Lecture 11.2

"Social Choice Functions" Implementable in Single Parameter Domain

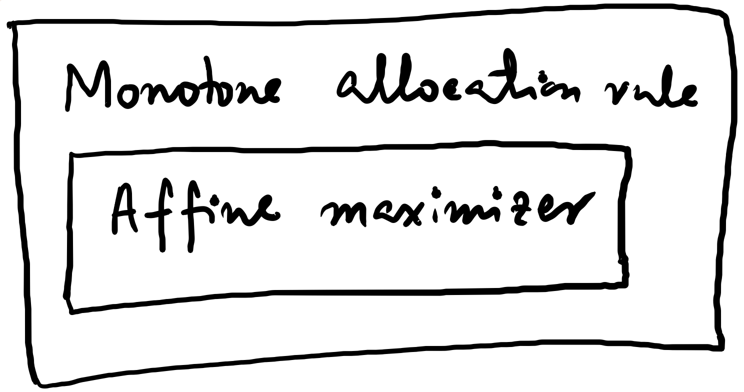
→ Allocation rules.

Monotone Allocation Rules in a Single Parameter Domain:

An allocation rule $k : \Theta \longrightarrow \mathcal{K}$ in a single parameter domain is called monotone in θ_i if, for every $\underline{\theta}_{-i} \in \Theta_{-i}$, $\theta_i, \theta_i' \in \Theta_i$, $\theta_i \leq \theta_i'$, $k(\theta_i, \underline{\theta}_{-i}) \in \mathcal{K}_i$, we have $k(\theta_i', \underline{\theta}_{-i}) \in \mathcal{K}_i$. That is, for a fixed type profile $\underline{\theta}_{-i}$

of other players, if player i wins with type θ_i, then player i continues to win with increase of its type.

- We will show that monotone allocation rules form the set of allocation rules implementable in a single parameter domain.



Critical Value Function :– Given an allocation rule in a single parameter domain, the critical value function $c_i(\theta_{-i})$ of player i is defined as

$$c_i(\theta_{-i}) = \sup\{\theta_i \in \Theta_i \mid k(\theta_i, \theta_{-i}) \notin K_i\} \in \mathbb{R}$$

$$c_i : \Theta_{-i} \longrightarrow \mathbb{R}$$

If for some $\theta_{-i} \in \Theta_{-i}$, the set $\{\theta_i \in \Theta_i \mid k(\theta_i, \theta_{-i}) \notin K_i\} = \phi$, then $c_i(\theta_{-i})$ is undefined.

Theorem: A social choice function $f(\cdot) = (k(\cdot), t_1(\cdot), \dots, t_n(\cdot))$ in a single parameter domain is dominant strategy incentive compatible and losers pay nothing if and only if the following conditions hold:

(i) The allocation rule $k(\cdot)$ is monotone.

(ii) Every winning player essentially pays its critical value.

That is,

$$t_i(\theta_i, \theta_{-i}) = \begin{cases} -c_i(\theta_{-i}) & \text{if } k(\theta_i, \theta_{-i}) \in K_i \\ 0 & \text{o/w.} \end{cases}$$

If $c_i(\theta_{-i})$ is undefined for some θ_{-i}, then we define $c_i(\theta_{-i}) = \lambda_i$ for any real number λ_i.

Proof: (If part) For a player $i \in [n]$, $\theta_i \in \Theta_i$, $\theta_{-i} \in \Theta_{-i}$, if player i wins, its utility is $\theta_i - c_i(\theta_{-i})$ and if player i loses then its utility is 0.

$$u_i(\theta_i, \theta_{-i}) = v_i(k(\theta_i, \theta_{-i}), \theta_i) + t_i(\theta_i, \theta_{-i})$$

<u>To show:</u>

$$v_i(k(\theta_i, \theta_{-i}), \theta_i) + t_i(\theta_i, \theta_{-i}) \geq v_i(k(\theta_i', \theta_{-i}), \theta_i) + t_i(\theta_i', \theta_{-i})$$

<u>case I:</u> $k(\theta_i, \theta_{-i}) \in R_i$

$$v_i(k(\theta_i, \theta_{-i}), \theta_i) + t_i(\theta_i, \theta_{-i}) = \theta_i - c_i(\theta_{-i}) \geq 0 \leftarrow$$

player prefers winning

if $k(\theta_i', \theta_{-i}) \in R_i$ ✓

if $k(\theta_i', \theta_{-i}) \notin R_i$

if and only if

$$\theta_i \geq c_i(\theta_{-i})$$