

CSCI 357: Algorithmic Game Theory

Lecture 5: Myerson's Lemma

Shikha Singh



Announcements and Logistics

- **Assignment 2** due by **11 pm today** (updated time)
 - After lecture 4-5 pm in this room
- **Assignment 1:** feedback returned, student solutions posted on GLOW
- **Assignment 3** will be a **partner** assignment
 - Work together on all problems and submit a joint write up
 - Please fill out <https://tinyurl.com/357partner> by tomorrow (Fri) noon
 - Form has a link to a google doc to help you coordinate!
 - Can also use Slack to coordinate if

Questions?

HW 1 Feedback

- Common issues:
 - Not justifying answers such as why an outcome is an equilibrium
 - Differentiating definitions for fixed a_{-i} vs for all a_{-i}
 - Action profile (a_i, a_{-i}) is important to distinguish i from other's actions
- Question 3 proofs: formalizing the contradiction
 - Intuitively, the idea is simple but the game changes at each step of the algorithm
 - Formalizing requires reasoning about the changes
 - We are trying to show what is left is NE of the original game
- Look at student solutions
 - Ask questions if the feedback is not clear

HW 1 Feedback

- If only a single strategy profile s survives iterated elimination of weakly dominated strategies, then s is a PNE of the original game.
- Proof sketch: Assume that s is not a pure-strategy Nash of the game
 - There exists a player i who is not playing a best response in s , that is, $s = (s_i, s_{-i})$ and $\exists s'_i$ s.t. $u_i(s'_i, s_{-i}) > u_i(s_i, s_{-i})$
 - Let S^t denote the action available to players at step t
 - Since only one strategy profile survives, s'_i must have been eliminated at some time step t because it was weakly dominated by some action s'' :
$$u_i(s'_i, a_{-i}) \leq u_i(s''_i, a_{-i}) \forall a_{-i} \in S^t_{-i}$$
 - s_{-i} must be part of S^t_{-i} : because it is never eliminated
 - Similarly a''_i must have been eliminated, and so on,
$$u_i(a'_i, a_{-i}) \leq u(a'_i, a_{-i}) \leq \dots \leq u_i(a_i, a_{-i}) \Rightarrow \Leftarrow \blacksquare$$

HW 2

- **Q2.** If S is the set of colluders then their total utility is

$$u_S(\mathbf{b}) = \begin{cases} (\sum_{i \in S} v_i) - p & \text{if any } j \in S \text{ wins the item and pays } p \\ 0 & \text{otherwise} \end{cases}$$

- Perfectly shareable item (or rather think of it as a service)
- Each bidder still needs to enter a bid (but colluders can coordinate)
- Question asks for necessary and sufficient conditions on valuations in S such that collusion gives better total u_S than honest bidding
- Break it up into case where truthful bidding would lead to win/loss
 - You may use $B = \max_{j \notin S} b_j$ (max bid outside) in your condition/analysis
- Q4c. Strategy is a mapping from **information available** to **action**

Last Time

- Analyze single item (sealed bid) auctions
- Second price (Vickrey auctions) are strategyproof (DSIC) and maximize surplus in linear time
- Ran a first price auction:
 - We will discuss the results on Monday, stay tuned!
- Meantime, think about
 - pros and cons of first-price vs second-price auction
 - downsides of sealed-bid auctions compared to ascending/descending

Today: Single Parameter

- General characterization of strategyproof (DSIC) mechanisms in single-parameter settings
 - Valuation for whatever allocation a bidder receives can be captured by a single number

n buyer with private valuations which can be described by a **single number** v_i



Multiple items



Example: k identical goods

- Simple example of single-parameter setting: we have k copies of an item
- Feasible allocation is then $X = (x_1, \dots, x_n) \subseteq \{0,1\}^n$, where $x_i = 1$ if bidder i gets a copy; 0 otherwise and $\sum_{i=1}^n x_i \leq k$

n buyers, each has private value v_i
for a single copy of the item



k identical items



Sponsored Search Model [Edelman & Varian]

- Every time someone searches a query, an auction **is run in real time** to decide: which advertisers links are shown, in what order, and how they are charged
- We look at a simplified but effective model to study sponsored search auction
- Items for sale are k slots for sponsored links on a page
- Bidders (advertisers) have a standing bid on a keyword that was searched on
- Slots higher up on the page are more valuable than low
 - Users more likely to click on them

The image shows a screenshot of a Google search results page for the query "mortgage". The page displays several sponsored advertisements. Three specific slots are highlighted with orange text and arrows:

- 1st slot:** Points to the top advertisement from www.lendingtree.com/home_mortgages with the headline "Compare Best Home Mortgages - See How Much You Can".
- 2nd slot:** Points to the second advertisement from www.bankrate.com/ with the headline "Today's Mortgage Rates - Sep 24, 2020".
- 3rd slot:** Points to the third advertisement from www.nerdwallet.com/ with the headline "Compare Mortgage Lenders - Find A Great Rate Today".

Below the 3rd slot, there is a link to "Mortgage Calculator | Bankrate - Bankrate.com" and a brief description of the calculator's functionality.

Sponsored Search Model [Edelman & Varian]

- Slots higher up on the page more likely to be clicked
 - Quantified through **click-through-rates (CTRs)**
 - CTR α_j of a slot j is the probability of clicks it is expected to receive
 - Reasonable to assume $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_n$
- **Simplifying assumption.** CTR of a slot is independent of its occupant, that is, doesn't depend on the quality of the ad
- Assume advertisers have a private valuation v_i **for each click on its link**: value derived from slot j by advertiser i is $v_i \cdot \alpha_j$

The image shows a screenshot of a Google search results page for the query 'mortgage'. The page displays search results and several advertisements. Three specific ads are highlighted with orange arrows and labels to the left, representing the '1st slot', '2nd slot', and '3rd slot' in the sponsored search model.

1st slot points to the top advertisement from www.lendingtree.com/home_mortgages with the headline 'Compare Best Home Mortgages - See How Much You Can'. Below the headline, it says 'Compare Your Best Mortgage Loans & Calculate Payments. Shop With LendingTree of lenders. View offers online. Free credit score. Rate comparisons. Services: Home Refinance, Home Equity Line.' There are also links for '1st Time Buyer' and 'Refinance'.

2nd slot points to the second advertisement from www.bankrate.com/ with the headline 'Today's Mortgage Rates - Sep 24, 2020'. Below the headline, it says 'Bankrate® Surveys Top Mortgage Lenders and Banks to Provide Today's Best Rates'.

3rd slot points to the third advertisement from www.nerdwallet.com/ with the headline 'Compare Mortgage Lenders - Find A Great Rate Today'. Below the headline, it says 'Compare Offers From Our Partners Side by Side And Find The Perfect Lender For You'.

Below the third ad, there is a link to 'Mortgage Calculator | Bankrate - Bankrate.com' with the text 'Use our free mortgage calculator to help you estimate your monthly mortgage payment. Account for interest rates and break down payments in an easy to use ...' and a list of links: '15-year or 30-year mortgage', 'Interest Only Mortgage', and 'Private Mortgage Insurance'.

Example: Sponsored Search

- A **feasible allocation** is an assignment of bidders to slots, such that each slot is assigned to at most one bidder and each bidder is assigned at most one slot, that is, $X = (x_1, x_2, \dots, x_n)$
 - where $x_i = \alpha_j$, the click through of slot j if bidder i is assigned to it; otherwise $x_i = 0$ if bidder is unassigned

n buyers, each has private value of v_i
"per click" they get



k slots, with different
click-through rates α_j



Example: Public Project

- Single parameter settings are **more general than auctions**
- For example, deciding whether or not to build a public project that can be used by everyone can be modeled by the allocation
 $X = \{(0, \dots, 0), (1, \dots, 1)\}$
- Auctions are a special case of general mechanisms
- Auctions involve transfer of goods and money but this is not necessary for the results we will study

| auction | mechanism |
|-----------|-----------|
| bidder | agent |
| bid | report |
| valuation | valuation |

Table 3.1: Correspondence of terms in auctions and mechanisms. An auction is the special case of a mechanism that is designed for the exchange of goods and money.

Sealed-Bid Mechanism



- We will focus on sealed-bid mechanisms that
 - Collect bids/reports $\mathbf{b} = (b_1, \dots, b_n)$
 - Choose a feasible allocation rule $\mathbf{x}(\mathbf{b}) \in X \subseteq \mathbb{R}^n$
 - Choose payments $\mathbf{p}(\mathbf{b}) \in \mathbb{R}^n$
- Such mechanisms are called **direct-revelation mechanism**
 - Mechanisms that ask agents to report their private value up front
- **Quasilinear utility:** $u_i(\mathbf{b}) = v_i \cdot x_i(\mathbf{b}) - p_i(\mathbf{b})$ on the bid profile \mathbf{b}
- We will focus on payment rules that satisfy
 - $p_i(\mathbf{b}) \geq 0$: sellers **can't pay the bidders**
 - $p_i(0, \mathbf{b}_{-i}) = 0$: a zero bid leads to a zero payment

Design Approach

- Our goal is to **maximize surplus** $\operatorname{argmax}_{(x_1, \dots, x_n) \in X} \sum_{i=1}^n v_i x_i$
- **Challenge:** jointly design two pieces: who gets what, and how much do they pay
 - Not enough to figure out who wins, if don't charge them the right amount
- Usually, the recipe we will follow:
 - **Step 1.** Assume **truthful bids**, and decide how to allocate so as to **maximize surplus** (in polynomial time)
 - **Step 2.** Using the allocation in step 1, decide how to charge payments so as that the mechanism is strategyproof (DSIC)

k identical goods: Allocation

- Collect sealed bids
- Who should we give the k items to maximize surplus (assuming truthful bids)
 - Top k bidders
- **Question.** What should we charge them so that truth telling is dominant strategy?

n buyers, each has private value v_i
for a single copy of the item



k identical items



Sponsored Search: Allocation

- How do we do we assign slots to maximize $\sum_{i=1}^n b_i x_i$?
 - Greedy allocation is optimal (can be showed by an exchange argument)
 - Recall that CTR rates $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_k$
 - Sort and relabel bids $b_1 \geq b_2 \geq \dots \geq b_n$
 - Assign j th highest bidder to j th highest slot
- Can we create a payment rule (an analog of second-price rule) that makes the greedy allocation incentive compatible?
 - What is the analog of the second-price auction here?

Towards a General Characterization

- **Question.** Can any allocation rule be paired with a payment rule such that the mechanism is strategyproof (truthtelling is a dominant strategy)?
 - When is this possible and how should we design the payment rule?
- Myerson's lemma gives a **general characterization** of allocation rules that can be turned into a truthful (DSIC) mechanism
- And tells us exactly how to design payment rules to achieve that

Myerson's Lemma: Informal

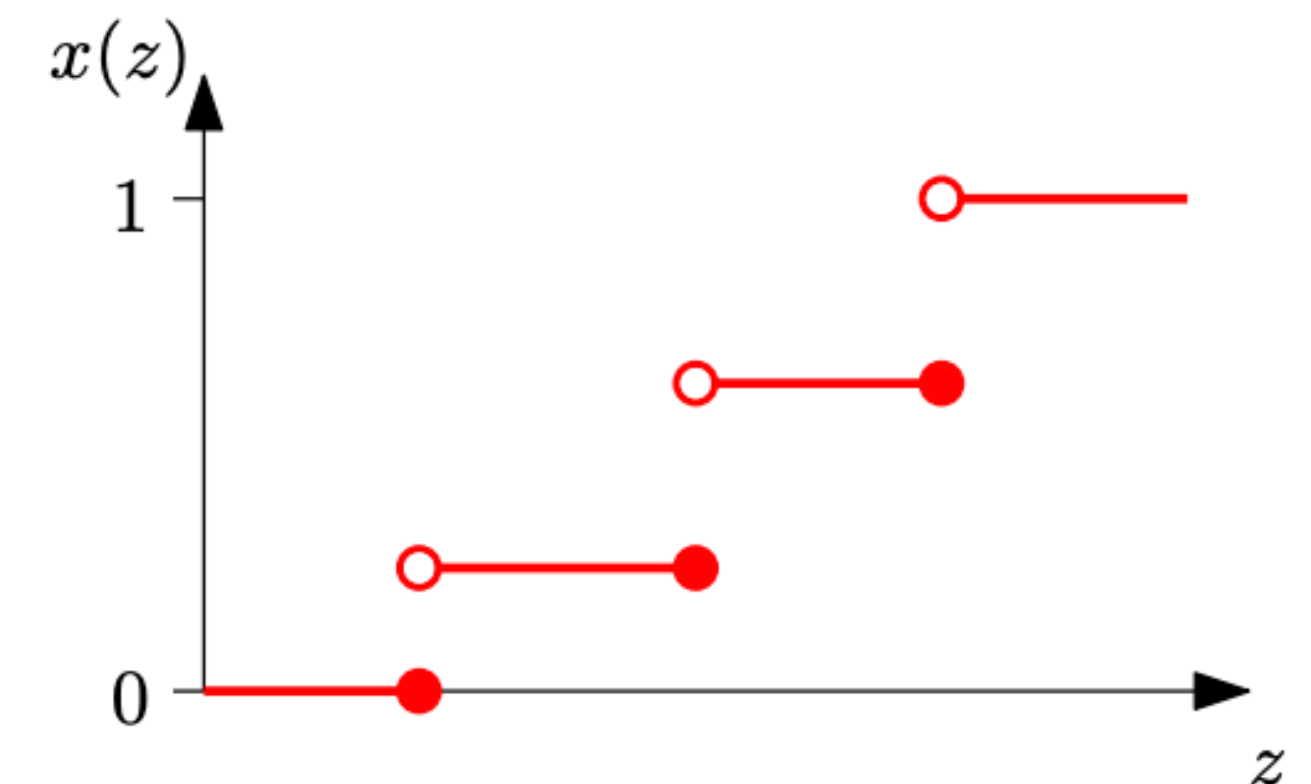
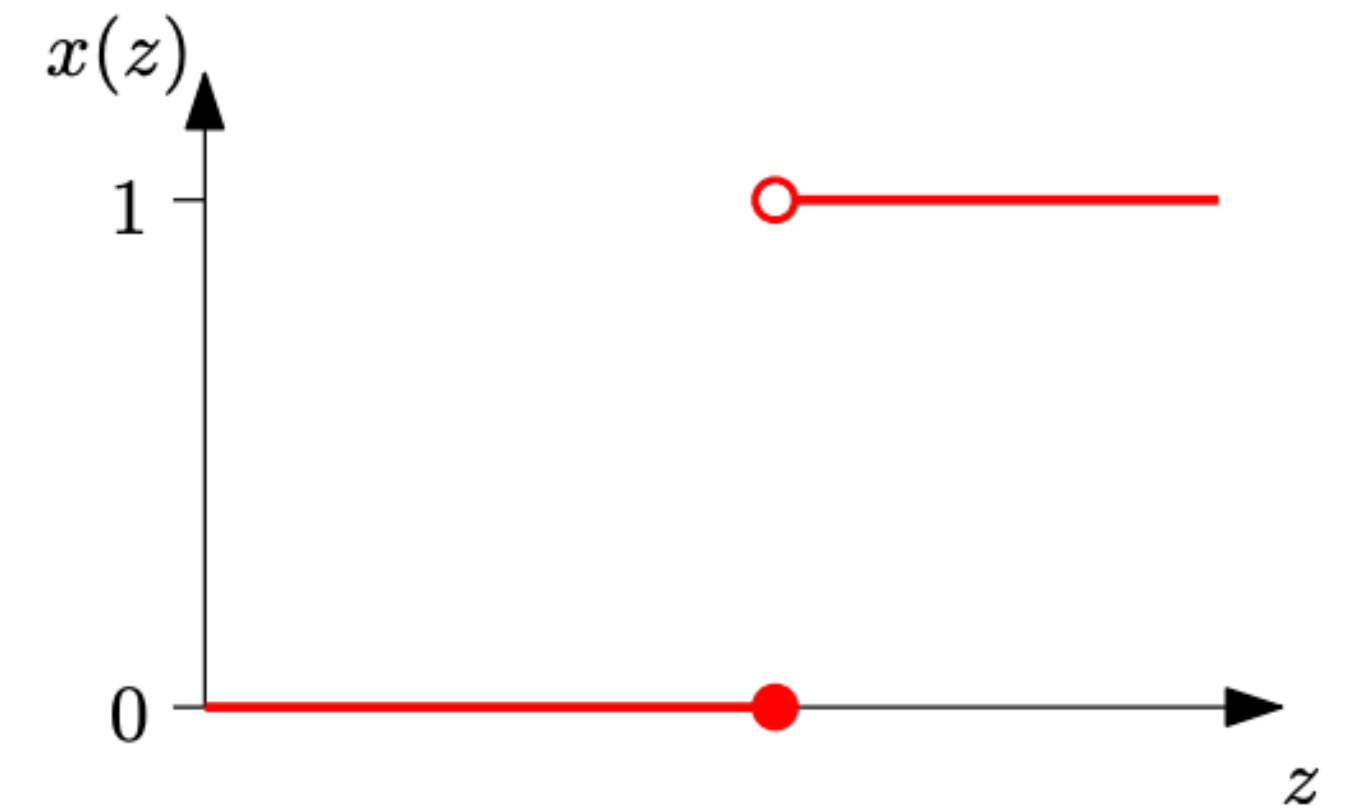
- In a fixed-parameter setting,
 - an allocation rule \mathbf{x} can be made dominant-strategy incentive compatible if and only if \mathbf{x} is **monotone (non decreasing)**, and
 - if \mathbf{x} is monotone, there is a **unique payment** rule \mathbf{p} such that (\mathbf{x}, \mathbf{p}) is DSIC.
- Question of whether there exists a payment that makes an allocation DSIC (a difficult to answer question) reduced to a question of whether a rule is "monotone" : a computation/ operational question

Monotone Allocation Rule

- **Definition (Monotone allocation rule).**

An allocation rule $\mathbf{x} = (x_1, \dots, x_n)$ for a single-parameter domain is monotone-non-decreasing if for every bidder i and bids \mathbf{b}_{-i} of other bidders, the allocation $x_i(z, \mathbf{b}_{-i})$ to i is non-decreasing in its bid z .

- That is, in a monotone allocation rule, bidder higher can only get you “more” stuff
- In any single-item auction that allocates the item to the highest bidder is monotone: if you’re the winner and you raise your bid (keeping other bids constant), you continue to win
- Is allocating the item to second-highest bidder monotone?



Myerson's Lemma

- Fix an single-parameter domain. We state the result for the continuous case.

(a) An allocation rule \mathbf{x} can be made dominant-strategy incentive compatible if and only if \mathbf{x} is monotone (non decreasing).

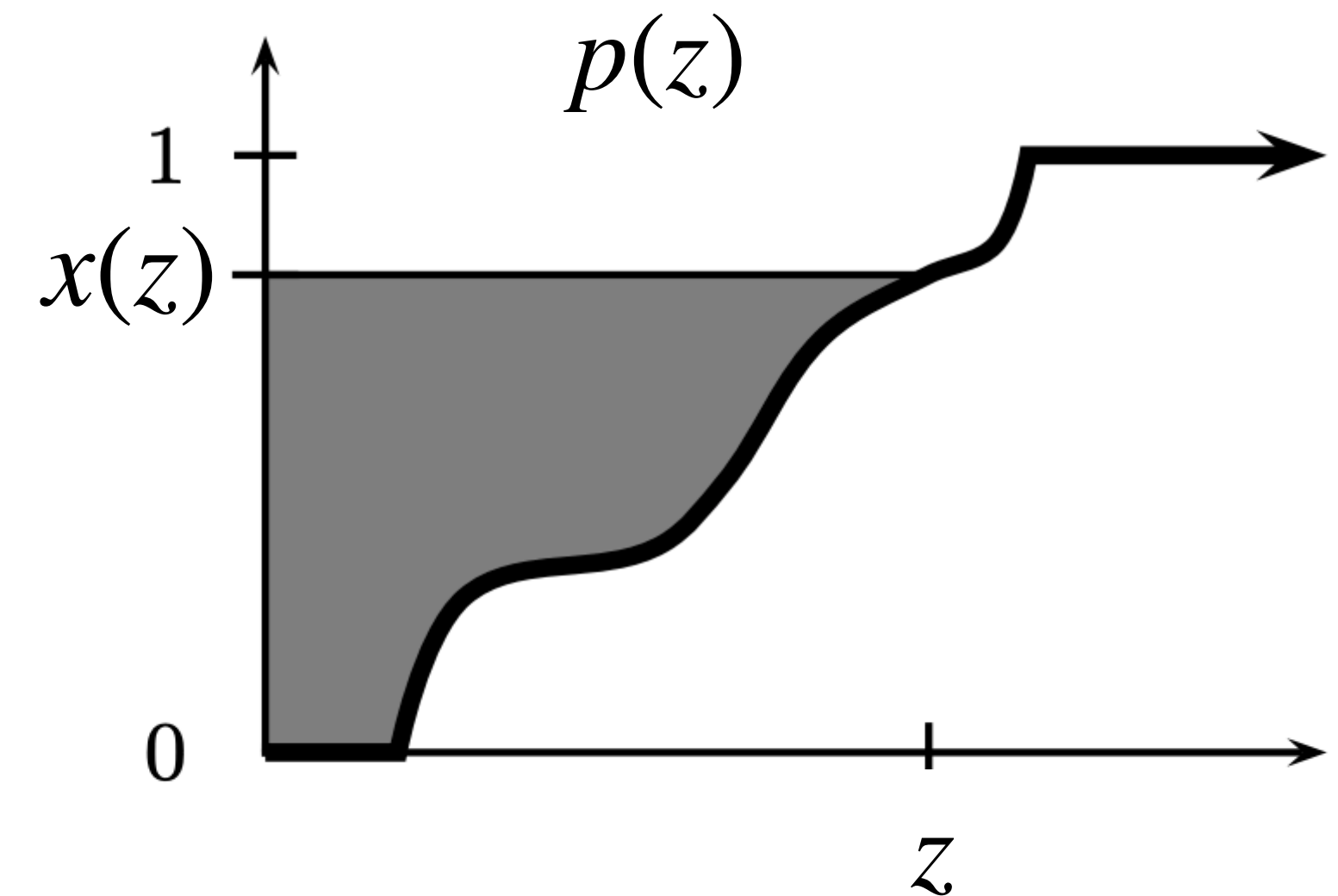
(b) If \mathbf{x} is monotone, there is a unique payment rule \mathbf{p} such that (\mathbf{x}, \mathbf{p}) is DSIC. This payment rule is given by the following expression for all i :

$$p_i(z, \mathbf{b}_{-i}) = z \cdot x_i(z, \mathbf{b}_{-i}) - \int_0^z x_i(z, \mathbf{b}_{-i}) dz$$

where player i bids z . Keeping \mathbf{b}_{-i} fixed, we can simplify:

$$p_i(z) = z \cdot x_i(z) - \int_0^z x_i(z) dz$$

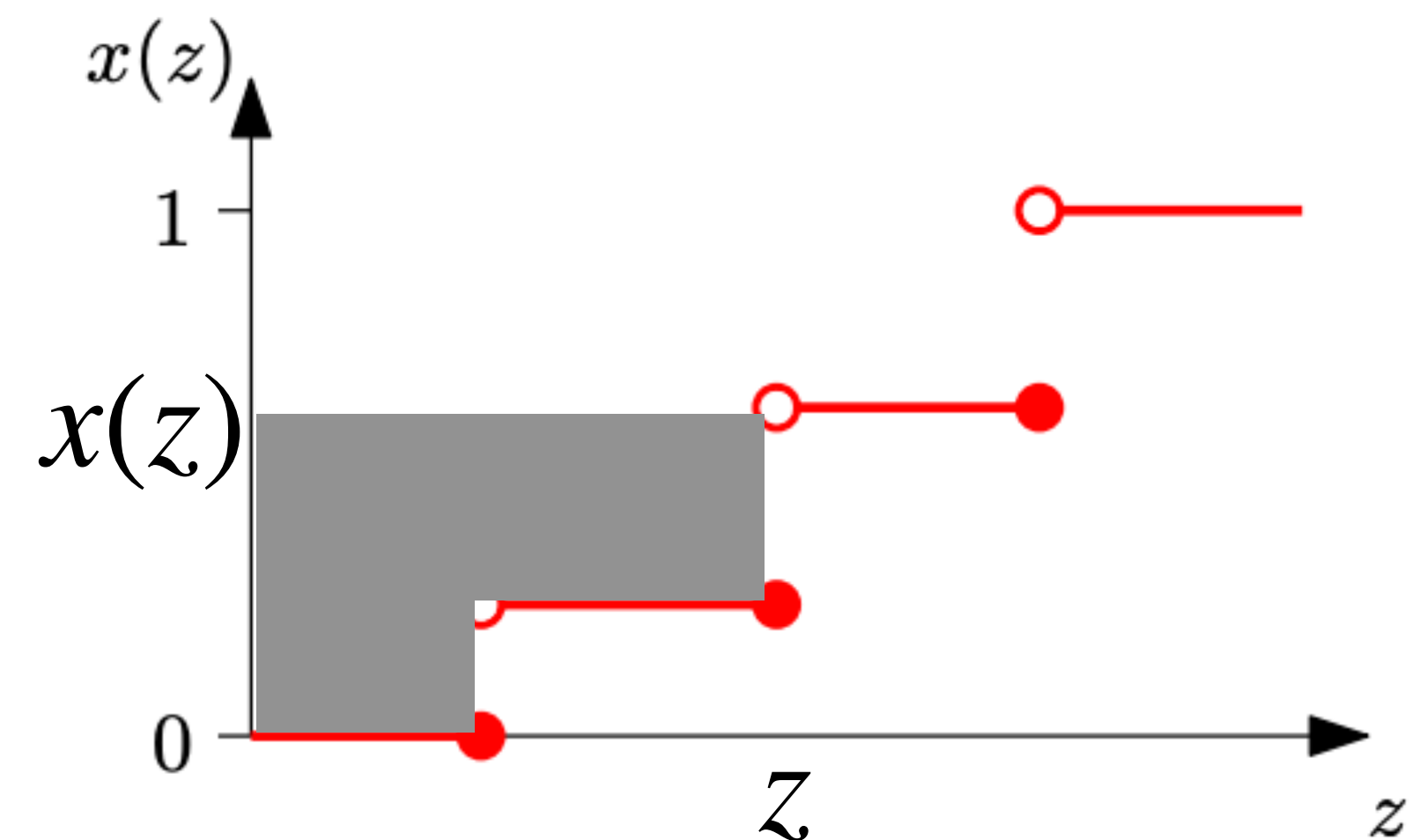
Assuming that $p_i(0) = 0$.



Myerson's Lemma

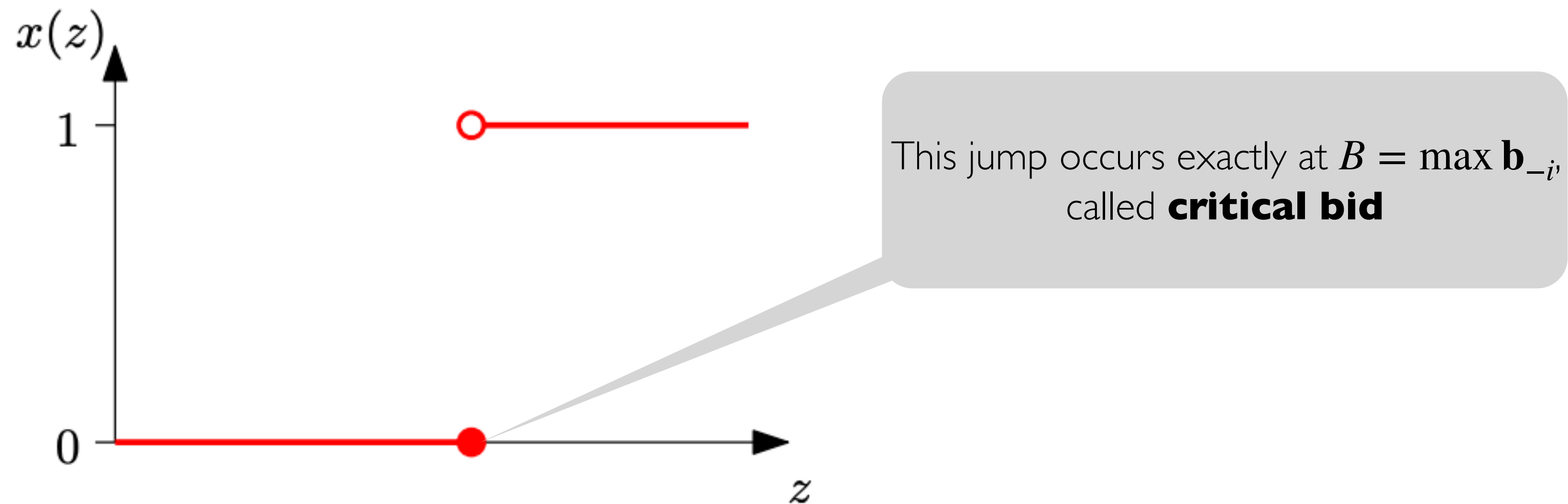
- Fix an single-parameter domain. We state the result for the **discrete** case.
 - (a) An allocation rule \mathbf{x} can be made dominant-strategy incentive compatible if and only if \mathbf{x} is monotone (non decreasing).
 - (b) If \mathbf{x} is monotone, there is a unique payment rule \mathbf{p} such that (\mathbf{x}, \mathbf{p}) is DSIC. This payment rule for all i :
- If there are ℓ points at which the allocation "jumps" before bid z , the payment at bid z

$$p_i(z) = \sum_{j=1}^{\ell} z_j \cdot [\text{jump in } x_i \text{ at } z_j]$$



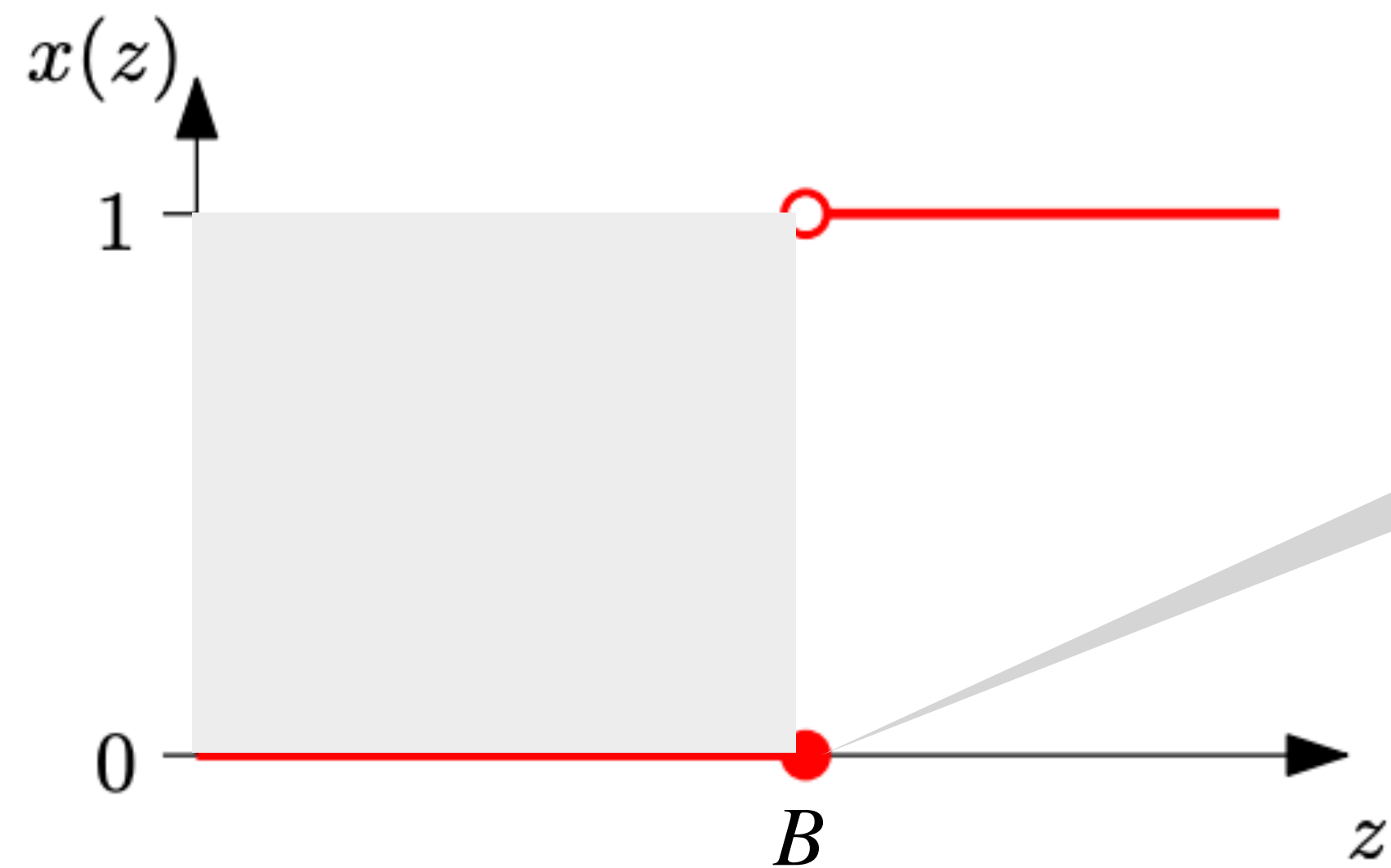
Single-Item Auction

- Let's apply Myerson's lemma to a single item auction that allocates the item to highest bidder
- This allocation rule is monotone: in fact a **0/1** monotone curve
- Fixing \mathbf{b}_{-i} , we can plot bidder i allocation wrt to bid:



Single-Item Auction

- If $z < B$: payment is 0
- If $z \geq B$: payment is given by shaded region, that is, B
- We have recreated the Vickrey auction from Myerson's lemma
- Moreover, this payment scheme is the only way to make the allocation rule (giving to highest bidder) truthful!

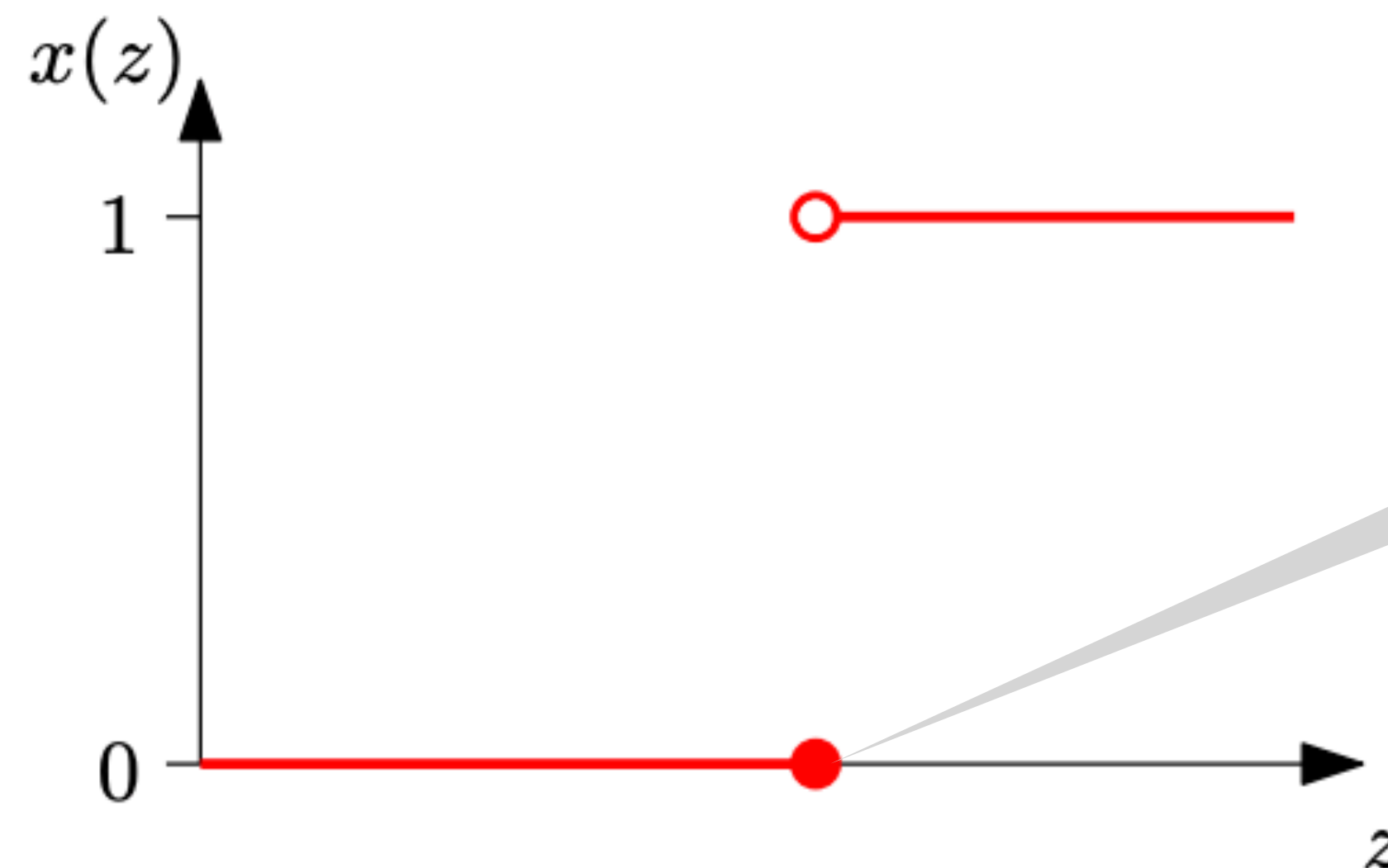


This jump occurs exactly at $B = \max \mathbf{b}_{-i}$, called **critical bid**

Any 0/1 Allocation Mechanism

- In a single-parameter environment, let X be any 0/1 feasible allocation (each player either wins $x_i = 0$ or loses $x_i = 1$)
 - Example: auctioning k units of the same item to n bidders
- In such auctions, what should the winners pay?

$$p(b_i, \mathbf{b}_{-i}) = \begin{cases} 0 & \text{if } x_i(b_i, \mathbf{b}_{-i}) = 0 \\ b_i^*(\mathbf{b}_{-i}) & \text{if } x_i(b_i, \mathbf{b}_{-i}) = 1 \end{cases}$$

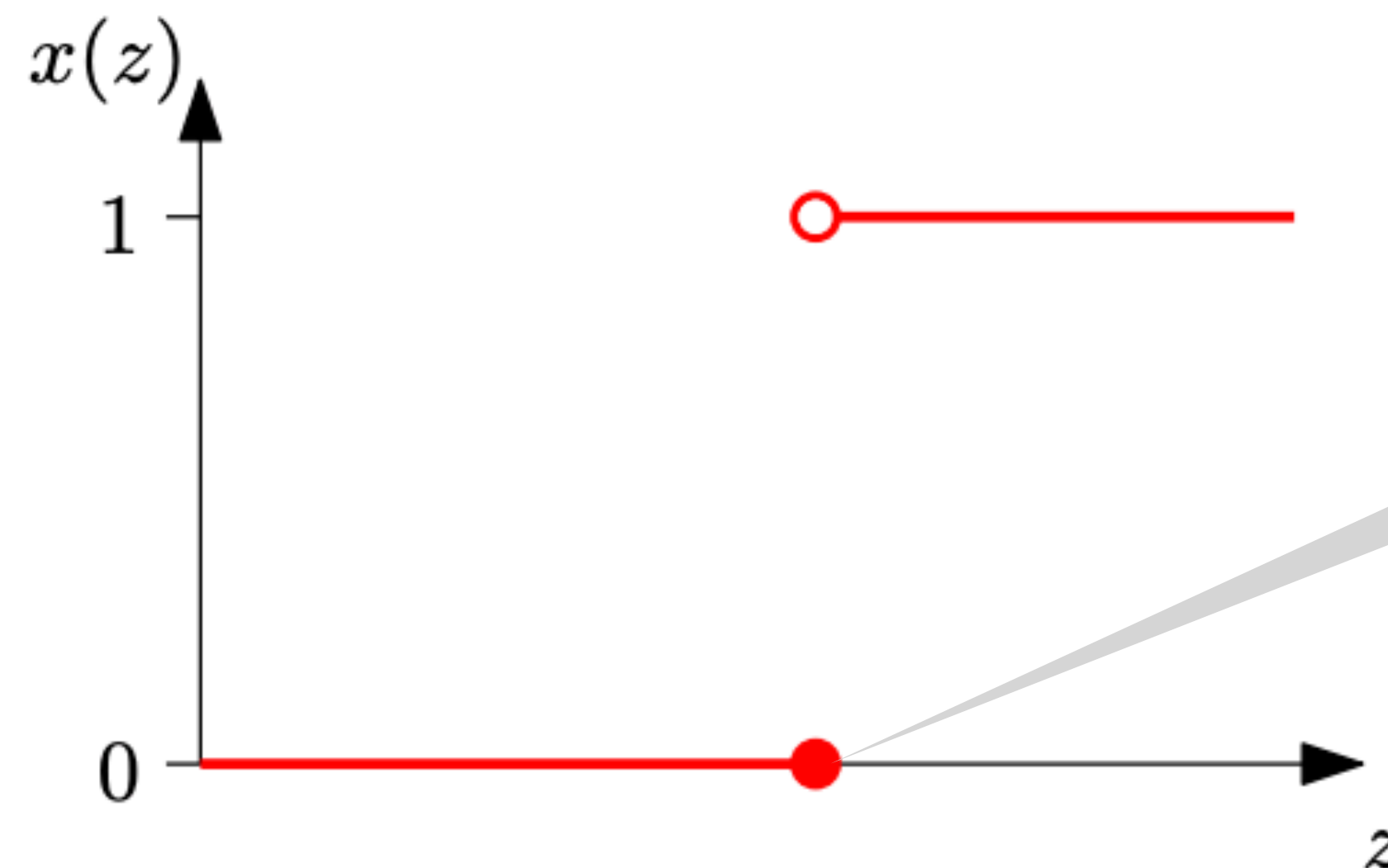


Critical bid: $b_i^*(\mathbf{b}_{-i})$ lowest bid at which i 's allocation goes from 0 to 1

Any 0/1 Allocation Mechanism

- In a single-parameter environment, let X be any 0/1 feasible allocation (each player either wins $x_i = 0$ or loses $x_i = 1$)
 - Example: auctioning k units of the same item to n bidders
- In such auctions, what should the winners pay?
 - $(k + 1)^{\text{st}}$ highest bid

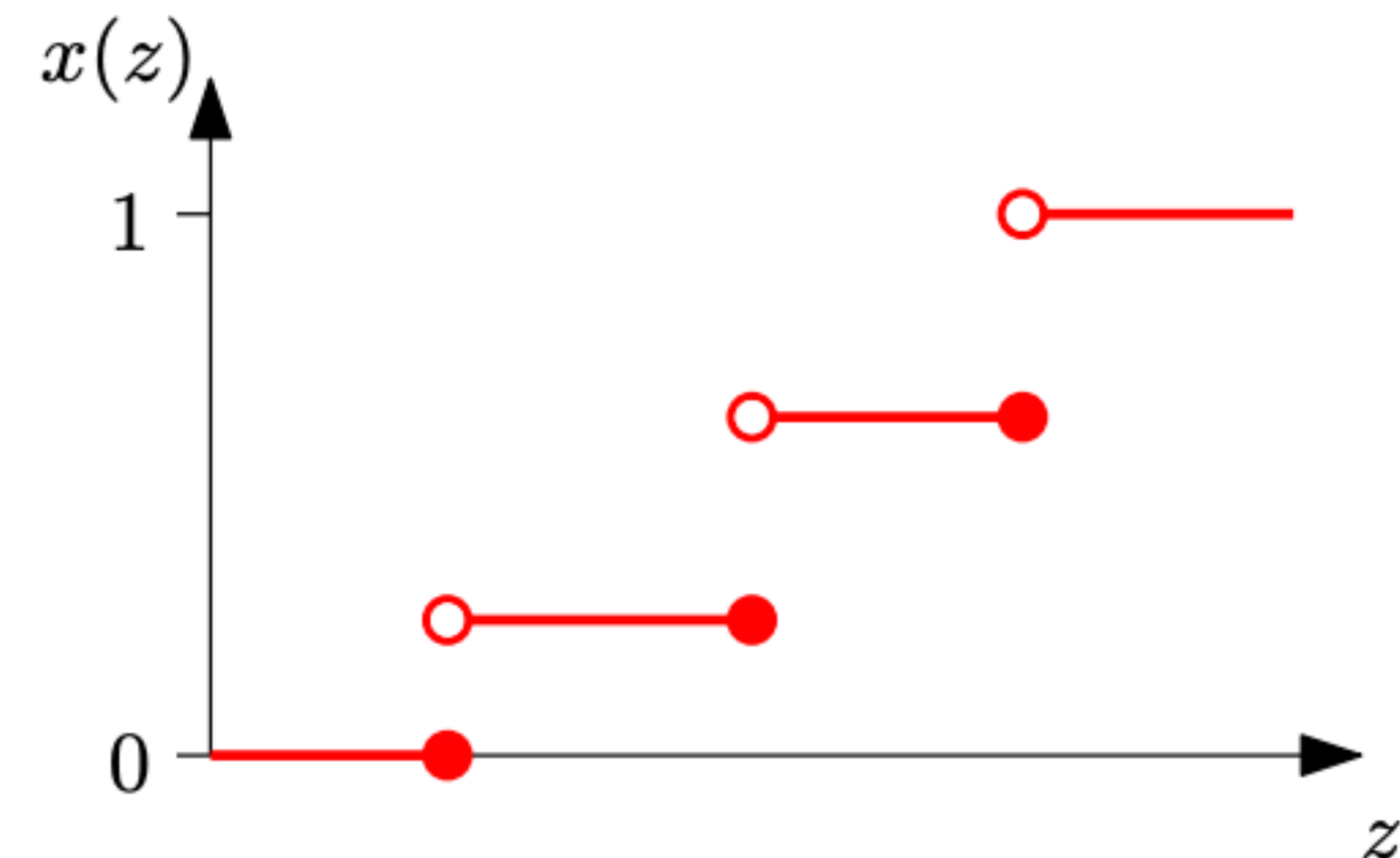
$$p(b_i, \mathbf{b}_{-i}) = \begin{cases} 0 & \text{if } x_i(b_i, \mathbf{b}_{-i}) = 0 \\ b_i^*(\mathbf{b}_{-i}) & \text{if } x_i(b_i, \mathbf{b}_{-i}) = 1 \end{cases}$$



Critical bid: $b_i^*(\mathbf{b}_{-i})$ lowest bid at which i 's allocation goes from 0 to 1

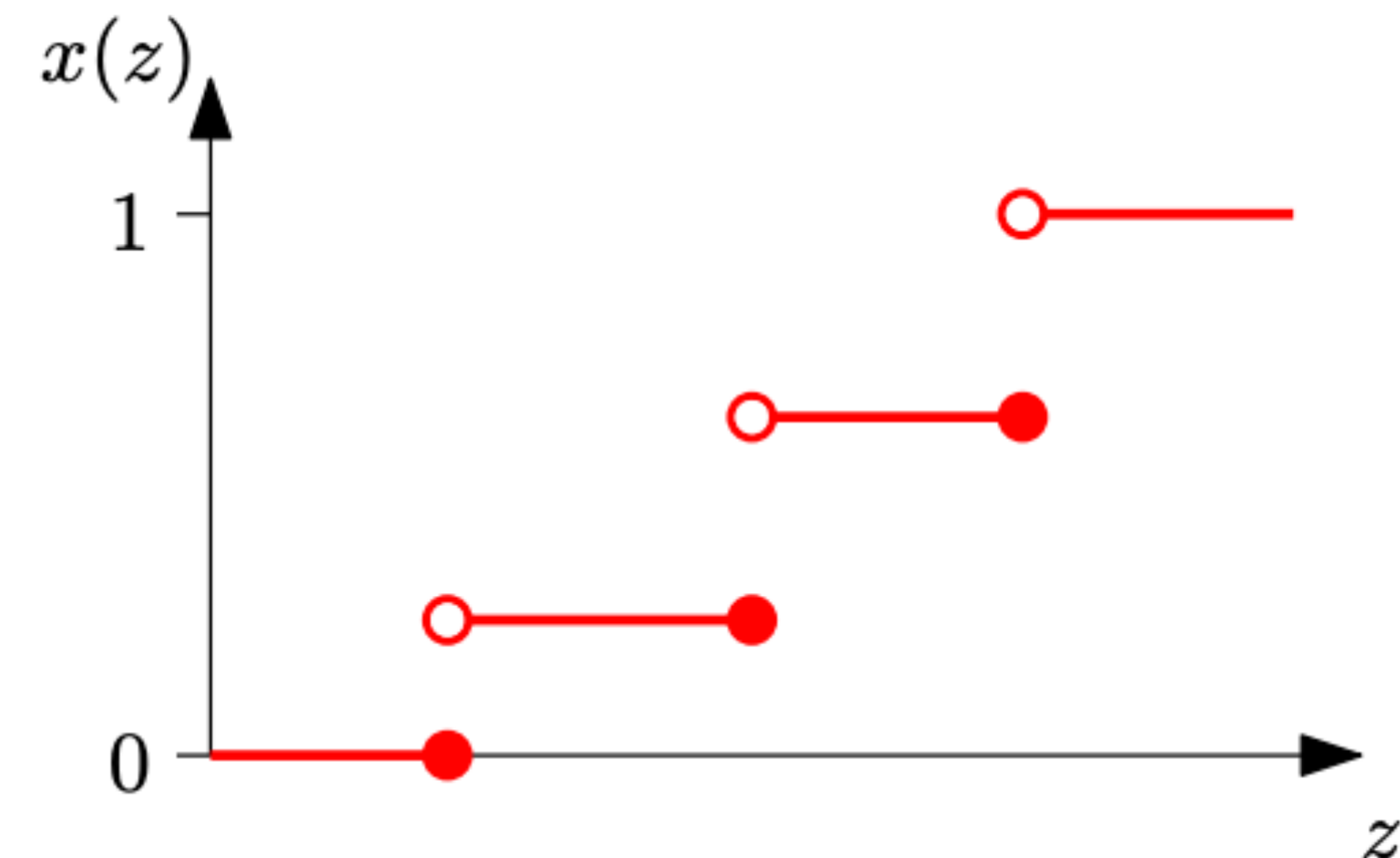
Sponsored-Search Auctions

- Sort bids $b_1 \geq b_2 \geq \dots \geq b_n$ (reorder bidders in this order)
- Assign slot 1 to bidder 1, slot 2 to bidder 2, etc.
- That is, CTR α_j of slot j gets assigned to bidder j
- What does the graph of such an allocation rule look like?
 - For intuition fix \mathbf{b}_{-i} and think of yourself as bidder 1 slowly raising your value from 0



Sponsored-Search DSIC Payment

- If you get no slot, you pay zero
- If you get last slot, you pay the “critical” bid that you beat out to get the slot (the bid of the person just below you in sorted order)
- If you get a lower slot j better than k , what do you pay?
 - **Exercise:** come with the expression for the payment p_j of bidder who wins slot j using Myerson’s rule?
- We will come back to this!



Myerson's Lemma: Proof

- **Part 1:** An allocation \mathbf{x} rule can be made DSIC only if \mathbf{x} is monotone
- **Part 2:** A mechanism (\mathbf{x}, \mathbf{p}) , where \mathbf{x} is monotone, is DSIC only if \mathbf{p} is given by the expression in Myerson's lemma
- **Part 3:** Finally, we show that if the allocation \mathbf{x} is monotone and the payment rule \mathbf{p} is as given by the expression in the lemma then, (\mathbf{x}, \mathbf{p}) is DSIC
- **Recall DSIC condition:** for every agent i , every possible private valuation v_i , every set of bids \mathbf{b}_{-i} by the other agents, i 's utility is maximized by bidding truthfully
- Fix an arbitrary player i and bid profile of others \mathbf{b}_{-i}
- Let $x(z)$ and $p(z)$ be shorthand for i 's allocation $x_i(z, \mathbf{b}_{-i})$ & payment $p_i(z, \mathbf{b}_{-i})$
- Throughout the proof, we will vary the bid z and see how it changes the allocation

Myerson's Lemma: Proof Part 1

- **Part 1.** An allocation rule \mathbf{x} can be made dominant-strategy incentive compatible only if \mathbf{x} is monotone non-decreasing
- If player i (with value v) deviates and bids as if she has value z , then her utility is $v \cdot x(z) - p(z)$
 - Notice: no control over your value v
- For truth telling to be a (weakly) dominant strategy for all values, must be that
 - $v \cdot x(v) - p(v) \geq v \cdot x(v^\dagger) - p(v^\dagger)$ for all v, v^\dagger
- We consider two possible values z_1, z_2 with $z_1 < z_2$
 - Case 1 (Underbidding): $v = z_2, v^\dagger = z_1$
 - Case 2 (Overbidding): $v = z_1, v^\dagger = z_2$

Myerson's Lemma: Proof Part 1

- In case (a), where $v = z_2$ and player underbids z_1

$$z_2 \cdot x(z_2) - p(z_2) \geq z_2 \cdot x(z_1) - p(z_1) \text{ — (Ineq 1)}$$

- In case (b), where $v = z_1$ and player overbids z_2

$$z_1 \cdot x(z_1) - p(z_1) \geq z_1 \cdot x(z_2) - p(z_2) \text{ — (Ineq 2)}$$

- Adding both: $z_2 \cdot x(z_2) + z_1 \cdot x(z_1) \geq z_2 \cdot x(z_1) + z_1 \cdot x(z_2)$

- Rearranging: $(z_2 - z_1) \cdot (x(z_2) - x(z_1)) \geq 0$

- Does this imply something about the allocation rule \mathbf{x} ?

- Since $z_2 > z_1$, this only holds if $x(z_2) \geq x(z_1)$: thus \mathbf{x} must be monotone non-decreasing ■ (Part 1)

Myerson's Lemma: Proof Part 2

- **Part 2.** A mechanism \mathbf{x}, \mathbf{p} , where \mathbf{x} is monotone can be made DSIC if and only if the payment rule \mathbf{p} satisfies:

$$p(z) = z \cdot x(z) - \int_0^z x(z) dz$$

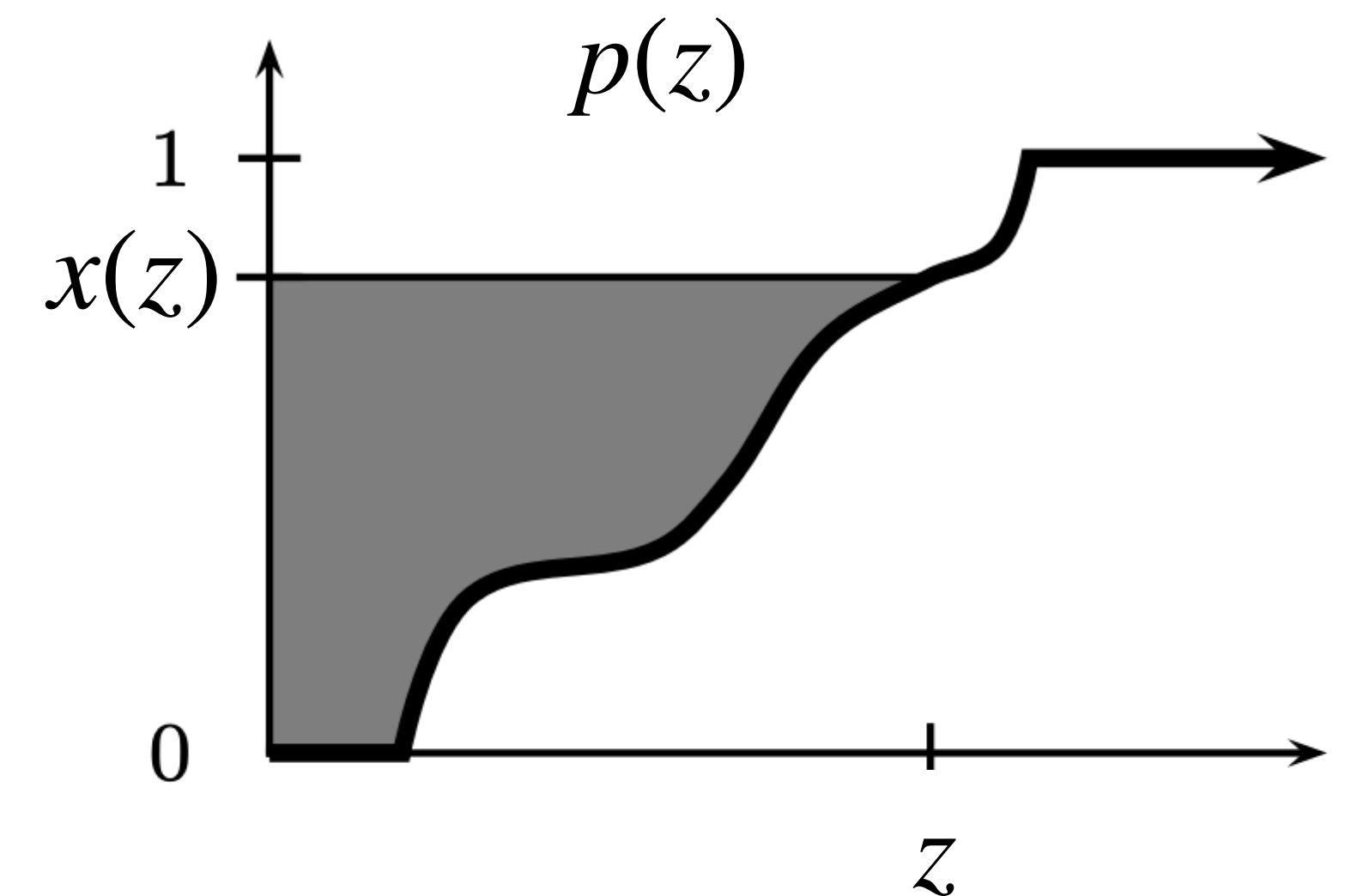
- We reuse the inequalities from part 2 of the proof:

$$z_2 \cdot x(z_2) - p(z_2) \geq z_2 \cdot x(z_1) - p(z_1) \text{ — (Ineq 1)}$$

$$z_1 \cdot x(z_1) - p(z_1) \geq z_1 \cdot x(z_2) - p(z_2) \text{ — (Ineq 2)}$$

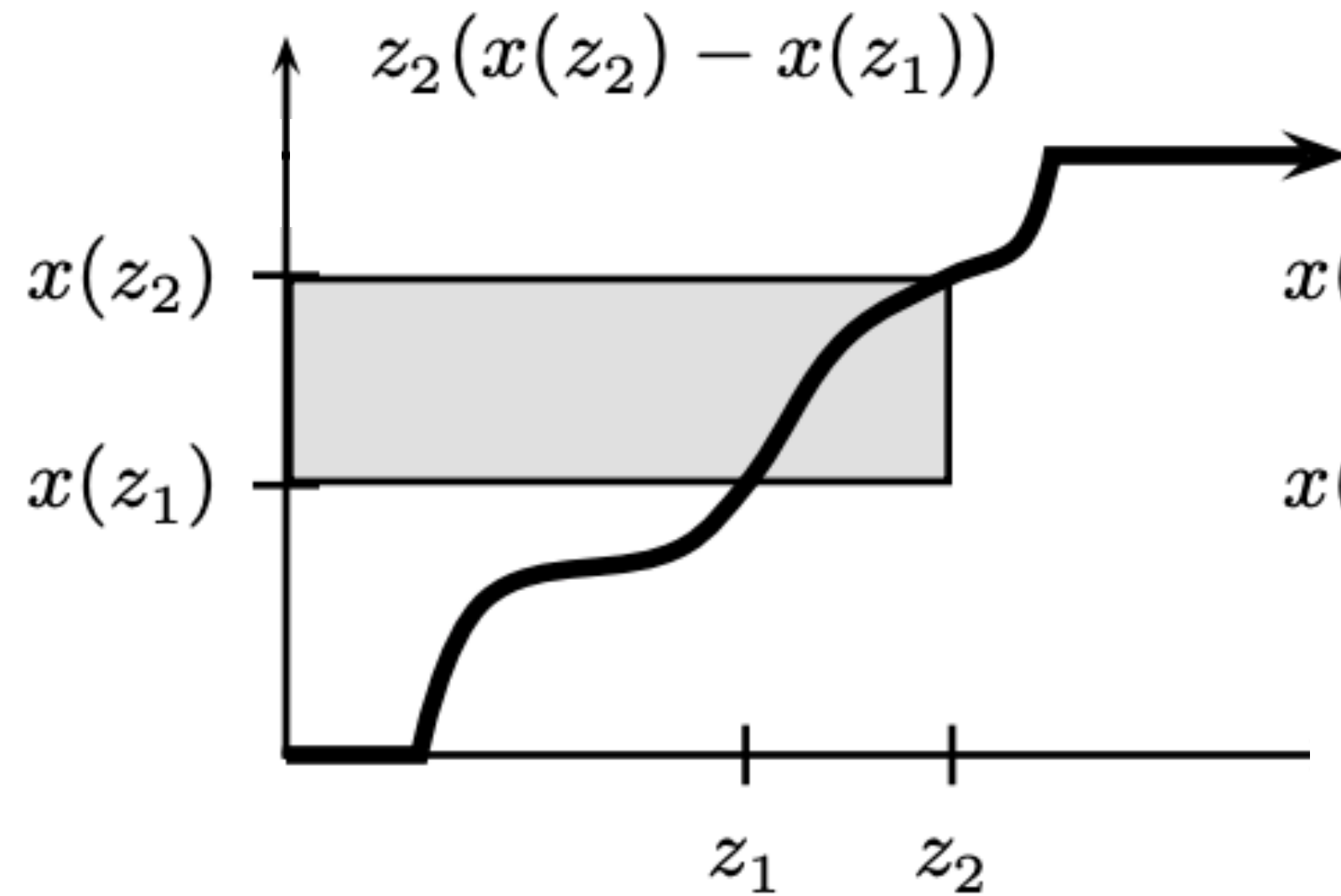
- We can upper and lower bound $p(z_2) - p(z_1)$ using them as

$$z_2 \cdot (x(z_2) - x(z_1)) \geq p(z_2) - p(z_1) \geq z_1 \cdot (x(z_1) - x(z_1))$$

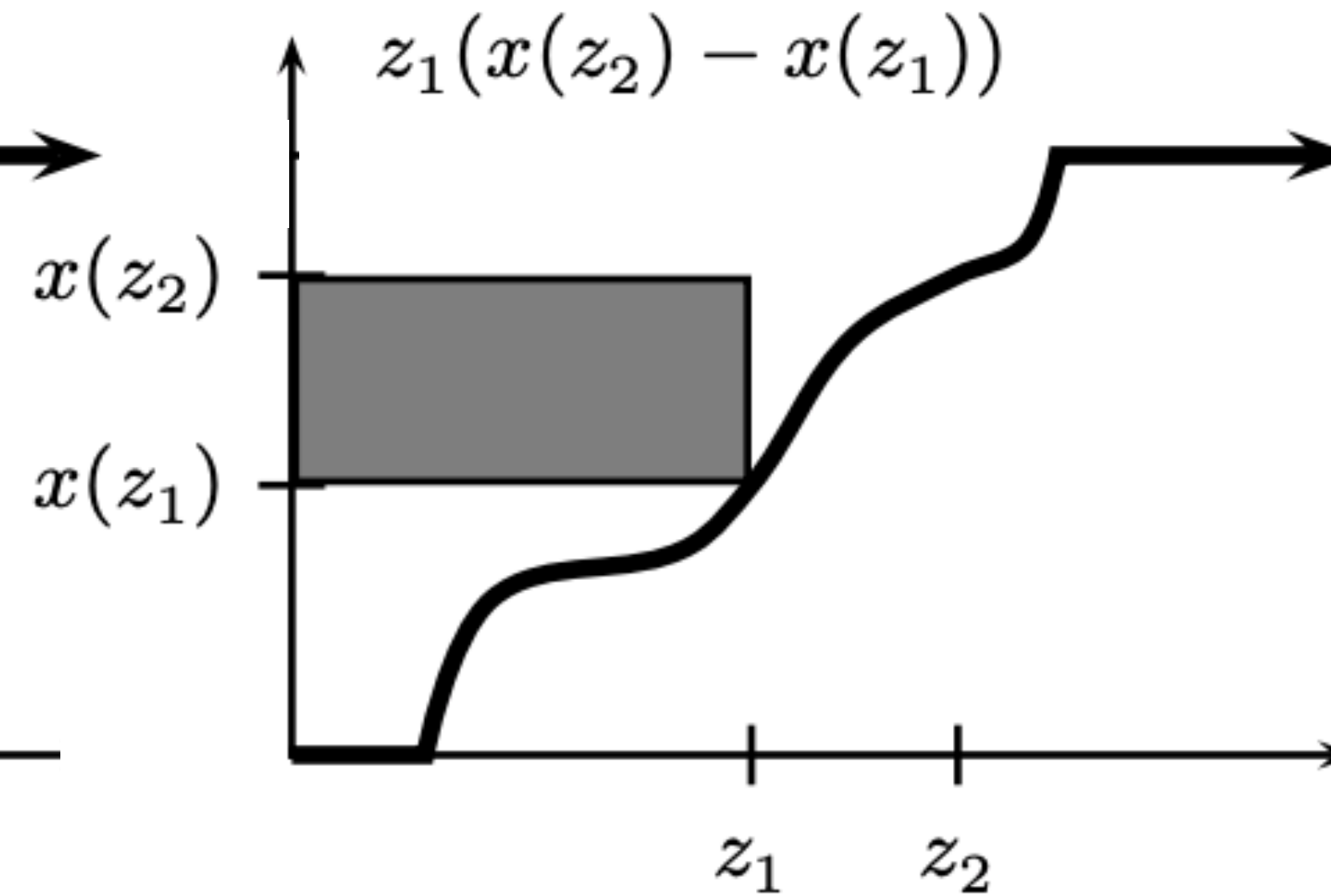


Myerson's Lemma: Proof Part 2

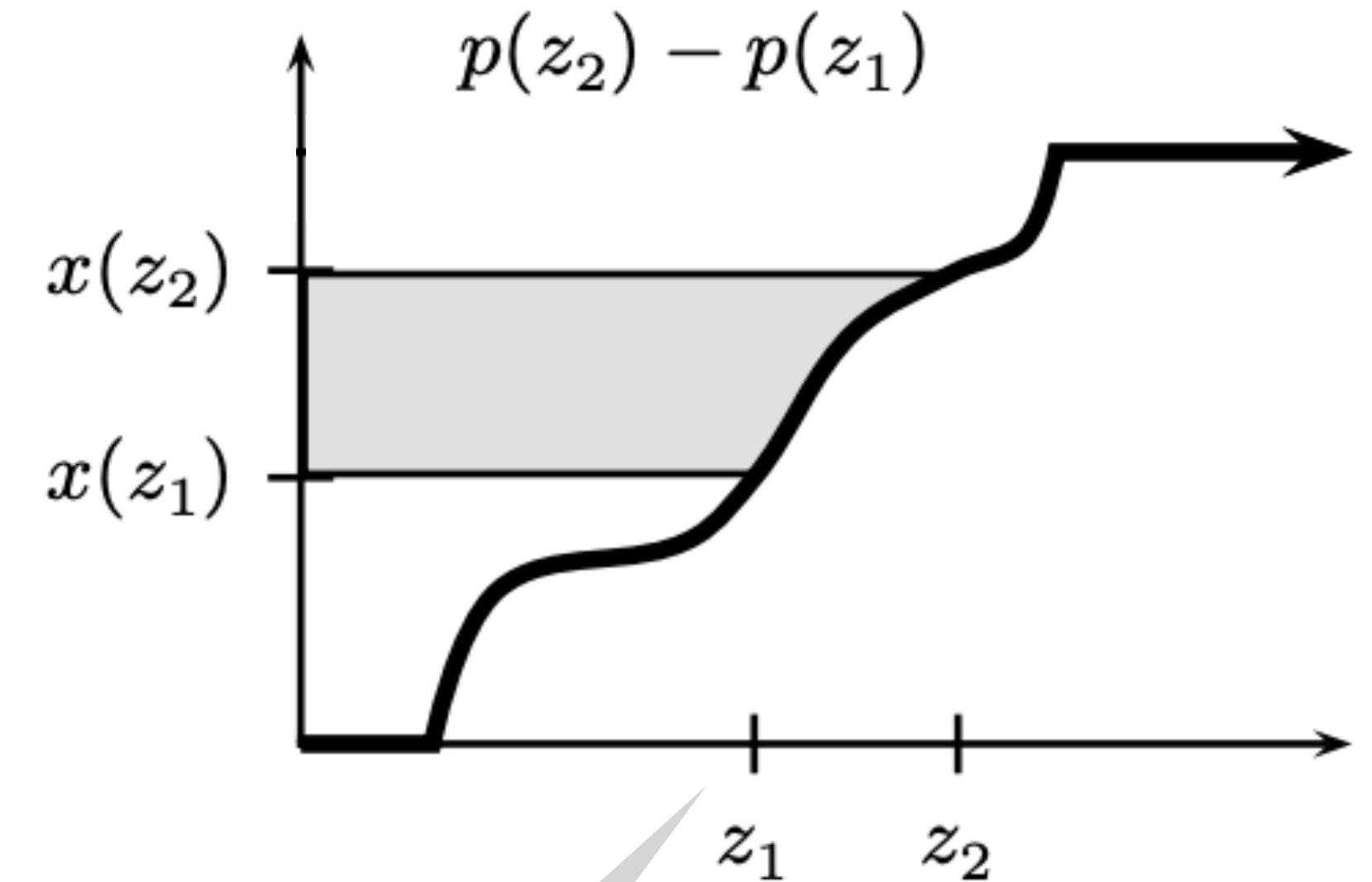
$$z_2 \cdot (x(z_2) - x(z_1)) \geq p(z_2) - p(z_1) \geq z_1 \cdot (x(z_2) - x(z_1))$$



(a) Payment upper bound



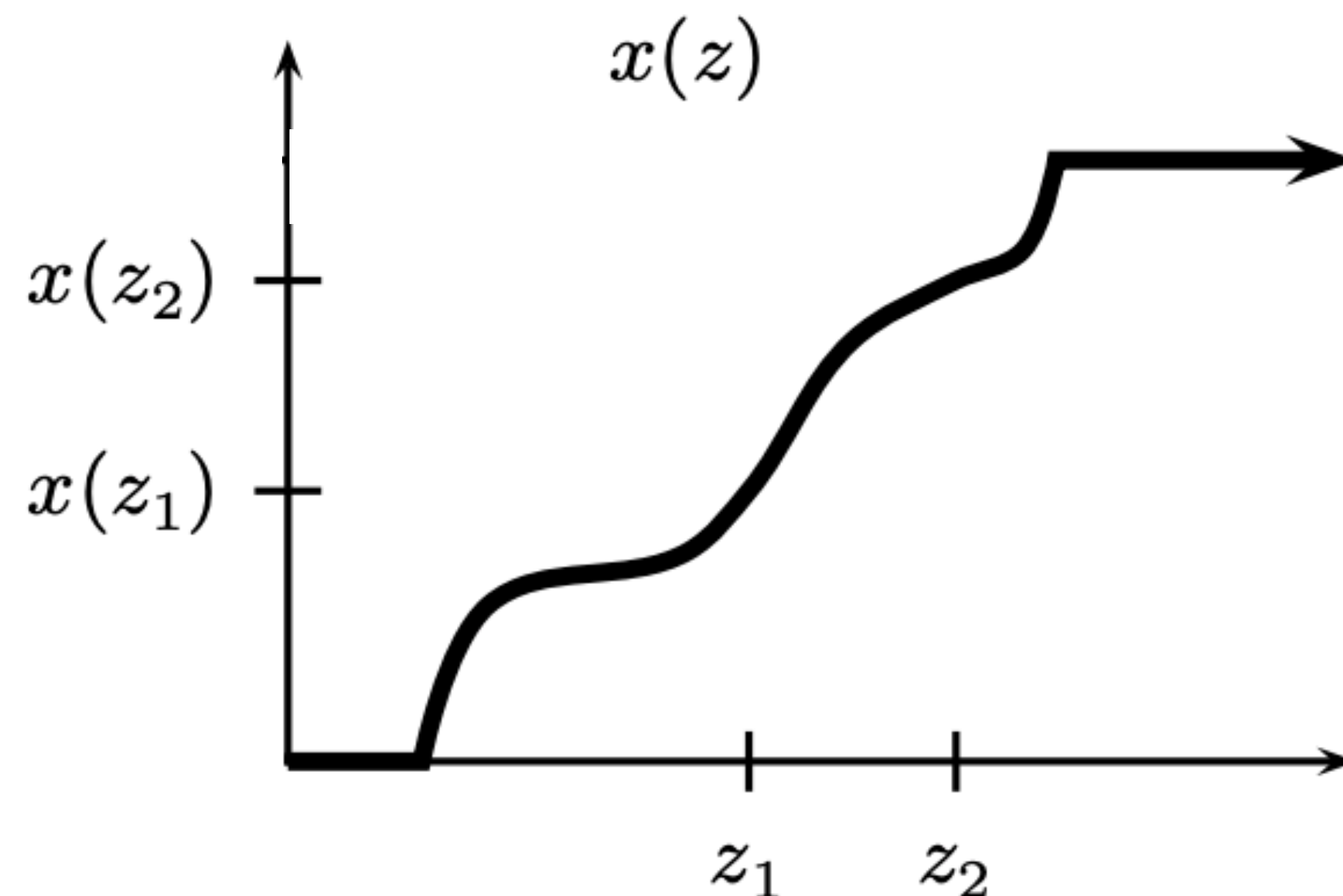
(b) Payment lower bound

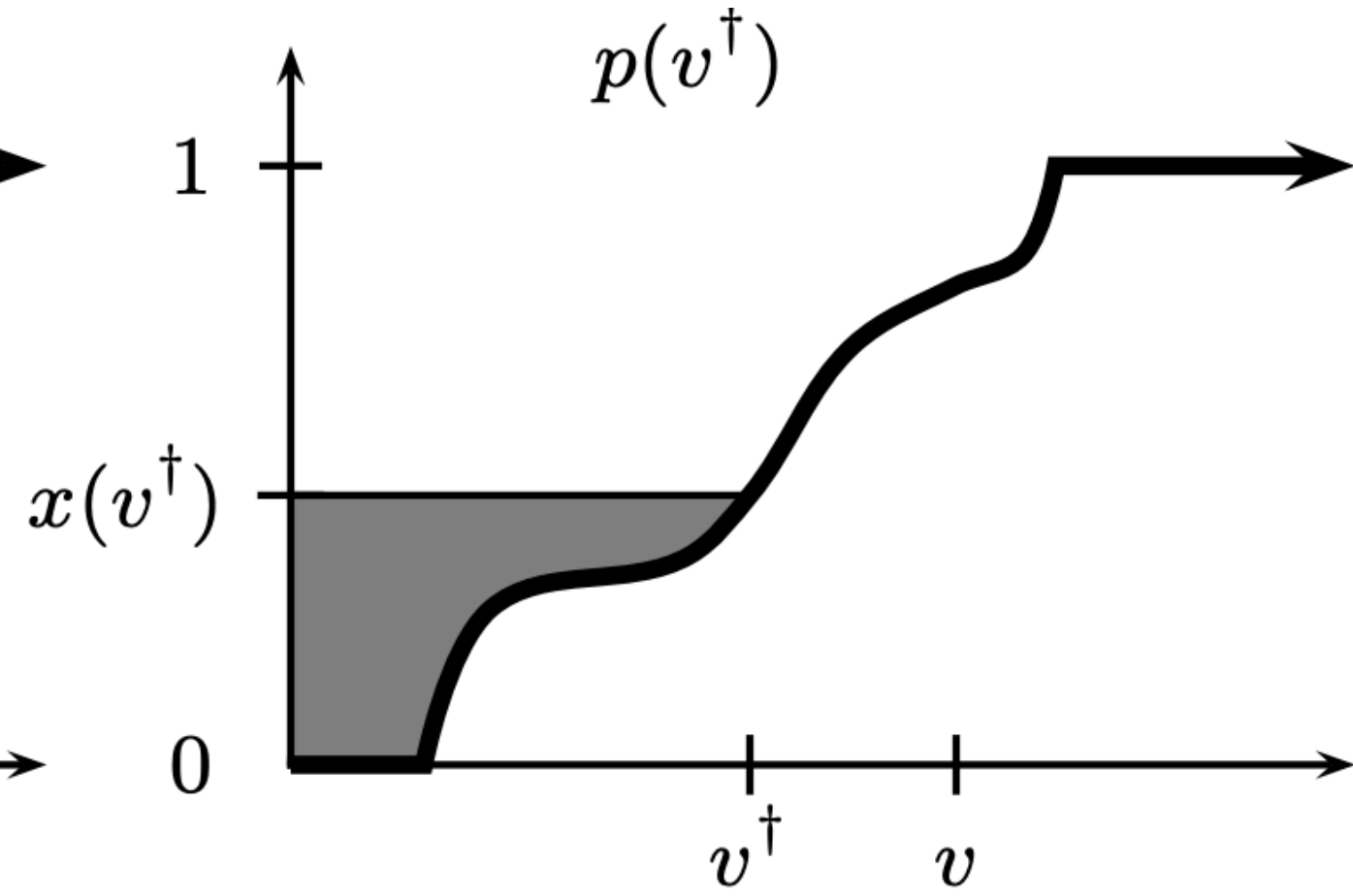
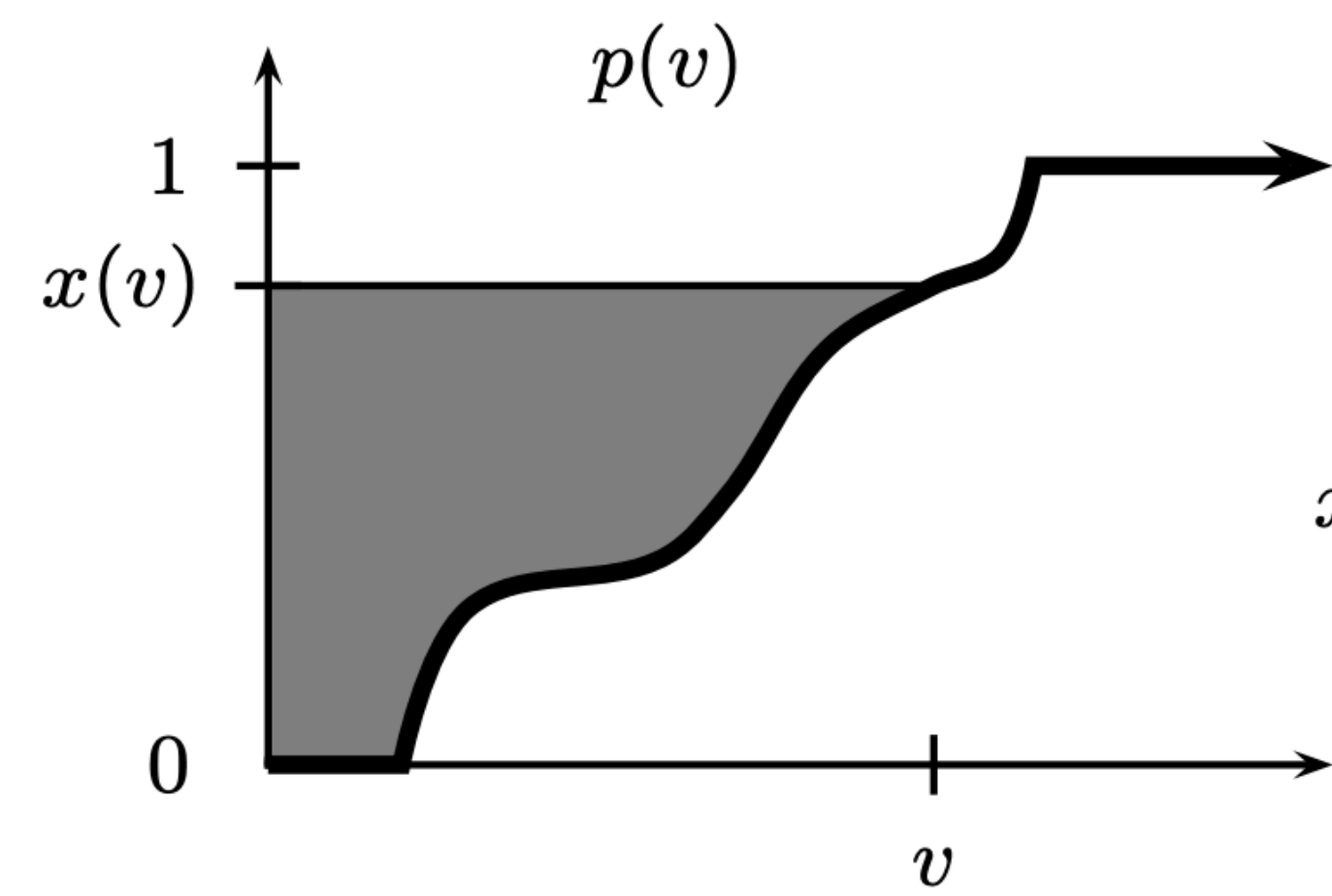
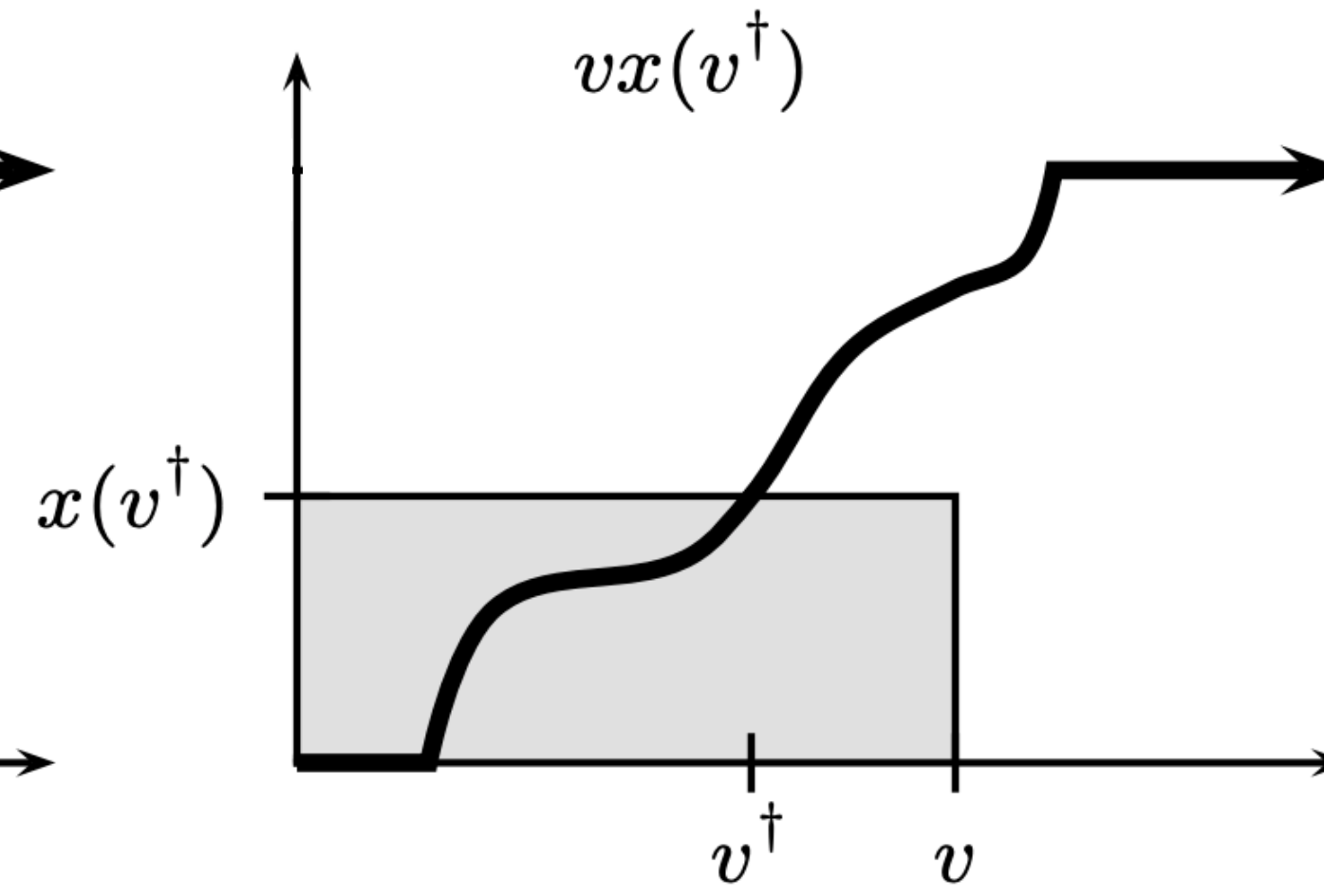
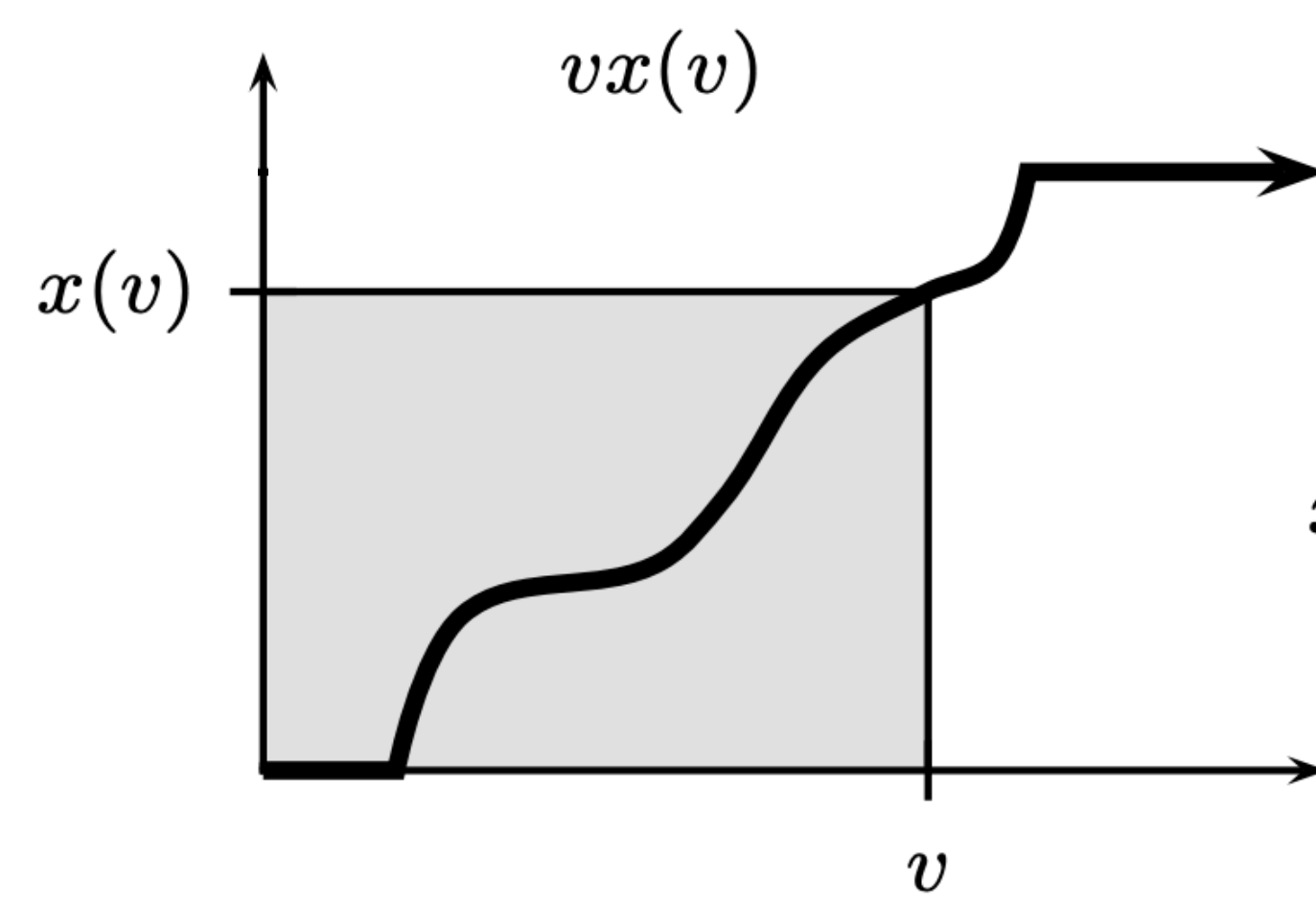


To finish,
set $z_1 = 0$ and $z_2 = v$

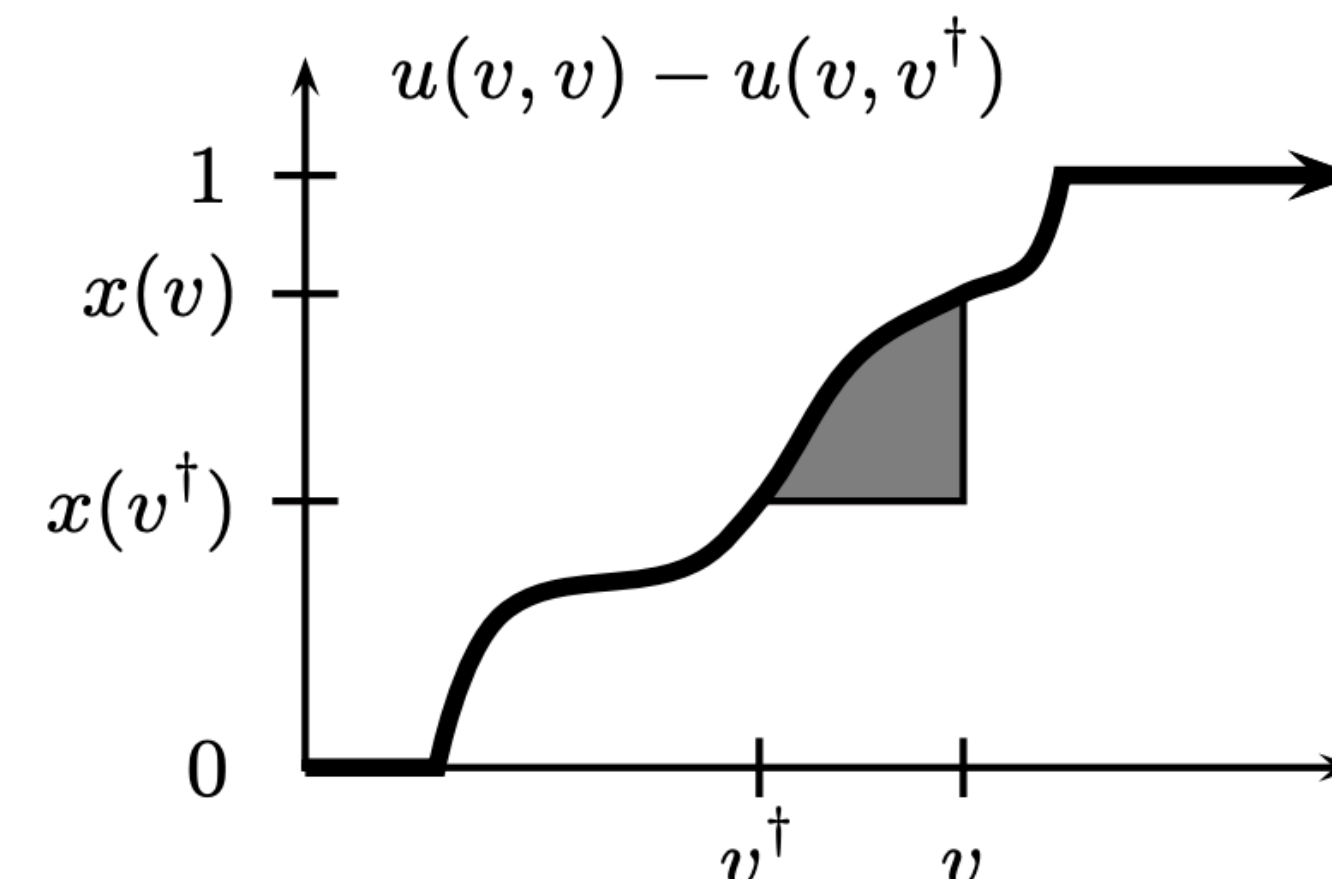
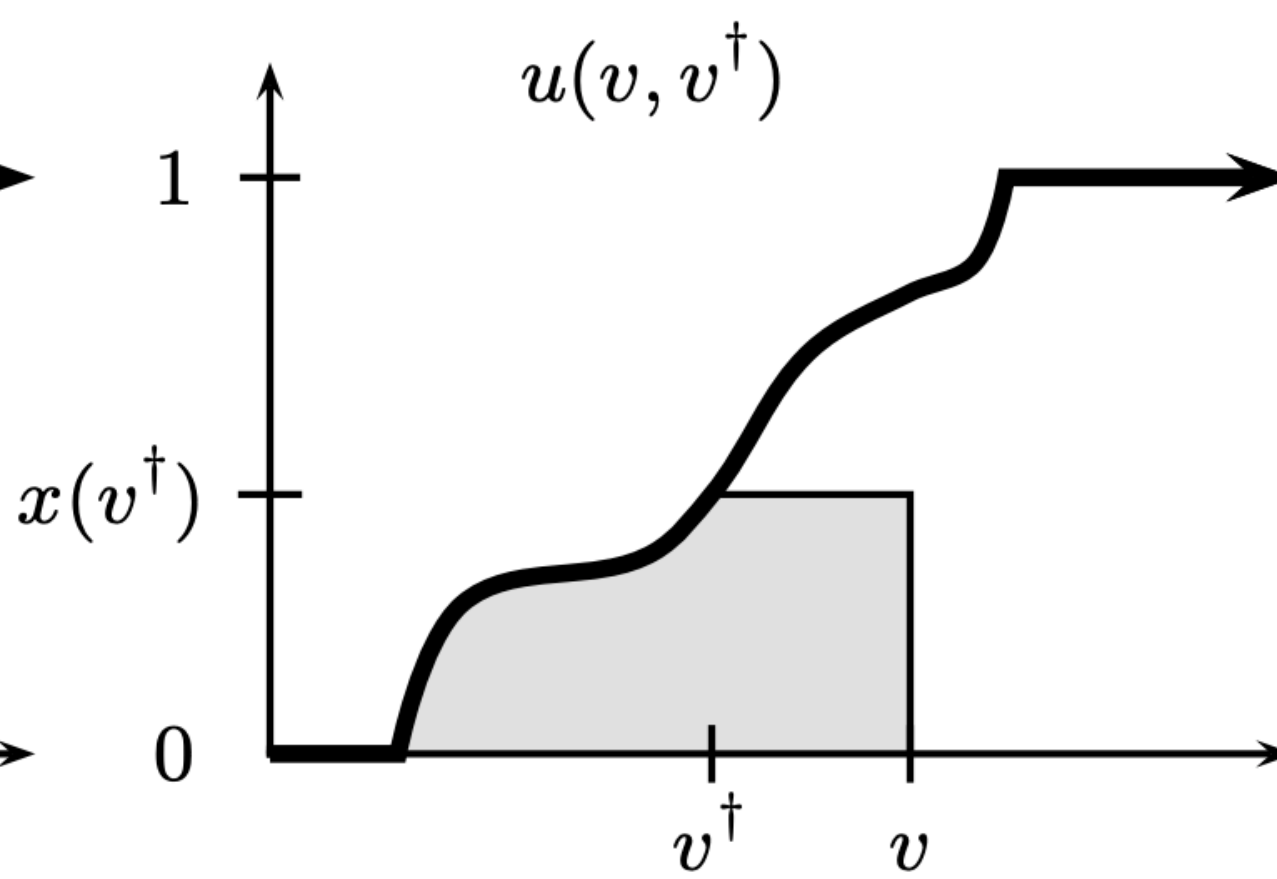
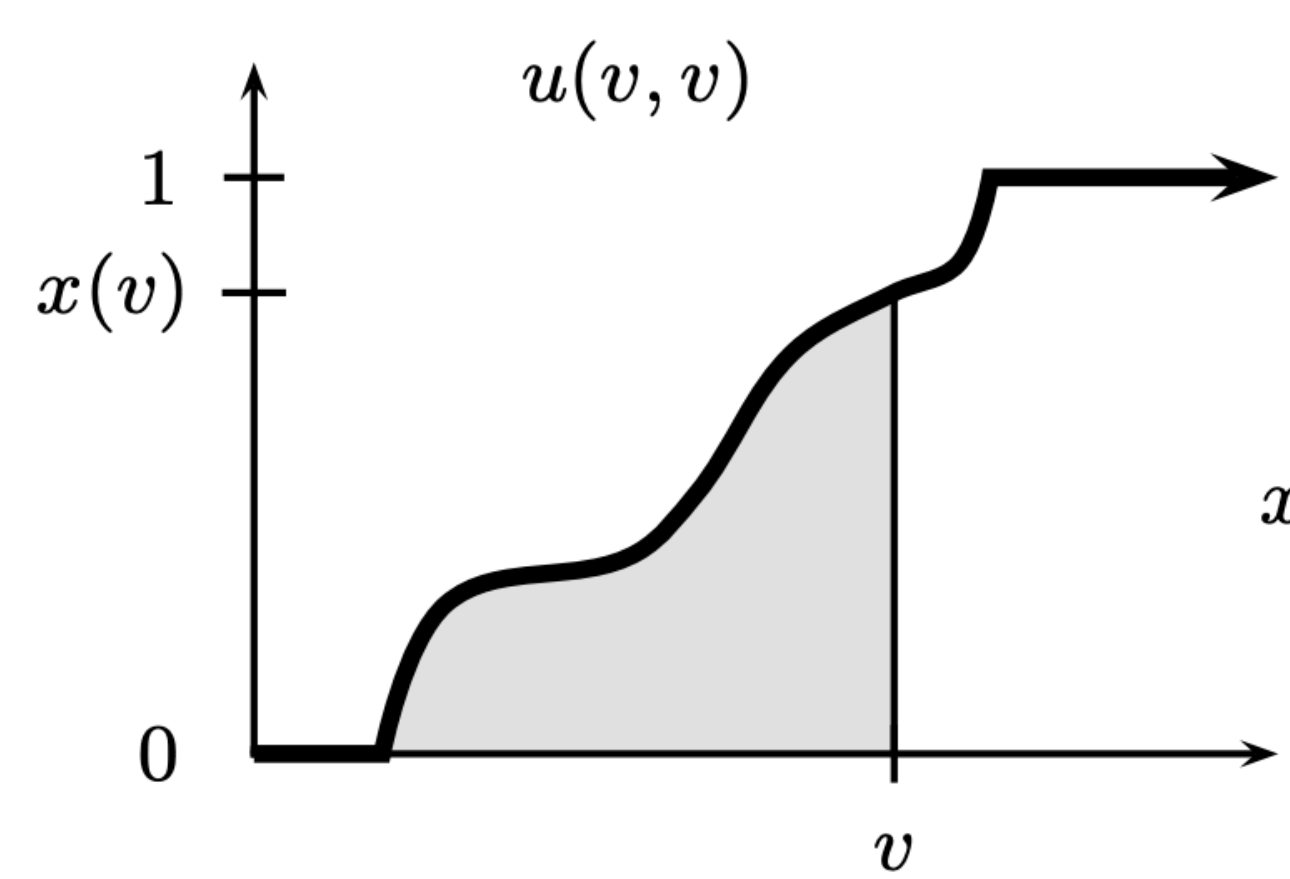
Myerson's Lemma: Proof Part 3

- **Part 3.** If the allocation \mathbf{x} is monotone and the payment rule \mathbf{p} is as given by the expression in the lemma then, (\mathbf{x}, \mathbf{p}) is DSIC
- Suppose Alice's value is $v = z_2$, and she **underbids** $v^\dagger = z_1$
- We will compare utilities $v \cdot x(v) - p(v)$ and $v \cdot x(v^\dagger) - p(v^\dagger)$



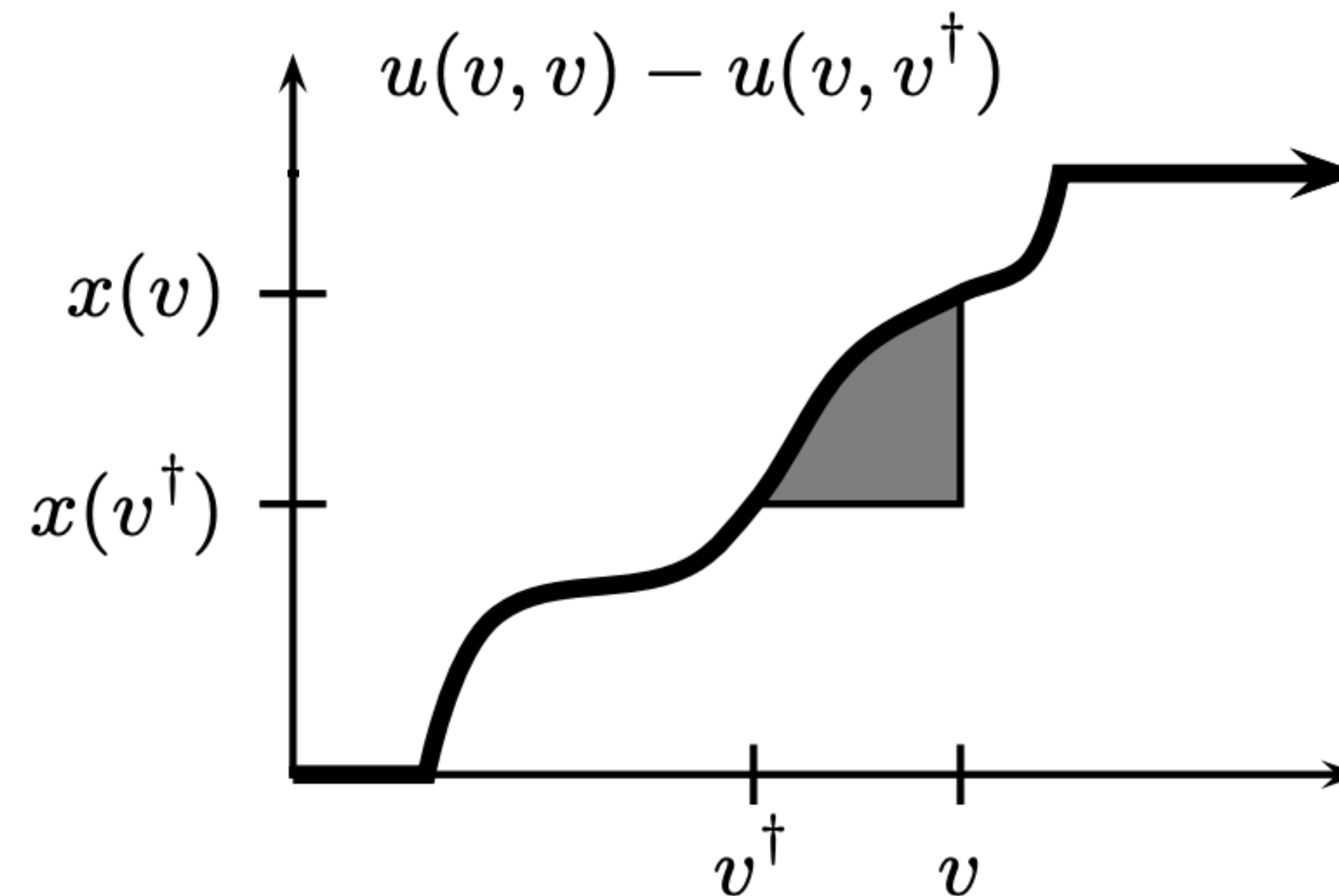


Why is $u(v, v) - u(v, v^\dagger) \geq 0$?



Myerson's Lemma: Proof Part 3

- $u(v, v) - u(v, v^\dagger) \geq 0$ because \mathbf{x} is monotone non-decreasing
 - Since $v > v^\dagger$, we have $x(v) \geq x(v^\dagger)$
- A similar argument proves the other case: where $v^\dagger > v$

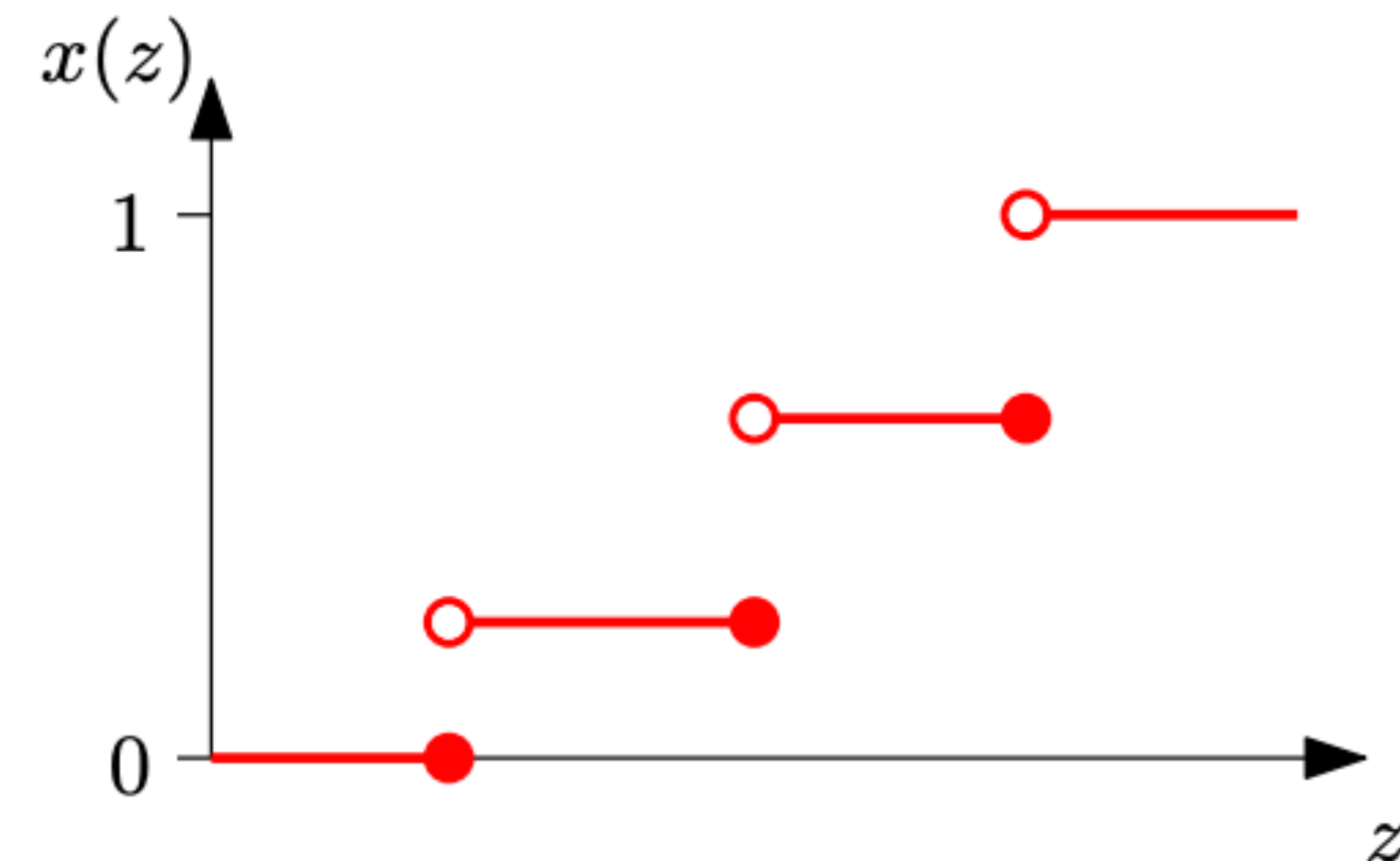


Implications of Myerson's Lemma

- Very powerful characterization
- Our initial design dilemma: can we make some allocation rule \mathbf{x} DSIC by pairing it with an appropriate payment rule?
 - Difficult to reason about
- Myerson's lemma takes this question and turns into one that is more wieldy and operational: checking if \mathbf{x} is monotone
 - Usually not difficult to check
- If an allocation rule is monotone, the lemma says there is **exactly one way to assign payments** to make it DSIC
 - And gives us a formula for the payments!

Sponsored-Search DSIC Payment

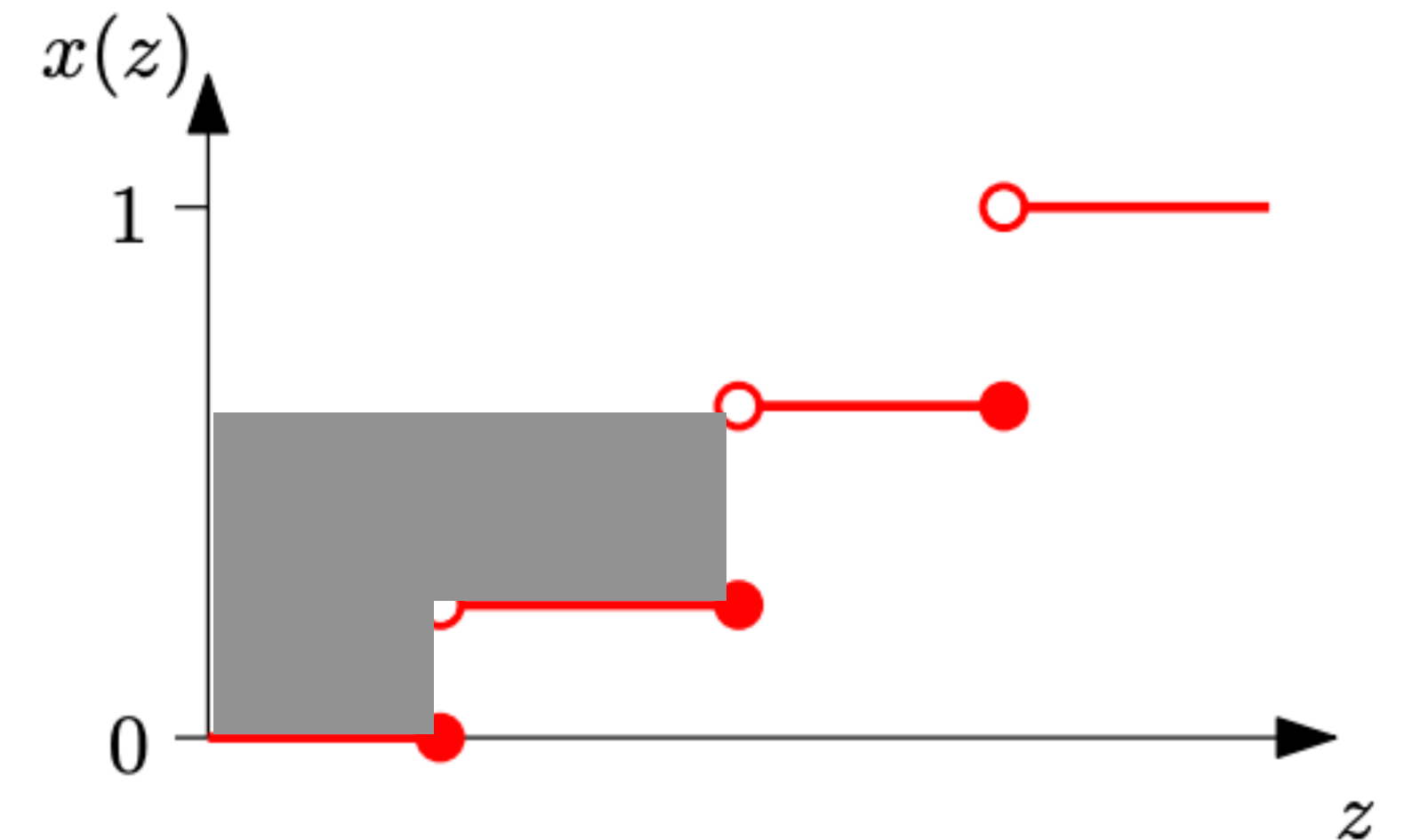
- If you get no slot, you pay zero
- If you get last slot, you pay the “critical” bid that you beat out to get the slot (the bid of the person just below you in sorted order)
- If you get slot $1 \leq j \leq k$, what do you pay?
 - **Exercise:** come with the expression for the payment of bidder who wins slot j using Myerson’s rule?
- We will come back to this!



Sponsored-Search DSIC Payment

- Myerson's payment rule of monotone piece-wise constant allocation
- If there are ℓ points at which the allocation "jumps" before bid z , the payment at bid z

$$p_i(z) = \sum_{j=1}^{\ell} z_j \cdot [\text{jump in } x_i \text{ at } z_j]$$



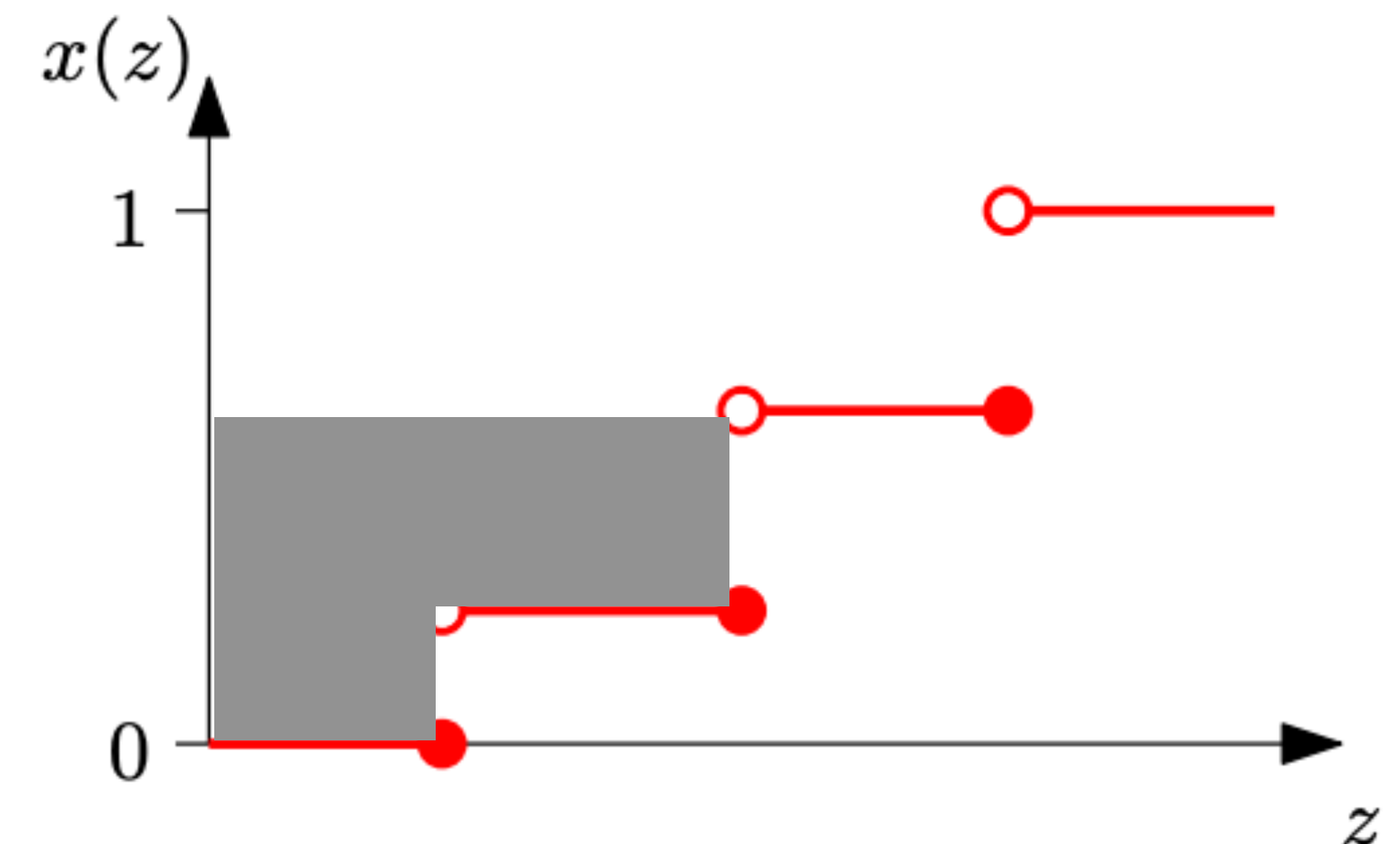
Sponsored-Search DSIC Payment

- Using Myerson's lemma, the i th highest bidder (who wins slot i) should pay:

$$p_i(\mathbf{b}) = \sum_{j=i}^k b_{j+1} \cdot (\alpha_j - \alpha_{j+1}) \text{ where } \alpha_{k+1} = 0$$

- The “**per click payment**” of bidder i who is in slot i is $\sum_{j=i}^k b_{j+1} \cdot \frac{\alpha_j - \alpha_{j+1}}{\alpha_i}$

- Payments have a nice interpretation:
 - If you win, you pay a suitable convex combination of lower bids!



Surplus Maximization & Externality

- Suppose our allocation rule is to maximize surplus, that is,

$$\mathbf{x}(\mathbf{b}) = \operatorname{argmax}_{x_1, \dots, x_n \in X} \sum_{i=1}^n b_i x_i$$

- Means pick x_1, \dots, x_n such that they are feasible (in X) **and** they maximize the sum $\sum_i b_i x_i$ for a given bid vector **b**
- In HW: show that this rule is monotone even for non-zero allocations
 - We can always find DSIC mechanisms for the objective of surplus maximization using Myerson's lemma
- When our goal is to allocate to maximize surplus, and we restrict ourselves to **0-1** allocations single-parameter environments, we can give an alternate form the Myerson's payment

Externality and Myerson's Payments

- An agent's **externality** is the change in social surplus excluding the agent, resulting from the agent's participation in the auction

$$\operatorname{argmax}_{(x_i=0, \mathbf{x}_{-i}) \in X} \sum_{j \neq i} x_j b_j \quad - \quad \operatorname{argmax}_{(x_i=1, \mathbf{x}_{-i}) \in X} \sum_{j \neq i} x_j b_j$$

Maximum possible surplus when i is **absent** (or $x_i = 0$)

Maximum possible surplus (**by other winners**) when i is **present** (and $x_i = 1$)

Externality and Myerson's Payments

- Myerson's payment for i in 0-1 allocations:

critical bid $b_i^*(\mathbf{b}_{-i})$, = agent i 's externality

- An agent must pay for the surplus loss it inflicts on others
- You will prove this in HW 3

Is this always non-negative?

$$\operatorname{argmax}_{(x_i=0, \mathbf{x}_{-i}) \in X} \sum_{j \neq i} x_j b_j \quad - \quad \operatorname{argmax}_{(x_i=1, \mathbf{x}_{-i}) \in X} \sum_{j \neq i} x_j b_j$$

Maximum surplus feasible
when i is absent (or $x_i = 0$)

Maximum surplus feasible (by others)
when i is present (and $x_i = 1$)

Payment computation has been reduced
to computing surplus maximization

Question. Are sponsored-search auctions in real life based on our (Myerson's) theory?

Generalized Second Price Auctions

- By “historical accident,” the sponsored search auctions in real life (called generalized-second price auction or GSP) are not DSIC
- In GSP, the **allocation rule is the same**
 - Allocate slots to highest bidders
- Payment rule: a bidder wins slot i pays the per-click bid of the winner of slot $i - 1$ or 0 if $i = k$ (rather than a convex combination of lower bids)
 - Some say Google incorrectly implemented Myerson’s lemma
 - Most likely reason is that the payment rule of GSP is much easier to explain to advertisers and share-holders
- Which one is better for revenue?
 - We’ll explore this question next week