Mechanism Design

Raj Aryan Agrawal
190050097
Undergraduate, Department of Computer Science
Indian Institute of Technology Bombay

May 12, 2020

Contents

1. Introduction

Mechanism design is concerned with settings where a policy maker faces the problem of aggregating the *announced preferences* of players into a system wide decision when the *actual preferences* of the players are not publicly known. That is we need to solve an optimization problem with incomplete information.
Mechanism design uses a technique that induces a game among the agents such that the equilibrium of the induced game - the desired system wide solution is implemented.

1.1. Cake Cutting Problem

Example 1.1. (Cake Cutting Problem) Consider the mother of 2 kids, who has to design a mechanism to make her kids share a cake equally. Here the mother is the social planner (policy maker). If the mother slices the cake into 2 equal pieces and distributes, this may not be an acceptable solution as each kid will have the perception he got the smaller piece.
Instead consider the mechanism

- One of the kids would slice the cake into 2 pieces
- The other kid gets the chance to pick up any of the pieces and leaves the other for first

Child 1 will slice cake into 2 equal halves (in *his* eyes), as any other division will leave him with a smaller piece as the remaining piece.
Child 2 is happy because he gets to choose and so will choose what appears larger among the 2 in *his* eyes.
Thus this mechanism gives desired outcome and both agents are happy.

1.1.1 Generalised Cake Cutting

This is a mechanism for n players dividing a cake. We say an allocation is *fair* if $u_i(piece_i) \geq 1/n$ for each player and envy-free if $u_i(piece_i) \geq u_i(piece_j)$ for each i and j. Then the algorithm is -
Banach - Knaster Discrete Protocol

- A player cuts a piece which he believes to be size $1/n$
- Every other player can trim the piece to what they consider $1/n$ or pass if they believe it is size $1/n$
- Last player to cut the piece gets to keep it

This method will give a fair and envy free distribution. In a round, in the eyes of all players the piece is of size $1/n$ at their round and any subsequent trimming is only reducing the size. For the last player (recieving player), the size of cake is exactly $1/n$, hence it is fair.

1.2. Social Choice Function

Definition 1.1. (Mechanism Design Setting) We define the setting as

- $N = \{1, 2, \ldots, n\}$ is the set of agents which are all rational and intelligent.
- X is the set of possible outcomes. Agents decide a collective choice from the set X
- Before making a collective choice, each agent privately observes his preferences over alternatives in X. This is modelled by supposing each agent i has a private value θ_i that determines his preferences and is private to player i
- Θ_i is the set of private values for agent i and $\Theta = \Theta_1 \times \cdots \times \Theta_n$ os set of all type profiles.
- $\mathbb{P} \in \Delta(\Theta)$ is a common prior. To ensure consistency of beliefs, individual belief functions $p_i : \Theta_i \mapsto \Delta(\Theta_{-i})$ can be all derived from common prior.
- Individual agents have preferences over the outcomes represented by utility function $u_i : X \times \Theta_i \mapsto \mathbb{R}$. Given $x \in X$ and $\theta_i \in \Theta_i$ the value $u_i(x, \theta_i)$ is the payoff for agent i. u_i can depend on types of other players as well, where $u_i : X \times \Theta \mapsto \mathbb{R}$.
- $X, N, \Theta_i, \mathbb{P}$ and u_i are assumed to be common knowledge among all players. The specific type θ_i is private information.

Definition 1.2. (Social Choice Function) Suppose $N = \{1, \ldots, n\}$ is a set of agents with type sets Θ_i for each i. given a set of outcomes X, a social choice function is a mapping $f : \Theta \mapsto X$. The outcome corresponding to a type profile is called the social choice.

Consider a social choice function $f : \Theta \mapsto X$, the types θ_i are private information.

Definition 1.3. (Preferance Elicitation Problem) For the social choice $f(\theta_1, \ldots, \theta_n)$ to be chosen when individual types are θ_i each agent must disclose true type to social planner. But all agents may not find it a best interest to reveal this. This is the *information relevation problem*

Definition 1.4. (Preference Aggregation Problem) On obtaining all the reported types, suppose θ_i is the true type and $\hat{\theta_i}$ is the reported type, then process of computing $f(\hat{\theta_1}, \ldots, \hat{\theta_n})$ is called the preference aggregation problem.

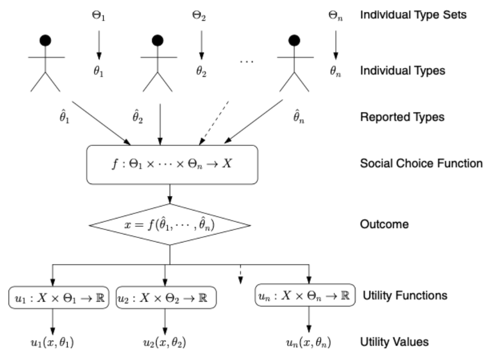

Figure 1: Mechanism Design Environment

Example 1.2. (Selling an Indivisible Object) Suppose there is 1 seller and n buyers, let sellet be agent 0 and the n buyers be ordered $1, 2, \dots, n$, hence $N = \{0, 1, \dots, n\}$. Then we can define the allocation

$$X = \{(y_0, y_1, \dots, y_n, t_0, t_1, \dots, t_n) | y_i \in \{0, 1\}; \sum_{i=0}^{n} y_i = 1; t_i \in R; i \in N\}$$

where y_i takes 1 if object allocated to player i and 0 otherwise. t_i is the payment received by agent i. For $x \in X$, we can define utility in a natural way as $u_i(x, \theta_i) = y_i\theta_i + t_i$, where $\theta_i \in \mathbb{R}_+$ can be viewed as agent i's valutation of the object. Such utility functions are called *quasi linear* as they are linear in some variables and possible non linear in other. Now we assume

- Seller has a known value θ_0, that is $\Theta_0 = \{\theta_0\}$
- $\Theta_i \subseteq \mathbb{R}_+$ is the set of all possible evaluations of buyer i, this can be viewed as the willingness to pay (above which buyer is not interested in paying)

Consider the Social Choice function

- Selling agent allocates the object to buyer with highest willingness to buy, in case multiple buyers, it is given to the agent with smallest i
- The allocated buyer i pays amount θ_i to selling agent.

Then the social choice function is $f(\theta) = \{y_0(\theta), \dots, y_n(\theta), t_0(\theta), \dots, t_n(\theta))$ which is

$$y_0 = 0$$

$$y_i(\theta) = \begin{cases} 1 & \text{if } \theta_i > \theta_j \ \forall j < i; \ \theta_i \geq \theta_j \ \forall j > i; \ i, j \in \{1, 2, \dots, n\} \\ 0 & \text{otherwise} \end{cases}$$

$$t_i(\theta) = -y_i(\theta)\theta_i \ \forall i \in \{1, \dots, n\}$$

$$t_0(\theta) = -\sum_{i=1}^{n} t_i(\theta)$$

2. Implementation of Social Choice Functions by Mechanisms

Mechanism design is an optimization problem where the specifications are first elicited and then the decision problem is solved. To elicit the type information in a truthful way there are 2 main approaches.

Definition 2.1. (Direct Mechanism) Suppose $f : \Theta \mapsto X$ is a social choice function. A direct mechanism (*direct revelation mechaism*) is the tuple $(\Theta_1, \dots, \Theta_n, f(.))$