Mechanism Design: Implementation

Game Theory Course:
Jackson, Leyton-Brown & Shoham

Bayesian Game Setting

- Extend the social choice setting to a new setting where agents can't be relied upon to disclose their preferences honestly.
- Start with a set of agents in a Bayesian game setting (but no actions).

Definition (Bayesian game setting)

A **Bayesian game setting** is a tuple (N, O, Θ, p, u), where

- N is a finite set of n agents;
- O is a set of outcomes;
- $\Theta = \Theta_1 \times \cdots \times \Theta_n$ is a set of possible joint type vectors;
- p is a (common prior) probability distribution on Θ; and
- $u = (u_1, \ldots, u_n)$, where $u_i : O \times \Theta \mapsto \mathbb{R}$ is the utility function for each player i.

Mechanism Design

Definition (Mechanism)

A mechanism (for a Bayesian game setting (N, O, Θ, p, u)) is a pair (A, M), where

- $A = A_1 \times \cdots \times A_n$, where A_i is the set of actions available to agent $i \in N$; and
- $M : A \mapsto \Pi(O)$ maps each action profile to a distribution over outcomes.

Thus, the designer gets to specify

- the action sets for the agents
- the mapping to outcomes, over which agents have utility
- can't change outcomes; agents' preferences or type spaces

What we're up to

- The problem is to pick a mechanism that will cause rational agents to behave in a desired way
  - each agent holds private information, in the Bayesian game sense
- Various equivalent ways of looking at this setting
  - perform an optimization problem, given that the values of (some of) the inputs are unknown
  - choose the Bayesian game out of a set of possible Bayesian games that maximizes some performance measure
  - design a game that *implements* a particular social choice function in equilibrium, given that the designer does not know agents' preferences and the agents might lie

Implementation in Dominant Strategies

Definition (Implementation in dominant strategies)

Given a Bayesian game setting (N, O, Θ, p, u), a mechanism (A, M) is an implementation in dominant strategies of a social choice function C (over N and O) if for any vector of utility functions u, the game has an equilibrium in dominant strategies, and in any such equilibrium a^* we have $M(a^*) = C(u)$.

Implementation in Bayes–Nash equilibrium

Definition (Bayes–Nash implementation)

Given a Bayesian game setting (N, O, Θ, p, u), a mechanism (A, M) is an implementation in Bayes–Nash equilibrium of a social choice function C (over N and O) if there exists a Bayes–Nash equilibrium of the game of incomplete information (N, A, Θ, p, u) such that for every $\theta \in \Theta$ and every action profile $a \in A$ that can arise given type profile θ in this equilibrium, we have that $M(a) = C(u(\cdot, \theta))$.

Bayes–Nash Implementation Comments

Bayes–Nash Equilibrium Problems:

- there could be more than one equilibrium
  - which one should I expect agents to play?
- agents could mis-coordinate and play none of the equilibria
- asymmetric equilibria are implausible

Refinements:

- Symmetric Bayes–Nash implementation
- *Ex-post* implementation

Implementation Comments

We can require that the desired outcome arises

- in the only equilibrium
- in every equilibrium
- in at least one equilibrium

Forms of implementation:

- Direct Implementation: agents each simultaneously send a single message to the center
- Indirect Implementation: agents send a sequence of messages; information may be (partially) revealed about the messages that were sent previously