Mechanism Design

J. Leite (adapted from Kevin Leyton-Brown)

Section 1

Mechanism Design

Can we Design a System to Efficiently Mediate Bargaining?

Can we avoid wasting resources and energy by designing the game?

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).
  - Encodes what we start with and cannot control.

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 (though they may be constrained by the environment)
- 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 no longer knows 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 miscoordinate and play none of the equilibria

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 may send a sequence of messages; in between, information may be (partially) revealed about the messages that were sent previously like extensive form

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 miscoordinate and play none of the equilibria

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 may send a sequence of messages; in between, information may be (partially) revealed about the messages that were sent previously like extensive form

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 miscoordinate and play none of the equilibria

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 may send a sequence of messages; in between, information may be (partially) revealed about the messages that were sent previously like extensive form

Section 2

Revelation Principle

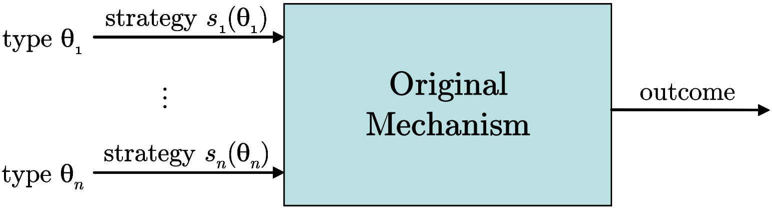
Revelation Principle

- It turns out that any social choice function that can be implemented by any mechanism can be implemented by a truthful, direct mechanism!

Revelation Principle

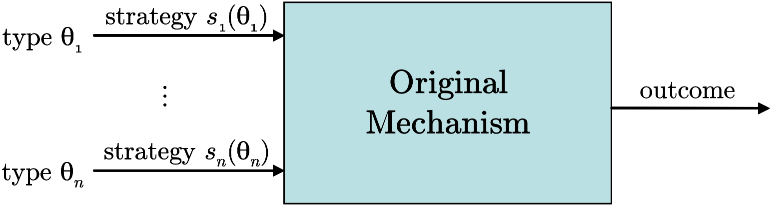
- It turns out that any social choice function that can be implemented by any mechanism can be implemented by a truthful, direct mechanism!
- Consider an arbitrary, non-truthful mechanism (e.g., may be indirect)

Revelation Principle



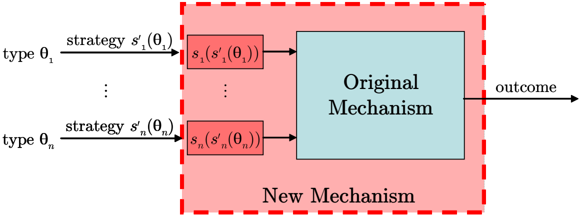
- It turns out that any social choice function that can be implemented by any mechanism can be implemented by a truthful, direct mechanism!
- Consider an arbitrary, non-truthful mechanism (e.g., may be indirect)

Revelation Principle



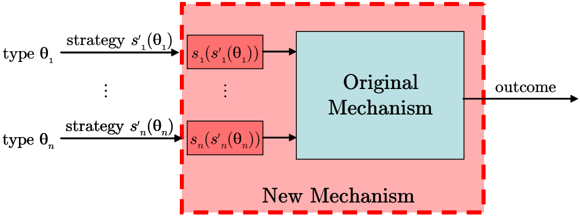
- It turns out that any social choice function that can be implemented by any mechanism can be implemented by a truthful, direct mechanism!
- Consider an arbitrary, non-truthful mechanism (e.g., may be indirect)
- Recall that a mechanism defines a game, and consider an equilibrium $s = (s_1, \ldots, s_n)$

Revelation Principle



- We can construct a new direct mechanism, as shown above
- This mechanism is truthful by exactly the same argument that s was an equilibrium in the original mechanism

Revelation Principle



- We can construct a new direct mechanism, as shown above
- This mechanism is truthful by exactly the same argument that s was an equilibrium in the original mechanism
- "The agents don't have to lie, because the mechanism already lies for them."

Computational Criticism of the Revelation Principle

- computation is pushed onto the center
  - often, agents' strategies will be computationally expensive
    - e.g., in the shortest path problem, agents may need to compute shortest paths, cutsets in the graph, etc.
  - since the center plays equilibrium strategies for the agents, the center now incurs this cost
- if computation is intractable, so that it cannot be performed by agents, then in a sense the revelation principle doesn't hold
  - agents can't play the equilibrium strategy in the original mechanism
  - however, in this case it's unclear what the agents will do

Discussion of the Revelation Principle

- The set of equilibria is not always the same in the original mechanism and revelation mechanism
  - of course, we've shown that the revelation mechanism does have the original equilibrium of interest
  - however, in the case of indirect mechanisms, even if the indirect mechanism had a unique equilibrium, the revelation mechanism can also have new, bad equilibria
- So what is the revelation principle good for?
  - recognition that truthfulness is not a restrictive assumption
  - recognition that indirect mechanisms can't do (inherently) better than direct mechanisms
  - for analysis purposes, we can consider only truthful mechanisms, and be assured that such a mechanism exists

Discussion of the Revelation Principle

- The set of equilibria is not always the same in the original mechanism and revelation mechanism
  - of course, we've shown that the revelation mechanism does have the original equilibrium of interest
  - however, in the case of indirect mechanisms, even if the indirect mechanism had a unique equilibrium, the revelation mechanism can also have new, bad equilibria
- So what is the revelation principle good for?
  - recognition that truthfulness is not a restrictive assumption
  - recognition that indirect mechanisms can't do (inherently) better than direct mechanisms
  - for analysis purposes, we can consider only truthful mechanisms, and be assured that such a mechanism exists

Section 3

Impossibility of General, Dominant-Strategy Implementation

Dominant Strategies and Mechanisms

- Consider a society N, O and any mechanism A, M for which every agent has a dominant strategy for each preference. There exists a social choice function C (a direct mechanism) for which truthful announcement of preferences is a dominant strategy.
- So, if we are considering implementation in dominant strategies, it is enough to look only at social choice functions for which truth is a dominant strategy: the set of non-manipulable or strategy-proof social choice functions.

Dominant Strategies and Mechanisms

- Consider a society N, O and any mechanism A, M for which every agent has a dominant strategy for each preference. There exists a social choice function C (a direct mechanism) for which truthful announcement of preferences is a dominant strategy.
- So, if we are considering implementation in dominant strategies, it is enough to look only at social choice functions for which truth is a dominant strategy: the set of non-manipulable or strategy-proof social choice functions.

Impossibility Result

Theorem (Gibbard-Satterthwaite)

Consider any social choice function C of N and O. If:

1. $|O| \geq 3$ *(there are at least three outcomes);*
2. C *is onto (surjective); that is, for every $o \in O$ there is a preference profile $[\succ]$ such that $C([\succ]) = o$*

C *is dominant-strategy truthful if and only if C is dictatorial.*

- So, any non-dictatorial social choice function on a full domain of preferences and with at least three alternatives will be manipulable by some agents for some preference profiles

Impossibility Result

Theorem (Gibbard-Satterthwaite)

Consider any social choice function C of N and O. If:

1. $|O| \geq 3$ *(there are at least three outcomes);*
2. C *is onto (surjective); that is, for every* $o \in O$ *there is a preference profile* $[\succ]$ *such that* $C([\succ]) = o$

C *is dominant-strategy truthful if and only if* C *is dictatorial.*

- So, any non-dictatorial social choice function on a full domain of preferences and with at least three alternatives will be manipulable by some agents for some preference profiles

What does this mean?

- We should be discouraged about the possibility of implementing arbitrary social-choice functions in mechanisms.
- However, in practice we can circumvent the Gibbard-Satterthwaite theorem in two ways:
  - use a weaker form of implementation
    - note: the result only holds for dominant strategy implementation, not e.g., Bayes-Nash implementation
  - relax the onto condition and the (implicit) assumption that agents are allowed to hold arbitrary preferences

Settings with Strategy-Proof Social Choice Functions

- Single-Peaked domains
  - median voting
  - or take the max of peaks, or the min of peaks...
- Trade
  - Have a private value for buying (or selling) an indivisible good
  - A price is fixed in advance
  - declare whether willing to buy (sell) at that price

Section 4

Transferable Utility

Transferable Utility

Definition (Quasilinear preferences with transferable utility)

Agents have quasilinear preferences with transferable utility in an n-player Bayesian game when the set of outcomes is

$$O = X \times \mathbb{R}^n$$

for a finite set X, if the utility of an agent i given joint type θ can be written

$$u_i(o, \theta) = u_i(x, \theta) - p_i,$$

where $o = (x, p)$ is an element of O, $u_i : X \times \Theta \mapsto \mathbb{R}$.

Transferable Utility Mechanisms

- When outcomes consist of basic outcomes and some transfers or payments: $u_i(o, \theta) = u_i(x, \theta) - p_i$
- We split the mechanism into a choice rule and a payment rule:
  - $x \in X$ is a discrete, non-monetary outcome
  - $p_i \in \mathbb{R}$ is a monetary payment (possibly negative) that agent i must make to the mechanism
- Implications:
  - $u_i(x, \theta)$ is not influenced by the amount of money an agent has
  - agents don't care how much others are made to pay

Transferable Utility Mechanisms

- When outcomes consist of basic outcomes and some transfers or payments: $u_i(o, \theta) = u_i(x, \theta) - p_i$
- We split the mechanism into a choice rule and a payment rule:
  - $x \in X$ is a discrete, non-monetary outcome
  - $p_i \in \mathbb{R}$ is a monetary payment (possibly negative) that agent i must make to the mechanism
- Implications:
  - $u_i(x, \theta)$ is not influenced by the amount of money an agent has
  - agents don't care how much others are made to pay

Quasilinear Mechanism

Definition (Quasilinear mechanism)

A mechanism in the quasilinear setting (for a Bayesian game setting $(N, O = X \times \mathbb{R}^n, \Theta, p, u)$) is a triple (A, χ, p), where

- $A = A_1 \times \cdots \times A_n$, where A_i is the set of actions available to agent $i \in N$,
- $\chi : A \mapsto \Pi(X)$ maps each action profile to a distribution over choices, and
- $p : A \mapsto \mathbb{R}^n$ maps each action profile to a payment for each agent.

Direct Quasilinear Mechanism

Definition (Direct quasilinear mechanism)

A direct quasilinear mechanism (for a Bayesian game setting $(N, O = X \times \mathbb{R}^n, \Theta, p, u)$) is a pair (χ, p). It defines a standard mechanism in the quasilinear setting, where for each i, $A_i = \Theta_i$.

Definition (Conditional utility independence)

A Bayesian game exhibits conditional utility independence if for all agents $i \in N$, for all outcomes $o \in O$ and for all pairs of joint types θ and $\theta' \in \Theta$ for which $\theta_i = \theta'_i$, it holds that $u_i(o, \theta) = u_i(o, \theta')$.

Quasilinear Mechanisms with Conditional Utility Independence

- Given conditional utility independence, we can write i's utility function as $u_i(o, \theta_i)$
  - it does not depend on the other agents' types
- An agent's valuation for choice $x \in X$: $v_i(x) = u_i(x, \theta_i)$
  - the maximum amount i would be willing to pay to get x
  - in fact, i would be indifferent between keeping the money and getting x
- Alternate definition of direct mechanism:
  - ask agents i to declare $v_i(x)$ for each $x \in X$
- Define $\hat{v}_i$ as the valuation that agent i declares to such a direct mechanism
  - may be different from his true valuation v_i
- Also define the tuples $\hat{v}$, $\hat{v}_{-i}$

Section 5

Mechanism Design as an Optimization Problem

Mechanism Design as an Optimization Problem

- We can understand mechanism design as the problem of finding the best possible mechanism, given constraints about how it operates.
- Well now consider some typical choices for
  - these constraints
  - this notion of best

Truthfulness

Definition (Truthfulness)

A quasilinear mechanism is truthful if it is direct and $\forall i \forall v_i,$ agent i's equilibrium strategy is to adopt the strategy $\hat{v}_i = v_i$.

Efficiency

Definition (Efficiency)

A quasilinear mechanism is strictly Pareto efficient, or just efficient, if in equilibrium it selects a choice x such that

$$\forall v \forall x', \sum_i v_i(x) \geq \sum_i v_i(x').$$

- An efficient mechanism selects the choice that maximizes the sum of agents' utilities, disregarding monetary payments.
- How is this related to Pareto efficiency from GT?
  - if we include the mechanism as an agent, all Pareto-efficient outcomes involve the same choice (and different payments)
  - any outcome involving another choice is Pareto-dominated: some agents could make a side-payment to others such that all would prefer the swap

Efficiency

Definition (Efficiency)

A quasilinear mechanism is strictly Pareto efficient, or just efficient, if in equilibrium it selects a choice x such that

$$\forall v \forall x', \sum_i v_i(x) \geq \sum_i v_i(x').$$

- An efficient mechanism selects the choice that maximizes the sum of agents' utilities, disregarding monetary payments.
- How is this related to Pareto efficiency from GT?
  - if we include the mechanism as an agent, all Pareto-efficient outcomes involve the same choice (and different payments)
  - any outcome involving another choice is Pareto-dominated: some agents could make a side-payment to others such that all would prefer the swap

Efficiency

Definition (Efficiency)

A quasilinear mechanism is strictly Pareto efficient, or just efficient, if in equilibrium it selects a choice x such that

$$\forall v \forall x', \sum_i v_i(x) \geq \sum_i v_i(x').$$

- Called economic efficiency to distinguish from other (e.g., computational) notions
- Also called social-welfare maximization
- Note: defined in terms of true (not declared) valuations.

Budget Balance

Definition (Budget balance)

A quasilinear mechanism is budget balanced when

$$\forall v, \sum_i p_i(s(v)) = 0,$$

where s is the equilibrium strategy profile.

- regardless of the agents' types, the mechanism collects and disburses the same amount of money from and to the agents

Budget Balance

Definition (Budget balance)

A quasilinear mechanism is budget balanced when

$$\forall v, \sum_i p_i(s(v)) = 0,$$

where s is the equilibrium strategy profile.

- regardless of the agents' types, the mechanism collects and disburses the same amount of money from and to the agents
- relaxed version: weak budget balance:

$$\forall v, \sum_i p_i(s(v)) \geq 0$$

  - the mechanism never takes a loss, but it may make a profit

Budget Balance

Definition (Budget balance)

A quasilinear mechanism is budget balanced when

$$\forall v, \sum_i p_i(s(v)) = 0,$$

where s is the equilibrium strategy profile.

- regardless of the agents' types, the mechanism collects and disburses the same amount of money from and to the agents
- Budget balance can be required to hold ex ante:

$$\mathbb{E}_v \sum_i p_i(s(v)) = 0$$

  - the mechanism must break even or make a profit only on expectation

Individual-Rationality

Definition (Ex interim individual rationality)

A mechanism is ex interim individual rational when $\forall i \forall v_i,\ \mathbb{E}_{v_{-i}|v_i} v_i(\chi(s_i(v_i), s_{-i}(v_{-i}))) - p_i(s_i(v_i), s_{-i}(v_{-i})) \geq 0$, where s is the equilibrium strategy profile.

- no agent loses by participating in the mechanism.
- ex interim because it holds for every possible valuation for agent i, but averages over the possible valuations of the other agents.

Definition (Ex post individual rationality)

A mechanism is ex post individual rational when $\forall i \forall v,\ v_i(\chi(s(v))) - p_i(s(v)) \geq 0$, where s is the equilibrium strategy profile.

Individual-Rationality

Definition (Ex interim individual rationality)

A mechanism is ex interim individual rational when $\forall i \forall v_i,\ \mathbb{E}_{v_{-i}|v_i} v_i(\chi(s_i(v_i), s_{-i}(v_{-i}))) - p_i(s_i(v_i), s_{-i}(v_{-i})) \geq 0$, where s is the equilibrium strategy profile.

- no agent loses by participating in the mechanism.
- ex interim because it holds for every possible valuation for agent i, but averages over the possible valuations of the other agents.

Definition (Ex post individual rationality)

A mechanism is ex post individual rational when $\forall i \forall v,\ v_i(\chi(s(v))) - p_i(s(v)) \geq 0$, where s is the equilibrium strategy profile.

Tractability

Definition (Tractability)

A mechanism is tractable when $\forall \hat{v}$, $\chi(\hat{v})$ and $p(\hat{v})$ can be computed in polynomial time.

- The mechanism is computationally feasible.

Revenue Maximization

We can also add an objective function to our mechanism. One example: revenue maximization.

Definition (Revenue maximization)

A mechanism is revenue maximizing when, among the set of functions χ and p that satisfy the other constraints, the mechanism selects the χ and p that maximize $\mathbb{E}_\theta \sum_i p_i(s(\theta))$, where $s(\theta)$ denotes the agents' equilibrium strategy profile.

- The mechanism designer can choose among mechanisms that satisfy the desired constraints by adding an objective function such as revenue maximization.

Revenue Minimization

- The mechanism may not be intended to make money.
- Budget balance may be impossible to satisfy.
- Set weak budget balance as a constraint and add the following objective.

Definition (Revenue minimization)

A quasilinear mechanism is revenue minimizing when, among the set of functions χ and p that satisfy the other constraints, the mechanism selects the χ and p that minimize $\max_v \sum_i p_i(s(v))$ in equilibrium, where $s(v)$ denotes the agents' equilibrium strategy profile.

- Note: this considers the worst case over valuations; we could consider average case instead.

Fairness

- Fairness is hard to define. What is fairer:
  - an outcome that fines all agents $100 and makes a choice that all agents hate equally?
  - an outcome that charges all agents $0 and makes a choice that some agents hate and some agents like?
- Maxmin fairness: make the least-happy agent the happiest.

Definition (Maxmin fairness)

A quasilinear mechanism is maxmin fair when, among the set of functions χ and p that satisfy the other constraints, the mechanism selects the χ and p that maximize

$$\mathbb{E}_v \left[\min_{i \in N} v_i(\chi(s(v))) - p_i(s(v)) \right],$$

where $s(v)$ denotes the agents' equilibrium strategy profile.

Fairness

- Fairness is hard to define. What is fairer:
  - an outcome that fines all agents \$100 and makes a choice that all agents hate equally?
  - an outcome that charges all agents \$0 and makes a choice that some agents hate and some agents like?
- Maxmin fairness: make the least-happy agent the happiest.

Definition (Maxmin fairness)

A quasilinear mechanism is maxmin fair when, among the set of functions χ and p that satisfy the other constraints, the mechanism selects the χ and p that maximize

$$\mathbb{E}_v \left[\min_{i \in N} v_i(\chi(s(v))) - p_i(s(v)) \right],$$

where $s(v)$ denotes the agents' equilibrium strategy profile.

Price of Anarchy Minimization

- When an efficient mechanism is impossible, we may want to get as close as possible
- Minimize the worst-case ratio between optimal social welfare and the social welfare achieved by the given mechanism.

Definition (Price-of-anarchy minimization)

A quasilinear mechanism minimizes the price of anarchy when, among the set of functions χ and p that satisfy the other constraints, the mechanism selects the χ and p that minimize

$$\max_{v \in V} \frac{\max_{x \in X} \sum_{i \in N} v_i(x)}{\sum_{i \in N} v_i\left(\chi(s(v))\right)},$$

where $s(v)$ denotes the agents' equilibrium strategy profile in the worst equilibrium of the mechanism—i.e., the one in which $\sum_{i \in N} v_i(\chi(s(v)))$ is the smallest.

Section 6

The Vickrey-Clarke-Groves (VCG) Mechanism

A positive result

- Recall that in the quasilinear utility setting, a mechanism can be defined as a choice rule and a payment rule.
- The VCG mechanism:
  - has truth as a dominant strategy (satisfies truthfulness, is strategy-proof)
  - makes efficient choices (not including payments)
- And, under additional assumptions about the setting, can satisfy
  - weak budget balanced
  - interim individual-rational

A positive result

- Recall that in the quasilinear utility setting, a mechanism can be defined as a choice rule and a payment rule.
- The VCG mechanism:
  - has truth as a dominant strategy (satisfies truthfulness, is strategy-proof)
  - makes efficient choices (not including payments)
- And, under additional assumptions about the setting, can satisfy
  - weak budget balanced
  - interim individual-rational

The Groves Mechanism

Definition (Groves mechanism)

The Groves mechanism is a direct quasilinear mechanism (χ, p), where

$$\chi(\hat{v}) \in \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

- So what's going on with the payment rule?
  - the agent i must pay some amount $h_i(\hat{v}_{-i})$ that doesn't depend on his own declared valuation
  - the agent i is paid $\sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$, the sum of the others' valuations for the chosen outcome

The Groves Mechanism

Definition (Groves mechanism)

The Groves mechanism is a direct quasilinear mechanism (χ, p), where

$$\chi(\hat{v}) \in \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

- So what's going on with the payment rule?
  - the agent i must pay some amount $h_i(\hat{v}_{-i})$ that doesn't depend on his own declared valuation
  - the agent i is paid $\sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$, the sum of the others' valuations for the chosen outcome

Groves Properties

Theorem

Truth telling is a dominant strategy under the Groves mechanism.

Theorem (Green–Laffont)

An efficient *mechanism* (χ, p) *such that* $\chi(\hat{v}) \in \arg\max_x \sum_i \hat{v}_i(x)$ *has truthful reporting as a dominant strategy for all agents and preferences only if it is a Grooves mechanism:* $p_i(\hat{v}) = h_i(\hat{v}_{-i}) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$

VCG

Definition (Clarke tax)

The Clarke tax sets the h_i term in a Groves mechanism as

$$h_i(\hat{v}_{-i}) = \sum_{j \neq i} \hat{v}_j \left(\chi(\hat{v}_{-i})\right).$$

Definition (Vickrey-Clarke-Groves (VCG) mechanism)

The Vickrey-Clarke-Groves mechanism is a direct quasilinear mechanism (χ, p), where

$$\chi(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j \left(\chi(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

VCG discussion

$$\chi(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\left(\chi(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))$$

- You get paid everyone's utility under the allocation that is actually chosen
  - except your own, but you get that directly as utility
- Then you get charged everyone's utility in the world where you don't participate
- Thus you pay your social cost

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\left(x(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\,(x(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\left(x(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\left(x(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\left(x(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG discussion

$$x(\hat{v}) = \arg\max_x \sum_i \hat{v}_i(x)$$

$$p_i(\hat{v}) = \sum_{j \neq i} \hat{v}_j\,(x(\hat{v}_{-i})) - \sum_{j \neq i} \hat{v}_j(x(\hat{v}))$$

Questions:

- who pays 0?
  - agents who don't affect the outcome
- who pays more than 0?
  - (pivotal) agents who make things worse for others by existing
- who gets paid?
  - (pivotal) agents who make things better for others by existing

VCG properties

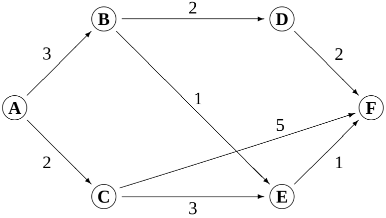
$$
\begin{aligned}
\chi(\hat{v}) &= \arg\max_x \sum_i \hat{v}_i(x) \\
p_i(\hat{v}) &= \sum_{j \neq i} \hat{v}_j \left(\chi(\hat{v}_{-i})\right) - \sum_{j \neq i} \hat{v}_j(\chi(\hat{v}))
\end{aligned}
$$

- Because only pivotal agents have to pay, VCG is also called the pivot mechanism
- It's dominant-strategy truthful, because it's a Groves mechanism

Section 7

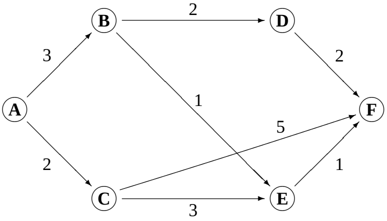
VCG example

Selfish routing example



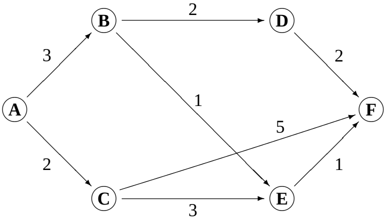
- What outcome will be selected by χ? path $ABEF$.
- How much will AC have to pay?
  - The shortest path taking his declaration into account has a length of 5, and imposes a cost of -5 on agents other than him (because it does not involve him). Likewise, the shortest path without AC's declaration also has a length of 5. Thus, his payment $p_{AC} = (-5) - (-5) = 0$.
  - This is what we expect, since AC is not pivotal.
  - Likewise, BD, CE, CF and DF will all pay zero.

Selfish routing example



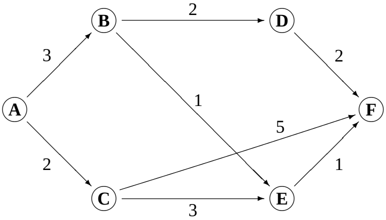
- What outcome will be selected by χ? path $ABEF$.
- How much will AC have to pay?
  - The shortest path taking his declaration into account has a length of 5, and imposes a cost of -5 on agents other than him (because it does not involve him). Likewise, the shortest path without AC's declaration also has a length of 5. Thus, his payment $p_{AC} = (-5) - (-5) = 0$.
  - This is what we expect, since AC is not pivotal.
  - Likewise, BD, CE, CF and DF will all pay zero.

Selfish routing example



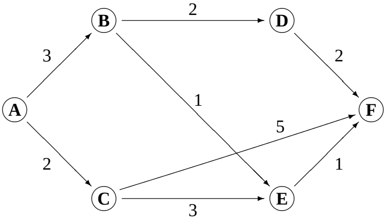
- What outcome will be selected by χ? path $ABEF$.
- How much will AC have to pay?
  - The shortest path taking his declaration into account has a length of 5, and imposes a cost of -5 on agents other than him (because it does not involve him). Likewise, the shortest path without AC's declaration also has a length of 5. Thus, his payment $p_{AC} = (-5) - (-5) = 0$.
  - This is what we expect, since AC is not pivotal.
  - Likewise, BD, CE, CF and DF will all pay zero.

Selfish routing example



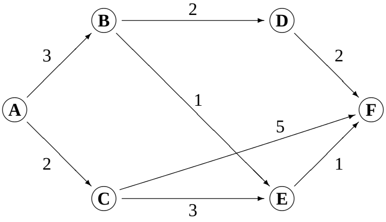
- What outcome will be selected by χ? path $ABEF$.
- How much will AC have to pay?
  - The shortest path taking his declaration into account has a length of 5, and imposes a cost of -5 on agents other than him (because it does not involve him). Likewise, the shortest path without AC's declaration also has a length of 5. Thus, his payment $p_{AC} = (-5) - (-5) = 0$.
  - This is what we expect, since AC is not pivotal.
  - Likewise, BD, CE, CF and DF will all pay zero.

Selfish routing example



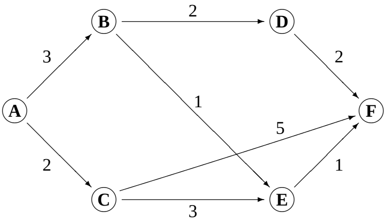
- How much will AB pay?
  - The shortest path taking AB's declaration into account has a length of 5, and imposes a cost of 2 on other agents.
  - The shortest path without AB is $ACEF$, which has a cost of 6.
  - Thus $p_{AB} = (-6) - (-2) = -4$.

Selfish routing example



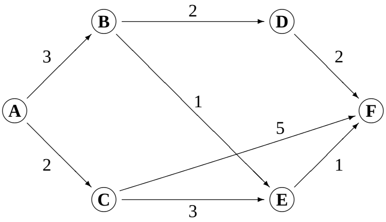
- How much will AB pay?
  - The shortest path taking AB's declaration into account has a length of 5, and imposes a cost of 2 on other agents.
  - The shortest path without AB is $ACEF$, which has a cost of 6.
  - Thus $p_{AB} = (-6) - (-2) = -4$.

Selfish routing example



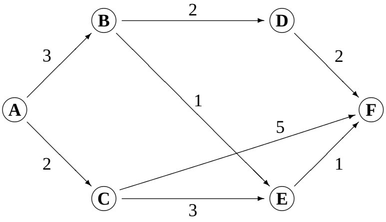
- How much will BE pay?
    -

Selfish routing example



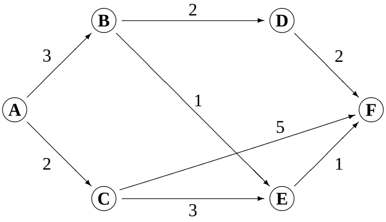
- How much will BE pay? $p_{BE} = (-6) - (-4) = -2$.
    -

Selfish routing example



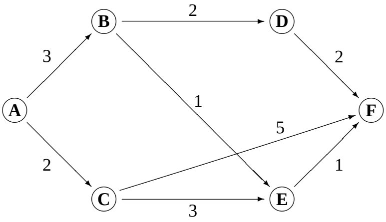
- How much will BE pay? $p_{BE} = (-6) - (-4) = -2$.
- How much will EF pay?
    -

Selfish routing example



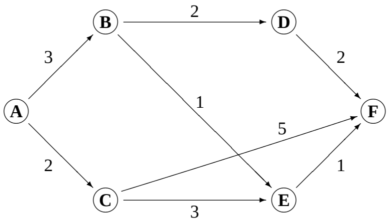
- How much will BE pay? $p_{BE} = (-6) - (-4) = -2$.
- How much will EF pay? $p_{EF} = (-7) - (-4) = -3$.
    -

Selfish routing example



- How much will BE pay? $p_{BE} = (-6) - (-4) = -2$.
- How much will EF pay? $p_{EF} = (-7) - (-4) = -3$.
  - EF and BE have the same costs but are paid different amounts. Why?

Selfish routing example



- How much will BE pay? $p_{BE} = (-6) - (-4) = -2$.
- How much will EF pay? $p_{EF} = (-7) - (-4) = -3$.
  - EF and BE have the same costs but are paid different amounts. Why?
  - EF has more market power: for the other agents, the situation without EF is worse than the situation without BE.

Section 8

VCG Limitations

1. Privacy

- VCG requires agents to fully reveal their private information
- This private information may have value to agents that extends beyond the current interaction
  - for example, the agents may know that they will compete with each other again in the future
- It is often preferable to elicit only as much information from agents as is required to determine the social welfare maximizing choice and compute the VCG payments.

2. Susceptibility to Collusion

Example

Agent	U(build road)	U(do not build road)	Payment
1	200	0	150
2	100	0	50
3	0	250	0

- What happens if agents 1 and 2 both increase their declared valuations by $50?

2. Susceptibility to Collusion

Example

Agent	U(build road)	U(do not build road)	Payment
1	250	0	
2	150	0	
3	0	250	0

- What happens if agents 1 and 2 both increase their declared valuations by $50?

2. Susceptibility to Collusion

Example

Agent	U(build road)	U(do not build road)	Payment
1	250	0	100
2	150	0	0
3	0	250	0

- What happens if agents 1 and 2 both increase their declared valuations by $50?

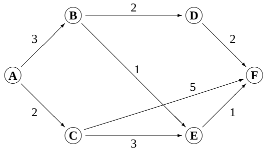
2. Susceptibility to Collusion

Example

Agent	U(build road)	U(do not build road)	Payment
1	250	0	100
2	150	0	0
3	0	250	0

- What happens if agents 1 and 2 both increase their declared valuations by $50?
- The choice is unchanged, but both of their payments are reduced.
- Thus, while no agent can gain by changing his declaration, groups can.

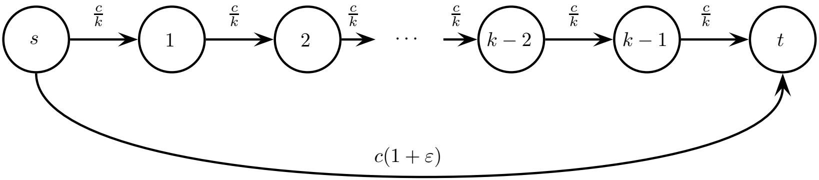
3. VCG is not Frugal



- VCG can end up paying arbitrarily more than an agent is willing to accept (or equivalently charging arbitrarily less than an agent is willing to pay)
- Consider AC, which is not part of the shortest path.
  - If the cost of this edge increased to 8, our payment to AB would increase to $p_{AB} = (-12) - (-2) = -10$.
  - If the cost were any $x \geq 2$, we would select the path $ABEF$ and would have to make a payment to AB of $p_{AB} = (-4 - x) - (-2) = -(x + 2)$.
  - The gap between agents' true costs and the payments that they could receive under VCG is unbounded.

3. VCG is not Frugal

Are VCG's payments at least close to the cost of the <u>second</u> shortest disjoint path?



- The top path has a total cost of c.
- VCG picks it, pays each of the k agents $c(1+\varepsilon) - (k-1)\frac{c}{k}$.
- Hence VCG's total payment is $c(1+k\varepsilon)$.
- For fixed ε, VCG's payment is $\Theta(k)$ times the cost of the second shortest disjoint path.

4. Revenue Monotonicity Violated

Revenue monotonicity: revenue always weakly increases as agents are added.

Example

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	90

4. Revenue Monotonicity Violated

Revenue monotonicity: revenue always weakly increases as agents are added.

Example

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	0
3	100	0	0

4. Revenue Monotonicity Violated

Revenue monotonicity: revenue always weakly increases as agents are added.

Example

Agent	U(build road)	U(do not build road)	Payment
1	0	90	0
2	100	0	0
3	100	0	0

- Adding agent 3 causes VCG to pick the same choice but to collect zero revenue!
- Agent 2 could pretend to be two agents and eliminate his payment.

5. Cannot Return All Revenue to Agents

- we may want to use VCG to induce agents to report their valuations honestly, but may not want to make a profit by collecting money from the agents.
- Thus, we might want to find some way of returning the mechanism's profits back the agents.
- However, the possibility of receiving a rebate after the mechanism has been run changes the agents' incentives.
- In fact, even if profits are given to a charity that the agents care about, or spent in a way that benefits the local economy and hence benefits the agents, the VCG mechanism is undermined.
- It is possible to return at least some of the revenues to the agents, but it must be done very carefully, and in general not all the money can be returned.

Section 9

Individual Rationality and Budget Balance

Individual Rationality and Budget Balance

- VCG gives rise to
  - Dominant Strategies
  - Efficient allocations
- In general it doesn't give rise to
  - Individual Rationality
  - Budget Balance
- However, under mild assumptions, we may get these properties.

Section 10

Individual Rationality

Two definitions

Definition (Choice-set monotonicity)

An environment exhibits choice-set monotonicity if $\forall i,\ X_{-i} \subseteq X$.

- removing any agent weakly decreases—that is, never increases—the mechanism's set of possible choices X

Definition (No negative externalities)

An environment exhibits no negative externalities if $\forall i \forall x \in X_{-i},\ v_i(x) \geq 0$.

- every agent has zero or positive utility for any choice that can be made without his participation

Example: road referendum

Example

Consider the problem of holding a referendum to decide whether or not to build a road.

- The set of choices is independent of the number of agents, satisfying choice-set monotonicity.
- No agent negatively values the project, though some might value the situation in which the project is not undertaken more highly than the situation in which it is.

Example: simple exchange

Example

Consider a market setting consisting of agents interested in buying a single unit of a good such as a share of stock, and another set of agents interested in selling a single unit of this good. The choices in this environment are sets of buyer-seller pairings (prices are imposed through the payment function).

- If a new agent is introduced into the market, no previously-existing pairings become infeasible, but new ones become possible; thus choice-set monotonicity is satisfied.
- Because agents have zero utility both for choices that involve trades between other agents and no trades at all, there are no negative externalities.

VCG Individual Rationality

Theorem

The VCG mechanism is ex-post individual rational when the choice set monotonicity and no negative externalities properties hold.

Proof.

All agents truthfully declare their valuations in equilibrium. Then

$$u_i = v_i(\chi(v)) - \left(\sum_{j \neq i} v_j(\chi(v_{-i})) - \sum_{j \neq i} v_j(\chi(v)) \right)$$
$$= \sum_i v_i(\chi(v)) - \sum_{j \neq i} v_j(\chi(v_{-i})) \tag{1}$$

$\chi(v)$ is the outcome that maximizes social welfare, and that this optimization could have picked $\chi(v_{-i})$ instead (by choice set monotonicity). Thus,

$$\sum_j v_j(\chi(v)) \geq \sum_j v_j(\chi(v_{-i})).$$

□

VCG Individual Rationality

Theorem

The VCG mechanism is ex-post individual rational when the choice set monotonicity and no negative externalities properties hold.

Proof.

$$\sum_j v_j(\chi(v)) \geq \sum_j v_j(\chi(v_{-i})).$$

Furthermore, from no negative externalities,

$$v_i(\chi(v_{-i})) \geq 0.$$

Therefore,

$$\sum_i v_i(\chi(v)) \geq \sum_{j \neq i} v_j(\chi(v_{-i})),$$

and thus Equation (1) is non-negative. □

Section 11

Budget Balance

Another property

Definition (No single-agent effect)

An environment exhibits no single-agent effect if $\forall i$, $\forall v_{-i}$, $\forall x \in \arg\max_y \sum_j v_j(y)$ there exists a choice x' that is feasible without i and that has $\sum_{j \neq i} v_j(x') \geq \sum_{j \neq i} v_j(x)$.

Welfare of agents other than i is weakly increased by dropping i.

Example

Consider a single-sided auction. Dropping an agent just reduces the amount of competition, making the others better off.

Good news

Theorem

The VCG mechanism is weakly budget-balanced when the no single-agent effect property holds.

Proof.

Assume truth-telling in equilibrium. We must show that the sum of transfers from agents to the center is greater than or equal to zero.

$$\sum_i p_i(v) = \sum_i \left(\sum_{j \neq i} v_j(\chi(v_{-i})) - \sum_{j \neq i} v_j(\chi(v)) \right)$$

From the no single-agent effect condition we have that

$$\forall i \sum_{j \neq i} v_j(\chi(v_{-i})) \geq \sum_{j \neq i} v_j(\chi(v)).$$

Thus the result follows directly. □

More good news

Theorem (Krishna & Perry, 1998)

In any Bayesian game setting in which VCG is ex post individually rational, VCG collects at least as much revenue as any other efficient and ex interim individually-rational mechanism.

- This is somewhat surprising: does not require dominant strategies, and hence compares VCG to all Bayes–Nash mechanisms.
- A useful corollary: VCG is as budget balanced as any efficient mechanism can be
  - it satisfies weak budget balance in every case where any dominant strategy, efficient and ex interim IR mechanism would be able to do so.

Section 12

Further MD topics

Computational applications of mechanism design

1. Task scheduling
   - allocate tasks among agents to minimize makespan
2. Bandwidth allocation in computer networks
   - allocate the real-valued capacity of a single network link among users with different demand curves
3. Multicast cost sharing
   - share the cost of a multicast transmission among the users who receive it
4. Two-sided matching
   - pair up members of two groups according to their preferences, without imposing any payments
   - e.g., students and advisors; hospitals and interns; kidney donors and recipients

Computational applications of mechanism design

1. Task scheduling
   - allocate tasks among agents to minimize makespan
2. Bandwidth allocation in computer networks
   - allocate the real-valued capacity of a single network link among users with different demand curves
3. Multicast cost sharing
   - share the cost of a multicast transmission among the users who receive it
4. Two-sided matching
   - pair up members of two groups according to their preferences, without imposing any payments
   - e.g., students and advisors; hospitals and interns; kidney donors and recipients

Computational applications of mechanism design

1. Task scheduling
   - allocate tasks among agents to minimize makespan
2. Bandwidth allocation in computer networks
   - allocate the real-valued capacity of a single network link among users with different demand curves
3. Multicast cost sharing
   - share the cost of a multicast transmission among the users who receive it
4. Two-sided matching
   - pair up members of two groups according to their preferences, without imposing any payments
   - e.g., students and advisors; hospitals and interns; kidney donors and recipients

Computational applications of mechanism design

1. Task scheduling
   - allocate tasks among agents to minimize makespan
2. Bandwidth allocation in computer networks
   - allocate the real-valued capacity of a single network link among users with different demand curves
3. Multicast cost sharing
   - share the cost of a multicast transmission among the users who receive it
4. Two-sided matching
   - pair up members of two groups according to their preferences, without imposing any payments
   - e.g., students and advisors; hospitals and interns; kidney donors and recipients