Mahindra University
Global Thinkers. Engaged Leaders.

Game Theory (CS4187)

Lectures 27, 31, and 32

Date: 22/10/2024
11/11/2024
12/11/2024

Instructor: Gourav Saha

Mechanism Design: The Setup

- Mechanism design is about **design rules of the games** in an **incomplete information game**, more specifically a **Bayesian Game**, so as to **make the players choose the desired actions**.

- So, let's recollect the definition of a Bayesian game.
  - We will consider **simultaneous move** Bayesian game for the time being.

Mechanism Design: The Setup

A Bayesian game setup consists of:

1. $\mathrm{N} = \{1, 2, \cdots, \mathrm{n}\}$ is the **set of players**.
2. $\mathrm{S_i}$ is the **set of actions** of the $\mathrm{i^{th}}$ player. The action of the $\mathrm{i^{th}}$ player is s_i where $s_i \in \mathrm{S_i}$.
   - Define $S = \mathrm{S_1} \times \mathrm{S_2} \times \cdots \times \mathrm{S_n}$ and $s = (s_1, s_2, \cdots, s_n)$.
3. Θ_i is the **set of types** of the $\mathrm{i^{th}}$ player. The type of the $\mathrm{i^{th}}$ player is θ_i where i.e. $\theta_i \in \Theta_\mathrm{i}$.
   - Define $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_\mathrm{n}$ and $\theta = (\theta_1, \theta_2, \cdots, \theta_n)$.
4. $P\colon \Theta \to \mathbb{R}$ is the **common prior**.
5. $\mathrm{u_i} : \Theta \times S \to \mathbb{R}$ is the utility function of the $\mathrm{i^{th}}$ player. The payoff the $\mathrm{i^{th}}$ player is $u_i(\theta, s)$.
6. N, $\mathrm{S_i}$,Θ_i, P, and u_i are **common knowledge**. Player i's type θ_i is its **private information**.

Mechanism Design: The Setup

The difference between a Bayesian game setup and a mechanism design setup is **highlighted in red**. The reason for this change is explained in the next slide.

A mechanism design setup consists of:

1. $\mathrm{N} = \{1, 2, \cdots, \mathrm{n}\}$ is the set of players.
2. X is the set of outcomes.
   - An outcome $x \in X$ is a function of the strategy profile $s = (s_1, s_2, \cdots, s_n)$ of the players.
3. Θ_i is the set of types of the i^{th} player. The type of the i^{th} player is θ_i where i.e. $\theta_i \in \Theta_\mathrm{i}$.
   - Define $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_\mathrm{n}$ and $\theta = (\theta_1, \theta_2, \cdots, \theta_n)$.
4. $P\colon \Theta \to \mathbb{R}$ is the common prior.
5. $\mathrm{u_i} : \Theta \times X \to \mathbb{R}$ is the utility function of the i^{th} player. The payoff the i^{th} player is $u_i(\theta, x)$.
6. N, X ,Θ_i, P, and u_i are common knowledge. Player i's type θ_i is its private information.

Mechanism Design: The Setup

Outcome vs Action:

- Consider a auction with only two bidders. Whether the bid is $\overset{\text{B1}}{100}, \overset{\text{B2}}{500}$ or $\overset{\text{B1}}{200}, \overset{\text{B2}}{300}$, bidder B2 will win the auction.
  - **Who won the auction** and the **amount the winner has to pay** is the outcome.
  - The individual bids were the actions. Off course, the bids decides the winner and the winner's payment.

- Since actions decides the outcome it is obvious that if the game designer can control the actions of the players, it can control the outcome. But in many cases, such a "high degree of control" over player's action is not required for the game designer to meet it's objective; it just wants to control the outcome.

- Needles to say, action an outcome can be the same, i.e. $X = S$.

Mechanism Design: The Setup

Social Choice Function:

- While designing the rules of the game, the game designer has a certain **objective**. This objective can is captured using **social choice function (SCF)**.

- A social choice function f is a mapping from the **type profile** to the **"desired" outcome**. Mathematically,

$$f : \boldsymbol{\Theta} \to \boldsymbol{X}.$$

  The outcome x corresponding to a type profile θ that is generated by SCF is called social choice of type profile θ.

- Example: Consider two bidders B1 and B2 with valuations v_1 and v_2 respectively. Then,

$$f(v_1, v_2) = \begin{cases} (B1, p_1) & , v_1 \geq v_2 \\ (B2, p_2) & , v_1 < v_2 \end{cases}$$

Winner

Winner's payment

In the above example, we are not explicitly worrying about the winner's payment but rather who is the winner in order to demonstrate the overall idea of SCF.

Mechanism Design: The Setup

Social Choice Function:

- While designing the rules of the game, the game designer has a certain **objective**. This objective can is captured using **social choice function (SCF)**.

- A social choice function f is a mapping from the **type profile** to the **"desired" outcome**. Mathematically,

$$f : \boldsymbol{\Theta} \rightarrow \boldsymbol{X}.$$

  The outcome x corresponding to a type profile θ that is generated by SCF is called social choice of type profile θ.

- SCF is quite a **general approach** to capture the objective of the game designer. In most cases, the objective of the game designer is written as a real-valued function. Example: For two player bidding, the objective of the game designer is to maximize $\boldsymbol{v_1 w_1 + v_2 w_2}$, where $w_1, w_2 \in \{0,1\}$ is the winner and hence either w_1 or w_2 can be 1.

Mechanism Design: The Setup

Social Choice Function:

[Example 1]

w

Mechanism Design: Different Kinds of Mechanism

- A mechanism design problem can be thought of as:
  - Step 1: Making the players reveal their types.
  - Step 2: Using the types from step 1, decide the outcome by solving an optimization problem. SCF is a function that captures this optimization problem.

- As far as step 1 is concerned, there are two kind of mechanisms that will be discussed next.

Definition (Direct mechanism): Direct mechanism is a tuple $(\Theta_1, \Theta_2, \cdots, \Theta_n, f)$ where f is the social choice function.

Mechanism Design: Different Kinds of Mechanism

Definition (Direct mechanism): Direct mechanism is a tuple $\mathcal{D} = (\Theta_1, \Theta_2, \cdots, \Theta_n, f)$ where f is the social choice function. We may use a short hand $\mathcal{D} = (\Theta, f)$ instead of $\mathcal{D} = (\Theta_1, \Theta_2, \cdots, \Theta_n, f)$.

Qualitative explanation of direct mechanism:

- In direct mechanism, the game designer asks the players to set it's type.
- Before the player reveals it's type, the game designer also tells the players the decision rule. For direct mechanism, the decision rule is the social choice function f.
- Then all player $i \in N$ reveal their type $\hat{\theta}_i \in \Theta_i$. Note that $\hat{\theta}_i$ may not be equal to player i's true type θ_i.
- After the player reveals their types $\hat{\theta}_i$, the game designer sets the outcome as $f(\hat{\theta})$ where $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2, \cdots, \hat{\theta}_n)$.

Mechanism Design: Different Kinds of Mechanism

Definition (Indirect mechanism): Indirect mechanism is a tuple $\mathcal{M} = (S_1, S_2, \cdots, S_n, g)$ where S_i is the set of actions of player i and $g : S_1 \times S_2 \times \cdots \times S_n \rightarrow X$. We may use a short hand $\mathcal{M} = (S, g)$ instead of $\mathcal{M} = (S_1, S_2, \cdots, S_n, g)$.

Qualitative explanation of indirect mechanism:

➢ In indirect mechanism, the game designer asks the players to take an **action**. The action s_i of player i is a mapping from it's type θ_i to it's action space S_i.

- Another reason why the terms **outcome** and **action** are used differently in context of mechanism design.

➢ Before the player takes action, the game designer also tells the players the decision rule g that will map the strategy profile $s = (s_1, s_2, \cdots, s_n)$ to an outcome in set X.

➢ Then all player i takes action $s_i(\theta_i)$ where θ_i is the true type of player i.

➢ After the player takes action, the game designer sets the outcome as $g(s)$.

Mechanism Design: Different Kinds of Mechanism

Definition (Direct mechanism): Direct mechanism is a tuple $\mathcal{D} = (\Theta_1, \Theta_2, \cdots, \Theta_n, f)$ where f is the social choice function.

Definition (Indirect mechanism): Indirect mechanism is a tuple $\mathcal{M} = (S_1, S_2, \cdots, S_n, g)$ where S_i is the set of actions of player i and $g : S_1 \times S_2 \times \cdots \times S_n \to X$.

NOTE: Direction mechanism is a special case of indirect mechanism where $S_i = \Theta_i, \forall i \in N$ and $g \equiv f$.

Mechanism Design: Different Kinds of Mechanism

Induced Bayesian Game:

- The moment a mechanism is set, this mechanism induces a Bayesian game among the players. This is called the induced game by the mechanism. In what follows, we will explain this idea using indirect mechanism as it is more general. The induced Bayesian game by mechanism $\mathcal{M} = (S, g)$:

1. $\mathrm{N} = \{1, 2, \cdots, \mathrm{n}\}$ is the **set of players**.

2. $\mathrm{S_i}$ is the **set of actions** of the $\mathrm{i^{th}}$ player. The action of the $\mathrm{i^{th}}$ player is s_i where $s_i \in \mathrm{S_i}$.
   - Define $S = \mathrm{S_1} \times \mathrm{S_2} \times \cdots \times \mathrm{S_n}$ and $s = (s_1, s_2, \cdots, s_n)$.

3. Θ_i is the **set of types** of the $\mathrm{i^{th}}$ player. The type of the $\mathrm{i^{th}}$ player is θ_i where i.e. $\theta_i \in \Theta_\mathrm{i}$.
   - Define $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_\mathrm{n}$ and $\theta = (\theta_1, \theta_2, \cdots, \theta_n)$.

4. $P\colon \Theta \to \mathbb{R}$ is the **common prior**.

5. $\mathrm{u_i} : \Theta \times S \to \mathbb{R}$ is the utility function of the $\mathrm{i^{th}}$ player. The payoff the $\mathrm{i^{th}}$ player is $u_i(\theta, \boldsymbol{g}(\boldsymbol{s}))$.

Mechanism Design: Different Kinds of Mechanism

Induced Bayesian Game:

- The moment a mechanism is set, this mechanism induces a Bayesian game among the players. This is called the induced game by the mechanism. In what follows, we will explain this idea using indirect mechanism as it is more general. The induced Bayesian game by mechanism $\mathcal{M} = (S, g)$:

1. $\mathrm{N} = \{1, 2, \cdots, \mathrm{n}\}$ is the set of players.

2. $\mathrm{S_i}$ is the set of actions of the $\mathrm{i^{th}}$ player. The action of the $\mathrm{i^{th}}$ player is s_i where $s_i \in \mathrm{S_i}$.
   - Define $S = \mathrm{S_1} \times \mathrm{S_2} \times \cdots \times \mathrm{S_n}$ and $s = (s_1, s_2, \cdots, s_n)$.

3. Θ_i is the set of types of the $\mathrm{i^{th}}$ player. The type of the $\mathrm{i^{th}}$ player is θ_i where i.e. $\theta_i \in \Theta_\mathrm{i}$.
   - Define $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_\mathrm{n}$ and $\theta = (\theta_1, \theta_2, \cdots, \theta_n)$.

4. $P: \Theta \to \mathbb{R}$ is the common prior.

This will be $f(\hat{\theta})$ for direct mechanism where $\hat{\theta}$ is the revealed type profile.

5. $\mathrm{u_i} : \Theta \times S \to \mathbb{R}$ is the utility function of the $\mathrm{i^{th}}$ player. The payoff the $\mathrm{i^{th}}$ player is $u_i(\theta, \boldsymbol{g}(\boldsymbol{s}))$.

Solution Concepts of the Induced Bayesian Game

- As discussed in the previous slide, as soon as a mechanism is decided, it induces as Bayesian game.

- Now that we have a Bayesian Game, we can talk about **solution concepts the induced Bayesian Game**. As far as mechanism design is concerned, we are interested in the following solution concepts:
  - **Dominant strategy equilibrium**.
  - **Bayesian Nash equilibrium**.

- We will now define these two solution concepts from the perspective of mechanism design.
  - The definition of these solution concepts is same as that discussed in modules 2 and 3 with the exception of dominant strategy equilibrium which has a small difference.

Solution Concepts of the Induced Bayesian Game

Dominant Strategy Equilibrium of the Induced Game

Definition (Dominant Strategy of a player in the induced Bayesian Game): The strategy s_i of player i is it's dominant strategy for type profile θ in the Bayesian game induced by mechanism $\mathcal{M} = (S, g)$ if,

$$u_i\left(\theta, g\left(s_i, s_{-i}\right)\right) \geq u_i\left(\theta, g\left(s_i^{'}, s_{-i}\right)\right), \forall s_i^{'} \in S_i, \forall s_{-i} \in S_{-i}$$

NOTE: The above definition of dominant strategy is for **very weakly dominant strategy** and NOT **weakly dominant strategy**.

Solution Concepts of the Induced Bayesian Game

Dominant Strategy Equilibrium of the Induced Game

Definition (Dominant Strategy of a player in the induced Bayesian Game): The strategy s_i of player i is it's dominant strategy for type profile θ in the Bayesian game induced by mechanism $\mathcal{M} = (S, g)$ if,

$$u_i\left(\theta, g\left(s_i, s_{-i}\right)\right) \geq u_i\left(\theta, g\left(s_i^{'}, s_{-i}\right)\right), \forall s_i^{'} \in S_i, \forall s_{-i} \in S_{-i}$$

Definition (Dominant Strategy Equilibrium of the Induced Game): A pure strategy profile $s(\theta) = \left(s_1(\theta_1), s_2(\theta_2), \cdots, s_n(\theta_n)\right)$ is a dominant strategy equilibrium of the Bayesian game induced by mechanism $\mathcal{M} = (S, g)$ if for all players $\boldsymbol{i} \in \boldsymbol{N}$,

$$u_i\left(\theta, g\left(s_i\left(\theta_i\right), s_{-i}\right)\right) \geq u_i\left(\theta, g\left(s_i^{'}, s_{-i}\right)\right), \forall s_i^{'} \in S_i, \forall s_{-i} \in S_{-i}, \forall \theta \in \Theta$$

Solution Concepts of the Induced Bayesian Game

Bayesian Nash Equilibrium (BNE) of the Induced Game

Definition (BNE of the Induced Game): A **pure** strategy profile $s(\theta) = \left(s_1(\theta_1), s_2(\theta_2), \cdots, s_n(\theta_n)\right)$ is a BNE of the Bayesian game induced by mechanism $\mathcal{M} = (S, g)$ if **for all players $i \in N$**,

$$U_i\left(g\left(s_i\left(\theta_i\right), s_{-i}\left(\theta_{-i}\right)\right) | \theta_i\right) \geq U_i\left(g\left(s_i^{'}, s_{-i}\left(\theta_{-i}\right)\right) | \theta_i\right), \; \forall s_i^{'} \in S_i, \forall \theta_i \in \Theta_i$$

NOTE: The functions U_i above are ex-interim utility given by,

$$U_i\left(g\left(s_i\left(\theta_i\right), s_{-i}\left(\theta_{-i}\right)\right) \middle| \theta_i\right) = \sum_{\theta_{-i} \in \Theta_{-i}} P\left[\theta_{-i} \middle| \theta_i\right] u_i\left(\theta, g\left(s_i\left(\theta_i\right), s_{-i}\left(\theta_{-i}\right)\right)\right)$$

Implementation of a SCF by a Mechanism

- Remember, a mechanism is an approach to obtain a certain objective.

- As we discussed before, this objective is defined by the SCF $\boldsymbol{f} : \boldsymbol{\Theta} \rightarrow \boldsymbol{X}$.

- The question is: When can we say that a mechanism $\mathcal{M} = (S, g)$ is successful in attaining it's objective? Answer: When it implements SCF f. But, what " implementing an SCF" means?

Definition (Implementation of a SCF): We say that a mechanism $\mathcal{M} = (S, g)$ implement SCF f is there exists a **pure strategy equilibrium** $s(\theta)$ such that,

$$g\left(s\left(\theta\right)\right) = f\left(\theta\right) \ , \forall \theta \in \Theta$$

The pure strategy equilibrium can be either: (i) Dominant strategy equilibrium, or (ii) BNE. This leads to the following cases that are discussed in the next two slides.

1. Dominant Strategy Implementable
2. Dominant Strategy Incentive Compatible **(important from exam perspective)**
3. Bayesian Implementable
4. Bayesian Incentive Compatible

Implementation of a SCF by a Mechanism

Definition (Dominant Strategy Implementable): An SCF f is said to be dominant strategy implementable by an **indirect mechanism** $\mathcal{M} = (S, g)$ if the game induced by $\mathcal{M}$ has a pure strategy profile $s(\theta) = (s_1(\theta_1), s_2(\theta_2), \cdots, s_n(\theta_n))$ that satisfies both the following conditions:

Condition 1: $u_i\left(\theta, g\left(s_i\left(\theta_i\right), s_{-i}\right)\right) \geq u_i\left(\theta, g\left(s_i^{'}, s_{-i}\right)\right), \forall s_i^{'} \in S_i, \forall s_{-i} \in S_{-i}, \forall \theta \in \Theta, \forall i \in N$

Condition for dominant strategy equilibrium

Condition 2: $g\left(s\left(\theta\right)\right) = f\left(\theta\right), \forall \theta \in \Theta$

Condition for implementation of SCF f

Definition (Dominant Strategy Incentive Compatible): A **direct mechanism** $\mathcal{D} = (\Theta, f)$ is said to be dominant strategy incentive compatible if,

$$u_i\left(\theta, f\left(\theta_i, \theta_{-i}\right)\right) \geq u_i\left(\theta, f\left(\theta_i^{'}, \theta_{-i}\right)\right), \forall \theta_i^{'} \in \Theta_i, \forall \theta_i \in \Theta_i, \forall \theta_{-i} \in \Theta_{-i}, \forall i \in N$$

IMPORTANT FROM EXAM PERSPECTIVE

Implementation of a SCF by a Mechanism

Definition (Bayesian Implementable): An SCF f is said to be Bayesian implementable by an **indirect mechanism** $\mathcal{M} = (S, g)$ if the game induced by $\mathcal{M}$ has a pure strategy profile $s(\theta) = \big(s_1(\theta_1), s_2(\theta_2), \cdots, s_n(\theta_n)\big)$ that satisfies both the following conditions:

Condition 1: $$U_i\left(g\left(s_i\left(\theta_i\right), s_{-i}\left(\theta_{-i}\right)\right) | \theta_i\right) \geq U_i\left(g\left(s_i^{'}, s_{-i}\left(\theta_{-i}\right)\right) | \theta_i\right), \forall s_i^{'} \in S_i, \forall \theta_i \in \Theta_i, \forall i \in N$$

Condition 2: $$g\left(s\left(\theta\right)\right) = f\left(\theta\right)\ , \forall \theta \in \Theta$$

Condition for implementation of SCF f

Condition for Bayesian Nash equilibrium (check definition of U_i in this slide).

Definition (Bayesian Incentive Compatible): A **direct mechanism** $\mathcal{D} = (\Theta, f)$ is said to be Bayesian incentive compatible if,

$$U_i\left(f\left(\theta_i, \theta_{-i}\right) | \theta_i\right) \geq U_i\left(f\left(\theta_i^{'}, \theta_{-i}\right) | \theta_i\right),\ \forall \theta_i^{'} \in S_i, \forall \theta_i \in \Theta_i, \forall i \in N$$

where the functions U_i above are ex-interim utility given by

$$U_i\left(f\left(\theta_i, \theta_{-i}\right) | \theta_i\right) = \sum_{\theta_{-i} \in \Theta_{-i}} P\left[\theta_{-i} | \theta_i\right] u_i\left(\theta, f\left(\theta_i, \theta_{-i}\right)\right)$$

Revelation Principle

- As you can well appreciate, direct mechanisms are more straightforward. Revelation Principle essentially states that we don't need to look for indirect mechanisms.
  - That does not mean that indirect mechanisms has no use. In some cases, even if direct mechanisms exists, they can be computationally difficult to implement compared to an indirect mechanism.

- Revelation Principle states the following: Say that SCF f is dominant strategy implementable by an indirect mechanism $\mathcal{M} = (S, g)$. Then there exists a direct mechanism $\mathcal{D} = (\Theta, f)$ that is dominant strategy incentive compatible.

- For the proof of the revelation principle refer to Theorem 16.1 (in chapter 16 of the book).

Thank You!