sells items at the "market-clearing price," where "supply equals demand." The supply is m, the number of items. The demand of a bidder depends on the selling price, with higher prices resulting in smaller demands. Formally, we define the *demand of bidder i at price p* as:

$$D_i(p) = \begin{cases} \min\left\{\left\lfloor \frac{B_i}{p} \right\rfloor, m\right\} & \text{if } p < v_i; \\ 0 & \text{if } p > v_i. \end{cases} \tag{9.1}$$

[1] We'd love to assume that budgets are private and thus also subject to misreporting, but private budgets make the mechanism design problem more difficult, even impossible in some senses (see also Exercise 9.3). Also, the special case of public budgets guides us to some elegant and potentially useful auction formats, which is the whole point of the endeavor.

To explain, recall that bidder i has value v_i for every item that she gets. If the price is above v_i, then she doesn't want any (i.e., $D_i(p) = 0$), while if the price is below v_i, she wants as many as she can afford (i.e., $D_i(p) = \min\{\lfloor \frac{B_i}{p} \rfloor, m\}$). When $v_i = p$, the bidder does not care how many items she gets, as long as her budget is respected. The auction can break ties arbitrarily and take $D_i(v_i)$ to be any convenient integer between 0 and $\min\{\lfloor \frac{B_i}{p} \rfloor, m\}$, inclusive.

As the price p increases, the demand $D_i(p)$ decreases, from $D_i(0) = m$ to $D_i(\infty) = 0$. A drop in demand can have two different forms: from an arbitrary positive integer to 0 (when p exceeds v_i), or by a single unit (when $\lfloor B_i/p \rfloor$ becomes one smaller).

For a price p different from all bidders' valuations, we define the *aggregate demand* by $A(p) = \sum_{i=1}^n D_i(p)$. In general, we define $A^-(p) = \lim_{q \uparrow p} \sum_{i=1}^n D_i(q)$ and $A^+(p) = \lim_{q \downarrow p} \sum_{i=1}^n D_i(q)$ as the limits of $A(p)$ from below and above, respectively.

The uniform-price auction picks the price p that equalizes supply and aggregate demand, and gives every bidder her demanded number of items at a price of p each.

The Uniform-Price Auction

1. Let p equalize supply and aggregate demand, meaning $A^-(p) \geq m \geq A^+(p)$.

2. Award $D_i(p)$ items to each bidder i, each at the price p. Define demands $D_i(p)$ for bidders i with $v_i = p$ so that all m items are allocated.

While we describe the uniform-price auction as a direct-revelation auction, it is straightforward to give an ascending implementation.

9.2.3 The Uniform-Price Auction Is Not DSIC

The good news is that, by the definition (9.1) of the demand $D_i(p)$, the uniform-price auction respects bidders' budgets. The bad news is that it is not DSIC. Similarly to simultaneous ascending auctions, it is vulnerable to demand reduction (Section 8.3.3).

Example 9.1 (Demand Reduction) Suppose there are two items and two bidders, with $B_1 = +\infty$, $v_1 = 6$, and $B_2 = v_2 = 5$. If both bidders bid truthfully, then the aggregate demand $A(p)$ is at least 3 until the price hits 5, at which point $D_1(5) = 2$ and $D_2(5) = 0$. The uniform-price auction thus allocates both items to the first bidder at a price of 5 each, for a utility of 2. If the first bidder falsely bids 3, she does better. The reason is that the second bidder's demand then drops to 1 at the price $\frac{5}{2}$ (she can no longer afford both), and the auction stops at the price 3, at which point $D_1(3)$ is defined as 1. The first bidder only gets one item, but the price is only 3, so her utility is 3, more than with truthful bidding.

Can we modify the uniform-price auction to restore the DSIC guarantee? Because the auction has a monotone allocation rule, we can replace the uniform price with the payments dictated by Myerson's lemma (Theorem 3.7). To obtain a DSIC multi-unit auction format with still better properties, the next section modifies both the allocation and payment rules of the uniform-price auction.

*9.3 The Clinching Auction

The *clinching auction* is a DSIC multi-unit auction for bidders with public budgets.[2] The idea is to sell items piecemeal, at increasing prices. In addition to the current price p, the auction keeps track of the current supply s (initially m) and the residual budget $\hat{B}_i$ (initially B_i) of each bidder i. The residual demand $\hat{D}_i(p)$ of bidder i at price $p \neq v_i$ is defined with respect to the residual budget and supply, analogous to (9.1):

$$\hat{D}_i(p) = \begin{cases} \min\left\{\left\lfloor \frac{\hat{B}_i}{p} \right\rfloor, s\right\} & \text{if } p < v_i \\ 0 & \text{if } p > v_i. \end{cases} \tag{9.2}$$

Define $\hat{D}_i^+(p) = \lim_{q \downarrow p} \hat{D}_i(q)$.

The clinching auction iteratively raises the current price p, and a bidder i "clinches" some items at the price p whenever they are uncontested, meaning the sum of others' residual demands is strictly less than the current supply s. Different items are sold in different iterations, at different prices. The auction continues until all of the items have been allocated.

[2]Again, we give a direct-revelation description, but there is also a natural ascending implementation.

The Clinching Auction

initialize $p = 0$, $s = m$, and $\hat{B}_i = B_i$ for every i
while $s > 0$ **do**
    increase p to the next-highest value of v_i or $\hat{B}_i/k$ for a positive integer k
    let i denote the bidder with the largest residual demand $\hat{D}_i^+(p)$, breaking ties arbitrarily
    while $\sum_{j \neq i} \hat{D}_j^+(p) < s$ **do**
        if $\sum_{j=1}^{n} \hat{D}_j^+(p) > s$ **then**
            award one item to bidder i at the price p
            `// this item is "clinched"`
            decrease $\hat{B}_i$ by p and s by 1
            recompute the bidder i with the largest residual demand at the price p, breaking ties arbitrarily
        else if $\sum_{j=1}^{n} \hat{D}_j^+(p) \leq s$ **then**
            award $\hat{D}_j^+(p)$ items to each bidder j at price p
            award any remaining items to the bidder ℓ that satisfies $v_\ell = p$, at a price of p per item
            decrease s to 0

The only relevant prices are those at which the residual demand of some bidder drops. Every such price p satisfies either $p = v_i$ or $p = \hat{B}_i/k$ for some bidder i and positive integer k. For simplicity, assume that all expressions of the form v_i and $\hat{B}_i/k$ for integers k that arise in the auction are distinct.

In the inner while loop, there are two cases. In the first case, the aggregate residual demand exceeds the residual supply, but the aggregate demand of bidders other than i is less than the supply. In this case, bidder i "clinches" an item at the current price p, and her budget is updated accordingly. Both the residual supply and i's residual demand decrease by 1 unit.

The second case can only occur when the aggregate demand

$\sum_{j=1}^{n} \hat{D}_j^+(p)$ drops by two or more at the price p. Assuming that all expressions of the form $\hat{B}_i/k$ for integers k are distinct, this can only happen if p equals the valuation v_ℓ of some bidder ℓ. In this case, when ℓ's demand drops to zero, there is no longer any competition for the remaining s items, so the residual demands of all of the bidders can be met simultaneously. There may be items remaining after satisfying all residual demands, in which case they are allocated to the indifferent bidder ℓ (at price $p = v_\ell$).

Example 9.2 (No Demand Reduction) Let's revisit Example 9.1: two items and two bidders, with $B_1 = +\infty$, $v_1 = 6$, and $B_2 = v_2 = 5$. Suppose both bidders bid truthfully. In the uniform-price auction (Example 9.1), the first bidder is awarded both items at a price of 5. In the clinching auction, because the demand $D_2(p)$ of the second bidder drops to 1 once $p = \frac{5}{2}$, the first bidder clinches one item at a price of $\frac{5}{2}$. The second item is sold to the first bidder at a price of 5, as before. The first bidder has utility $\frac{9}{2}$ when she bids truthfully in the clinching auction, and no false bid can be better (Theorem 9.4).

Exercise 9.1 asks you to prove the following proposition.

Proposition 9.3 (Clinching Auction Is Feasible) *The clinching auction always stops, allocates exactly m items, and charges payments that are at most bidders' budgets.*

We now turn to the clinching auction's incentive guarantee.

Theorem 9.4 (Clinching Auction Is DSIC) *The clinching auction for bidders with public budgets is DSIC.*

Proof: We could verify that the auction's allocation rule is monotone and that the payments conform to Myerson's payment formula (3.5), but it's easier to just verify the DSIC condition directly. So, fix a bidder i and bids $\mathbf{b}_{-i}$ by the others. Since bidder i's budget is public, she cannot affect the term $\lfloor \hat{B}_i/p \rfloor$ of her residual demand $\hat{D}_i^+(p)$. She can only affect the time at which she is kicked out of the auction, meaning $\hat{D}_i^+(p) = 0$ forevermore. Every item clinched by bidder i when $p < v_i$ contributes positively to her utility, while every item clinched when $p > v_i$ contributes negatively. Truthful bidding guarantees nonnegative utility.

First, compare the utility earned by a bid $b_i < v_i$ to that earned by a truthful bid. Imagine running the clinching auction twice in parallel, once when i bids b_i and once when i bids v_i. By induction on the number of iterations, the execution of the clinching auction is identical in the two scenarios as the price ascends from 0 to b_i. Thus, by bidding b_i, the bidder can only lose out on items that she otherwise would have clinched (for nonnegative utility) in the price interval $[b_i, v_i]$.

Similarly, if i bids $b_i > v_i$ instead of v_i, the only change is that she might acquire some additional items for nonpositive utility in the price interval $[v_i, b_i]$. Thus, no false bid nets i more utility than a truthful one does. ■

If budgets are private and the clinching auction is run with reported budgets, then it is no longer DSIC (Exercise 9.3).

Is the allocation computed by the clinching auction "good" in some sense? (If only the DSIC condition mattered, then we could give away all the items for free to a random bidder.) There are several ways to formulate this question; see the Notes for details.

9.4 Mechanism Design without Money

There are a number of important applications where incentives matter and the use of money is infeasible or illegal. In these settings, all agents effectively have a budget of zero. Mechanism design without money is relevant for designing and understanding methods for voting, organ donation, and school choice. The designer's hands are tied without money, even tighter than with budget constraints. Despite this and strong impossibility results in general settings, some of mechanism design's greatest hits are for applications without money.

A representative example is the *house allocation problem*. There are n agents, and each initially owns one house. Each agent's preferences are represented by a total ordering over the n houses rather than by numerical valuations. An agent need not prefer her own house over the others. How can we sensibly reallocate the houses to make the agents better off? One answer is given by the *Top Trading Cycle (TTC) algorithm*.

Top Trading Cycle (TTC) Algorithm

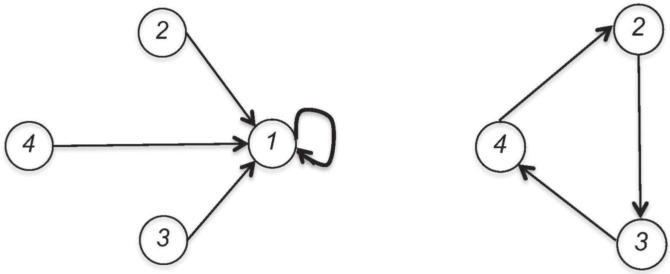
initialize N to the set of all agents
while $N \neq \emptyset$ **do**
  form the directed graph G with vertex set N and
   edge set $\{(i, \ell)$:
   i's favorite house within N is owned by $\ell\}$
  compute the directed cycles $C_1, \ldots, C_h$ of G[3]
```
// self-loops count as directed cycles
// cycles are disjoint
```
  for each edge (i, ℓ) of each cycle $C_1, \ldots, C_h$ **do**
    reallocate ℓ's house to agent i
  remove the agents of $C_1, \ldots, C_h$ from N

The following lemma, which follows immediately from the description of the TTC algorithm, is crucial for understanding the algorithm's properties.

Lemma 9.5 *Let N_k denote the set of agents removed in the kth iteration of the TTC algorithm. Every agent of N_k receives her favorite house outside of those owned by $N_1 \cup \cdots \cup N_{k-1}$, and the original owner of this house is in N_k.*

Example 9.6 (The TTC Algorithm) Suppose $N = \{1, 2, 3, 4\}$, that every agent prefers agent 1's house to the other three, and that the second-favorite houses of agents 2, 3, and 4 are those owned by agents 3, 4, and 2, respectively. (The rest of the agents' preferences do not matter for the example.) Figure 9.1(a) depicts the graph G in the first iteration of the TTC algorithm. There is only one cycle, the self-loop with agent 1. In the notation of Lemma 9.5, $N_1 = \{1\}$. Figure 9.1(b) shows the graph G in the second iteration of the TTC algorithm, after agent 1 and her house have been removed. All agents now participate in a single cycle, and each gets her favorite house among those owned by the agents in $N_2 = \{2, 3, 4\}$.

[3] G has at least one directed cycle, since traversing a sequence of outgoing edges must eventually repeat a vertex. Because all out-degrees are 1, these cycles are disjoint.



(a) First iteration (b) Second iteration

Figure 9.1: The Top Trading Cycle algorithm (Example 9.6).

When agents' total orderings are private, we can consider the direct-revelation mechanism that accepts reported total orderings from the agents and then applies the TTC algorithm. There is no incentive for agents to misreport their preferences to this mechanism.

Theorem 9.7 (TTC Is DSIC) *The TTC algorithm induces a DSIC mechanism.*

Proof: Fix an agent i and reports by the others. Define the sets N_k as in Lemma 9.5, assuming that i reports truthfully. Suppose that $i \in N_j$. The key point is that no misreport can net agent i a house originally owned by an agent of $N_1 \cup \cdots \cup N_{j-1}$. For in each iteration $k = 1, 2, \ldots, j-1$, no agent $\ell \in N_k$ points to i's house—otherwise, i would belong to the same directed cycle as ℓ, and hence to N_k instead of N_j. No agent of N_k points to i's house in an iteration prior to k, either—if she did, she would still point to i's house in iteration k. Thus, whatever agent i reports, she cannot join any cycle involving the agents of $N_1 \cup \cdots \cup N_{j-1}$. Lemma 9.5 then implies that she has no incentive to misreport. ■

Theorem 9.7 by itself is not impressive. For example, the mechanism that never reallocates anything is also DSIC. Our next result gives a sense in which the TTC algorithm is "optimal."

Consider an assignment of one distinct house to each agent. A subset of agents forms a *blocking coalition* for this assignment if they can internally reallocate their original houses to make some member better off while making no member worse off. For example, in an assignment where an agent i receives a house worse than her initial house, $\{i\}$ forms a blocking coalition. A *core allocation* is an assignment with no blocking coalitions.

Theorem 9.8 (TTC and Core Allocations) *For every house allocation problem, the allocation computed by the TTC algorithm is the unique core allocation.*

Proof: We first prove that the only possible core allocation is the one computed by the TTC algorithm. Define the sets N_k as in Lemma 9.5. In the TTC allocation, every agent of N_1 receives her first choice. Thus N_1 forms a blocking coalition for every allocation that differs from the TTC allocation on an agent of N_1. Similarly, in the TTC allocation, all agents of N_2 receive their first choice outside of the houses owned by N_1 (Lemma 9.5). Given that every core allocation agrees with the TTC allocation on the agents of N_1, such allocations must also agree on the agents of N_2—otherwise, N_2 forms a blocking coalition. Continuing inductively, we conclude that every allocation that differs from the TTC allocation is not a core allocation.

To verify that the TTC allocation is a core allocation, consider an arbitrary subset S of agents and an internal reallocation of the houses owned by S. This reallocation partitions S into directed cycles. If some such cycle contains agents from two different N_k's, then the reallocation gives at least one agent i from a set N_j a house originally owned by an agent from a set N_ℓ with $\ell > j$, leaving i worse off than in the TTC allocation (Lemma 9.5). Similarly, for a cycle contained in N_k, any agent that doesn't receive her favorite house from N_k is worse off than in the TTC allocation. We conclude that if the internal reallocation of the houses of S differs from the allocation computed by the TTC algorithm, then some agent of S is worse off. Since S is arbitrary, the TTC allocation has no blocking coalitions and is a core allocation. ■

The Upshot

☆ In many important mechanism design problems, payments are restricted or forbidden. Payment constraints make mechanism design significantly harder.

- ✰ With multiple identical items and bidders with budgets, the uniform-price auction sells all of the items at a common price that equalizes supply and demand.
- ✰ The clinching auction is a more complex auction that sells items piecemeal at increasing prices. Unlike the uniform-price auction, the clinching auction is DSIC.
- ✰ The Top Trading Cycle (TTC) algorithm is a method for reallocating objects owned by agents (one per agent) to make the agents as well off as possible.
- ✰ The TTC algorithm leads to a DSIC mechanism and it computes the unique core allocation.

Notes

The original clinching auction, due to Ausubel (2004), is an ascending implementation of the VCG mechanism in multi-unit auctions with downward-sloping valuations and no budgets (see Problem 9.2). The version in this lecture, with constant valuations-per-item and public budgets, is from Dobzinski et al. (2012).

There are several ways to argue that the clinching auction is in some sense near-optimal. The first way is to follow the development of revenue-maximizing mechanisms (Lecture 5), by positing a distribution over bidders' valuations and solving for the DSIC mechanism that maximizes expected social welfare subject to the given budget constraints. Common budgets are better understood than general budgets are, and in this case the clinching auction is provably near-optimal (Devanur et al., 2013). A second approach, explored in Exercise 9.4, is to modify the social welfare objective function to take budgets into account, replacing $\sum_i v_i x_i$ by $\sum_i \min\{B_i, v_i x_i\}$. The clinching auction is provably near-optimal with respect to this objective function (Dobzinski and Paes Leme, 2014). The third way is to study Pareto optimality rather than an objective function.[4] Dobzinski et al. (2012) prove that the clinching auction is the unique deterministic DSIC auction that always computes a Pareto-optimal allocation. One caveat is that some desirable mechanisms, such as the Bayesian-optimal mechanisms produced by the first approach, need not be Pareto optimal.

Shapley and Scarf (1974) define the house allocation problem, and credit the TTC algorithm to D. Gale. Theorems 9.7 and 9.8 are from Roth (1982b) and Roth and Postlewaite (1977), respectively. Single-peaked preferences (Problem 9.3) are studied by Moulin (1980).

Exercises

Exercise 9.1 Prove Proposition 9.3.

Exercise 9.2 Extend the clinching auction and its analysis to the general case, where the valuations v_i and expressions of the form $\hat{B}_i/k$ for positive integers k need not be distinct.

Exercise 9.3 *(H)* Consider a multi-unit auction where bidders have private valuations per unit *and* private budgets. Prove that the clinching auction, executed with reported valuations and reported budgets, is not DSIC.

Exercise 9.4 Consider a single-parameter environment (Section 3.1) in which each bidder i has a publicly known budget B_i. Consider the allocation rule that, given bids $\mathbf{b}$, chooses the feasible outcome that maximizes the "truncated welfare" $\sum_{i=1}^{n} \min\{b_i x_i, B_i\}$. Ties between outcomes with equal truncated welfare are broken arbitrarily but consistently.

(a) Prove that this allocation rule is monotone, and that the corresponding DSIC mechanism, with payments given by Myerson's

[4] An allocation is *Pareto optimal* if there's no way to reassign items and payments to make some agent (a bidder or the seller) better off without making another worse off, where the seller's utility is her revenue.

payment formula (3.5), never charges a bidder more than her budget.

(b) Consider a single-item environment. Argue informally that the auction in (a) generally results in a "reasonable" outcome.

(c) *(H)* Consider a multi-unit auction with m identical items, where each bidder i has a private valuation v_i per item. Explain why the truncated welfare objective function might assign the same value to almost all of the feasible allocations, and therefore the auction in (a) can easily lead to "unreasonable" outcomes.

Exercise 9.5 Another mechanism for the house allocation problem, familiar from the assignment of dorm rooms to college students, is the *random serial dictatorship*.[5]

Random Serial Dictatorship

```
initialize H to the set of all houses
randomly order the agents
for i = 1, 2, 3, ..., n do
    assign the ith agent her favorite house h from
      among those in H
    delete h from H
```

(a) Does an analog of Theorem 9.7 hold for the random serial dictatorship, no matter which random ordering is chosen by the mechanism?

(b) Does an analog of Theorem 9.8 hold for the random serial dictatorship, no matter which random ordering is chosen by the mechanism?

Problems

Problem 9.1 Consider single-item auctions with n bidders with known bidder budgets.

[5] Some prefer the less hostile term *random priority mechanism*.

(a) Give a DSIC auction, possibly randomized, that always uses nonnegative payments and respects bidders' budgets and achieves (expected) welfare at least $\frac{1}{n}$ times the highest valuation.

(b) *(H)* Prove that there is a constant $c > 0$ such that, for arbitrarily large n and suitable choices of bidders' budgets, for every DSIC single-item auction (possibly randomized) with nonnegative payments that always respects bidders' budgets, there is a valuation profile on which its expected welfare is at most c/n times the highest valuation.[6]

Problem 9.2 In this problem we modify the multi-unit auction setting studied in lecture in two ways. First, we make the mechanism design problem easier by assuming that bidders have no budgets. Along a different axis, we make the problem more general: rather than having a common value v_i for every item that she gets, a bidder i has a private marginal valuation v_{ij} for her jth item, given that she already has $j-1$ items. Thus, if i receives k items at a combined price of p, her utility is $(\sum_{j=1}^{k} v_{ij}) - p$. We assume that every bidder i has a *downward-sloping* valuation, meaning that successive items offer diminishing returns: $v_{i1} \geq v_{i2} \geq v_{i3} \geq \cdots \geq v_{im}$. For simplicity, assume that all of the bidders' marginal valuations are distinct.

(a) Give a simple greedy algorithm for implementing the allocation rule of the VCG mechanism. Does your algorithm still work if bidders' valuations are not downward-sloping?

(b) Give a simple description of the payment of a bidder in the VCG mechanism, as a sum of marginal valuations reported by the other bidders.

(c) Adapt the clinching auction of Section 9.3 to the present setting by redefining bidders' demand functions appropriately. Prove that the allocation and payment rules of your auction are the same as in the VCG mechanism.

[6] Randomized DSIC mechanisms are defined in Problem 6.4.

Problem 9.3 Consider a mechanism design problem where the set of outcomes is the unit interval $[0, 1]$ and each agent i has *single-peaked preferences*, meaning that there is an agent-specific "peak" $x_i \in [0, 1]$ such that i strictly prefers y to z whenever $z < y \leq x$ or $x \geq y > z$. Thus an agent with single-peaked preferences wants the chosen outcome to be as close to her peak as possible.[7]

(a) Is the mechanism that accepts a reported peak from each agent and outputs the average DSIC?

(b) Is the mechanism that accepts a reported peak from each agent and outputs the median DSIC? Feel free to assume that the number of agents is odd.

(c) *(H)* The two mechanisms above are *anonymous*, meaning that the outcome depends only on the unordered set of reported peaks and not on the identity of who reported which peak. They are also *onto*, meaning that for every $x \in [0, 1]$ there is a profile of reported preferences such that x is the outcome of the mechanism. For n agents, can you find more than n different direct-revelation mechanisms that are deterministic, DSIC, anonymous, and onto?

[7] One natural interpretation of $[0, 1]$ is as the political spectrum, spanning the gamut from radical to reactionary.