Mechanism Design with Monetary Transfers

Swaprava Nath

Economics and Planning Unit
Indian Statistical Institute, New Delhi

Workshop on Static and Dynamic Mechanism Design
Indian Statistical Institute, New Delhi

August 2, 2015

Outline of the Talk

Outline of the Talk

Outline of the Talk

The Gibbard-Satterthwaite Setting

- Voters can have arbitrary *strict ordinal* preferences over the set of alternatives
- Set of alternatives $X = \{a, b, c, d\}$

Voter 1	Voter 2	Voter 3	Voter 4
a	d	c	d
b	b	b	b
c	a	a	c
d	c	d	a

The Gibbard-Satterthwaite Setting

- Voters can have arbitrary *strict ordinal* preferences over the set of alternatives
- Set of alternatives $X = \{a, b, c, d\}$
- **Goal**: elicit the preferences truthfully from the agents

Voter 1	Voter 2	Voter 3	Voter 4
a	d	c	d
b	b	b	b
c	a	a	c
d	c	d	a

The Gibbard-Satterthwaite Setting

- Voters can have arbitrary *strict ordinal* preferences over the set of alternatives
- Set of alternatives $X = \{a, b, c, d\}$
- **Goal**: elicit the preferences truthfully from the agents

Voter 1	Voter 2	Voter 3	Voter 4
a	d	c	d
b	b	b	b
c	a	a	c
d	c	d	a

Theorem (Gibbard (1973), Satterthwaite (1975))

If $|X| \geq 3$, an onto social choice function *is strategyproof if and only if it is dictatorial.*

Outline of the Talk

1. The Setup
   - Unrestricted Preferences
   - Restricted Preferences
2. Mechanisms in Quasi-linear Domain
   - Structure of a Mechanism
   - Some Definitions
3. Results
   - Groves Class of Mechanisms
   - What Other Mechanisms are Incentive Compatible
   - Revenue Equivalence
   - Uniqueness of Groves for Efficiency
   - Budget Balance
   - Bayesian Incentive Compatibility
4. Summary

Quasi-linear Preferences

- An alternative $x \in X$ is a tuple (a, p)
- Allocation a belongs to the set of allocations A

Quasi-linear Preferences

- An alternative $x \in X$ is a tuple (a, p)
- Allocation a belongs to the set of allocations A
- Payments p belong to $\mathbb{R}^n$

Quasi-linear Preferences

- An alternative $x \in X$ is a tuple (a, p)
- Allocation a belongs to the set of allocations A
- Payments p belong to $\mathbb{R}^n$
- Each agent i has a valuation function $v_i : A \to \mathbb{R}$ belonging to the set V_i

Quasi-linear Preferences

- An alternative $x \in X$ is a tuple (a, p)
- Allocation a belongs to the set of allocations A
- Payments p belong to $\mathbb{R}^n$
- Each agent i has a valuation function $v_i : A \to \mathbb{R}$ belonging to the set V_i
- Agents' utilities are given by

$$u_i(x) = u_i(a, p) = v_i(a) - p_i$$

Example: Public Good

Alternatives →		
Alice	10	80
Bob	100	30
Carol	40	50

Photo courtesy: wikimedia.org and nimsuniversity.org

Example: Public Good

Alternatives →		
Alice	10	80
Bob	100	30
Carol	40	50

- Valuations: $v_A(F) = 10, v_A(L) = 80$
- Social planner takes the decision of building `F` or `L`
- Can *tax* people differently depending on their preferences

Quasi-linear preferences

Example: Resource Allocation

Commodities →	IBMSmartCloud	hp Cloud	cisco
Alice	0.2	0.8	0.5
Bob	0.3	0.1	0.2
Carol	0.5	0.1	0.3

Photo courtesy: individual organizations

Example: Resource Allocation

Commodities $\rightarrow$	IBMSmartCloud	hp Cloud	cisco
Alice	0.2	0.8	0.5
Bob	0.3	0.1	0.2
Carol	0.5	0.1	0.3

- Set of allocations $A = \{x \in [0,1]^{n \times m} : \sum_{j=1}^{m} x_{i,j} = 1\}$
- Items are divisible among the agents
- Agents' valuations reflect their preferences over different allocations
- They are charged monetary transfers for every allocation

Quasi-linear preferences

Example: Resource Allocation

Commodities →	IBMSmartCloud	hp Cloud	CISCO
Alice	0.2	0.8	0.5
Bob	0.3	0.1	0.2
Carol	0.5	0.1	0.3

Selfish valuations

Why Quasi-linearity avoids GS Impossibility

- GS theorem is valid for unrestricted preferences
- In quasi-linear domain, agents' preferences are restricted

Why Quasi-linearity avoids GS Impossibility

- GS theorem is valid for unrestricted preferences
- In quasi-linear domain, agents' preferences are restricted
- Example:
  - Set of alternatives $X = A \times \mathbb{R}^n$ consists of (a, p) pairs
  - Allocation $a \in A$ and payment vector $p \in \mathbb{R}^n$

Why Quasi-linearity avoids GS Impossibility

- GS theorem is valid for unrestricted preferences
- In quasi-linear domain, agents' preferences are restricted
- Example:
  - Set of alternatives $X = A \times \mathbb{R}^n$ consists of (a, p) pairs
  - Allocation $a \in A$ and payment vector $p \in \mathbb{R}^n$
  - Consider two alternatives $x_1 = (a, p_1)$ and $x_2 = (a, p_2)$, where $p_1 < p_2$

Why Quasi-linearity avoids GS Impossibility

- GS theorem is valid for unrestricted preferences
- In quasi-linear domain, agents' preferences are restricted
- Example:
  - Set of alternatives $X = A \times \mathbb{R}^n$ consists of (a, p) pairs
  - Allocation $a \in A$ and payment vector $p \in \mathbb{R}^n$
  - Consider two alternatives $x_1 = (a, p_1)$ and $x_2 = (a, p_2)$, where $p_1 < p_2$
  - For all agents, $x_1 \succ x_2$ for any valuation profile

Why Quasi-linearity avoids GS Impossibility

- GS theorem is valid for unrestricted preferences
- In quasi-linear domain, agents' preferences are restricted
- Example:
  - Set of alternatives $X = A \times \mathbb{R}^n$ consists of (a, p) pairs
  - Allocation $a \in A$ and payment vector $p \in \mathbb{R}^n$
  - Consider two alternatives $x_1 = (a, p_1)$ and $x_2 = (a, p_2)$, where $p_1 < p_2$
  - For all agents, $x_1 \succ x_2$ for any valuation profile
  - There is no preference profile where $x_2 \succ x_1$

An Example of a Truthful Mechanism

Alice	10	80	40
Bob	100	20	50
Carol	0	40	30

An Example of a Truthful Mechanism

Alice	10	80	40
Bob	100	20	50
Carol	0	40	30

- Consider the mechanism:
  - pick the alternative a^* that maximizes the sum of the valuations (with arbitrary tie-breaking rule)
  - pay every agent i an amount $\sum_{j \neq i} v_j(a^*)$

An Example of a Truthful Mechanism

Alice	10	80	40
Bob	100	20	50
Carol	0	40	30

- Consider the mechanism:
  - pick the alternative a^* that maximizes the sum of the valuations (with arbitrary tie-breaking rule)
  - pay every agent i an amount $\sum_{j \neq i} v_j(a^*)$
- The mechanism is truthful, even though not a dictatorship

Outline of the Talk

Outline of the Talk

Structure of a Mechanism

- Set of agents $N = \{1, \ldots, n\}$
- Set of allocations A, finite (for this tutorial)
- Valuation of agent i, $v_i : A \to \mathbb{R}$, the set of valuations is denoted by V_i

Structure of a Mechanism

- Set of agents $N = \{1, \dots, n\}$
- Set of allocations A, finite (for this tutorial)
- Valuation of agent i, $v_i : A \to \mathbb{R}$, the set of valuations is denoted by V_i
- A mechanism in quasi-linear (QL) domain is a pair of functions:
  - allocation function, $a : \prod_j V_j \to A$
  - payment function, $p_i : \prod_j V_j \to \mathbb{R}$, for all $i \in N$
- Agent i's payoff is given by:

$$v_i(a(v)) - p_i(v)$$

Structure of a Mechanism

- Set of agents $N = \{1, \ldots, n\}$
- Set of allocations A, finite (for this tutorial)
- Valuation of agent i, $v_i : A \to \mathbb{R}$, the set of valuations is denoted by V_i
- A mechanism in quasi-linear (QL) domain is a pair of functions:
  - allocation function, $a : \prod_j V_j \to A$
  - payment function, $p_i : \prod_j V_j \to \mathbb{R}$, for all $i \in N$
- Agent i's payoff is given by:

$$v_i(a(v)) - p_i(v)$$

- Only *direct revelation mechanisms* (DRM) **(this talk)**

Outline of the Talk

Social Choice Function

Definition (Social Choice Function)

A *social choice function* (SCF) f is a mapping from the set of valuation profiles to the set of allocations, i.e., $f : V \to A$, where $V = \prod_j V_j$.

- Note that the outcome is only the allocations

Social Choice Function

Definition (Social Choice Function)

A *social choice function* (SCF) f is a mapping from the set of valuation profiles to the set of allocations, i.e., $f : V \to A$, where $V = \prod_j V_j$.

- Note that the outcome is only the allocations
- In QL domain:
  A mechanism $M = (a, p)$ *implements* a SCF f if:
  - $a(v) = f(v), \forall v \in V$ and,
  - for every agent $i \in N$, reporting v_i truthfully is an *equilibrium*
- Even though the SCF is concerned with only allocations, payments can also be characterized by *revenue equivalence* (defined later)

Incentive Compatibility

Definition (Dominant Strategy Incentive Compatibility (DSIC))

A mechanism $(f, p_1, \ldots, p_n)$ is *dominant strategy incentive compatible* if for all $i \in N$ and for all $v_{-i} \in V_{-i} := \prod_{j \neq i} V_j$,

$$v_i(f(v_i, v_{-i})) - p_i(v_i, v_{-i}) \geq v_i(f(v_i', v_{-i})) - p_i(v_i', v_{-i}),\ \forall v_i, v_i' \in V_i.$$

In this case, payments $p_i, i \in N$ implement f in dominant strategies

Incentive Compatibility (Contd.)

- In a Bayesian game, the valuations v are generated through a prior P
- Each agent i knows her own realized valuation v_i and P
- Her belief on the valuations of other agents v_{-i} is given by $P(v_{-i}|v_i)$ derived by Baye's rule

Incentive Compatibility (Contd.)

- In a Bayesian game, the valuations v are generated through a prior P
- Each agent i knows her own realized valuation v_i and P
- Her belief on the valuations of other agents v_{-i} is given by $P(v_{-i}|v_i)$ derived by Baye's rule

Definition (Bayesian Incentive Compatibility (BIC))

A mechanism $(f, p_1, \ldots, p_n)$ is *Bayesian incentive compatible* for a prior P if for all $i \in N$,

$$\mathbb{E}_{v_{-i}|v_i}[v_i(f(v_i, v_{-i})) - p_i(v_i, v_{-i})] \geq \mathbb{E}_{v_{-i}|v_i}[v_i(f(v_i', v_{-i})) - p_i(v_i', v_{-i})] \quad \forall v_i, v_i' \in V_i.$$

In this case, payments $p_i, i \in N$ implement f in a Bayesian Nash equilibrium

Observations on IC

- A DSIC mechanism is always BIC

Observations on IC

- A DSIC mechanism is always BIC

For a DSIC mechanism $(f, p_1, \ldots, p_n)$, let valuations of agents other than i is fixed at v_{-i}

- If v_i, v_i' be such that $f(v_i, v_{-i}) = f(v_i', v_{-i})$, then $p_i(v_i, v_{-i}) = p_i(v_i', v_{-i})$

Observations on IC

- A DSIC mechanism is always BIC

For a DSIC mechanism $(f, p_1, \dots, p_n)$, let valuations of agents other than i is fixed at v_{-i}

- If v_i, v_i' be such that $f(v_i, v_{-i}) = f(v_i', v_{-i})$, then $p_i(v_i, v_{-i}) = p_i(v_i', v_{-i})$
- Consider another payment $q_i(v_i, v_{-i}) = p_i(v_i, v_{-i}) + h_i(v_{-i})$,

$$v_i(f(v_i, v_{-i})) - q_i(v_i, v_{-i}) \geq v_i(f(v_i', v_{-i})) - q_i(v_i', v_{-i}), \ \forall v_i, v_i' \in V_i.$$

Efficiency

Definition (Efficiency)

An SCF f is *efficient* if for all $v \in V$,

$$f(v) \in \underset{a \in A}{\operatorname{argmax}} \sum_{i \in N} v_i(a).$$

An efficient SCF ensures that the 'social welfare' is maximized

Revenue Equivalence

- This property characterizes the payment functions

Definition (Revenue Equivalence)

An SCF f satisfies *revenue equivalence* if for any two payment rules p and p' that implement f, there exist functions $\alpha_i : V_{-i} \to \mathbb{R}$, such that,

$$p_i(v_i, v_{-i}) = p'_i(v_i, v_{-i}) + \alpha_i(v_{-i}), \ \forall v_i \in V_i, \forall v_{-i} \in V_{-i}, \forall i \in N.$$

Revenue Equivalence

- This property characterizes the payment functions

Definition (Revenue Equivalence)

An SCF f satisfies *revenue equivalence* if for any two payment rules p and p' that implement f, there exist functions $\alpha_i : V_{-i} \to \mathbb{R}$, such that,

$$p_i(v_i, v_{-i}) = p'_i(v_i, v_{-i}) + \alpha_i(v_{-i}), \ \forall v_i \in V_i, \forall v_{-i} \in V_{-i}, \forall i \in N.$$

- Saw an example of a payment of agent i being different by a factor not dependent on i's valuation
- This property says more: pick *any* two payments that implement f - they must be different by a similar factor

Budget Balance

Definition (Budget Balance)

A set of payments $p_i : V \to \mathbb{R}, i \in N$ is budget balanced if,

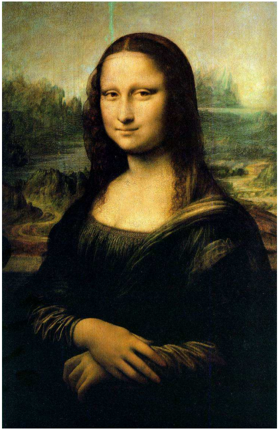
$$\sum_{i \in N} p_i(v) = 0, \forall v \in V.$$

- This property ensures that the mechanism does not produce any monetary surplus
- Hard to satisfy with incentive compatibility

Outline of the Talk

Outline of the Talk

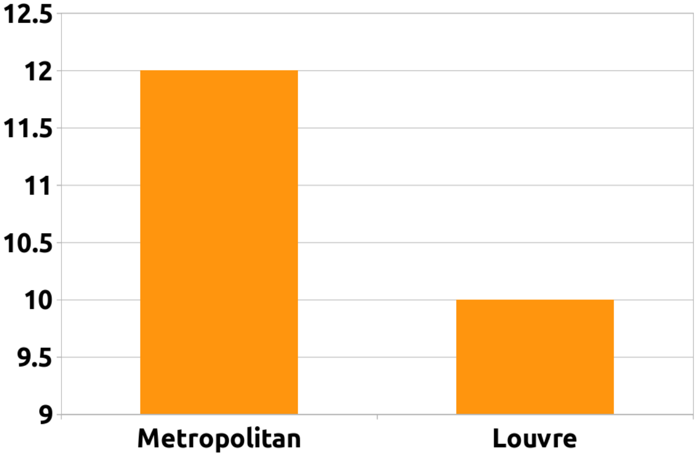
Single Indivisible Item Auction



Buyer 1
Metropolitan Museum of Arts

Buyer 2
Louvre

Second Price Auction



- Metropolitan wins, but pays second highest bid
- The mechanism is DSIC (why?)

Groves Class of Mechanisms

- Allocation rule is efficient:

$$a^*(v) \in \operatorname*{argmax}_{a \in A} \sum_{i \in N} v_i(a)$$

- Payment rule is given by:

$$p_i^*(v_i, v_{-i}) = h_i(v_{-i}) - \sum_{j \in N \setminus \{i\}} v_j(a^*(v)),$$

where $h_i : V_{-i} \to \mathbb{R}$ is any arbitrary function that does not depend on v_i

Groves Class of Mechanisms

- Allocation rule is efficient:

$$a^*(v) \in \underset{a \in A}{\text{argmax}} \sum_{i \in N} v_i(a)$$

- Payment rule is given by:

$$p_i^*(v_i, v_{-i}) = h_i(v_{-i}) - \sum_{j \in N \setminus \{i\}} v_j(a^*(v)),$$

where $h_i : V_{-i} \to \mathbb{R}$ is any arbitrary function that does not depend on v_i

Claim

Groves class of mechanisms are DSIC

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i})
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i}))
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i})) \\
& = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - h_i(v_{-i})
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i})) \\
& = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - h_i(v_{-i}) \\
& \geq \sum_{j \in N} v_j(a^*(v_i', v_{-i})) - h_i(v_{-i}) \text{ (by definition of } a^*\text{)}
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i})) \\
& = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - h_i(v_{-i}) \\
& \geq \sum_{j \in N} v_j(a^*(v_i', v_{-i})) - h_i(v_{-i}) \text{ (by definition of } a^*) \\
& = v_i(a^*(v_i', v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i', v_{-i}))
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i})) \\
& = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - h_i(v_{-i}) \\
& \geq \sum_{j \in N} v_j(a^*(v_i', v_{-i})) - h_i(v_{-i}) \text{ (by definition of } a^*) \\
& = v_i(a^*(v_i', v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i', v_{-i})) \\
& = v_i(a^*(v_i', v_{-i})) - p_i^*(v_i', v_{-i})
\end{aligned}
$$

Incentive Compatibility of Groves

- Utility of agent i according to Groves class of mechanisms:

$$
\begin{aligned}
& u_i^{(a^*,p^*)}(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - p_i^*(v_i, v_{-i}) \\
& = v_i(a^*(v_i, v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i, v_{-i})) \\
& = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - h_i(v_{-i}) \\
& \geq \sum_{j \in N} v_j(a^*(v_i', v_{-i})) - h_i(v_{-i}) \text{ (by definition of } a^*) \\
& = v_i(a^*(v_i', v_{-i})) - h_i(v_{-i}) + \sum_{j \in N \setminus \{i\}} v_j(a^*(v_i', v_{-i})) \\
& = v_i(a^*(v_i', v_{-i})) - p_i^*(v_i', v_{-i}) \\
& = u_i^{(a^*,p^*)}(v_i', v_{-i})
\end{aligned}
$$

Pivot Mechanism

- A special case of Groves class when the payment is given by:

$$h_i(v_{-i}) = \sum_{j \in N \setminus \{i\}} v_j(a^*_{-i}(v_{-i})),$$

where the allocation $a^*_{-i}(v_{-i})$ is given by:

$$a^*_{-i}(v_{-i}) \in \operatorname*{argmax}_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a)$$

- The allocation a^*_{-i} maximizes the sum of valuations in the absence of agent i
- The function h_i is the maximum value of this sum
- The payment is therefore:

$$p_i(v_i, v_{-i}) = \max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a) - \sum_{j \in N \setminus \{i\}} v_j(a^*(v))$$

Interpretations of the Pivot Mechanism

$$p_i(v_i, v_{-i}) = \max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a) - \sum_{j \in N \setminus \{i\}} v_j(a^*(v))$$

Two Interpretations:

1. Externality:
   - $\max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a)$ is what the agents $N \setminus \{i\}$ can achieve
   - $\sum_{j \in N \setminus \{i\}} v_j(a^*(v))$ is what they achieve under the efficient rule
   - The mechanism asks agent i to pay the difference

Interpretations of the Pivot Mechanism

$$p_i(v_i, v_{-i}) = \max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a) - \sum_{j \in N \setminus \{i\}} v_j(a^*(v))$$

Two Interpretations:

1. Externality:
   - $\max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a)$ is what the agents $N \setminus \{i\}$ can achieve
   - $\sum_{j \in N \setminus \{i\}} v_j(a^*(v))$ is what they achieve under the efficient rule
   - The mechanism asks agent i to pay the difference
2. Marginal contribution:
   - Net utility of agent i in pivot mechanism:

     $$u_i(v_i, v_{-i}) = \sum_{j \in N} v_j(a^*(v_i, v_{-i})) - \max_{a \in A} \sum_{j \in N \setminus \{i\}} v_j(a)$$

     i.e., the difference in sum valuation in presence of agent i and in her absence
   - Net utility is agent i's *marginal contribution*

What is Pivotal about it?

Alternatives →		
Alice	10	70
Bob	100	10
Carol	10	50

What is Pivotal about it?

Alternatives →		
Alice	10	70
Bob	100	10
Carol	10	50

- Outcome: `L`

What is Pivotal about it?

Alternatives →		
~~Alice~~	~~10~~	~~70~~
Bob	100	10
Carol	10	50

- Outcome: `L`
- Alice pays $(100 + 10) - (10 + 50) = 50$

What is Pivotal about it?

Alternatives →		
Alice	10	70
Bob	100	10
~~Carol~~	~~10~~	~~50~~

- Outcome: `L`
- Alice pays $(100 + 10) - (10 + 50) = 50$
- Carol pays $(10 + 100) - (70 + 10) = 30$

What is Pivotal about it?

Alternatives $\rightarrow$		
Alice	10	70
~~Bob~~	~~100~~	~~10~~
Carol	10	50

- Outcome: `L`
- Alice pays $(100 + 10) - (10 + 50) = 50$
- Bob pays $(70 + 50) - (70 + 50) = 0$
- Carol pays $(10 + 100) - (70 + 10) = 30$

What is Pivotal about it?

Alternatives →		
Alice	10	70
Bob	100	10
Carol	10	50

- Outcome: L
- Alice pays $(100 + 10) - (10 + 50) = 50$
- Bob pays $(70 + 50) - (70 + 50) = 0$
- Carol pays $(10 + 100) - (70 + 10) = 30$

Outline of the Talk

Affine Maximizers

An important class of SCFs is that of affine maximizers

Definition (Affine Maximizer)

An SCF $f : V \to A$ is an *affine maximizer* if there exists $w_i \geq 0, i \in N$, not all zero, and a function $\kappa : A \to \mathbb{R}$ such that,

$$f(v) \in \underset{a \in A}{\operatorname{argmax}} \left(\sum_{i \in N} w_i v_i(a) + \kappa(a) \right).$$

Affine Maximizers

An important class of SCFs is that of affine maximizers

Definition (Affine Maximizer)

An SCF $f : V \to A$ is an *affine maximizer* if there exists $w_i \geq 0, i \in N$, not all zero, and a function $\kappa : A \to \mathbb{R}$ such that,

$$f(v) \in \operatorname*{argmax}_{a \in A} \left(\sum_{i \in N} w_i v_i(a) + \kappa(a) \right).$$

Special cases:

- $w_i = 1, \forall\ i$ and $\kappa \equiv 0$: **efficient** SCF

Affine Maximizers

An important class of SCFs is that of affine maximizers

Definition (Affine Maximizer)

An SCF $f : V \to A$ is an *affine maximizer* if there exists $w_i \geq 0, i \in N$, not all zero, and a function $\kappa : A \to \mathbb{R}$ such that,

$$f(v) \in \underset{a \in A}{\operatorname{argmax}} \left(\sum_{i \in N} w_i v_i(a) + \kappa(a) \right).$$

Special cases:

- $w_i = 1, \forall\ i$ and $\kappa \equiv 0$: **efficient** SCF
- $w_d = 1$, for some d, $w_i = 0, \forall\ i \neq d$ and $\kappa \equiv 0$: **dictatorial** SCF

Affine Maximizers (Contd.)

- An affine maximizer f satisfies *independence of irrelevant agents* (IIA) if for every i with $w_i = 0$ and for every $v_{-i} \in V_{-i}$,

$$f(v_i, v_{-i}) = f(v_i', v_{-i}), \forall v_i, v_i' \in V_i$$

- This is a consistency condition for tie-breaking

Affine Maximizers (Contd.)

- An affine maximizer f satisfies *independence of irrelevant agents* (IIA) if for every i with $w_i = 0$ and for every $v_{-i} \in V_{-i}$,

$$f(v_i, v_{-i}) = f(v_i', v_{-i}), \forall v_i, v_i' \in V_i$$

- This is a consistency condition for tie-breaking
- Every affine maximizer satisfying IIA is implementable

Affine Maximizers (Contd.)

- An affine maximizer f satisfies *independence of irrelevant agents* (IIA) if for every i with $w_i = 0$ and for every $v_{-i} \in V_{-i}$,

$$f(v_i, v_{-i}) = f(v_i', v_{-i}), \forall v_i, v_i' \in V_i$$

- This is a consistency condition for tie-breaking
- Every affine maximizer satisfying IIA is implementable
- In particular, payments are of the following form: for all $i \in N$

$$p_i(v_i, v_{-i}) = \begin{cases} \frac{1}{w_i} \left(\sum_{j \neq i} w_j v_j(f(v)) + \kappa(f(v)) + h_i(v_{-i}) \right), & w_i > 0 \\ 0 & w_i = 0 \end{cases}$$

f is an affine maximizer

Roberts' Theorem

Theorem (Roberts 1979)

Let the allocation space A be finite with $|A| \geq 3$. If the space of valuations V is unrestricted, *then an onto and dominant strategy implementable SCF $f : V \to A$ is an affine maximizer.*

Roberts' Theorem

Theorem (Roberts 1979)

Let the allocation space A be finite with $|A| \geq 3$. If the space of valuations V is unrestricted, *then an onto and dominant strategy implementable SCF $f : V \to A$ is an affine maximizer.*

Understanding Roberts' Theorem:

- Groves' or pivotal mechanisms are implementable, but this result is giving a necessary condition for implementability

Roberts' Theorem

Theorem (Roberts 1979)

Let the allocation space A be finite with $|A| \geq 3$. If the space of valuations V is unrestricted, *then an onto and dominant strategy implementable SCF $f : V \rightarrow A$ is an affine maximizer.*

Understanding Roberts' Theorem:

- Groves' or pivotal mechanisms are implementable, but this result is giving a necessary condition for implementability
- Moreover, it provides a functional form characterization of the DSIC mechanisms (as opposed to Myerson's monotonicity characterization)

Roberts' Theorem

Theorem (Roberts 1979)

Let the allocation space A be finite with $|A| \geq 3$. If the space of valuations V is unrestricted, *then an onto and dominant strategy implementable SCF $f : V \to A$ is an affine maximizer.*

Understanding Roberts' Theorem:

- Groves' or pivotal mechanisms are implementable, but this result is giving a necessary condition for implementability
- Moreover, it provides a functional form characterization of the DSIC mechanisms (as opposed to Myerson's monotonicity characterization)
- If payments are enforced to be zero for every valuation profile v, then the only implementable mechanism is dictatorial - GS theorem is a corollary of this result

Some Observations and Implications

- If an SCF f is implementable in a valuation space V, it is implementable in every valuation space $V' \subseteq V$ - same payments implement them and the number of incentive compatibility constraints reduce

Some Observations and Implications

- If an SCF f is implementable in a valuation space V, it is implementable in every valuation space $V' \subseteq V$ - same payments implement them and the number of incentive compatibility constraints reduce
- Efficient SCF is implementable in any valuation space

Some Observations and Implications

- If an SCF f is implementable in a valuation space V, it is implementable in every valuation space $V' \subseteq V$ - same payments implement them and the number of incentive compatibility constraints reduce
- Efficient SCF is implementable in any valuation space
- Unrestricted valuation space is crucial for Roberts' theorem - some recent results show that the affine maximizer characterization is true even for certain restricted valuation spaces

Some Observations and Implications

- If an SCF f is implementable in a valuation space V, it is implementable in every valuation space $V' \subseteq V$ - same payments implement them and the number of incentive compatibility constraints reduce
- Efficient SCF is implementable in any valuation space
- Unrestricted valuation space is crucial for Roberts' theorem - some recent results show that the affine maximizer characterization is true even for certain restricted valuation spaces
- Characterization of implementability in restricted domains is an active research area

[A proof by pictures]

Outline of the Talk

Revenue Equivalence

- If p and p' implement f in dominant strategies, then

$$p_i(v) = p'_i(v) + \alpha_i(v_{-i}), \forall v \in V$$

Revenue Equivalence

- If p and p' implement f in dominant strategies, then

$$p_i(v) = p'_i(v) + \alpha_i(v_{-i}), \forall v \in V$$

Theorem (Rockafeller 1997; Krishna and Maenner (2001))

If the type space is convex and the valuations are linear in type, then an SCF, implementable in dominant strategies, satisfies revenue equivalence.

Revenue Equivalence

- If p and p' implement f in dominant strategies, then

$$p_i(v) = p'_i(v) + \alpha_i(v_{-i}), \forall v \in V$$

Theorem (Rockafeller 1997; Krishna and Maenner (2001))

If the type space is convex and the valuations are linear in type, then an SCF, implementable in dominant strategies, satisfies revenue equivalence.

Theorem (Chung and Olszewski (2007))

Suppose the type space $T \subseteq \mathbb{R}^n$ is a connected set, A is finite and the valuations are continuous in type. If an SCF is implementable in dominant strategies, then it satisfies revenue equivalence.

Outline of the Talk

Green-Laffont-Holmström Characterization

- An efficient SCF f chooses an alternative in $\text{argmax}_{a \in A} \sum_{j \in N} v_j(a)$

Green-Laffont-Holmström Characterization

- An efficient SCF f chooses an alternative in $\text{argmax}_{a \in A} \sum_{j \in N} v_j(a)$

Theorem (Green and Laffont (1979), Holmström (1979))

If the valuation space is convex and smoothly connected, every efficient and DSIC mechanism is a Groves mechanism.

- Shows uniqueness of Groves class in the space of efficient, DSIC mechanisms

[A proof outline]

Outline of the Talk

Green-Laffont Impossibility

Theorem (Green and Laffont (1979))

No Groves mechanism is budget balanced *(BB), i.e.,*
$\nexists p_i^{\textsf{Groves}} s.t. \sum_{i \in N} p_i^{\textsf{Groves}}(v) = 0, \forall v \in V.$

- This leads to the following corollary

Corollary

If the valuation space is convex and smoothly connected, no efficient *mechanism can be both DSIC and BB.*

Outline of the Talk

AGV Mechanism

- If the equilibrium condition is relaxed to BIC, we have a positive result
- Payment is defined via a function $\delta_i, i \in N$:

$$\delta_i(v_i) = \mathbb{E}_{v_{-i}|v_i} \left(\sum_{j \in N \setminus \{i\}} v_j(a^*(v)) \right),$$

  where a^* is an efficient allocation
- Payment is:

$$p_i^{\text{AGV}}(v) = \sum_{j \in N \setminus \{i\}} \delta_j(v_j) - \delta_i(v_i)$$

Theorem (d'Aspremont and Gerard-Varet (1979), Arrow (1979))

The AGV mechanism is BIC, efficient, and budget-balanced

Outline of the Talk

1. The Setup
   - Unrestricted Preferences
   - Restricted Preferences
2. Mechanisms in Quasi-linear Domain
   - Structure of a Mechanism
   - Some Definitions
3. Results
   - Groves Class of Mechanisms
   - What Other Mechanisms are Incentive Compatible
   - Revenue Equivalence
   - Uniqueness of Groves for Efficiency
   - Budget Balance
   - Bayesian Incentive Compatibility
4. Summary

Summary

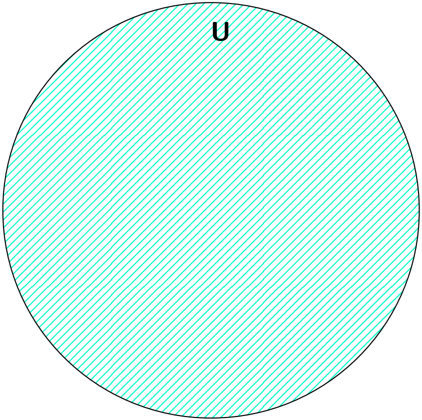
DSIC mechanisms

Valuation / Type Space

Mechanism Space

Summary

DSIC mechanisms

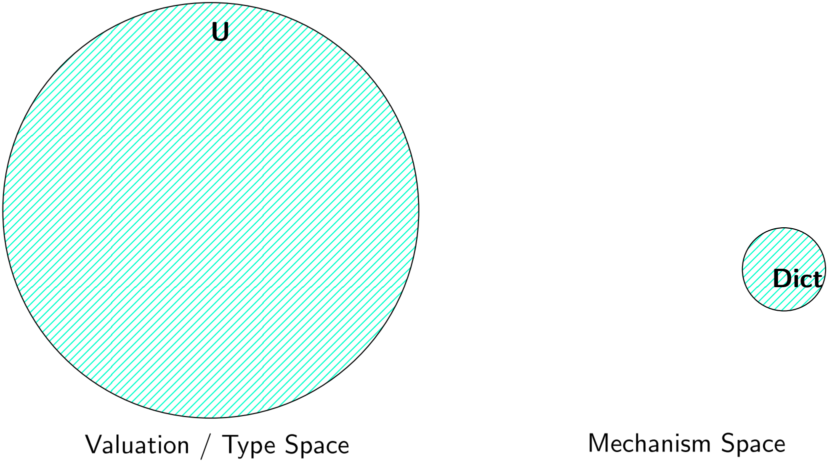


Valuation / Type Space

Mechanism Space

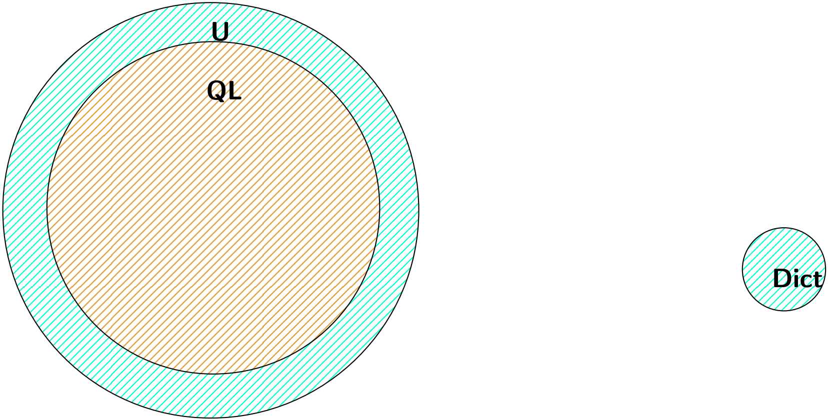
Summary

DSIC mechanisms



Summary

DSIC mechanisms

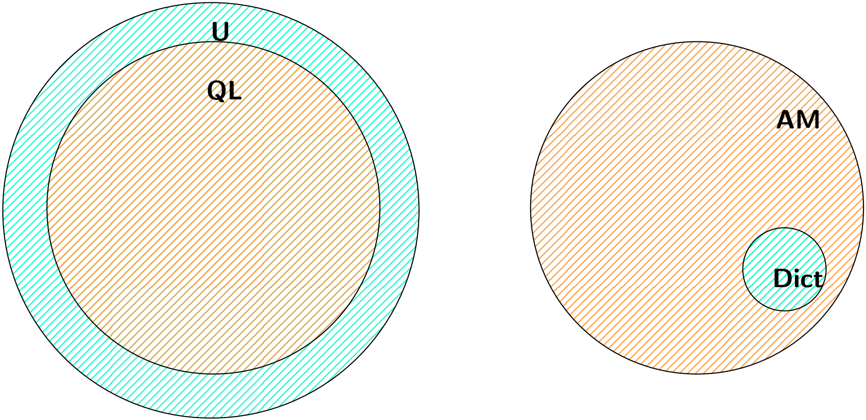


Valuation / Type Space

Mechanism Space

Summary

DSIC mechanisms

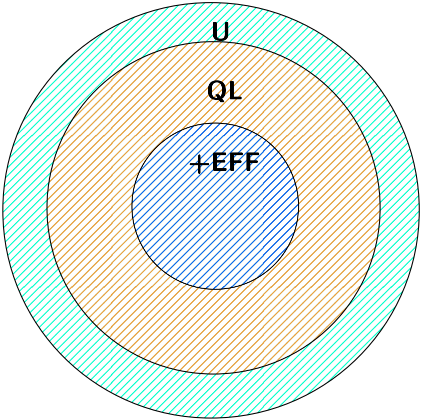


Valuation / Type Space

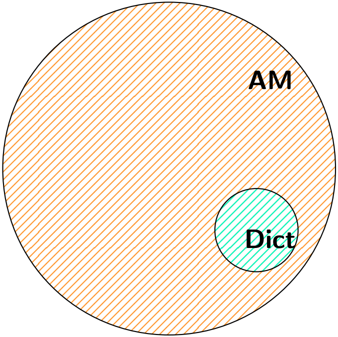
Mechanism Space

Summary

DSIC mechanisms



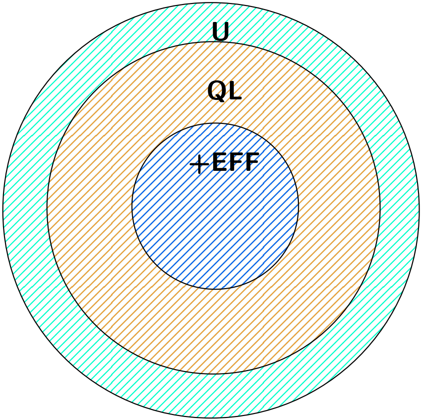
Valuation / Type Space



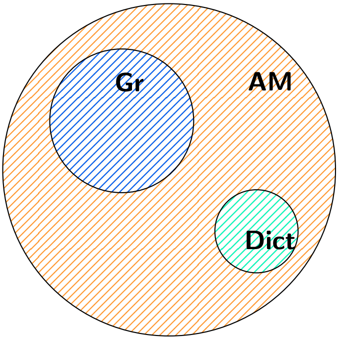
Mechanism Space

Summary

DSIC mechanisms



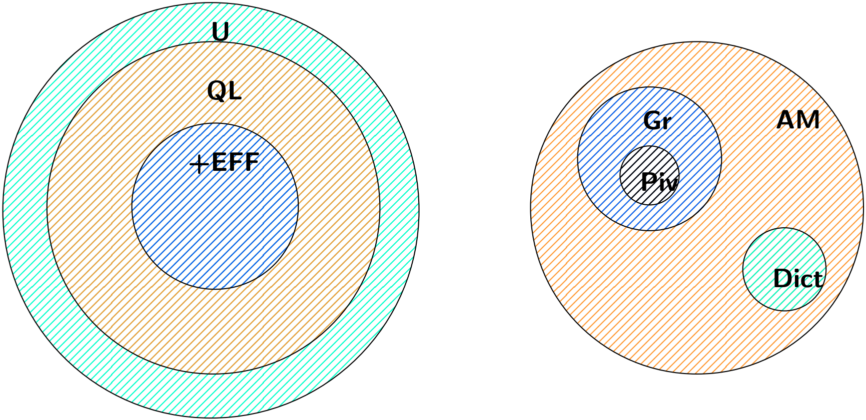
Valuation / Type Space



Mechanism Space

Summary

DSIC mechanisms

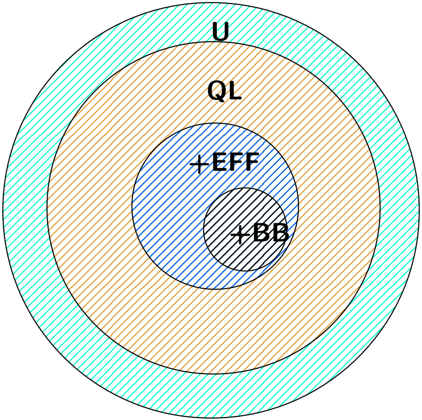


Valuation / Type Space

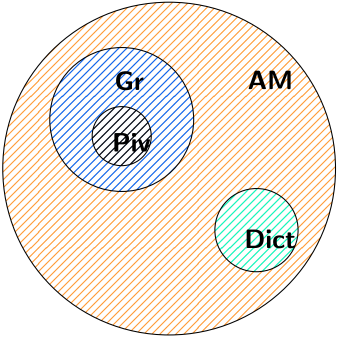
Mechanism Space

Summary

DSIC mechanisms



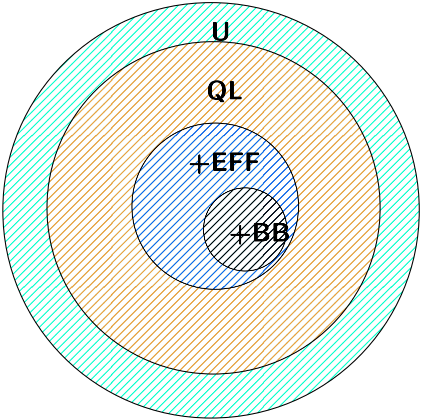
Valuation / Type Space



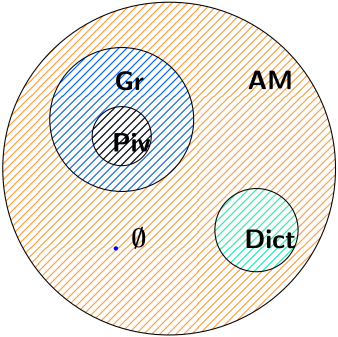
Mechanism Space

Summary

DSIC mechanisms



Valuation / Type Space



Mechanism Space

Thank you!

✉ swaprava@gmail.com

http://www.isid.ac.in/∼swaprava

Value Difference Set

Q: What does affine maximizer mean?

Value Difference Set

Q: What does affine maximizer mean?
A: If $f(v) = y$ then

$$w^\top v(y) + \kappa(y) \geq w^\top v(z) + \kappa(z), \forall z \in A \setminus \{y\}$$
$$\Rightarrow w^\top (v(y) - v(z)) \geq \kappa(z) - \kappa(y), \forall z \in A \setminus \{y\}$$

Value Difference Set

Q: What does affine maximizer mean?
A: If $f(v) = y$ then

$$
\begin{aligned}
w^\top v(y) + \kappa(y) &\geq w^\top v(z) + \kappa(z), \forall z \in A \setminus \{y\} \\
\Rightarrow w^\top (v(y) - v(z)) &\geq \kappa(z) - \kappa(y), \forall z \in A \setminus \{y\} \\
w^\top \alpha &\geq \beta \qquad \textbf{half-space}
\end{aligned}
$$

Value Difference Set

Q: What does affine maximizer mean?
A: If $f(v) = y$ then

$$w^\top v(y) + \kappa(y) \geq w^\top v(z) + \kappa(z), \forall z \in A \setminus \{y\}$$
$$\Rightarrow w^\top (v(y) - v(z)) \geq \kappa(z) - \kappa(y), \forall z \in A \setminus \{y\}$$
$$w^\top \alpha \geq \beta \qquad \textbf{half-space}$$

- Define the *value difference set* for any pair of *distinct* alternatives $y, z \in A$.

$$P(y, z) = \{\alpha \in \mathbb{R}^n : \exists\ v \in V \text{ s.t. } v(y) - v(z) = \alpha \text{ and } f(v) = y\}.$$

Value Difference Set

Q: What does affine maximizer mean?
A: If $f(v) = y$ then

$$\begin{aligned} w^\top v(y) + \kappa(y) &\geq w^\top v(z) + \kappa(z), \forall z \in A \setminus \{y\} \\ \Rightarrow w^\top (v(y) - v(z)) &\geq \kappa(z) - \kappa(y), \forall z \in A \setminus \{y\} \\ w^\top \alpha &\geq \beta \qquad \textbf{half-space} \end{aligned}$$

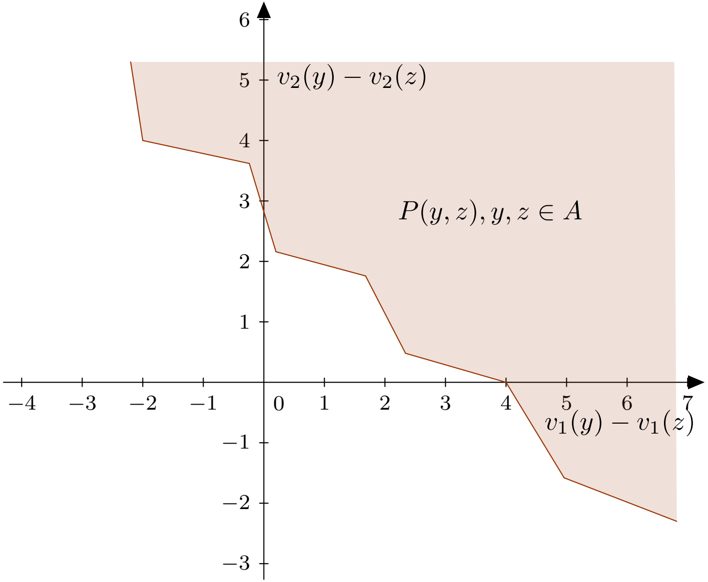
- Define the *value difference set* for any pair of *distinct* alternatives $y, z \in A$.

$$P(y,z) = \{\alpha \in \mathbb{R}^n : \exists\ v \in V \text{ s.t. } v(y) - v(z) = \alpha \text{ and } f(v) = y\}.$$

Claim

If $\alpha \in P(y,z)$, and $\delta > \mathbf{0} \in \mathbb{R}^n$, then $\alpha + \delta \in P(y,z)$, for all distinct $y, z \in A$.

Graphical Illustration for Two Players



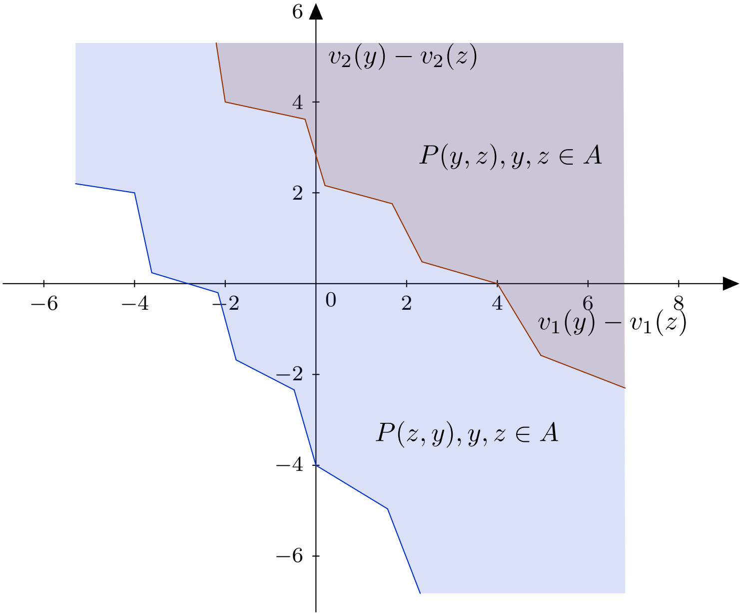
Complementary Structures of $P(y, z)$ and $P(z, y)$

Claim

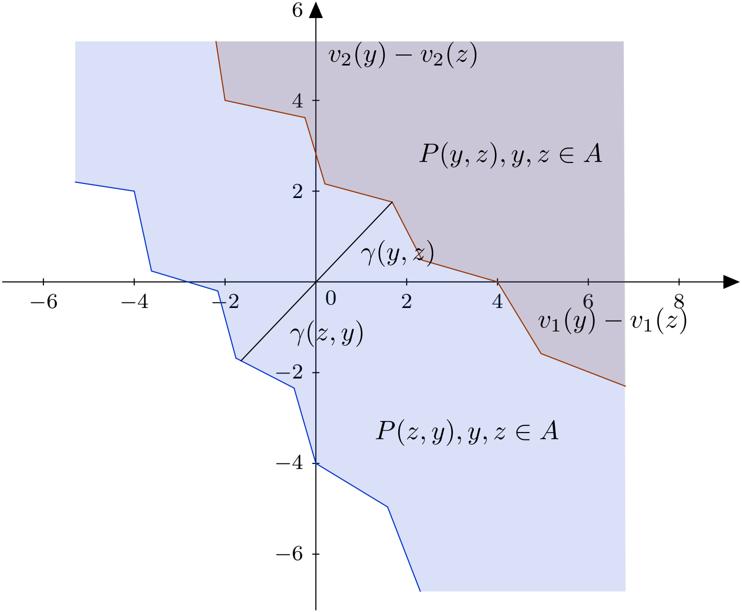
For every $\alpha, \epsilon \in \mathbb{R}^n,\ \epsilon > \mathbf{0}$*, and for all* $y, z \in A$*,*

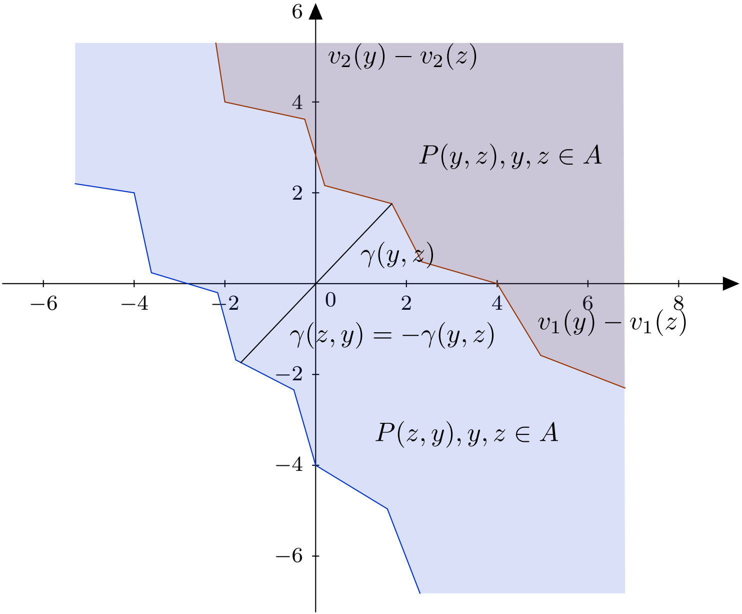
(a) $\alpha - \epsilon \in P(y, z) \quad \Rightarrow \quad -\alpha \notin P(z, y)$.

(b) $\alpha \notin P(y, z) \quad \Rightarrow \quad -\alpha \in P(z, y)$.



6
4
2
0
−2
−4
−6
−6
−4
−2
2
4
6
8
$v_2(y) - v_2(z)$
$v_1(y) - v_1(z)$
$P(y, z), y, z \in A$
$P(z, y), y, z \in A$


6
4
2
−2
−4
−6
−6
−4
−2
0
2
4
6
8
$v_2(y) - v_2(z)$
$v_1(y) - v_1(z)$
$P(y, z), y, z \in A$
$P(z, y), y, z \in A$
$\gamma(y, z)$
$\gamma(z, y)$



6
4
2
−2
−4
−6
−6
−4
−2
0
2
4
6
8
$v_2(y) - v_2(z)$
$P(y,z), y,z \in A$
$\gamma(y,z)$
$v_1(y) - v_1(z)$
$\gamma(z,y) = -\gamma(y,z)$
$P(z,y), y,z \in A$

Independence of $\mathring{C}$ from the Alternatives in A

- Define the translated set $C(y,z) = P(y,z) - \gamma(y,z)\mathbf{1}$
- Denote the 'interior' of $C(y,z)$ by $\mathring{C}(y,z)$

Independence of $\mathring{C}$ from the Alternatives in A

- Define the translated set $C(y,z) = P(y,z) - \gamma(y,z)\mathbf{1}$
- Denote the 'interior' of $C(y,z)$ by $\mathring{C}(y,z)$

Claim

$\mathring{C}(y,z) = \mathring{C}(w,l)$, *for any* $y,z,w,l \in A$, $y \neq z$ *and* $w \neq l$.

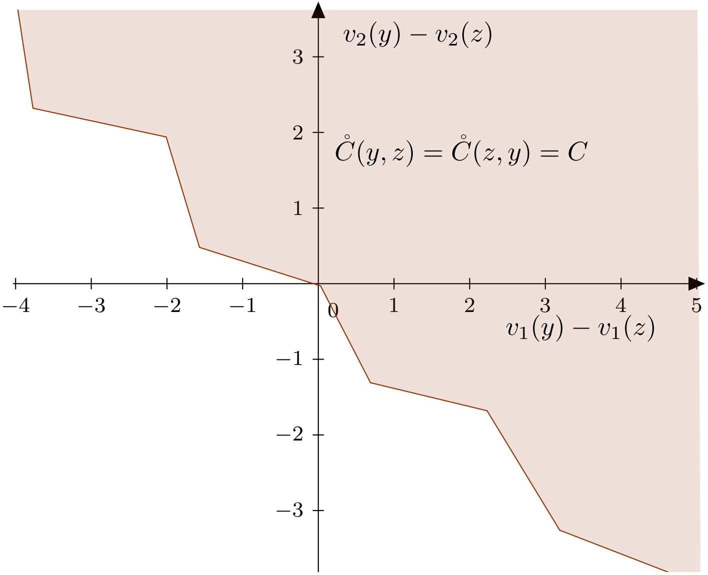
Independence of $\mathring{C}$ from the Alternatives in A

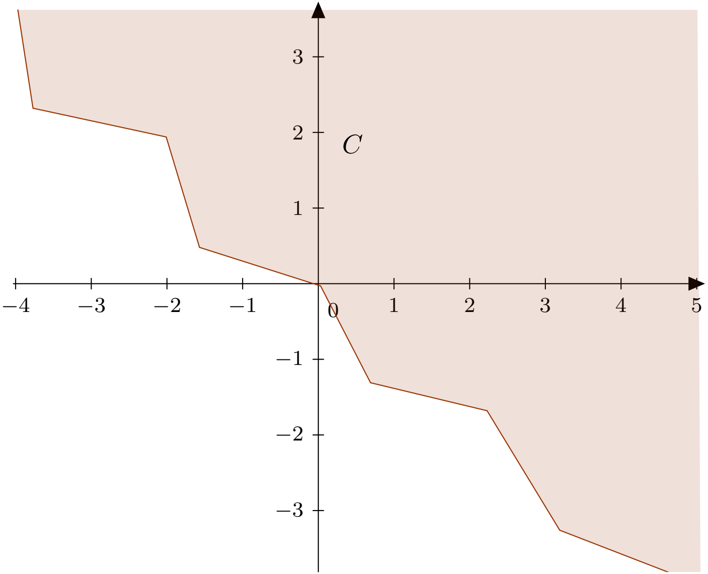
- Define the translated set $C(y,z) = P(y,z) - \gamma(y,z)\mathbf{1}$
- Denote the 'interior' of $C(y,z)$ by $\mathring{C}(y,z)$

Claim

$\mathring{C}(y,z) = \mathring{C}(w,l)$, *for any* $y,z,w,l \in A$, $y \neq z$ *and* $w \neq l$.

Remark: Note that this result, in particular, includes the cases, $\mathring{C}(y,z) = \mathring{C}(l,z) = \mathring{C}(l,y) = \mathring{C}(z,y)$. Therefore, the claim holds even without y,z,w,l being all distinct.

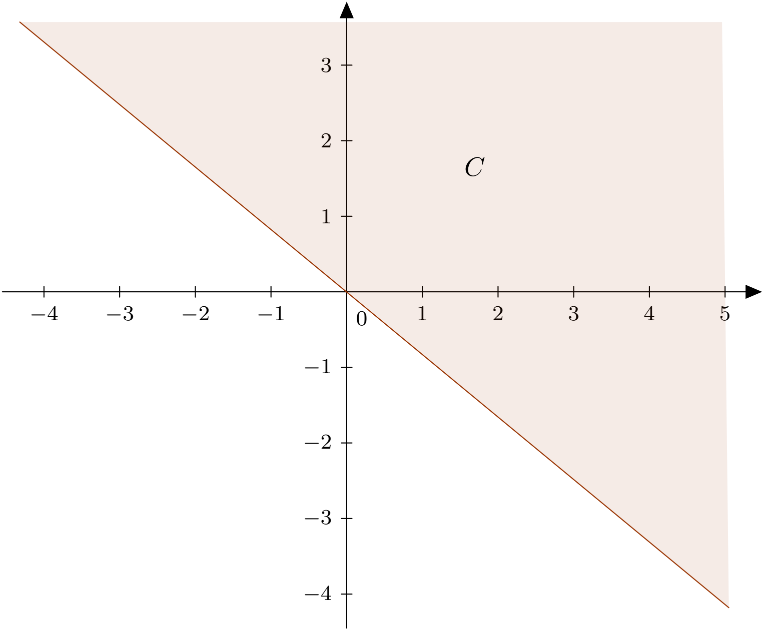

$v_2(y) - v_2(z)$
$\mathring{C}(y,z) = \mathring{C}(z,y) = C$
$v_1(y) - v_1(z)$
3
2
1
−1
−2
−3
−4
−3
−2
−1
0
1
2
3
4
5


C
3
2
1
−1
−2
−3
−4
−3
−2
−1
0
1
2
3
4
5

Convexity of C

Claim

The set C is convex.



[Back]

Holmström Characterization

- Set of allocations $A = \{a, b\}$
- Social welfares at these two allocations are $\sum_{j \in N} v_j(a)$ and $\sum_{j \in N} v_j(b)$

Holmström Characterization

- Set of allocations $A = \{a, b\}$
- Social welfares at these two allocations are $\sum_{j \in N} v_j(a)$ and $\sum_{j \in N} v_j(b)$
- Efficiency requires that if a is chosen, then $\sum_{j \in N} v_j(a) \geq \sum_{j \in N} v_j(b)$

Holmström Characterization

- Set of allocations $A = \{a, b\}$
- Social welfares at these two allocations are $\sum_{j \in N} v_j(a)$ and $\sum_{j \in N} v_j(b)$
- Efficiency requires that if a is chosen, then $\sum_{j \in N} v_j(a) \geq \sum_{j \in N} v_j(b)$
- Fix valuations of agents other than i at v_{-i}
- Fix valuations of agent i except allocation a, i.e., at b at $v_i(b)$

Holmström Characterization

- Set of allocations $A = \{a, b\}$
- Social welfares at these two allocations are $\sum_{j \in N} v_j(a)$ and $\sum_{j \in N} v_j(b)$
- Efficiency requires that if a is chosen, then $\sum_{j \in N} v_j(a) \geq \sum_{j \in N} v_j(b)$
- Fix valuations of agents other than i at v_{-i}
- Fix valuations of agent i except allocation a, i.e., at b at $v_i(b)$
- There exists some threshold $v_i^*(a)$ such that
  - for all $v_i(a) \geq v_i^*(a)$, a is the outcome
  - for all $v_i(a) < v_i^*(a)$, b is the outcome

Holmström Characterization

- Set of allocations $A = \{a, b\}$
- Social welfares at these two allocations are $\sum_{j \in N} v_j(a)$ and $\sum_{j \in N} v_j(b)$
- Efficiency requires that if a is chosen, then $\sum_{j \in N} v_j(a) \geq \sum_{j \in N} v_j(b)$
- Fix valuations of agents other than i at v_{-i}
- Fix valuations of agent i except allocation a, i.e., at b at $v_i(b)$
- There exists some threshold $v_i^*(a)$ such that
  - for all $v_i(a) \geq v_i^*(a)$, a is the outcome
  - for all $v_i(a) < v_i^*(a)$, b is the outcome
- Consider $v_i(a) = v_i^*(a) + \epsilon$, $\epsilon > 0$, and write the DSIC constraint:

$$v_i^*(a) + \epsilon - p_{i,a} \geq v_i(b) - p_{i,b} \quad (1)$$

outcome does not change $\Rightarrow$ payment does not change

Holmström Characterization

- Consider $v_i(a) = v_i^*(a) - \delta$, $\delta > 0$, and similarly:

$$v_i(b) - p_{i,b} \geq v_i^*(a) - \delta - p_{i,a} \tag{2}$$

- Combining Equations (1) and (2) and taking limits $\epsilon, \delta \to 0$, we get,

$$v_i(b) - p_{i,b} = v_i^*(a) - p_{i,a}$$

Holmström Characterization

- Consider $v_i(a) = v_i^*(a) - \delta$, $\delta > 0$, and similarly:

$$v_i(b) - p_{i,b} \geq v_i^*(a) - \delta - p_{i,a} \tag{2}$$

- Combining Equations (1) and (2) and taking limits $\epsilon, \delta \to 0$, we get,

$$v_i(b) - p_{i,b} = v_i^*(a) - p_{i,a}$$

- Since $v_i^*(a)$ is a threshold of change of *efficient* outcome,

$$v_i^*(a) + \sum_{j \in N \setminus \{i\}} v_j(a) = v_i(b) + \sum_{j \in N \setminus \{i\}} v_j(b)$$

Holmström Characterization

- Consider $v_i(a) = v_i^*(a) - \delta$, $\delta > 0$, and similarly:

$$v_i(b) - p_{i,b} \geq v_i^*(a) - \delta - p_{i,a} \tag{2}$$

- Combining Equations (1) and (2) and taking limits $\epsilon, \delta \to 0$, we get,

$$v_i(b) - p_{i,b} = v_i^*(a) - p_{i,a}$$

- Since $v_i^*(a)$ is a threshold of change of *efficient* outcome,

$$v_i^*(a) + \sum_{j \in N \setminus \{i\}} v_j(a) = v_i(b) + \sum_{j \in N \setminus \{i\}} v_j(b)$$

- Substituting:

$$p_{i,a} - p_{i,b} = -\left(\sum_{j \in N \setminus \{i\}} v_j(b) - \sum_{j \in N \setminus \{i\}} v_j(a) \right)$$

[Back]