# CS 497: Electronic Market Design

Kate Larson

David R. Cheriton School of Computer Science
University of Waterloo

# Outline

1. Introduction
2. Game Theory
3. Mechanism Design

# Introduction

- Kate Larson
    - Faculty Member in CS
    - Member of the AI research group
- Research Interests: Multiagent Systems
    - Strategic Reasoning
        - bounded rationality/limited resources
    - Electronic market design

# Introduction

- Growth in settings where there are multiple *self-interested* interacting parties
  - Networks
  - Electronic markets
  - Game playing...
- To act optimally, participants must take into account how other agents are going to act
- We want to be able to
  - Understand the ways in which agents will interact and behave
  - Design systems so that agents behave the way we would like

# Introduction

- Growth in settings where there are multiple *self-interested* interacting parties
  - Networks
  - Electronic markets
  - Game playing...
- To act optimally, participants must take into account how other agents are going to act
- We want to be able to
  - Understand the ways in which agents will interact and behave
  - Design systems so that agents behave the way we would like

# Introduction

- Growth in settings where there are multiple *self-interested* interacting parties
  - Networks
  - Electronic markets
  - Game playing...
- To act optimally, participants must take into account how other agents are going to act
- We want to be able to
  - Understand the ways in which agents will interact and behave
  - Design systems so that agents behave the way we would like

# Two Communities

**Economics**

- Traditional emphasis on game theoretic rationality
- Describing how agents should behave
- Multiple self-interested agents

**Computer Science**

- Traditional emphasis on computational and informational constraints
- Building agents
- Individual or cooperative agents

# Two Communities

**Economics**

- Traditional emphasis on game theoretic rationality
- Describing how agents should behave
- Multiple self-interested agents

**Computer Science**

- Traditional emphasis on computational and informational constraints
- Building agents
- Individual or cooperative agents

# New Research Problems

- How do we use game theory and mechanism design in computer science settings?
- How do we resolve conflicts between game-theoretic and computational constraints?
- Development of new theories, methodologies and models

# New Research Area

Explosion of research in the area (Algorithmic game theory, computational mechanism design, Distributed algorithmic mechanism design, computational game theory,...)

- Papers appearing in AAAI, AAMAS, UAI, NIPS, PODC, SIGCOMM, INFOCOMM, SODA, STOC, FOCS, ...
- Papers by CS researchers appearing in Games and Economic Behavior, Journal of Economic Theory, Econometrica,...
- Numerous workshops and meetings,...

# Today's Lecture

Today I will provide an overview of some key game theory and mechanism design concepts:

- What is a game?
- What is a solution concept for a game?
- What is a mechanism?

# Self-Interested Agents

We are interested in **self-interested** agents.

It does not mean that

- they want to harm other agents
- they only care about things that benefit them

It means that

- the agent has its *own* description of states of the world that it likes, and that its actions are motivated by this description

# Self-Interested Agents

We are interested in **self-interested** agents.

It does not mean that

- they want to harm other agents
- they only care about things that benefit them

It means that

- the agent has its *own* description of states of the world that it likes, and that its actions are motivated by this description

# Self-Interested Agents

We are interested in **self-interested** agents.

It does not mean that

- they want to harm other agents
- they only care about things that benefit them

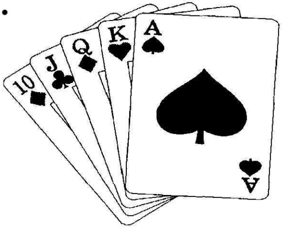
It means that

- the agent has its *own* description of states of the world that it likes, and that its actions are motivated by this description

# What is game theory?

The study of games!

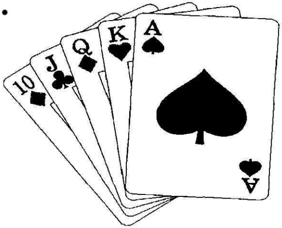
- Bluffing in poker
- What move to make in chess
- How to play Rock-Scissors-Paper




Also study of auction design, strategic deterrence, election laws, coaching decisions, routing protocols,...

# What is game theory?

The study of games!

- Bluffing in poker
- What move to make in chess
- How to play Rock-Scissors-Paper
- 



Also study of auction design, strategic deterrence, election laws, coaching decisions, routing protocols,...

# What is game theory?

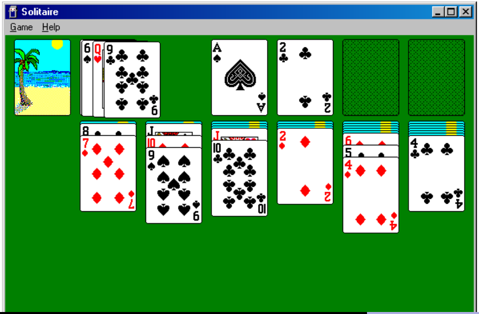
Game theory is a formal way to analyze **interactions** among a **group** of **rational** agents who behave **strategically**.

# What is game theory?

Game theory is a formal way to analyze **interactions** among a **group** of **rational** agents who behave **strategically**.

**Group:** Must have more than one decision maker

- Otherwise you have a decision problem, not a game




**Solitaire is not a game.**

# What is game theory?

Game theory is a formal way to analyze **interactions** among a **group** of **rational** agents who behave **strategically**.

**Interaction:** What one agent does directly affects at least one other agent

**Strategic:** Agents take into account that their actions influence the game

**Rational:** An agent chooses its best action (maximizes its expected utility)

# Normal Form

## aka Strategic Form

A normal form game is defined by

- Finite set of agents (or players) $N$, $|N| = n$
- Each agent $i$ has an action space $A_i$
  - $A_i$ is non-empty and finite
- Outcomes are defined by action profiles ($a = (a_1, \ldots, a_n)$) where $a_i$ is the action taken by agent $i$
  - Notation: $a_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, a_n)$ and $a = (a_i, a_{-i})$
- Each agent has a utility function $u_i : A_1 \times \ldots \times A_n \mapsto \mathbb{R}$

# Examples

**Prisoners' Dilemma**

| | C | D |
|---|---|---|
| C | -1,-1 | -4,0 |
| D | 0,-4 | -3,-3 |

**Pure coordination game**

$\forall$ action profiles $a \in A_1 \times \ldots \times A_n$ and $\forall i, j$, $u_i(a) = u_j(a)$.

| | L | R |
|---|---|---|
| L | 1,1 | 0,0 |
| R | 0,0 | 1,1 |

Agents do not have conflicting interests. Their sole challenge is to coordinate on an action which is good for all.

# Examples

**Prisoners' Dilemma**

| | C | D |
|---|---|---|
| C | -1,-1 | -4,0 |
| D | 0,-4 | -3,-3 |

**Pure coordination game**
$\forall$ action profiles $a \in A_1 \times \ldots \times A_n$ and $\forall i, j$, $u_i(a) = u_j(a)$.

| | L | R |
|---|---|---|
| L | 1,1 | 0,0 |
| R | 0,0 | 1,1 |

Agents do not have conflicting interests. Their sole challenge is to coordinate on an action which is good for all.

# Zero-sum games

$\forall a \in A_1 \times A_2, u_1(a) + u_2(a) = 0.$ That is, one player gains at the other player's expense.

**Matching Pennies**

| | H | T |
|---|---|---|
| H | 1,-1 | -1, 1 |
| T | -1,1 | 1,-1 |

| | H | T |
|---|---|---|
| H | 1 | -1 |
| T | -1 | 1 |

Given the utility of one agent, the other's utility is known.

# More Examples

Most games have elements of both cooperation and competition.

**BoS**

| | H | S |
|---|---|---|
| H | 2,1 | 0,0 |
| S | 0,0 | 1,2 |

**Hawk-Dove**

| | D | H |
|---|---|---|
| D | 3,3 | 1,4 |
| H | 4,1 | 0,0 |

# More Examples

Most games have elements of both cooperation and competition.

**BoS**

| | H | S |
|---|---|---|
| H | 2,1 | 0,0 |
| S | 0,0 | 1,2 |

**Hawk-Dove**

| | D | H |
|---|---|---|
| D | 3,3 | 1,4 |
| H | 4,1 | 0,0 |

# Strategies

**Notation:** Given set $X$, let $\Delta X$ be the set of all probability distributions over $X$.

**Definition**

*A strategy $s_i$ is a probability distribution over $A_i$. $s_i(a_i)$ is the probability action $a_i$ will be played by mixed strategy $s_i$.*

**Definition**

*A pure strategy, $s_i$, is a strategy such that there exists an action $a_j \in A_i$ and $s_i(a_j) = 1$. We often use $s_i = a_j$ to denote a pure strategy.*

# Strategies

**Notation:** Given set $X$, let $\Delta X$ be the set of all probability distributions over $X$.

## Definition

*A strategy $s_i$ is a probability distribution over $A_i$. $s_i(a_i)$ is the probability action $a_i$ will be played by mixed strategy $s_i$.*

## Definition

*A pure strategy, $s_i$, is a strategy such that there exists an action $a_j \in A_i$ and $s_i(a_j) = 1$. We often use $s_i = a_j$ to denote a pure strategy.*

# Expected Utility

The expected utility of agent $i$ given strategy profile $s$ is

$$u_i(s) = \sum_{a \in A} u_i(a) \Pi_{j=1}^{n} s_j(a_j)$$

**Example**

| | C | D |
|---|---|---|
| C | -1,-1 | -4,0 |
| D | 0, -4 | -3,-3 |

Given strategy profile
$s = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{10}, \frac{9}{10}))$

$$u_1 = -1(\frac{1}{2})(\frac{1}{10}) - 4(\frac{1}{2})(\frac{9}{10}) - 3(\frac{1}{2})(\frac{9}{10}) = -3.2$$

$$u_2 = -1(\frac{1}{2})(\frac{1}{10}) - 4(\frac{1}{2})(\frac{1}{10}) - 3(\frac{1}{2})(\frac{9}{10}) = -1.6$$

# Best-response

Given a game, what strategy should an agent choose?
We first consider only pure strategies.

**Definition**

*Given $a_{-i}$, the best-response for agent $i$ is $a_i \in A_i$ such that*

$$u_i(a_i^*, a_{-i}) \geq u_i(a_i', a_{-i}) \forall a_i' \in A_i$$

*Note that the best response may not be unique.*
*A best-response set is*

$$B_i(a_{-i}) = \{a_i \in A_i | u_i(a_i, a_{-i}) \geq u_i(a_i', a_{-i}) \forall a_i' \in A_i\}$$

# Nash Equilibrium

**Definition**

*A profile $a^*$ is a Nash equilibrium if $\forall i$, $a_i^*$ is a best response to $a_{-i}^*$. That is*

$$\forall i u_i(a_i^*, a_{-i}^*) \geq u_i(a_i', a_{-i}^*) \; \forall a_i' \in A_i$$

*Equivalently, $a^*$ is a Nash equilibrium if $\forall i$*

$$a_i^* \in B(a_{-i}^*)$$

# Examples

PD

| | C | D |
|---|---|---|
| C | -1,-1 | -4,0 |
| D | 0,-4 | -3,-3 |

BoS

| | H | T |
|---|---|---|
| H | 2,1 | 0,0 |
| T | 0,0 | 1,2 |

Matching Pennies

| | H | T |
|---|---|---|
| H | 1,-1 | -1,1 |
| T | -1,1 | 1,-1 |

# Nash Equilibria

We need to extend the definition of a Nash equilibrium.
Strategy profile $s^*$ is a Nash equilibrium is for all $i$

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i', s_{-i}^*) \ \forall s_i' \in S_i$$

Similarly, a best-response set is

$$B(s_{-i}) = \{s_i \in S_i | u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i}) \forall s_i' \in S_i\}$$

# Examples

PD

| | C | D |
|---|---|---|
| C | -1,-1 | -4,0 |
| D | 0,-4 | -3,-3 |

BoS

| | H | T |
|---|---|---|
| H | 2,1 | 0,0 |
| T | 0,0 | 1,2 |

Matching Pennies

| | H | T |
|---|---|---|
| H | 1,-1 | -1,1 |
| T | -1,1 | 1,-1 |

# Existence of Nash Equilibria

**Theorem (Nash, 1950)**

*Every finite normal form game has a Nash equilibrium.*

Nobel Prize in Economics (1994)
Shared with Harsanyi and Selten.

# Proof

Beyond scope of course of todays lecture.

**Basic idea:** Define set $X$ to be all mixed strategy profiles.
Show that it has nice properties (compact and convex).
Define $f : X \mapsto 2^X$ to be the best-response set function, i.e. given $s$, $f(s)$ is the set all strategy profiles $s' = (s'_1, \ldots, s'_n)$ such that $s'_i$ is $i$'s best response to $s'_{-i}$.
Show that $f$ satisfies required properties of a fixed point theorem (Kakutani's or Brouwer's).
Then, $f$ has a fixed point, i.e. there exists $s$ such that $f(s) = s$.
This $s$ is mutual best-response – NE!

# Interpretations of Nash Equilibria

- Consequence of rational inference
- Focal point
- Self-enforcing agreement
- Stable social convention
- ...

# The existence proof is non-constructive.

For some games we can find equilibria easily:

- Zero-sum games can be represented by a linear program

For arbitrary games, the problem is PPAD-complete. (What does this mean?)

- 2-player games have the same complexity as $k$-player games and finding fixed points
- There is some evidence that no efficient solutions exist for such problems

Finding equilibria with certain properties is often NP-hard.

# The existence proof is non-constructive.

For some games we can find equilibria easily:

- Zero-sum games can be represented by a linear program

For arbitrary games, the problem is PPAD-complete. (What does this mean?)

- 2-player games have the same complexity as $k$-player games and finding fixed points
- There is some evidence that no efficient solutions exist for such problems

Finding equilibria with certain properties is often NP-hard.

# The existence proof is non-constructive.

For some games we can find equilibria easily:

- Zero-sum games can be represented by a linear program

For arbitrary games, the problem is PPAD-complete. (What does this mean?)

- 2-player games have the same complexity as $k$-player games and finding fixed points
- There is some evidence that no efficient solutions exist for such problems

Finding equilibria with certain properties is often NP-hard.

# The existence proof is non-constructive.

For some games we can find equilibria easily:

- Zero-sum games can be represented by a linear program

For arbitrary games, the problem is PPAD-complete. (What does this mean?)

- 2-player games have the same complexity as $k$-player games and finding fixed points
- There is some evidence that no efficient solutions exist for such problems

Finding equilibria with certain properties is often NP-hard.

# Finding Nash Equilibria

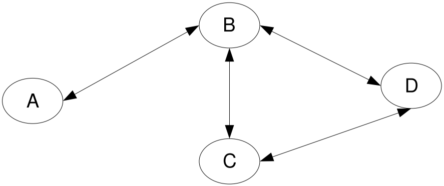
- Games can be represented as LCPs
  - Lemke-Howson algorithm
- Reduce games by removing (dominated) strategies

# Finding Nash Equilibria

- Games can be represented as LCPs
  - Lemke-Howson algorithm
- Reduce games by removing (dominated) strategies

# Finding Nash Equilibria

- Games descriptions can be abstracted
  - Currently do this in poker playing programs
- Take advantage in underlying structure of the agents' interactions
  - Graphical Games (Kearns et al) and Action-Graph Games (Bhat et al)



# Other Solution Concepts

Often Nash is too weak a solution concept

- Implicit knowledge (and common knowledge) assumptions
- Fragile
- Multiple Nash equilibria (does not always remove "unreasonable" outcomes)

Other solution concepts

- Dominant strategy equilibria
- Subgame perfect equilibria
- Bayes-Nash equilibria
- etc.

# Other Solution Concepts

Often Nash is too weak a solution concept

- Implicit knowledge (and common knowledge) assumptions
- Fragile
- Multiple Nash equilibria (does not always remove “unreasonable” outcomes)

Other solution concepts

- Dominant strategy equilibria
- Subgame perfect equilibria
- Bayes-Nash equilibria
- etc.

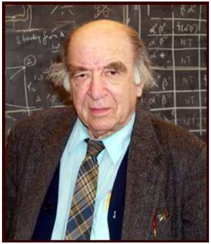
# Mechanism Design

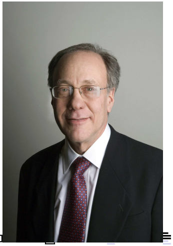
Game Theory asks

- Given a game, what should rational agents do?

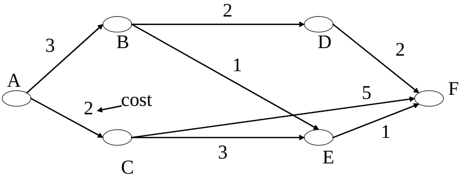
Mechanism Design asks

- Given rational agents, what sort of game should we design?
- Can we guarantee that agents will reach an outcome with the properties *we* want
  - maximize social welfare, maximize revenue, fairness criteria,...





# Example




- We want to find the least-cost route from $S$ to $T$.
- Costs are private information – we do not know them
- We do know that agents (nodes) are interested in maximizing revenue
- How can we use this to figure out the least-cost route?

# Fundamentals

- Set of possible outcomes $O$
- Set of agents $N$, $|N| = n$
  - Each agent $i$ has type $\theta_i \in \Theta_i$
  - Type captures all private information that is relevant to the agent's decision making
- Utility $u_i(o, \theta_i)$ over outcome $o \in O$
- Recall: goal is to implement some system wide solution
  - Captured by a social choice function

$$f : \Theta_1 \times \ldots \times \Theta_n \rightarrow O$$

where $f(\theta_1, \ldots, \theta_n) = o$ is a collective choice

# Fundamentals

- Set of possible outcomes $O$
- Set of agents $N$, $|N| = n$
  - Each agent $i$ has type $\theta_i \in \Theta_i$
  - Type captures all private information that is relevant to the agent's decision making
- Utility $u_i(o, \theta_i)$ over outcome $o \in O$
- Recall: goal is to implement some system wide solution
  - Captured by a social choice function

$$f : \Theta_1 \times \ldots \times \Theta_n \to O$$

where $f(\theta_1, \ldots, \theta_n) = o$ is a collective choice

# Fundamentals

- Set of possible outcomes $O$
- Set of agents $N$, $|N| = n$
  - Each agent $i$ has type $\theta_i \in \Theta_i$
  - Type captures all private information that is relevant to the agent's decision making
- Utility $u_i(o, \theta_i)$ over outcome $o \in O$
- Recall: goal is to implement some system wide solution
  - Captured by a social choice function

$$f : \Theta_1 \times \ldots \times \Theta_n \to O$$

where $f(\theta_1, \ldots, \theta_n) = o$ is a collective choice

# Fundamentals

- Set of possible outcomes $O$
- Set of agents $N$, $|N| = n$
  - Each agent $i$ has type $\theta_i \in \Theta_i$
  - Type captures all private information that is relevant to the agent's decision making
- Utility $u_i(o, \theta_i)$ over outcome $o \in O$
- Recall: goal is to implement some system wide solution
  - Captured by a social choice function

$$f : \Theta_1 \times \ldots \times \Theta_n \to O$$

where $f(\theta_1, \ldots, \theta_n) = o$ is a collective choice

# Fundamentals

- Set of possible outcomes $O$
- Set of agents $N$, $|N| = n$
  - Each agent $i$ has type $\theta_i \in \Theta_i$
  - Type captures all private information that is relevant to the agent's decision making
- Utility $u_i(o, \theta_i)$ over outcome $o \in O$
- Recall: goal is to implement some system wide solution
  - Captured by a social choice function

$$f : \Theta_1 \times \ldots \times \Theta_n \rightarrow O$$

where $f(\theta_1, \ldots, \theta_n) = o$ is a collective choice

# Examples of Social Choice Functions

- **Voting:**
  - Choose a candidate among a group
- **Public project:**
  - Decide whether to build a swimming pool whose cost must be funded by the agents themselves
- **Allocation:**
  - Allocate a single, indivisible item to one agent in a group

# Examples of Social Choice Functions

- **Voting:**
  - Choose a candidate among a group
- **Public project:**
  - Decide whether to build a swimming pool whose cost must be funded by the agents themselves
- **Allocation:**
  - Allocate a single, indivisible item to one agent in a group

# Examples of Social Choice Functions

- **Voting:**
  - Choose a candidate among a group
- **Public project:**
  - Decide whether to build a swimming pool whose cost must be funded by the agents themselves
- **Allocation:**
  - Allocate a single, indivisible item to one agent in a group

# Mechanisms

Recall that we want to implement a social choice function

- Need to know agents' preferences
- They may not reveal them to us truthfully

Example:

- One item to allocate, and want to give it to agent who values it the most
- If we just ask agents to tell us their true preferences, they may lie

*I want the bear!*

*I want it more!*

# Mechanism Design Problem

- By having agents interact through an institution we might be able to solve the problem
- Mechanism:
$$M = (S_1, \ldots, S_n, g(\cdot))$$
where
  - $S_i$ is the strategy space of agent $i$
  - $g : S_1 \times \ldots \times S_n \rightarrow O$ is the outcome function

# Mechanism Design Problem

- By having agents interact through an institution we might be able to solve the problem
- Mechanism:
$$M = (S_1, \ldots, S_n, g(\cdot))$$
where
  - $S_i$ is the strategy space of agent $i$
  - $g : S_1 \times \ldots \times S_n \rightarrow O$ is the outcome function

# Implementation

**Definition**

*A mechanism* $M = (S_1, \ldots, S_n, g(\cdot))$ **implements** *social choice function* $f(\Theta)$ *if there is an equilibrium strategy profile*

$$s^* = (s_1^*(\theta_1), \ldots, s_n^*(\theta_n))$$

*of the game induced by* $M$ *such that*

$$g(s_1^*(\theta_1), \ldots, s_n^*(\theta_n)) = f(\theta_1, \ldots, \theta_n)$$

*for all*

$$(\theta_1, \ldots, \theta_n) \in \Theta_1 \times \ldots \times \Theta_n$$

# Implementation

We did not specify the type of equilibrium in the definition

- Nash

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Bayes-Nash

$$E[u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)] \geq E[u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)]$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Dominant

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*, \forall s_{-i}$

# Implementation

We did not specify the type of equilibrium in the definition

- Nash

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Bayes-Nash

$$E[u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)] \geq E[u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)]$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Dominant

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*, \forall s_{-i}$

# Implementation

We did not specify the type of equilibrium in the definition

- Nash

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Bayes-Nash

$$E[u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)] \geq E[u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)]$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*$

- Dominant

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i)$$

$\forall i, \forall \theta_i, \forall s_i' \neq s_i^*, \forall s_{-i}$

# Direct Mechanisms

## Definition

*A* **direct mechanism** *is a mechanism where*

$$S_i = \Theta_i \text{ for all } i$$

*and*

$$g(\theta) = f(\theta) \text{ for all } \theta \in \Theta_1 \times \ldots \times \Theta_n$$

# Incentive Compatibility

## Definition

*A direct mechanism is* **incentive compatible** *if it has an equilibrium $s^*$ where*

$$s_i^*(\theta_i) = \theta_i$$

*for all $\theta_i \in \Theta_i$ and for all $i$. That is, truth-telling by all agents is an equilibrium.*

## Definition

*A direct mechanism is* **strategy-proof** *if it is incentive compatible and the equilibrium is a dominant strategy equilibrium.*

# Incentive Compatibility

## Definition

*A direct mechanism is* **incentive compatible** *if it has an equilibrium* $s^*$ *where*

$$s_i^*(\theta_i) = \theta_i$$

*for all* $\theta_i \in \Theta_i$ *and for all* $i$. *That is, truth-telling by all agents is an equilibrium.*

## Definition

*A direct mechanism is* **strategy-proof** *if it is incentive compatible and the equilibrium is a dominant strategy equilibrium.*

# Revelation Principle

**Theorem**

*Suppose there exists a mechanism $M = (S_1, \ldots, S_n, g(\cdot))$ that implements social choice function $f$ in dominant strategies. Then there is a direct strategy-proof mechanism $M'$ which also implements $f$.*
*[Gibbard 73; Green & Laffont 77; Myerson 79]*

*"The computations that go on within the mind of any bidder in the nondirect mechanism are shifted to become part of the mechanism in the direct mechanism."*
*[McAfee & McMillan 87]*

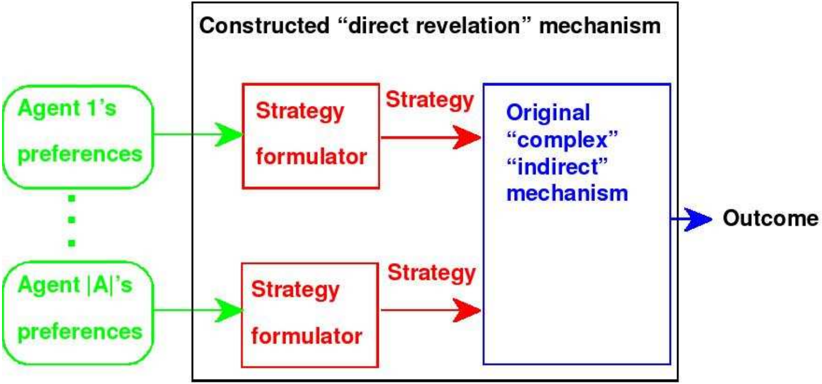
# Revelation Principle

**Theorem**

*Suppose there exists a mechanism* $M = (S_1, \ldots, S_n, g(\cdot))$ *that implements social choice function* $f$ *in dominant strategies. Then there is a direct strategy-proof mechanism* $M'$ *which also implements* $f$.
*[Gibbard 73; Green & Laffont 77; Myerson 79]*

> *"The computations that go on within the mind of any bidder in the nondirect mechanism are shifted to become part of the mechanism in the direct mechanism."*
> *[McAfee & McMillan 87]*

# Revelation Principle: Intuition



# Theoretical Implications

- **Literal interpretation:** Need only study direct mechanisms
  - A modeler can limit the search for an optimal mechanism to the class of direct IC mechanisms
  - If no direct mechanism can implement social choice function $f$ then no mechanism can
  - Useful because the space of possible mechanisms is huge

# Theoretical Implications

- **Literal interpretation:** Need only study direct mechanisms
  - A modeler can limit the search for an optimal mechanism to the class of direct IC mechanisms
  - If no direct mechanism can implement social choice function $f$ then no mechanism can
  - Useful because the space of possible mechanisms is huge

# Theoretical Implications

- **Literal interpretation:** Need only study direct mechanisms
  - A modeler can limit the search for an optimal mechanism to the class of direct IC mechanisms
  - If no direct mechanism can implement social choice function $f$ then no mechanism can
  - Useful because the space of possible mechanisms is huge

# Theoretical Implications

- **Literal interpretation:** Need only study direct mechanisms
  - A modeler can limit the search for an optimal mechanism to the class of direct IC mechanisms
  - If no direct mechanism can implement social choice function $f$ then no mechanism can
  - Useful because the space of possible mechanisms is huge

# Practical Implications

- Incentive-compatibility is "free"
  - Any outcome implemented by mechanism $M$ can be implemented by incentive-compatible mechanism $M'$
- "Fancy" mechanisms are unnecessary
  - Any outcome implemented by a mechanism with complex strategy space $S$ can be implemented by a direct mechanism

**BUT** Lots of mechanisms used in practice are not direct and incentive-compatible!

# Practical Implications

- Incentive-compatibility is “free”
  - Any outcome implemented by mechanism $M$ can be implemented by incentive-compatible mechanism $M'$
- “Fancy” mechanisms are unnecessary
  - Any outcome implemented by a mechanism with complex strategy space $S$ can be implemented by a direct mechanism

**BUT** Lots of mechanisms used in practice are not direct and incentive-compatible!

# Practical Implications

- Incentive-compatibility is "free"
  - Any outcome implemented by mechanism $M$ can be implemented by incentive-compatible mechanism $M'$
- "Fancy" mechanisms are unnecessary
  - Any outcome implemented by a mechanism with complex strategy space $S$ can be implemented by a direct mechanism

**BUT** Lots of mechanisms used in practice are not direct and incentive-compatible!

# Quick Review

We now know

- What a mechanism is
- What it means for a SCF to be dominant-strategy implementable
- Revelation Principle

We do not yet know

- What types of SCF are dominant-strategy implementable

# Quick Review

We now know

- What a mechanism is
- What it means for a SCF to be dominant-strategy implementable
- Revelation Principle

We do not yet know

- What types of SCF are dominant-strategy implementable

# Gibbard-Satterthwaite Impossibility

## Theorem

*Assume that*

- *$O$ is finite and $|O| \geq 3$,*
- *each $o \in O$ can be achieved by SCF $f$ for some $\theta$, and*
- *$\Theta$ includes all possible strict orderings over $O$.*

*Then $f$ is implementable in dominant strategies (strategy-proof) if and only if it is dictatorial.*

## Definition

*SCF $f$ is* **dictatorial** *if there is an agent $i$ such that for all $\theta$*

$$f(\theta) \in \{o \in O | u_i(o, \theta_i) \geq u_i(o', \theta_i) \forall o' \in O\}$$

# Gibbard-Satterthwaite Impossibility

## Theorem

*Assume that*

- *$O$ is finite and $|O| \geq 3$,*
- *each $o \in O$ can be achieved by SCF $f$ for some $\theta$, and*
- *$\Theta$ includes all possible strict orderings over $O$.*

*Then $f$ is implementable in dominant strategies (strategy-proof) if and only if it is dictatorial.*

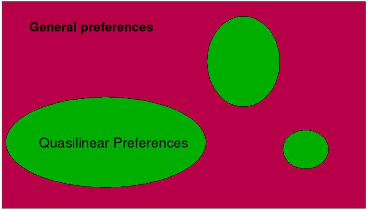
## Definition

*SCF $f$ is* **dictatorial** *if there is an agent $i$ such that for all $\theta$*

$$f(\theta) \in \{o \in O | u_i(o, \theta_i) \geq u_i(o', \theta_i) \forall o' \in O\}$$

# Circumventing Gibbard-Satterthwaite

- Use a weaker equilibrium concept
- Design mechanisms where computing a beneficial manipulation is hard
- Randomization
- Restrict the structure of agents' preferences



# Quasi-linear preferences

- Outcome $o = (x, t_1, \ldots, t_n)$
  - $x$ is a "project choice"
  - $t_i \in \mathbb{R}$ are transfers (money)
- Utility function of agent $i$

$$u_i(o, \theta_i) = v_i(x, \theta_i) - t_i$$

- Quasi-linear mechanism

$$M = (S_1, \ldots, S_n, g(\cdot))$$

where

$$g(\cdot) = (x(\cdot), t_1(\cdot), \ldots, t_n(\cdot))$$

# Quasi-linear preferences

- Outcome $o = (x, t_1, \ldots, t_n)$
  - $x$ is a "project choice"
  - $t_i \in \mathbb{R}$ are transfers (money)
- Utility function of agent $i$

$$u_i(o, \theta_i) = v_i(x, \theta_i) - t_i$$

- Quasi-linear mechanism

$$M = (S_1, \ldots, S_n, g(\cdot))$$

where

$$g(\cdot) = (x(\cdot), t_1(\cdot), \ldots, t_n(\cdot))$$

# Quasi-linear preferences

- Outcome $o = (x, t_1, \ldots, t_n)$
  - $x$ is a "project choice"
  - $t_i \in \mathbb{R}$ are transfers (money)
- Utility function of agent $i$

$$u_i(o, \theta_i) = v_i(x, \theta_i) - t_i$$

- Quasi-linear mechanism

$$M = (S_1, \ldots, S_n, g(\cdot))$$

where

$$g(\cdot) = (x(\cdot), t_1(\cdot), \ldots, t_n(\cdot))$$

# Social Choice Functions and Quasi-linearity

- SCF is **efficient** if for all $\theta$

$$\sum_{i=1}^{n} v_i(x(\theta), \theta_i) \geq \sum_{i=1}^{n} v_i(x'(\theta), \theta_i) \forall x'(\theta)$$

  This is also known as **social welfare maximizing**

- SCF is **budget-balanced** if

$$\sum_{i=1}^{n} t_i(\theta) = 0$$

  **Weakly budget-balanced** if

$$\sum_{i=1}^{n} t_i(\theta) \geq 0$$

# Social Choice Functions and Quasi-linearity

- SCF is **efficient** if for all $\theta$

$$\sum_{i=1}^{n} v_i(x(\theta), \theta_i) \geq \sum_{i=1}^{n} v_i(x'(\theta), \theta_i) \forall x'(\theta)$$

  This is also known as **social welfare maximizing**

- SCF is **budget-balanced** if

$$\sum_{i=1}^{n} t_i(\theta) = 0$$

  **Weakly budget-balanced** if

$$\sum_{i=1}^{n} t_i(\theta) \geq 0$$

# Groves Mechanisms [Groves 73]

A **Groves mechanism** $M = (S_1, \ldots, S_n, (x, t_1, \ldots, t_n))$ is defined by

- Choice rule

$$x^*(\theta) = \arg\max_x \sum_i v_i(x, \theta_i)$$

- Transfer rules

$$t_i(\theta) = h_i(\theta_{-i}) - \sum_{j \neq i} v_j(x^*(\theta), \theta_j)$$

where $h_i(\cdot)$ is an (arbitrary) function that does not depend on the reported type $\theta_i'$ of agent $i$.

# Groves Mechanisms [Groves 73]

A **Groves mechanism** $M = (S_1, \ldots, S_n, (x, t_1, \ldots, t_n))$ is defined by

- Choice rule

$$x^*(\theta) = \arg\max_x \sum_i v_i(x, \theta_i)$$

- Transfer rules

$$t_i(\theta) = h_i(\theta_{-i}) - \sum_{j \neq i} v_j(x^*(\theta), \theta_j)$$

where $h_i(\cdot)$ is an (arbitrary) function that does not depend on the reported type $\theta_i'$ of agent $i$.

# Groves Mechanisms [Groves 73]

A **Groves mechanism** $M = (S_1, \dots, S_n, (x, t_1, \dots, t_n))$ is defined by

- Choice rule

$$x^*(\theta) = \arg\max_x \sum_i v_i(x, \theta_i)$$

- Transfer rules

$$t_i(\theta) = h_i(\theta_{-i}) - \sum_{j \neq i} v_j(x^*(\theta), \theta_j)$$

where $h_i(\cdot)$ is an (arbitrary) function that does not depend on the reported type $\theta_i'$ of agent $i$.

# Groves Mechanisms

**Theorem**

*Groves mechanisms are strategy-proof and efficient.*

We have gotten around Gibbard-Satterthwaite.

# Proof

Agent $i$'s utility for strategy $\hat{\theta}_i$, given $\hat{\theta}_{-i}$ from agents $j \neq i$ is

$$\begin{aligned} u_i(\hat{\theta}_i) &= v_i(x^*(\hat{\theta}, \theta_i)) - t_i(\hat{\theta}) \\ &= v_i(x^*(\hat{\theta}, \theta_i)) + \sum_{j \neq i} v_j(x^*(\hat{\theta}, \hat{\theta}_j) - h_i(\hat{\theta}_{-i}) \end{aligned}$$

Ignore $h_i(\hat{\theta}_{-i})$ and notice $x^*(\hat{\theta}) = \arg\max_x \sum_i v_i(x, \hat{\theta}_i)$
i.e it maximizes the sum of reported values. Therefore, agent $i$ should announce $\hat{\theta}_i = \theta_i$ to maximize its own payoff.

**Thm:** Groves mechanisms are unique (up to $h_i(\theta_{-i})$).

# Vickrey-Clarke-Groves Mechanism

## aka Clarke mechanism, aka Pivotal mechanism

- Implement efficient outcome

$$x^* = \arg\max_x \sum_i v_i(x, \theta_i)$$

- Compute transfers

$$t_i(\theta) = \sum_{j \neq i} v_j(x^{-i}, \theta_j) - \sum_{j \neq i} v_j(x^*, \theta_j)$$

where $x^{-i} = \arg\max_x \sum_{j \neq i} v_j(x, \theta_j)$

**VCG are efficient and strategy-proof.**

# Vickrey-Clarke-Groves Mechanism

aka Clarke mechanism, aka Pivotal mechanism

- Implement efficient outcome

$$x^* = \arg\max_x \sum_i v_i(x, \theta_i)$$

- Compute transfers

$$t_i(\theta) = \sum_{j \neq i} v_j(x^{-i}, \theta_j) - \sum_{j \neq i} v_j(x^*, \theta_j)$$

where $x^{-i} = \arg\max_x \sum_{j \neq i} v_j(x, \theta_j)$

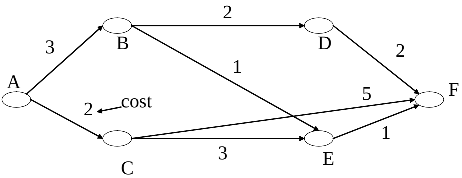
**VCG are efficient and strategy-proof.**

# VCG Mechanism

Agent's equilibrium utility is

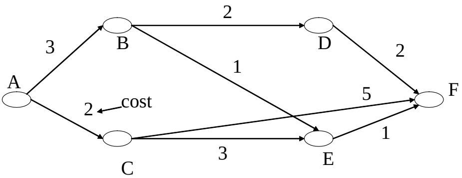
$$\begin{aligned} u_i((x^*,t),\theta_i) &= v_i(x^*,\theta_i) - \left[\sum_{j\neq i} v_j(x^{-i},\theta_j) - \sum_{j\neq i} v_j(x^*,\theta_j)\right] \\ &= \sum_{j=1}^{n} v_j(x^*,\theta_j) - \sum_{j\neq i} v_j(x^{-i},\theta_j) \\ &= \text{marginal contribution to the welfare of the system} \end{aligned}$$

# VCG Example



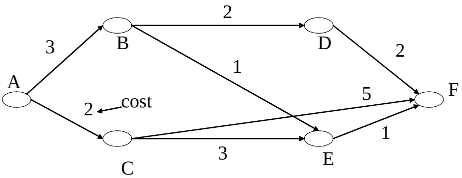

- What outcome will be chosen by $M$? path $ABEF$
- How much will $AC$ have to pay?
  - The shortest path taking its declaration into account has a length of 5, and imposes a cost of -5 on agents other than it (since it does not involve it). Likewise, the shortest path without AC's declaration also has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

# VCG Example



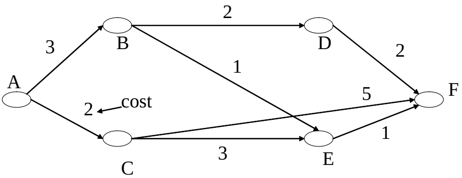

- What outcome will be chosen by $M$? path $ABEF$
- How much will $AC$ have to pay?
    - The shortest path taking its declaration into account has a length of 5, and imposes a cost of -5 on agents other than it (since it does not involve it). Likewise, the shortest path without AC's declaration also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

# VCG Example



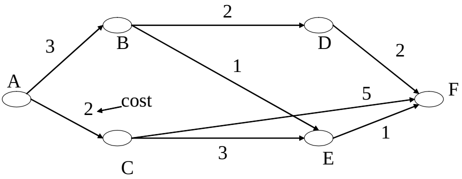

- What outcome will be chosen by $M$? path $ABEF$
- How much will $AC$ have to pay?
  - The shortest path taking its declaration into account has a length of 5, and imposes a cost of -5 on agents other than it (since it does not involve it). Likewise, the shortest path without AC's declaration also has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

# VCG Example



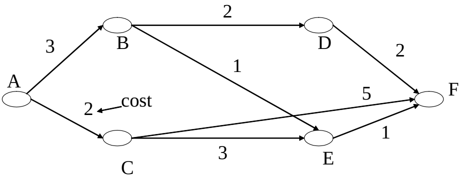

- What outcome will be chosen by $M$? path *ABEF*
- How much will *AC* have to pay?
  - The shortest path taking its declaration into account has a length of 5, and imposes a cost of -5 on agents other than it (since it does not involve it). Likewise, the shortest path without AC's declaration also has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

# VCG Example



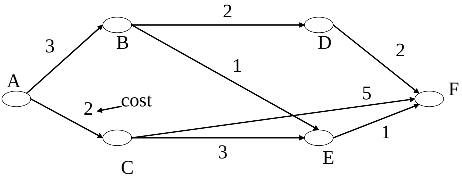

- How much will *AB* pay?
  - The shortest path taking *AB*'s declaration into account has a length of 5, and imposes a cost of 2 on other agents.
  - The shortest path without *AB* is *ACEF*, which has cost of 6.
  - Thus $P_{AB} = (-6) - (-2) = -4$.

# VCG Example



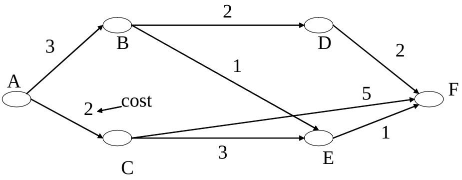

- How much will $AB$ pay?
  - The shortest path taking $AB$'s declaration into account has a length of 5, and imposes a cost of 2 on other agents.
  - The shortest path without $AB$ is $ACEF$, which has cost of 6.
  - Thus $P_{AB} = (-6) - (-2) = -4$.

# VCG Example



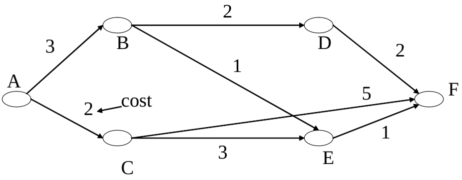

- How much will *BE* pay? $p_{BE} = (-6) - (-4) = -2$
- How much will *EF* pay? $p_{EF} = (-7) - (-4) = -3$
  - *EF* and *BE* have the same costs but are paid different amounts. Why?
  - *EF* has more *market power* – for the other agents, the situation without *EF* is worse than the situation without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

# VCG Example



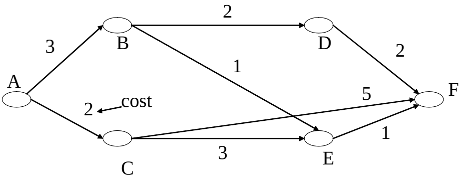

- How much will $BE$ pay? $p_{BE} = (-6) - (-4) = -2$
- How much will $EF$ pay? $p_{EF} = (-7) - (-4) = -3$
  - $EF$ and $BE$ have the same costs but are paid different amounts. Why?
  - $EF$ has more *market power* – for the other agents, the situation without $EF$ is worse than the situation without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

# VCG Example



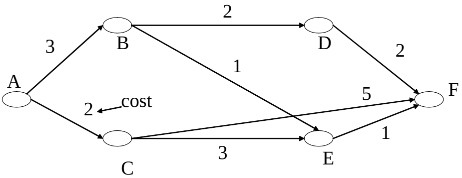

- How much will *BE* pay? $p_{BE} = (-6) - (-4) = -2$
- How much will *EF* pay? $p_{EF} = (-7) - (-4) = -3$
  - *EF* and *BE* have the same costs but are paid different amounts. Why?
  - *EF* has more *market power* – for the other agents, the situation without *EF* is worse than the situation without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

# VCG Example



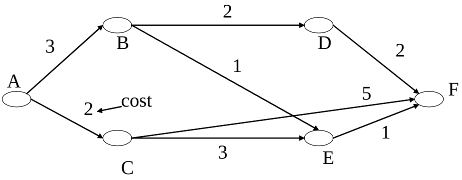

- How much will $BE$ pay?$p_{BE} = (-6) - (-4) = -2$
- How much will $EF$ pay?$p_{EF} = (-7) - (-4) = -3$
  - $EF$ and $BE$ have the same costs but are paid different amounts. Why?
  - $EF$ has more *market power* – for the other agents, the situation without $EF$ is worse than the situation without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

# VCG 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*.

# VCG 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*.