Quasi linear preferences
The SCF is decomposed into two components

Allocation rule component

\[ f : \Theta_1 \times \Theta_2 \times \ldots \times \Theta_n \rightarrow A \]

When the types are \( \Theta_i, i \in \mathbb{N} \), \( f(\Theta_1, \ldots, \Theta_n) = a \in A \)

Payment function

\[ p_i : \Theta_1 \times \Theta_2 \times \ldots \times \Theta_n \rightarrow \mathbb{R}, \forall i \in \mathbb{N} \]

When the types are \( \Theta_i, i \in \mathbb{N} \), \( p_i(\Theta_1, \ldots, \Theta_n) = \Pi_i \in \mathbb{R} \)

Examples of allocation rules

1. Constant rule, \( f^c(\Theta) = a \) for all \( \Theta \in \Theta \)

2. Dictatorial rule, \( f^d(\Theta) \in \arg \max_a \arg \max_{\Theta_d} (a, \Theta_d) \) for some \( d \in \mathbb{N} \)

3. Allocatively efficient rule/utilitarian rule

\[ f^{AE}(\Theta) \in \arg \max_{a \in A} \sum_{i \in \mathbb{N}} p_i(a, \Theta_i) \]

Note: this is different from Pareto efficiency (PE is a property defined for the outcome which also considers the payment)

4. Affine maximizer rule:

\[ f^{AM}(\Theta) \in \arg \max_{a \in A} \left( \sum_{i \in \mathbb{N}} \lambda_i p_i(a, \Theta_i) + K(a) \right), \lambda_i > 0, \text{not all zero.} \]
5. Max-min/egalitarian

\[ f^{\text{MM}}(\theta) \in \arg \max_{a \in A} \min_{i \in N} v_i(a, \theta_i) \]

Examples of payment rules

1. No deficit: \[ \sum_{i \in N} p_i(\theta) > 0 \quad \forall \theta \in \Theta. \]

2. No subsidy: \[ p_i(\theta) > 0, \quad \forall \theta \in \Theta, \quad \forall i \in N. \]

3. Budget balanced: \[ \sum_{i \in N} p_i(\theta) = 0, \quad \forall \theta \in \Theta. \]

Recall: Incentive Compatibility

A mechanism \((f, p)\) is incentive compatible (DSIC) if

\[ v_i(f(\theta_i, \tilde{\theta}_i), \theta_i) - p_i(\theta_i, \tilde{\theta}_i) \geq v_i(f(\theta_i, \tilde{\theta}_i), \theta_i) - p_i(\theta_i, \tilde{\theta}_i), \]

\[ \forall \theta_i \in \Theta_i, \quad \forall \theta_i, \tilde{\theta}_i \in \Theta_i. \]

DSIC means truth-telling is a weakly DSE.

We say that The payment rule \( p \) implements \( f \) in dominant strategies OR \( f \) is implementable in dominant strategies (by a payment rule).

In QL domain, we are often more interested in the allocation rule than the whole SCF (includes payment).
What needs to be satisfied for a DSIC mechanism \((f, p)\)?

\(N = \{1, 2\}, \Theta_1 = \Theta_2 = \{\Theta^H, \Theta^L\}, f: \Theta_1 \times \Theta_2 \rightarrow A\)

The following conditions must hold:

\[
\nu_i(f(\Theta^H, \Theta_2), \Theta^H) - p_1(\Theta^H, \Theta_2) \geq \nu_i(f(\Theta^L, \Theta_2), \Theta^H) - p_1(\Theta^L, \Theta_2), \forall \Theta_2
\]

\[
\nu_i(f(\Theta^H, \Theta_2), \Theta^L) - p_1(\Theta^H, \Theta_2) \geq \nu_i(f(\Theta^H, \Theta_2), \Theta^L) - p_1(\Theta^H, \Theta_2), \forall \Theta_2
\]

for player 2:

\[
\nu_2(f(\Theta_1, \Theta^H), \Theta^H) - p_2(\Theta_1, \Theta^H) \geq \nu_2(f(\Theta_1, \Theta^L), \Theta^H) - p_2(\Theta_1, \Theta^L), \forall \Theta_1
\]

\[
\nu_2(f(\Theta_1, \Theta^L), \Theta^L) - p_2(\Theta_1, \Theta^L) \geq \nu_2(f(\Theta_1, \Theta^H), \Theta^L) - p_2(\Theta_1, \Theta^H), \forall \Theta_1
\]

Properties of the payment that implements an allocation rule

1. Say \((f, p)\) is incentive compatible. Consider another payment

\[
q_i(\Theta_i, \Theta_i) = p_i(\Theta_i, \Theta_i) + h_i(\Theta_i) \quad \forall \Theta, \forall i \in N.
\]

Q: is \((f, q)\) DSIC?

A: Yes.

\[
\nu_i(f(\Theta_i, \Theta_i), \Theta_i) - p_i(\Theta_i, \Theta_i) - h_i(\Theta_i) \\
\geq \nu_i(f(\Theta_i, \Theta_i), \Theta_i) - p_i(\Theta_i, \Theta_i) - h_i(\Theta_i) \\
\quad \forall \Theta_i, \Theta_i', \Theta_i', \forall i \in N.
\]

If we can find a payment that implements an allocation rule, there exists uncountably many payments that can implement it.

The converse question: when do the payments that implement \(f\) differ only by a factor \(h_i(\Theta_i)\)?
2. Implication of incentive compatibility on payment

Suppose the allocation is same in two type profiles \( \theta \) and \( \tilde{\theta} = (\tilde{\theta}_i, \tilde{\theta}_j) \)

\[ f(\theta) = f(\tilde{\theta}) = a \]

then if \( P \) implements \( f \), then

\[ p_i(\theta) = p_i(\tilde{\theta}) \]  [exercise]