

भारतीय प्रौद्योगिकी संस्थान मुंबई

Indian Institute of Technology Bombay

CS 6001: Game Theory and Algorithmic Mechanism Design

Week 7 – Mechanism Design and Arrow's Impossibility Theorem

Swaprava Nath

September, 2026

ज्ञानम् परमम् ध्येयम्

Knowledge is the supreme goal

Week 7 Roadmap

Lecture	Chunks (20–25 min each)	Quizzes
Lecture 13	1. Mechanism Design as Reverse Game Theory 2. Social Choice Functions and Examples 3. Mechanisms, DSIC, and the Revelation Principle 4. Bayesian Implementation and BIC	Quiz 1 after Chunk 2 Quiz 2 after Chunk 4
Lecture 14	1. Aggregating Preferences: ASWFs and Pareto 2. Independence of Irrelevant Alternatives 3. Arrow's Theorem and Decisive Sets 4. Proof: Field Expansion and Group Contraction	Quiz 1 after Chunk 2 Quiz 2 after Chunk 4

Contents

- Lecture 13: Mechanism Design
- Lecture 14: Arrow's Impossibility Theorem

Chunk 1

Mechanism Design as Inverse Game Theory

(≈ 20–25 minutes)

From Predicting Games to Designing Them

So far, we were handed the rules of a game and asked to predict the outcome.

From Predicting Games to Designing Them



So far, we were handed the rules of a game and asked to predict the outcome.

Mechanism design flips this: we start with a goal and design rules so that self-interested play reaches it. This is why it is called **inverse game theory**.

From Predicting Games to Designing Them

So far, we were handed the rules of a game and asked to predict the outcome.

Mechanism design flips this: we start with a goal and design rules so that self-interested play reaches it. This is why it is called **inverse game theory**.

Where this shows up

A ride-hailing platform wants trips matched to the drivers who value them most. A recommendation engine wants advertisers bidding honestly on ad slots. A blockchain wants validators reporting the true state of the ledger. In each case, the designer picks the rules; the participants play them.

Why Can’t the Designer Just Decide?

Question

Why can’t the designer just impose the outcome directly?

Why Can't the Designer Just Decide?

Question

Why can't the designer just impose the outcome directly?

Answer

The information needed (values, costs, preferences) is private to the agents. The designer must build a game that gets this information revealed, honestly.

The Formal Framework

Every mechanism design problem has the same skeleton.

- $N = \{1, \ldots, n\}$: the players

The Formal Framework

Every mechanism design problem has the same skeleton.

- $N = \{1, \ldots, n\}$: the players
- X: the set of possible **outcomes** – who wins the auction, which candidate is elected, which route a rider is matched to

The Formal Framework

Every mechanism design problem has the same skeleton.

- $N = \{1, \ldots, n\}$: the players
- X: the set of possible **outcomes** – who wins the auction, which candidate is elected, which route a rider is matched to
- $\theta_i \in \Theta_i$: player i's **type** – everything private that shapes their preference, e.g., a driver's true cost of taking a trip, or a bidder's true value for an ad slot

The Formal Framework

Every mechanism design problem has the same skeleton.

- $N = \{1, \ldots, n\}$: the players
- X: the set of possible **outcomes** – who wins the auction, which candidate is elected, which route a rider is matched to
- $\theta_i \in \Theta_i$: player i's **type** – everything private that shapes their preference, e.g., a driver's true cost of taking a trip, or a bidder's true value for an ad slot
- $\theta = (\theta_1, \ldots, \theta_n) \in \Theta = \Theta_1 \times \cdots \times \Theta_n$: a full type profile

The Formal Framework

Every mechanism design problem has the same skeleton.

- $N = \{1, \ldots, n\}$: the players
- X: the set of possible **outcomes** – who wins the auction, which candidate is elected, which route a rider is matched to
- $\theta_i \in \Theta_i$: player i's **type** – everything private that shapes their preference, e.g., a driver's true cost of taking a trip, or a bidder's true value for an ad slot
- $\theta = (\theta_1, \ldots, \theta_n) \in \Theta = \Theta_1 \times \cdots \times \Theta_n$: a full type profile

The core difficulty

If the designer already knew every θ_i, there would be nothing to design – just compute the best outcome and impose it. Because types are private, the designer needs a game that makes truth-telling worthwhile.

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much
- **Cardinal**: θ_i pins down a utility function u_i over outcomes

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much
- **Cardinal**: θ_i pins down a utility function u_i over outcomes
  - **Private value model**: $u_i : X \times \Theta_i \to \mathbb{R}$ – a bidder's enjoyment of a painting doesn't depend on anyone else's valuation

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much
- **Cardinal**: θ_i pins down a utility function u_i over outcomes
  - **Private value model**: $u_i : X \times \Theta_i \to \mathbb{R}$ – a bidder's enjoyment of a painting doesn't depend on anyone else's valuation
  - **Interdependent value model**: $u_i : X \times \Theta \to \mathbb{R}$ – in a spectrum auction, each bidder's own profit depends partly on *everyone's* signal about how congested the band will be

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much
- **Cardinal**: θ_i pins down a utility function u_i over outcomes
  - **Private value model**: $u_i : X \times \Theta_i \to \mathbb{R}$ – a bidder's enjoyment of a painting doesn't depend on anyone else's valuation
  - **Interdependent value model**: $u_i : X \times \Theta \to \mathbb{R}$ – in a spectrum auction, each bidder's own profit depends partly on *everyone's* signal about how congested the band will be

Question

An ML company auctions off GPU-cluster time to internal teams. Is each team's value for a training slot more naturally private or interdependent?

How Types Become Preferences

- **Ordinal**: θ_i is just a ranking over outcomes, e.g., a driver ranks trip requests $a \succ_i b \succ_i c$ with no sense of by how much
- **Cardinal**: θ_i pins down a utility function u_i over outcomes
  - **Private value model**: $u_i : X \times \Theta_i \to \mathbb{R}$ – a bidder's enjoyment of a painting doesn't depend on anyone else's valuation
  - **Interdependent value model**: $u_i : X \times \Theta \to \mathbb{R}$ – in a spectrum auction, each bidder's own profit depends partly on *everyone's* signal about how congested the band will be

Question

An ML company auctions off GPU-cluster time to internal teams. Is each team's value for a training slot more naturally private or interdependent?

Answer

Mostly private – a team's value depends on its own project deadline, not on other teams' training runs. (Interdependence would creep in only if teams shared model checkpoints.)

Chunk 2

Two Running Examples, and Social Choice Functions

(≈ 20–25 minutes)

Example 1: Voting

A funding agency must pick one proposal from a shortlist to fund next.

- **Outcomes** X: the set of candidates, e.g., $X = \{\text{Alice, Bob, Carol}\}$

Example 1: Voting

A funding agency must pick one proposal from a shortlist to fund next.

- **Outcomes** X: the set of candidates, e.g., $X = \{\text{Alice, Bob, Carol}\}$
- **Type** θ_i: a voter's ranking over candidates, e.g., $\theta_i = (\text{Alice} \succ \text{Bob} \succ \text{Carol})$

Example 1: Voting

A funding agency must pick one proposal from a shortlist to fund next.

- **Outcomes** X: the set of candidates, e.g., $X = \{\text{Alice, Bob, Carol}\}$
- **Type** θ_i: a voter's ranking over candidates, e.g., $\theta_i = (\text{Alice} \succ \text{Bob} \succ \text{Carol})$

Same structure as electing a class representative, or a token-holder vote on a protocol upgrade – only the label on X changes.

Example 2: Single-Object Allocation

A cloud provider auctions a single high-demand GPU node to n competing ML teams.

- Outcome $x = (a, p) \in X$: $a = (a_1, \ldots, a_n)$, $a_i \in \{0, 1\}$, $\sum_i a_i \leqslant 1$ is who gets the node; $p = (p_1, \ldots, p_n)$ is what each team pays

Example 2: Single-Object Allocation

A cloud provider auctions a single high-demand GPU node to n competing ML teams.

- Outcome $x = (a, p) \in X$: $a = (a_1, \ldots, a_n)$, $a_i \in \{0, 1\}$, $\sum_i a_i \leqslant 1$ is who gets the node; $p = (p_1, \ldots, p_n)$ is what each team pays
- Type θ_i: team i's private value for the node – their maximum willingness to pay

Example 2: Single-Object Allocation

A cloud provider auctions a single high-demand GPU node to n competing ML teams.

- Outcome $x = (a, p) \in X$: $a = (a_1, \ldots, a_n)$, $a_i \in \{0, 1\}$, $\sum_i a_i \leqslant 1$ is who gets the node; $p = (p_1, \ldots, p_n)$ is what each team pays
- Type θ_i: team i's private value for the node – their maximum willingness to pay
- Utility: $u_i(x, \theta_i) = a_i \theta_i - p_i$

Example 2: Single-Object Allocation

A cloud provider auctions a single high-demand GPU node to n competing ML teams.

- Outcome $x = (a, p) \in X$: $a = (a_1, \ldots, a_n)$, $a_i \in \{0, 1\}$, $\sum_i a_i \leqslant 1$ is who gets the node; $p = (p_1, \ldots, p_n)$ is what each team pays
- Type θ_i: team i's private value for the node – their maximum willingness to pay
- Utility: $u_i(x, \theta_i) = a_i \theta_i - p_i$

Reading the utility

Win the node ($a_i = 1$): net value is $\theta_i - p_i$. Lose ($a_i = 0$): utility is 0, since a losing team pays nothing.

Social Choice Function

The designer's goal, stated formally, is a **Social Choice Function (SCF)**.

Definition (Social Choice Function)

A social choice function is a map $f : \Theta \to X$ that specifies the ideal outcome for every possible profile of agent types.

Social Choice Function

The designer's goal, stated formally, is a **Social Choice Function (SCF)**.

Definition (Social Choice Function)

A social choice function is a map $f : \Theta \to X$ that specifies the ideal outcome for every possible profile of agent types.

"If I knew the true types were θ, I would want outcome $f(\theta)$."

Social Choice Function: Examples



Examples

- **Voting**: pick the candidate who would beat every other candidate head-to-head, if one exists

Social Choice Function: Examples



Examples

- **Voting**: pick the candidate who would beat every other candidate head-to-head, if one exists
- **Public project**: pick the project maximizing total value, $f(\theta) \in \arg\max_{a \in X} \sum_{i \in N} \theta_i(a)$ – e.g., which of several open-source proposals a foundation should fund

Social Choice Function: Examples



Examples

- **Voting**: pick the candidate who would beat every other candidate head-to-head, if one exists
- **Public project**: pick the project maximizing total value, $f(\theta) \in \arg\max_{a \in X} \sum_{i \in N} \theta_i(a)$ – e.g., which of several open-source proposals a foundation should fund

Question

We know f. How do we get an equilibrium of some game to land exactly on $f(\theta)$, without knowing θ ourselves?

In-Class Quiz 1

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

Chunk 3

Mechanisms, DSIC, and the Revelation Principle

(≈ 20–25 minutes)

Mechanisms

Definition (Mechanism)

A mechanism is a pair $\langle (M_1, \ldots, M_n), g \rangle$: a message space M_i for each agent, and an outcome function $g : M_1 \times \cdots \times M_n \to X$.

Mechanisms

Definition (Mechanism)

A mechanism is a pair $\langle (M_1, \ldots, M_n), g \rangle$: a message space M_i for each agent, and an outcome function $g : M_1 \times \cdots \times M_n \to X$.

Agent i, knowing only θ_i, picks a message $m_i \in M_i$ to maximize $u_i(g(m_1, \ldots, m_n), \theta_i)$.

Mechanisms

Definition (Mechanism)

A mechanism is a pair $\langle (M_1, \ldots, M_n), g \rangle$: a message space M_i for each agent, and an outcome function $g : M_1 \times \cdots \times M_n \to X$.

Agent i, knowing only θ_i, picks a message $m_i \in M_i$ to maximize $u_i(g(m_1, \ldots, m_n), \theta_i)$.

Indirect mechanism

M_i can be anything: a bidding ladder in an ascending auction, a sequence of GitHub pull requests in a governance vote, a multi-round negotiation. Complex, but very general.

Mechanisms

Definition (Mechanism)

A mechanism is a pair $\langle (M_1, \ldots, M_n), g \rangle$: a message space M_i for each agent, and an outcome function $g : M_1 \times \cdots \times M_n \to X$.

Agent i, knowing only θ_i, picks a message $m_i \in M_i$ to maximize $u_i(g(m_1, \ldots, m_n), \theta_i)$.

Indirect mechanism

M_i can be anything: a bidding ladder in an ascending auction, a sequence of GitHub pull requests in a governance vote, a multi-round negotiation. Complex, but very general.

Definition (Direct mechanism)

A direct mechanism sets $M_i = \Theta_i$ for every i, and $g = f$: agents are simply asked to report their type, and the SCF computes the outcome.

Weakly Dominant

Definition

In a mechanism $\langle M_1, M_2, \ldots, M_n, g \rangle$, a message m_i is **weakly dominant** for player i at θ_i if

$$u_i(g(m_i, \tilde{m}_{-i}), \theta_i) \geqslant u_i(g(m_i', \tilde{m}_{-i}), \theta_i), \forall \tilde{m}_{-i}, \forall m_i'$$

All subsequent definitions assume cardinal preferences, however they can be replaced with ordinal, e.g., the above one could be defined as

$$g(m_i, \tilde{m}_{-i}) \; \theta_i \; g(m_i', \tilde{m}_{-i}), \forall \tilde{m}_{-i}, \forall m_i'$$

Dominant Strategy Implementable (DSI)



Definition

An SCF $f : \Theta \to X$ is **implemented in dominant strategies** by $\langle M_1, M_2, \ldots, M_n, g \rangle$ if

Dominant Strategy Implementable (DSI)

Definition

An SCF $f : \Theta \to X$ is **implemented in dominant strategies** by $\langle M_1, M_2, \ldots, M_n, g \rangle$ if

- $\exists$ message mappings $s_i : \Theta_i \to M_i$, s.t, $s_i(\theta_i)$ is a dominant strategy for agent i at θ_i, $\forall \theta_i \in \Theta_i$, $\forall i \in N$.

Dominant Strategy Implementable (DSI)



Definition

An SCF $f : \Theta \to X$ is **implemented in dominant strategies** by $\langle M_1, M_2, \ldots, M_n, g \rangle$ if

- $\exists$ message mappings $s_i : \Theta_i \to M_i$, s.t, $s_i(\theta_i)$ is a dominant strategy for agent i at θ_i, $\forall \theta_i \in \Theta_i$, $\forall i \in N$.
- $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$, $\forall \theta \in \Theta$

Dominant Strategy Implementable (DSI)

Definition

An SCF $f : \Theta \to X$ is **implemented in dominant strategies** by $\langle M_1, M_2, \ldots, M_n, g \rangle$ if

- $\exists$ message mappings $s_i : \Theta_i \to M_i$, s.t, $s_i(\theta_i)$ is a dominant strategy for agent i at θ_i, $\forall \theta_i \in \Theta_i$, $\forall i \in N$.
- $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$, $\forall \theta \in \Theta$

We call this an indirect implementation, i.e., SCF f is **dominant strategy implementable (DSI)** by $\langle M_1, M_2, \ldots, M_n, g \rangle$.

Dominant Strategy Incentive Compatible (DSIC)



Definition

A direct mechanism $\langle \Theta_1, \Theta_2, \ldots, \Theta_n, f \rangle$ is **dominant strategy incentive compatible (DSIC)** if

$$u_i(f(\theta_i, \tilde{\theta}_{-i}), \theta_i) \geqslant u_i(f(\theta_i', \tilde{\theta}_{-i}), \theta_i), \forall \tilde{\theta}_{-i}, \theta_i', \theta_i, \forall i \in N$$

Dominant Strategy Incentive Compatible (DSIC)

Definition

A direct mechanism $\langle \Theta_1, \Theta_2, \ldots, \Theta_n, f \rangle$ is **dominant strategy incentive compatible (DSIC)** if

$$u_i(f(\theta_i, \tilde{\theta}_{-i}), \theta_i) \geqslant u_i(f(\theta_i', \tilde{\theta}_{-i}), \theta_i), \forall \tilde{\theta}_{-i}, \theta_i', \theta_i, \forall i \in N$$

- To find if an SCF f is dominant strategy implementable, we need to search over all possible indirect mechanisms $\langle M_1, M_2, \ldots, M_n, g \rangle$.

Dominant Strategy Incentive Compatible (DSIC)

Definition

A direct mechanism $\langle \Theta_1, \Theta_2, \ldots, \Theta_n, f \rangle$ is **dominant strategy incentive compatible (DSIC)** if

$$u_i(f(\theta_i, \tilde{\theta}_{-i}), \theta_i) \geqslant u_i(f(\theta_i', \tilde{\theta}_{-i}), \theta_i), \forall \tilde{\theta}_{-i}, \theta_i', \theta_i, \forall i \in N$$

- To find if an SCF f is dominant strategy implementable, we need to search over all possible indirect mechanisms $\langle M_1, M_2, \ldots, M_n, g \rangle$.
- But luckily, there is a result that reduces the search space.

The Revelation Principle

Theorem (Revelation Principle)

An SCF f is dominant strategy implementable (DSI) by some mechanism if and only if its direct mechanism is DSIC.

The Revelation Principle

Theorem (Revelation Principle)

An SCF f is dominant strategy implementable (DSI) by some mechanism if and only if its direct mechanism is DSIC.

Why this is powerful

To check whether a goal f is achievable at all under dominant strategies, we never need to search over exotic indirect mechanisms – English auctions, sealed-bid schemes, multi-round protocols. One test suffices: is "just report your type" itself truthful?

The Revelation Principle

Theorem (Revelation Principle)

An SCF f is dominant strategy implementable (DSI) by some mechanism if and only if its direct mechanism is DSIC.

Why this is powerful

To check whether a goal f is achievable at all under dominant strategies, we never need to search over exotic indirect mechanisms – English auctions, sealed-bid schemes, multi-round protocols. One test suffices: is "just report your type" itself truthful?

Question

If the direct mechanism for f is *not* DSIC, can some cleverer indirect mechanism still implement f in dominant strategies?

The Revelation Principle

Theorem (Revelation Principle)

An SCF f is dominant strategy implementable (DSI) by some mechanism if and only if its direct mechanism is DSIC.

Why this is powerful

To check whether a goal f is achievable at all under dominant strategies, we never need to search over exotic indirect mechanisms – English auctions, sealed-bid schemes, multi-round protocols. One test suffices: is "just report your type" itself truthful?

Question

If the direct mechanism for f is *not* DSIC, can some cleverer indirect mechanism still implement f in dominant strategies?

Answer

No – that is exactly what the theorem rules out.

Proof: Direct Mechanism DSIC $\implies$ f is DSI



($\Leftarrow$) If the direct mechanism is DSIC, truth-telling is already a dominant strategy for every player. So the direct mechanism itself implements f in dominant strategies. Done.

Proof: f is DSI $\implies$ Direct Mechanism DSIC



($\Rightarrow$) Suppose f is DSI via some mechanism $\mathcal{M} = \langle M, g \rangle$: each player i has a dominant strategy $s_i(\theta_i) \in M_i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$.

Proof: f is DSI $\implies$ Direct Mechanism DSIC



($\Rightarrow$) Suppose f is DSI via some mechanism $\mathcal{M} = \langle M, g \rangle$: each player i has a dominant strategy $s_i(\theta_i) \in M_i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$.

Fix player i with true type θ_i, comparing truth (θ_i) against a lie (θ_i') in the *direct* mechanism.

Proof: f is DSI $\implies$ Direct Mechanism DSIC



($\Rightarrow$) Suppose f is DSI via some mechanism $\mathcal{M} = \langle M, g \rangle$: each player i has a dominant strategy $s_i(\theta_i) \in M_i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$.

Fix player i with true type θ_i, comparing truth (θ_i) against a lie (θ_i') in the *direct* mechanism.

Reporting θ_i or θ_i' directly corresponds to player i playing $s_i(\theta_i)$ or $s_i(\theta_i')$ in $\mathcal{M}$, against $s_{-i}(\theta_{-i})$.

Proof: f is DSI $\implies$ Direct Mechanism DSIC



($\Rightarrow$) Suppose f is DSI via some mechanism $\mathcal{M} = \langle M, g \rangle$: each player i has a dominant strategy $s_i(\theta_i) \in M_i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$.

Fix player i with true type θ_i, comparing truth (θ_i) against a lie (θ_i') in the *direct* mechanism.

Reporting θ_i or θ_i' directly corresponds to player i playing $s_i(\theta_i)$ or $s_i(\theta_i')$ in $\mathcal{M}$, against $s_{-i}(\theta_{-i})$.

Since $s_i(\theta_i)$ is dominant in $\mathcal{M}$, it beats $s_i(\theta_i')$ against *any* opponent play, in particular against $s_{-i}(\theta_{-i})$:

$$u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \geqslant u_i(g(s_i(\theta_i'), s_{-i}(\theta_{-i})), \theta_i)$$

Proof: f is DSI $\implies$ Direct Mechanism DSIC

($\Rightarrow$) Suppose f is DSI via some mechanism $\mathcal{M} = \langle M, g \rangle$: each player i has a dominant strategy $s_i(\theta_i) \in M_i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$.

Fix player i with true type θ_i, comparing truth (θ_i) against a lie (θ_i') in the *direct* mechanism.

Reporting θ_i or θ_i' directly corresponds to player i playing $s_i(\theta_i)$ or $s_i(\theta_i')$ in $\mathcal{M}$, against $s_{-i}(\theta_{-i})$.

Since $s_i(\theta_i)$ is dominant in $\mathcal{M}$, it beats $s_i(\theta_i')$ against *any* opponent play, in particular against $s_{-i}(\theta_{-i})$:

$$u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \geqslant u_i(g(s_i(\theta_i'), s_{-i}(\theta_{-i})), \theta_i)$$

$$\implies u_i(f(\theta_i, \theta_{-i})), \theta_i) \geqslant u_i(f(\theta_i', \theta_{-i})), \theta_i)$$

which is precisely the DSIC condition.

Chunk 4

Bayesian Implementation and BIC

(≈ 20–25 minutes)

When Dominant Strategies Are Too Much to Ask

DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

When Dominant Strategies Are Too Much to Ask

DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

Bayesian implementation relaxes this: a strategy only needs to be optimal *on average*, given beliefs about others.

When Dominant Strategies Are Too Much to Ask



DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

Bayesian implementation relaxes this: a strategy only needs to be optimal *on average*, given beliefs about others.

- All players share a common prior P over type profiles θ

When Dominant Strategies Are Too Much to Ask



DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

Bayesian implementation relaxes this: a strategy only needs to be optimal *on average*, given beliefs about others.

- All players share a common prior P over type profiles θ
- Nature draws θ from P; player i privately learns only θ_i

When Dominant Strategies Are Too Much to Ask



DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

Bayesian implementation relaxes this: a strategy only needs to be optimal *on average*, given beliefs about others.

- All players share a common prior P over type profiles θ
- Nature draws θ from P; player i privately learns only θ_i
- Player i forms posterior beliefs about θ_{-i} via Bayes' rule

When Dominant Strategies Are Too Much to Ask

DSIC is the gold standard: truthful no matter what anyone else does. Many desirable SCFs simply cannot achieve it.

Bayesian implementation relaxes this: a strategy only needs to be optimal *on average*, given beliefs about others.

- All players share a common prior P over type profiles θ
- Nature draws θ from P; player i privately learns only θ_i
- Player i forms posterior beliefs about θ_{-i} via Bayes' rule

Where this fits

A recommendation platform doesn't know an advertiser's exact budget, but it knows the *distribution* of advertiser budgets from historical auctions – that distribution is the common prior.

Bayesian Implementable SCF

Definition (Bayesian implementable SCF)

f is implementable in Bayesian strategies if there is a mechanism $\langle M, g \rangle$ and a strategy profile s forming a **Bayesian Nash Equilibrium**, such that

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(m_i', s_{-i}(\theta_{-i})), \theta)\right]$$

for all $m_i' \in M_i, \theta_i, i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$ for every θ.

Bayesian Implementable SCF

Definition (Bayesian implementable SCF)

f is implementable in Bayesian strategies if there is a mechanism $\langle M, g \rangle$ and a strategy profile s forming a **Bayesian Nash Equilibrium**, such that

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(m_i', s_{-i}(\theta_{-i})), \theta)\right]$$

for all $m_i' \in M_i, \theta_i, i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$ for every θ.

Question

Is every DSI mechanism automatically Bayesian implementable?

Bayesian Implementable SCF

Definition (Bayesian implementable SCF)

f is implementable in Bayesian strategies if there is a mechanism $\langle M, g \rangle$ and a strategy profile s forming a **Bayesian Nash Equilibrium**, such that

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(g(m_i', s_{-i}(\theta_{-i})), \theta)\right]$$

for all $m_i' \in M_i, \theta_i, i$, and $g(s_1(\theta_1), \ldots, s_n(\theta_n)) = f(\theta)$ for every θ.

Question

Is every DSI mechanism automatically Bayesian implementable?

Answer

Yes – optimal against *every* opponent action is in particular optimal *on average*.

Bayesian Incentive Compatibility



Definition (BIC)

A direct mechanism $\langle \Theta, f \rangle$ is **Bayesian Incentive Compatible (BIC)** if truth-telling is a Bayesian Nash Equilibrium:

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i, \theta_{-i}), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i', \theta_{-i}), \theta)\right], \ \forall \theta_i, \theta_i', i.$$

Bayesian Incentive Compatibility



Definition (BIC)

A direct mechanism $\langle \Theta, f \rangle$ is **Bayesian Incentive Compatible (BIC)** if truth-telling is a Bayesian Nash Equilibrium:

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i, \theta_{-i}), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i', \theta_{-i}), \theta)\right], \ \forall \theta_i, \theta_i', i.$$

Unlike DSIC (truthful against every possible report from others), BIC only asks truthful *on average*, weighted by belief.

Bayesian Incentive Compatibility

Definition (BIC)

A direct mechanism $\langle \Theta, f \rangle$ is **Bayesian Incentive Compatible (BIC)** if truth-telling is a Bayesian Nash Equilibrium:

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i, \theta_{-i}), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i', \theta_{-i}), \theta)\right], \ \forall \theta_i, \theta_i', i.$$

Unlike DSIC (truthful against every possible report from others), BIC only asks truthful *on average*, weighted by belief.

Theorem (Revelation Principle for Bayesian Implementation)

An SCF f is implementable in a Bayesian equilibrium by some mechanism if and only if its direct mechanism is BIC.

Bayesian Incentive Compatibility

Definition (BIC)

A direct mechanism $\langle \Theta, f \rangle$ is **Bayesian Incentive Compatible (BIC)** if truth-telling is a Bayesian Nash Equilibrium:

$$\mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i, \theta_{-i}), \theta)\right] \geqslant \mathbb{E}_{\theta_{-i}|\theta_i}\left[u_i(f(\theta_i', \theta_{-i}), \theta)\right], \ \forall \theta_i, \theta_i', i.$$

Unlike DSIC (truthful against every possible report from others), BIC only asks truthful *on average*, weighted by belief.

Theorem (Revelation Principle for Bayesian Implementation)

An SCF f is implementable in a Bayesian equilibrium by some mechanism if and only if its direct mechanism is BIC.

Same payoff as before: to check Bayesian implementability, test whether "report your type" is truthful *in expectation*. The proof mirrors the DSI case, with every utility comparison replaced by an expected utility comparison.

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal
- Mechanisms $\langle M, g \rangle$; direct mechanisms ask agents to report θ_i directly

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal
- Mechanisms $\langle M, g \rangle$; direct mechanisms ask agents to report θ_i directly
- DSIC: truthful reporting beats every lie, against every possible report from others

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal
- Mechanisms $\langle M, g \rangle$; direct mechanisms ask agents to report θ_i directly
- DSIC: truthful reporting beats every lie, against every possible report from others
- Revelation Principle: f is DSI $\iff$ its direct mechanism is DSIC – search only direct mechanisms

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal
- Mechanisms $\langle M, g \rangle$; direct mechanisms ask agents to report θ_i directly
- DSIC: truthful reporting beats every lie, against every possible report from others
- Revelation Principle: f is DSI $\iff$ its direct mechanism is DSIC – search only direct mechanisms
- Bayesian relaxation: BIC and the Bayesian Revelation Principle, when DSIC is unreachable

Lecture 13: What We Built

- Mechanism design: fix the goal (f), design the game so self-interested play reaches it
- Framework: players, outcomes X, private types θ_i, ordinal or cardinal preferences, private or interdependent values
- Social Choice Function $f : \Theta \to X$ encodes the designer's goal
- Mechanisms $\langle M, g \rangle$; direct mechanisms ask agents to report θ_i directly
- DSIC: truthful reporting beats every lie, against every possible report from others
- Revelation Principle: f is DSI $\iff$ its direct mechanism is DSIC – search only direct mechanisms
- Bayesian relaxation: BIC and the Bayesian Revelation Principle, when DSIC is unreachable
- **Next:** Arrow's impossibility theorem for aggregating preferences

In-Class Quiz 2

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

End of Lecture 13

See you next class.

Contents

- Lecture 13: Mechanism Design
- Lecture 14: Arrow's Impossibility Theorem

Chunk 1

Aggregating Preferences: ASWFs and Pareto

(≈ 20–25 minutes)

Forget Honesty for a Moment

Getting agents to report truthfully is hard. Suppose we solve that problem completely – everyone tells us their real preference.

Forget Honesty for a Moment

Getting agents to report truthfully is hard. Suppose we solve that problem completely – everyone tells us their real preference.

Question

Can we always combine everyone's honest preferences into one clear, fair group ranking?

Forget Honesty for a Moment

Getting agents to report truthfully is hard. Suppose we solve that problem completely – everyone tells us their real preference.

Question

Can we always combine everyone's honest preferences into one clear, fair group ranking?

This is the question of **social choice theory**, pioneered by Kenneth Arrow.

Forget Honesty for a Moment

Getting agents to report truthfully is hard. Suppose we solve that problem completely – everyone tells us their real preference.

Question

Can we always combine everyone's honest preferences into one clear, fair group ranking?

This is the question of **social choice theory**, pioneered by Kenneth Arrow.

A concrete version

A large open-source foundation's steering committee members each rank a set of competing technical proposals. Is there always a **reasonable** way to merge n individual rankings into a single committee ranking?

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Definition (Preference ordering)

R_i is an ordering if it satisfies

1. **Completeness**: for any a, b, either aR_ib or bR_ia (or both)

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Definition (Preference ordering)

R_i is an ordering if it satisfies

1. **Completeness**: for any a, b, either aR_ib or bR_ia (or both)
2. **Transitivity**: aR_ib and bR_ic implies aR_ic

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Definition (Preference ordering)

R_i is an ordering if it satisfies

1. **Completeness**: for any a, b, either aR_ib or bR_ia (or both)
2. **Transitivity**: aR_ib and bR_ic implies aR_ic

Preference Orderings

- Alternatives $A = \{a_1, \ldots, a_m\}$, $|A| \geqslant 3$
- Agents (voters) $N = \{1, \ldots, n\}$
- Agent i's preference R_i: aR_ib means "i finds a at least as good as b"

Definition (Preference ordering)

R_i is an ordering if it satisfies

1. **Completeness**: for any a, b, either aR_ib or bR_ia (or both)
2. **Transitivity**: aR_ib and bR_ic implies aR_ic

Any R_i splits into a strict part P_i (aP_ib: aR_ib but not bR_ia) and an indifference part I_i (aI_ib: both hold).

Arrovian Social Welfare Function

Definition (ASWF)

An Arrovian Social Welfare Function is $F : \mathcal{R}^n \to \mathcal{R}$, taking a profile $(R_1, \ldots, R_n)$ of individual orderings to a single *collective* ordering, itself complete and transitive.

Arrovian Social Welfare Function

Definition (ASWF)

An Arrovian Social Welfare Function is $F : \mathcal{R}^n \to \mathcal{R}$, taking a profile $(R_1, \ldots, R_n)$ of individual orderings to a single *collective* ordering, itself complete and transitive.

We don't just want a winner – we want a full social ranking of every alternative. Write $F(R)$'s strict part as $\hat{F}(R)$.

Arrovian Social Welfare Function

Definition (ASWF)

An Arrovian Social Welfare Function is $F : \mathcal{R}^n \to \mathcal{R}$, taking a profile $(R_1, \ldots, R_n)$ of individual orderings to a single *collective* ordering, itself complete and transitive.

We don't just want a winner – we want a full social ranking of every alternative. Write $F(R)$'s strict part as $\hat{F}(R)$.

Question

A committee could just always output "member 1's ranking, unchanged." Is that an ASWF?

Arrovian Social Welfare Function

Definition (ASWF)

An Arrovian Social Welfare Function is $F : \mathcal{R}^n \to \mathcal{R}$, taking a profile $(R_1, \ldots, R_n)$ of individual orderings to a single *collective* ordering, itself complete and transitive.

We don't just want a winner – we want a full social ranking of every alternative. Write $F(R)$'s strict part as $\hat{F}(R)$.

Question

A committee could just always output "member 1's ranking, unchanged." Is that an ASWF?

Answer

Yes – F maps profiles to orderings; nothing in the definition yet rules out ignoring everyone but one member. We'll return to exactly this case.

Weak and Strong Pareto

Definition (Weak Pareto)

F satisfies Weak Pareto if, whenever *every* agent strictly prefers a to b, the social ranking agrees:

$$[\forall i \in N,\ aP_i b] \implies [a\hat{F}(R)b]$$

Weak and Strong Pareto

Definition (Weak Pareto)

F satisfies Weak Pareto if, whenever *every* agent strictly prefers a to b, the social ranking agrees:

$$[\forall i \in N,\ aP_i b] \implies [a\hat{F}(R)b]$$

Definition (Strong Pareto)

F satisfies Strong Pareto if whenever every agent finds a at least as good as b, and at least one agent strictly prefers a:

$$[(\forall i, aR_i b) \text{ and } (\exists j, aP_j b)] \implies [a\hat{F}(R)b]$$

Weak and Strong Pareto

Definition (Weak Pareto)

F satisfies Weak Pareto if, whenever *every* agent strictly prefers a to b, the social ranking agrees:

$$[\forall i \in N,\ aP_i b] \implies [a\hat{F}(R)b]$$

Definition (Strong Pareto)

F satisfies Strong Pareto if whenever every agent finds a at least as good as b, and at least one agent strictly prefers a:

$$[(\forall i, aR_i b) \text{ and } (\exists j, aP_j b)] \implies [a\hat{F}(R)b]$$

Strong Pareto $\implies$ Weak Pareto (Weak Pareto's hypothesis is a special case of Strong Pareto's).

Weak and Strong Pareto

Definition (Weak Pareto)

F satisfies Weak Pareto if, whenever *every* agent strictly prefers a to b, the social ranking agrees:

$$[\forall i \in N,\ aP_i b] \implies [a\hat{F}(R)b]$$

Definition (Strong Pareto)

F satisfies Strong Pareto if whenever every agent finds a at least as good as b, and at least one agent strictly prefers a:

$$[(\forall i, aR_i b) \text{ and } (\exists j, aP_j b)] \implies [a\hat{F}(R)b]$$

Strong Pareto $\implies$ Weak Pareto (Weak Pareto's hypothesis is a special case of Strong Pareto's).

Reading it

If literally every committee member ranks proposal a over proposal b, it would be perverse for the aggregated ranking to say otherwise.

Chunk 2

Independence of Irrelevant Alternatives

(≈ 20–25 minutes)

Two Profiles That Agree on (a, b)

Four proposals a, b, c, d before the steering committee, in two different rounds (two different sets of individual rankings) R and R':

R			
a	a	c	d
b	c	b	c
c	b	a	b
d	d	d	a

R'			
d	c	b	b
a	a	c	a
b	b	a	d
c	d	d	c

Two Profiles That Agree on (a, b)

Four proposals a, b, c, d before the steering committee, in two different rounds (two different sets of individual rankings) R and R':

R			
a	a	c	d
b	c	b	c
c	b	a	b
d	d	d	a

R'			
d	c	b	b
a	a	c	a
b	b	a	d
c	d	d	c

For every agent, the relative order of a and b is unchanged between R and R' – both profiles **agree** on (a, b), even though c and d move around.

IIA, Formally

Definition (Agreement on a pair)

R_i and R_i' agree on (a, b) if $aP_ib \Leftrightarrow aP_i'b$, and $bP_ia \Leftrightarrow bP_i'a$, and $aI_ib \Leftrightarrow aI_i'b$. Shorthand: $R_i|_{\{a,b\}} = R_i'|_{\{a,b\}}$.

IIA, Formally

Definition (Agreement on a pair)

R_i and R_i' agree on (a, b) if $aP_ib \Leftrightarrow aP_i'b$, and $bP_ia \Leftrightarrow bP_i'a$, and $aI_ib \Leftrightarrow aI_i'b$. Shorthand: $R_i|_{\{a,b\}} = R_i'|_{\{a,b\}}$.

Definition (Independence of Irrelevant Alternatives)

F satisfies IIA if, whenever $R|_{\{a,b\}} = R'|_{\{a,b\}}$ for every agent, then $F(R)|_{\{a,b\}} = F(R')|_{\{a,b\}}$.

IIA, Formally

Definition (Agreement on a pair)

R_i and R_i' agree on (a, b) if $aP_ib \Leftrightarrow aP_i'b$, and $bP_ia \Leftrightarrow bP_i'a$, and $aI_ib \Leftrightarrow aI_i'b$. Shorthand: $R_i|_{\{a,b\}} = R_i'|_{\{a,b\}}$.

Definition (Independence of Irrelevant Alternatives)

F satisfies IIA if, whenever $R|_{\{a,b\}} = R'|_{\{a,b\}}$ for every agent, then $F(R)|_{\{a,b\}} = F(R')|_{\{a,b\}}$.

The social ranking of a versus b should depend only on how agents rank a versus b – not on where the "irrelevant" alternatives c, d happen to sit.

IIA, Formally

Definition (Agreement on a pair)

R_i and R_i' agree on (a, b) if $aP_ib \Leftrightarrow aP_i'b$, and $bP_ia \Leftrightarrow bP_i'a$, and $aI_ib \Leftrightarrow aI_i'b$. Shorthand: $R_i|_{\{a,b\}} = R_i'|_{\{a,b\}}$.

Definition (Independence of Irrelevant Alternatives)

F satisfies IIA if, whenever $R|_{\{a,b\}} = R'|_{\{a,b\}}$ for every agent, then $F(R)|_{\{a,b\}} = F(R')|_{\{a,b\}}$.

The social ranking of a versus b should depend only on how agents rank a versus b – not on where the "irrelevant" alternatives c, d happen to sit.

Why platforms care

A recommender system ranking two songs A and B for a user shouldn't flip their relative order just because an unrelated third song C was added to the catalog. That flip is exactly an IIA violation.

Scoring Rules

A **scoring rule** assigns a score to each position (1st, 2nd, . . .) and sums scores across agents; the aggregate order follows total score.

Scoring Rules

A **scoring rule** assigns a score to each position (1st, 2nd, ...) and sums scores across agents; the aggregate order follows total score.

Try $s_1 = 3, s_2 = 2, s_3 = 1, s_4 = 0$ on our profile R:

- $a : 3+3+1+0 = 7$ $b : 2+1+2+1 = 6$ $c : 1+2+3+2 = 8$ $d : 0+0+0+3 = 3$

Scoring Rules

A **scoring rule** assigns a score to each position (1st, 2nd, ...) and sums scores across agents; the aggregate order follows total score.

Try $s_1 = 3, s_2 = 2, s_3 = 1, s_4 = 0$ on our profile R:

- $a : 3+3+1+0 = 7 \quad b : 2+1+2+1 = 6 \quad c : 1+2+3+2 = 8 \quad d : 0+0+0+3 = 3$

So $F(R) = (c \succ a \succ b \succ d)$.

Scoring Rules

A **scoring rule** assigns a score to each position (1st, 2nd, ...) and sums scores across agents; the aggregate order follows total score.

Try $s_1 = 3, s_2 = 2, s_3 = 1, s_4 = 0$ on our profile R:

- $a : 3+3+1+0 = 7 \quad b : 2+1+2+1 = 6 \quad c : 1+2+3+2 = 8 \quad d : 0+0+0+3 = 3$

So $F(R) = (c \succ a \succ b \succ d)$.

Free property

Any scoring rule automatically satisfies Weak Pareto: if everyone prefers a to b (or ties), a's total score can only be $\geqslant b$'s.

Does Any Scoring Rule Satisfy IIA?

Try **Plurality**: top position scores 1, everything else scores 0.

Does Any Scoring Rule Satisfy IIA?

Try **Plurality**: top position scores 1, everything else scores 0.

$$F_{\text{plu}}(R) = \begin{pmatrix} a \\ c, d \\ b \end{pmatrix} \qquad F_{\text{plu}}(R') = \begin{pmatrix} b \\ c, d \\ a \end{pmatrix}$$

Does Any Scoring Rule Satisfy IIA?

Try **Plurality**: top position scores 1, everything else scores 0.

$$F_{\text{plu}}(R) = \begin{pmatrix} a \\ c, d \\ b \end{pmatrix} \qquad F_{\text{plu}}(R') = \begin{pmatrix} b \\ c, d \\ a \end{pmatrix}$$

Recall R and R' *agree* on (a, b) for every agent. But $F_{\text{plu}}(R)$ ranks a over b, while $F_{\text{plu}}(R')$ ranks b over a.

Does Any Scoring Rule Satisfy IIA?

Try **Plurality**: top position scores 1, everything else scores 0.

$$F_{\text{plu}}(R) = \begin{pmatrix} a \\ c, d \\ b \end{pmatrix} \qquad F_{\text{plu}}(R') = \begin{pmatrix} b \\ c, d \\ a \end{pmatrix}$$

Recall R and R' *agree* on (a, b) for every agent. But $F_{\text{plu}}(R)$ ranks a over b, while $F_{\text{plu}}(R')$ ranks b over a.

IIA is violated. Is there *any* scoring rule that avoids this?

Does Any Scoring Rule Satisfy IIA?

Try **Plurality**: top position scores 1, everything else scores 0.

$$F_{\text{plu}}(R) = \begin{pmatrix} a \\ c, d \\ b \end{pmatrix} \qquad F_{\text{plu}}(R') = \begin{pmatrix} b \\ c, d \\ a \end{pmatrix}$$

Recall R and R' *agree* on (a, b) for every agent. But $F_{\text{plu}}(R)$ ranks a over b, while $F_{\text{plu}}(R')$ ranks b over a.

IIA is violated. Is there *any* scoring rule that avoids this?

Unfortunately, no!

No Scoring Rule Satisfies IIA

Take *any* scoring rule on three alternatives: $s_1 \geqslant s_2 \geqslant s_3$, with $s_1 > s_3$. (If $s_1 = s_3$ every alternative always scores the same, and the rule ranks everything as tied.)

No Scoring Rule Satisfies IIA

Take *any* scoring rule on three alternatives: $s_1 \geqslant s_2 \geqslant s_3$, with $s_1 > s_3$. (If $s_1 = s_3$ every alternative always scores the same, and the rule ranks everything as tied.)

Three voters, two profiles:

$$R = \begin{pmatrix} a & b & b \\ b & c & c \\ c & a & a \end{pmatrix} \qquad R' = \begin{pmatrix} a & b & c \\ c & a & b \\ b & c & a \end{pmatrix}$$

No Scoring Rule Satisfies IIA

Take *any* scoring rule on three alternatives: $s_1 \geqslant s_2 \geqslant s_3$, with $s_1 > s_3$. (If $s_1 = s_3$ every alternative always scores the same, and the rule ranks everything as tied.)

Three voters, two profiles:

$$R = \begin{pmatrix} a & b & b \\ b & c & c \\ c & a & a \end{pmatrix} \qquad R' = \begin{pmatrix} a & b & c \\ c & a & b \\ b & c & a \end{pmatrix}$$

Voter 1 ranks a above b in both. Voters 2 and 3 rank b above a in both. **Only c moves** – so $R|_{\{a,b\}} = R'|_{\{a,b\}}$.

The Scores Flip Anyway

In R:

$$\sigma(a) = s_1 + s_3 + s_3, \qquad \sigma(b) = s_2 + s_1 + s_1$$

$$\sigma(b) - \sigma(a) = (s_1 - s_3) + (s_2 - s_3) \; > \; 0 \quad \Longrightarrow \quad b \text{ is ranked strictly above } a.$$

The Scores Flip Anyway

In R:

$$\sigma(a) = s_1 + s_3 + s_3, \qquad \sigma(b) = s_2 + s_1 + s_1$$

$$\sigma(b) - \sigma(a) = (s_1 - s_3) + (s_2 - s_3) \; > \; 0 \quad \Longrightarrow \quad b \text{ is ranked strictly above } a.$$

In R':

$$\sigma(a) = s_1 + s_2 + s_3, \qquad \sigma(b) = s_3 + s_1 + s_2 \quad \Longrightarrow \quad a \text{ and } b \text{ are tied.}$$

The Scores Flip Anyway

In R:

$$\sigma(a) = s_1 + s_3 + s_3, \qquad \sigma(b) = s_2 + s_1 + s_1$$

$$\sigma(b) - \sigma(a) = (s_1 - s_3) + (s_2 - s_3) \; > \; 0 \quad \Longrightarrow \quad b \text{ is ranked strictly above } a.$$

In R':

$$\sigma(a) = s_1 + s_2 + s_3, \qquad \sigma(b) = s_3 + s_1 + s_2 \quad \Longrightarrow \quad a \text{ and } b \text{ are tied.}$$

IIA fails

Every voter ranks a against b identically in R and R', yet the social ranking of (a, b) changes from "b strictly above a" to "a tied with b". This holds for *every* scoring rule with $s_1 > s_3$ – Borda, plurality, veto, and everything in between.

Is a Dictatorship the Escape?

Question

Is a dictatorship the scoring rule that escapes this?

Is a Dictatorship the Escape?

Question

Is a dictatorship the scoring rule that escapes this?

Answer

No – a dictatorship is not a scoring rule at all, once $n \geqslant 2$. Scoring rules are **anonymous**: a total score is a sum over voters, so swapping two voters' ballots leaves every total unchanged. A dictatorship is not anonymous.

Is a Dictatorship the Escape?

Question

Is a dictatorship the scoring rule that escapes this?

Answer

No – a dictatorship is not a scoring rule at all, once $n \geqslant 2$. Scoring rules are **anonymous**: a total score is a sum over voters, so swapping two voters' ballots leaves every total unchanged. A dictatorship is not anonymous.

The only escape

The one scoring rule satisfying IIA is the trivial $s_1 = s_2 = s_3$, which scores every alternative equally and declares everything tied – and so fails Weak Pareto. There is no scoring rule with both properties.

An Aggregation Rule That Satisfies IIA

Question

What is the one aggregation rule that trivially satisfies IIA – the social ranking of (a, b) depends on only one agent's ranking of (a, b), always?

An Aggregation Rule That Satisfies IIA

Question

What is the one aggregation rule that trivially satisfies IIA – the social ranking of (a, b) depends on only one agent's ranking of (a, b), always?

Answer

A **dictatorship**: a fixed agent d such that $F(R) = R_d$ for every profile. It turns out this is the *only* way, among scoring rules, to get IIA – a first hint of the impossibility to come.

In-Class Quiz 1

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

Chunk 3

Arrow's Theorem and Decisive Sets

(≈ 20–25 minutes)

Arrow's Impossibility Theorem

Theorem (Arrow, 1951)

For $|A| \geqslant 3$, if an ASWF F satisfies Weak Pareto and IIA, then F must be a dictatorship.

Arrow's Impossibility Theorem

Theorem (Arrow, 1951)

For $|A| \geqslant 3$, if an ASWF F satisfies Weak Pareto and IIA, then F must be a dictatorship.

Not "most" aggregation rules fail these two mild-sounding axioms together – *every* rule that satisfies both is a dictatorship, full stop.

Arrow's Impossibility Theorem

Theorem (Arrow, 1951)

For $|A| \geqslant 3$, if an ASWF F satisfies Weak Pareto and IIA, then F must be a dictatorship.

Not "most" aggregation rules fail these two mild-sounding axioms together – *every* rule that satisfies both is a dictatorship, full stop.

The shape of the proof

Show any F satisfying WP and IIA has a *decisive* group; then shrink that group, one member at a time, until only a single dictator remains.

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

1. G is **almost decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G, \quad bP_j a, \forall j \in N \setminus G] \implies [a\hat{F}(R)b].$$

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

1. G is **almost decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G, \quad bP_j a, \forall j \in N \setminus G] \implies [a\hat{F}(R)b].$$

We will write this with the shorthand $\overline{D}_G(a, b)$: "G is almost decisive over $\{a, b\}$ w.r.t. F".

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

1. G is **almost decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G, \quad bP_j a, \forall j \in N \setminus G] \implies [a\hat{F}(R)b].$$

   We will write this with the shorthand $\overline{D}_G(a, b)$: "G is almost decisive over $\{a, b\}$ w.r.t. F".

2. G is **decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G] \implies [a\hat{F}(R)b].$$

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

1. G is **almost decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G, \quad bP_j a, \forall j \in N \setminus G] \implies [a\hat{F}(R)b].$$

   We will write this with the shorthand $\overline{D}_G(a, b)$: "G is almost decisive over $\{a, b\}$ w.r.t. F".

2. G is **decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G] \implies [a\hat{F}(R)b].$$

   We will write this with the shorthand $D_G(a, b)$: "G is decisive over $\{a, b\}$ w.r.t. F".

Decisiveness

Definition

Let $F : \mathcal{R}^n \to \mathcal{R}$ be given, $G \subseteq N, G \neq \emptyset$.

1. G is **almost decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G, \quad bP_j a, \forall j \in N \setminus G] \implies [a\hat{F}(R)b].$$

   We will write this with the shorthand $\overline{D}_G(a, b)$: "G is almost decisive over $\{a, b\}$ w.r.t. F".

2. G is **decisive** over $\{a, b\}$ if for every R:

$$[aP_i b, \forall i \in G] \implies [a\hat{F}(R)b].$$

   We will write this with the shorthand $D_G(a, b)$: "G is decisive over $\{a, b\}$ w.r.t. F".

Observation: $D_G(a, b) \implies \overline{D}_G(a, b)$

Reading Decisiveness

Reading a decisive set

If a small clique on the steering committee is decisive over (proposal A, proposal B), i.e. $D_G(A, B)$, that clique's preference for A over B always wins – regardless of what the rest of the committee thinks.

Reading Decisiveness

Reading a decisive set

If a small clique on the steering committee is decisive over (proposal A, proposal B), i.e. $D_G(A, B)$, that clique's preference for A over B always wins – regardless of what the rest of the committee thinks.

Reading an almost decisive set

$\overline{D}_G(A, B)$ is weaker: the clique wins only in the extreme situation where *everyone else* unanimously prefers B to A.

Reading Decisiveness

Reading a decisive set

If a small clique on the steering committee is decisive over (proposal A, proposal B), i.e. $D_G(A, B)$, that clique's preference for A over B always wins – regardless of what the rest of the committee thinks.

Reading an almost decisive set

$\overline{D}_G(A, B)$ is weaker: the clique wins only in the extreme situation where *everyone else* unanimously prefers B to A.

Decisiveness is not symmetric: $D_G(a, b)$ says nothing about $D_G(b, a)$.

Proof of Arrow's theorem

The proof proceeds in two parts:

Proof of Arrow's theorem

The proof proceeds in two parts:

Part 1: (**Field expansion lemma**) If a group is **almost decisive** over a pair of alternatives, it is **decisive** over all pairs of alternatives

Proof of Arrow's theorem

The proof proceeds in two parts:

Part 1: (**Field expansion lemma**) If a group is **almost decisive** over a pair of alternatives, it is **decisive** over all pairs of alternatives

Part 2: (**Group contraction lemma**) If a group is decisive, then a strict non-empty subset of that group is also decisive.

Proof of Arrow's theorem

The proof proceeds in two parts:

Part 1: (**Field expansion lemma**) If a group is **almost decisive** over a pair of alternatives, it is **decisive** over all pairs of alternatives

Part 2: (**Group contraction lemma**) If a group is decisive, then a strict non-empty subset of that group is also decisive.

Note: these two lemmas immediately prove the theorem

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a, b) \Rightarrow D_G(x, y).$$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a, b) \Rightarrow D_G(x, y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a, b) \Rightarrow D_G(x, y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a, b) \Rightarrow D_G(a, y), y \neq a, b$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a, b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a, b$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
3. $\overline{D}_G(a,b) \Rightarrow D_G(x,y), x \neq a,b, \quad y \neq a,b$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
3. $\overline{D}_G(a,b) \Rightarrow D_G(x,y), x \neq a,b, \quad y \neq a,b$
4. $\overline{D}_G(a,b) \Rightarrow D_G(x,a), x \neq a,b$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
3. $\overline{D}_G(a,b) \Rightarrow D_G(x,y), x \neq a,b, \quad y \neq a,b$
4. $\overline{D}_G(a,b) \Rightarrow D_G(x,a), x \neq a,b$
5. $\overline{D}_G(a,b) \Rightarrow D_G(b,y), y \neq a,b$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
3. $\overline{D}_G(a,b) \Rightarrow D_G(x,y), x \neq a,b, \quad y \neq a,b$
4. $\overline{D}_G(a,b) \Rightarrow D_G(x,a), x \neq a,b$
5. $\overline{D}_G(a,b) \Rightarrow D_G(b,y), y \neq a,b$
6. $\overline{D}_G(a,b) \Rightarrow D_G(a,b)$

Field expansion lemma

Lemma

Let F satisfy WP and IIA, then $\forall a, b, x, y, G \subseteq N, G \neq \emptyset, a \neq b, x \neq y$

$$\overline{D}_G(a,b) \Rightarrow D_G(x,y).$$

It implies that under WP and IIA, the two notions of decisiveness are equivalent.

Cases to consider (ordered for the convenience of the proof):

1. $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
2. $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
3. $\overline{D}_G(a,b) \Rightarrow D_G(x,y), x \neq a,b, \quad y \neq a,b$
4. $\overline{D}_G(a,b) \Rightarrow D_G(x,a), x \neq a,b$
5. $\overline{D}_G(a,b) \Rightarrow D_G(b,y), y \neq a,b$
6. $\overline{D}_G(a,b) \Rightarrow D_G(a,b)$
7. $\overline{D}_G(a,b) \Rightarrow D_G(b,a)$

Proof of FEL



- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_i y,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_iy,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$
- Construct R' s.t.

G		$N \setminus G$	
a	a	b	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	y
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y	y	y	a

positions of a and y in $N \setminus G$ s.t. $R'|_{a,y} = R|_{a,y}$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_iy,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$
- Construct R' s.t.

G		$N \setminus G$	
a	a	b	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	y
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y	y	y	a

positions of a and y in $N \setminus G$ s.t. $R'|_{a,y} = R|_{a,y}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_iy,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$
- Construct R' s.t.

G		$N \setminus G$	
a	a	b	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	y
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y	y	y	a

positions of a and y in $N \setminus G$ s.t. $R'|_{a,y} = R|_{a,y}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $b, y \Rightarrow b\hat{F}(R')y$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_iy,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$
- Construct R' s.t.

G		$N \setminus G$	
a	a	b	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	y
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y	y	y	a

positions of a and y in $N \setminus G$ s.t. $R'|_{a,y} = R|_{a,y}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $b,y \Rightarrow b\hat{F}(R')y$
- Transitivity $\Rightarrow a\hat{F}(R')y$

Proof of FEL

- **Case 1:** $\overline{D}_G(a,b) \Rightarrow D_G(a,y), y \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $aP_iy,\ \forall i \in G$ — Need to show: $a\hat{F}(R)y$
- Construct R' s.t.

G		$N \setminus G$	
a	a	b	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	y
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y	y	y	a

positions of a and y in $N \setminus G$ s.t. $R'|_{a,y} = R|_{a,y}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $b,y \Rightarrow b\hat{F}(R')y$
- Transitivity $\Rightarrow a\hat{F}(R')y$
- IIA $\Rightarrow a\hat{F}(R)y$. Hence, $D_G(a,y)$

Proof of FEL (contd.)



- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., xP_ib, $\forall i \in G$ — Need to show: $x\hat{F}(R)b$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $xP_ib,\ \forall i \in G$ — Need to show: $x\hat{F}(R)b$
- Construct R' s.t.

G		$N \setminus G$	
x	x	x	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
a	a	b	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	a

positions of x and b in $N \setminus G$ s.t. $R'|_{x,b} = R|_{x,b}$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $xP_ib,\ \forall i \in G$ — Need to show: $x\hat{F}(R)b$
- Construct R' s.t.

G		$N \setminus G$	
x	x	x	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
a	a	b	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	a

positions of x and b in $N \setminus G$ s.t. $R'|_{x,b} = R|_{x,b}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $xP_ib,\ \forall i \in G$ — Need to show: $x\hat{F}(R)b$
- Construct R' s.t.

G		$N \setminus G$	
x	x	x	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
a	a	b	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	a

positions of x and b in $N \setminus G$ s.t. $R'|_{x,b} = R|_{x,b}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $x,a \Rightarrow x\hat{F}(R')a$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $xP_ib,\ \forall i \in G$ — Need to show: $x\hat{F}(R)b$
- Construct R' s.t.

G		$N \setminus G$	
x	x	x	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
a	a	b	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	a

positions of x and b in $N \setminus G$ s.t. $R'|_{x,b} = R|_{x,b}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $x,a \Rightarrow x\hat{F}(R')a$
- Transitivity $\Rightarrow x\hat{F}(R')b$

Proof of FEL (contd.)

- **Case 2:** $\overline{D}_G(a,b) \Rightarrow D_G(x,b), x \neq a,b$
- Pick an arbitrary $R \in \mathcal{R}^n$, s.t., $xP_ib,\ \forall i \in G$ — Need to show: $x\hat{F}(R)b$
- Construct R' s.t.

G		$N \setminus G$	
x	x	x	b
$\vdots$	$\vdots$	$\vdots$	$\vdots$
a	a	b	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$
b	b	a	a

positions of x and b in $N \setminus G$ s.t. $R'|_{x,b} = R|_{x,b}$

- $\overline{D}_G(a,b) \Rightarrow a\hat{F}(R')b$
- WP over $x,a \Rightarrow x\hat{F}(R')a$
- Transitivity $\Rightarrow x\hat{F}(R')b$
- IIA $\Rightarrow x\hat{F}(R)b$. Hence, $D_G(x,b)$

Proof of FEL (other cases)

- **Case 3:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,y)\ (x \neq a,y)$

Proof of FEL (other cases)

- **Case 3:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,y)\ (x \neq a,y)$
- **Case 4:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(x,a)\ (x \neq a,b)$

Proof of FEL (other cases)

- **Case 3:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,y)\ (x \neq a,y)$
- **Case 4:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(x,a)\ (x \neq a,b)$
- **Case 5:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(b,y)\ (y \neq a,b)$

Proof of FEL (other cases)

- **Case 3:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,y)\ (x \neq a,y)$
- **Case 4:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(x,a)\ (x \neq a,b)$
- **Case 5:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(b,y)\ (y \neq a,b)$
- **Case 6:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(a,b)$

Proof of FEL (other cases)

- **Case 3:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,y)\ (x \neq a,y)$
- **Case 4:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(x,a)\ (x \neq a,b)$
- **Case 5:** $\overline{D}_G(a,b) \overset{\text{(case 1)}}{\Longrightarrow} D_G(a,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(a,y) \overset{\text{(case 2)}}{\Longrightarrow} D_G(b,y)\ (y \neq a,b)$
- **Case 6:** $\overline{D}_G(a,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(x,b)\ (x \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(x,b) \overset{\text{(case 2)}}{\Longrightarrow} D_G(a,b)$
- **Case 7:** $\overline{D}_G(a,b) \overset{\text{(case 5)}}{\Longrightarrow} D_G(b,y)\ (y \neq a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_G(b,y) \overset{\text{(case 1)}}{\Longrightarrow} D_G(b,a)$

Chunk 4

Group Contraction, and Assembling the Proof

(≈ 25–30 minutes)

Group contraction lemma

Lemma

Let F satisfy WP and IIA, and let $G \subseteq N, G \neq \emptyset, |G| \geqslant 2$ be decisive. Then $\exists G' \subset G, G' \neq \emptyset$ which is also decisive.

Proof:

- $G, |G| \geqslant 2$ is given. Let $G_1 \subset G, G_2 = G \setminus G_1$, $G_1, G_2 \neq \emptyset$, arbitrary.

Group contraction lemma

Lemma

Let F satisfy WP and IIA, and let $G \subseteq N, G \neq \emptyset, |G| \geqslant 2$ be decisive. Then $\exists G' \subset G, G' \neq \emptyset$ which is also decisive.

Proof:

- $G, |G| \geqslant 2$ is given. Let $G_1 \subset G, G_2 = G \setminus G_1$, $G_1, G_2 \neq \emptyset$, arbitrary.
- Construct R

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

$aP_ib, \forall i \in G$ and G decisive $\Rightarrow a\hat{F}(R)b$

Group contraction lemma

Lemma

Let F satisfy WP and IIA, and let $G \subseteq N, G \neq \emptyset, |G| \geqslant 2$ be decisive. Then $\exists G' \subset G, G' \neq \emptyset$ which is also decisive.

Proof:

- $G, |G| \geqslant 2$ is given. Let $G_1 \subset G, G_2 = G \setminus G_1$, $G_1, G_2 \neq \emptyset$, arbitrary.
- Construct R

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

$aP_ib, \forall i \in G$ and G decisive $\Rightarrow a\hat{F}(R)b$

- Where can c stand in $F(R)$ w.r.t. a? We will show in every possible case, either G_1 or G_2 will be decisive

Proof of GCL

Case 1: $a\hat{F}(R)c$

- Consider G_1

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

Proof of GCL

Case 1: $a\hat{F}(R)c$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- Consider G_1
- $aP_ic, \forall i \in G_1, cP_ia, \forall i \in N \setminus G_1$

Proof of GCL

Case 1: $a\hat{F}(R)c$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- Consider G_1
- $aP_ic, \forall i \in G_1, cP_ia, \forall i \in N \setminus G_1$
- Consider each R' where the above relation holds

Proof of GCL

Case 1: $a\hat{F}(R)c$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- Consider G_1
- $aP_ic, \forall i \in G_1, cP_ia, \forall i \in N \setminus G_1$
- Consider each R' where the above relation holds
- by IIA $a\hat{F}(R')c$

Proof of GCL

Case 1: $a\hat{F}(R)c$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- Consider G_1
- $aP_ic, \forall i \in G_1, cP_ia, \forall i \in N \setminus G_1$
- Consider each R' where the above relation holds
- by IIA $a\hat{F}(R')c$
- Hence $\overline{D}_{G_1}(a,c) \overset{\text{(FEL)}}{\Longrightarrow} D_{G_1}$

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$
- Consider G_2

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$
- Consider G_2
- $cP_ib, \forall i \in G_2, bP_ic, \forall i \in N \setminus G_2$

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$
- Consider G_2
- $cP_ib, \forall i \in G_2, bP_ic, \forall i \in N \setminus G_2$
- Consider each R' where the above relation holds

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$
- Consider G_2
- $cP_ib, \forall i \in G_2, bP_ic, \forall i \in N \setminus G_2$
- Consider each R' where the above relation holds
- by IIA $c\hat{F}(R')b$

Proof of GCL (contd.)

Case 2: $\neg(a\hat{F}(R)c) \implies cF(R)a$

G_1	G_2	$N \setminus G$
a	c	b
b	a	c
c	b	a

have seen $\Rightarrow a\hat{F}(R)b$

- $a\hat{F}(R)b$ and $cF(R)a$ give $c\hat{F}(R)b$
- Consider G_2
- $cP_ib, \forall i \in G_2, bP_ic, \forall i \in N \setminus G_2$
- Consider each R' where the above relation holds
- by IIA $c\hat{F}(R')b$
- Hence $\overline{D}_{G_2}(c,b) \overset{\text{(FEL)}}{\implies} D_{G_2}$

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a, b)$ for every pair (if everyone prefers a to b, so does society)

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a, b)$ for every pair (if everyone prefers a to b, so does society)
- GCL repeatedly: N decisive $\Rightarrow$ a strict subset decisive $\Rightarrow \cdots \Rightarrow$ a singleton $\{d\}$ decisive

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a,b)$ for every pair (if everyone prefers a to b, so does society)
- GCL repeatedly: N decisive $\Rightarrow$ a strict subset decisive $\Rightarrow \cdots \Rightarrow$ a singleton $\{d\}$ decisive
- $D_{\{d\}}(a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_{\{d\}}(a,b) \overset{\text{(FEL)}}{\Longrightarrow} D_{\{d\}}(x,y)$ for every pair (x,y)

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a,b)$ for every pair (if everyone prefers a to b, so does society)
- GCL repeatedly: N decisive $\Rightarrow$ a strict subset decisive $\Rightarrow \cdots \Rightarrow$ a singleton $\{d\}$ decisive
- $D_{\{d\}}(a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_{\{d\}}(a,b) \overset{\text{(FEL)}}{\Longrightarrow} D_{\{d\}}(x,y)$ for every pair (x,y)
- So d's strict preference decides every pair – F is a dictatorship

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a,b)$ for every pair (if everyone prefers a to b, so does society)
- GCL repeatedly: N decisive $\Rightarrow$ a strict subset decisive $\Rightarrow \cdots \Rightarrow$ a singleton $\{d\}$ decisive
- $D_{\{d\}}(a,b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_{\{d\}}(a,b) \overset{\text{(FEL)}}{\Longrightarrow} D_{\{d\}}(x,y)$ for every pair (x,y)
- So d's strict preference decides every pair – F is a dictatorship

Assembling Arrow's Theorem

- WP $\Rightarrow$ N is decisive: $D_N(a, b)$ for every pair (if everyone prefers a to b, so does society)
- GCL repeatedly: N decisive $\Rightarrow$ a strict subset decisive $\Rightarrow \cdots \Rightarrow$ a singleton $\{d\}$ decisive
- $D_{\{d\}}(a, b) \overset{\text{(definition)}}{\Longrightarrow} \overline{D}_{\{d\}}(a, b) \overset{\text{(FEL)}}{\Longrightarrow} D_{\{d\}}(x, y)$ for every pair (x, y)
- So d's strict preference decides every pair – F is a dictatorship

The uncomfortable takeaway

Any aggregation rule satisfying two innocuous-looking axioms – unanimity and independence from irrelevant options – collapses to one person's opinion. Real voting and ranking systems must give up WP, IIA, unrestricted domain, or transitivity of the output to escape this.

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking
- IIA: the social ranking of a vs. b should depend only on how agents rank a vs. b

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking
- IIA: the social ranking of a vs. b should depend only on how agents rank a vs. b
- No scoring rule satisfies IIA except a dictatorship

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking
- IIA: the social ranking of a vs. b should depend only on how agents rank a vs. b
- No scoring rule satisfies IIA except a dictatorship
- Arrow's theorem: WP + IIA + $|A| \geqslant 3 \implies$ dictatorship, with no exceptions

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking
- IIA: the social ranking of a vs. b should depend only on how agents rank a vs. b
- No scoring rule satisfies IIA except a dictatorship
- Arrow's theorem: WP + IIA + $|A| \geqslant 3 \implies$ dictatorship, with no exceptions
- Proof via decisive sets: Field Expansion Lemma (power spreads across pairs) and Group Contraction Lemma (decisive groups shrink) together isolate a single dictator

Lecture 14: What We Built

- ASWF $F : \mathcal{R}^n \to \mathcal{R}$ aggregates individual preference orderings into one social ordering
- Weak and Strong Pareto: unanimous agreement should carry over to the social ranking
- IIA: the social ranking of a vs. b should depend only on how agents rank a vs. b
- No scoring rule satisfies IIA except a dictatorship
- Arrow's theorem: WP + IIA + $|A| \geqslant 3 \implies$ dictatorship, with no exceptions
- Proof via decisive sets: Field Expansion Lemma (power spreads across pairs) and Group Contraction Lemma (decisive groups shrink) together isolate a single dictator
- **Next week:** escaping the impossibility – the Gibbard–Satterthwaite theorem and domain restrictions

In-Class Quiz 2

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

End of Lecture 14

See you next class.



भारतीय प्रौद्योगिकी संस्थान मुंबई

Indian Institute of Technology Bombay