Algorithmic Game Theory
COMP6207

Lecture 3: Intro to Algorithmic Mechanism Design

Bahar Rastegari
b.rastegari@soton.ac.uk
Electronics and Computer Science
University of Southampton

Learning Outcomes

By the end of this session, you should be able to

- **Define** what is a *mechanism* and what is the goal of *mechanism design*.
- **Describe** the differences between *mechanism design* and *algorithmic mechanism design*
- **Define** what a *Bayesian game* is and what a *Bayesian game setting* is and **outline** the differences between the two

Mechanism Design

We are designing a system where participating agents are strategic.

- We need to be aware of, and consider, agents' incentives.

Mechanism Design

We are designing a system where participating agents are strategic.

- We need to be aware of, and consider, agents' incentives.

System designer has some goals, which may not be aligned with the goals of the participants.

Mechanism Design

We are designing a system where participating agents are strategic.

- We need to be aware of, and consider, agents' incentives.

System designer has some goals, which may not be aligned with the goals of the participants.

Example. A football cup is a system with strategic participants.

- The goal of participating teams is to win the cup.
- The goal of the designer of the tournament is . . . . . .

Mechanism Design

We are designing a system where participating agents are strategic.

- We need to be aware of, and consider, agents' incentives.

System designer has some goals, which may not be aligned with the goals of the participants.

Example. A football cup is a system with strategic participants.

- The goal of participating teams is to win the cup.
- The goal of the designer of the tournament is ......
  - to provide entertainment for the supporters and viewers, and keep running the tournaments for many years to come (by ensuring financial security)
- It is usually expected in a football game that teams should want to score a goal into the opponent's net, not their own!

Mechanism Design: a failed example

Fun video (click): 1994 Caribbean cup qualification

Mechanism Design: a failed example

Fun video (click): 1994 Caribbean cup qualification

What went wrong?

Fun video (click): 1994 Caribbean cup qualification

What went wrong?

Rule (unusual variant of golden rule)

Every game must have a winner!

- *If a game ends with a draw, it goes to extra-time!*
- *The first goal in extra-time wins the match, and*
- *counts as double!*

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq \mathbf{2}$
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference ≥ 2
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!
   Why?

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!
   Why? Because, if the game goes to extra time and they score, their goal counts double, making the final score 4–2, which means they will go through!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!
   Why? Because, if the game goes to extra time and they score, their goal counts double, making the final score 4–2, which means they will go through!
5. Grenada figures out what Barbados is planning to do, and they come up with a plan of their own: if they score a goal in either net before 90 minutes is up, they go through!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!
   Why? Because, if the game goes to extra time and they score, their goal counts double, making the final score 4–2, which means they will go through!
5. Grenada figures out what Barbados is planning to do, and they come up with a plan of their own: if they score a goal in either net before 90 minutes is up, they go through!
6. Barbados has to defend both nets from the goal! No goal is scored!

Barbados vs. Grenada, 1994: recap of what happened!

Rule (unusual variant of golden rule)

If a game ends with a draw, it goes to extra-time! The first goal in extra-time wins the match and counts as double!

1. To go through, Barbados needs to win with goal difference $\geq$**2**
2. Barbados leads **2–0**. So far, they go through, and Grenada is out!
3. Grenada scores, making it **2–1**. So, Barbados is out!
4. Barbados deliberately scores into own net, making it **2–2**!
   Why? Because, if the game goes to extra time and they score, their goal counts double, making the final score 4–2, which means they will go through!
5. Grenada figures out what Barbados is planning to do, and they come up with a plan of their own: if they score a goal in either net before 90 minutes is up, they go through!
6. Barbados has to defend both nets from the goal! No goal is scored!
7. Barbados scores a goal in overtime, which counts twice, makes it **4–2** and qualifies to the next round. Grenada is out!

Mechanism Design: well-developed science of rule-making

In systems with strategic participants, the rules matter!

- The system designer must anticipate strategic behaviour to create a good and reliable design.
- We cannot expect the participants to behave against their own interests.

Mechanism Design: well-developed science of rule-making

In systems with strategic participants, the rules matter!

- The system designer must anticipate strategic behaviour to create a good and reliable design.
- We cannot expect the participants to behave against their own interests.

Goal: design rules so that strategic behaviour by participants leads to a desirable outcome.

Mechanism Design: well-developed science of rule-making

In systems with strategic participants, the rules matter!

- The system designer must anticipate strategic behaviour to create a good and reliable design.
- We cannot expect the participants to behave against their own interests.

Goal: design rules so that strategic behaviour by participants leads to a desirable outcome.

Roughly speaking, assuming unknown individual utilities, we ask whether we can design a game such that, no matter what the secret utilities of the agents actually are, the equilibrium of the game is guaranteed to have a (set of) certain desired properties.

Applications

Almost everywhere in various aspects of our lives, but to name a few big and well-known systems:

- Elections
- Internet search auctions (ad auctions)
- Wireless spectrum auctions
- Matching medical residents or interns to hospitals
- Matching children to schools
- Kidney exchange markets

Algorithmic Mechanism Design (AMD)

Algorithmic Mechanism Design (AMD)

- A mechanism is an algorithm the input of which is withheld by strategic selfish agents.

Algorithmic Mechanism Design (AMD)

- A mechanism is an algorithm the input of which is withheld by strategic selfish agents.
- AMD lies at the intersection of economic game theory and computer science.

Algorithmic Mechanism Design (AMD)

- A mechanism is an algorithm the input of which is withheld by strategic selfish agents.
- AMD lies at the intersection of economic game theory and computer science.
- It was first coined by **Noam Nisan** and **Amir Ronen** in a research paper published in STOC'99.
  - Actually, the paper was titled "Algorithmic Mechanism Design"!

Algorithmic Mechanism Design (AMD)

- A mechanism is an algorithm the input of which is withheld by strategic selfish agents.
- AMD lies at the intersection of economic game theory and computer science.
- It was first coined by **Noam Nisan** and **Amir Ronen** in a research paper published in STOC'99.
  - Actually, the paper was titled "Algorithmic Mechanism Design"!
- In 2007, only 8 years later, a book titled "Algorithmic Game Theory" was published, with Noam and few others as editors (and various leading researchers of the field as authors of different sections).
- Algorithmic Game Theory is a broader field (it includes AMD), but the whole field started with that STOC'99 paper.

Realm of Algorithmic Mechanism Design

Settings in which

- a center wants to solve an optimization problem, but

Realm of Algorithmic Mechanism Design

Settings in which

- a center wants to solve an optimization problem, but
- the inputs to this problem are the private information of self-interested agents.

Realm of Algorithmic Mechanism Design

Settings in which

- a center wants to solve an optimization problem, but
- the inputs to this problem are the private information of self-interested agents.

The center must design a mechanism that

Realm of Algorithmic Mechanism Design

Settings in which

- a center wants to solve an optimization problem, but
- the inputs to this problem are the private information of self-interested agents.

The center must design a mechanism that

- solves the optimization problem while

Realm of Algorithmic Mechanism Design

Settings in which

- a center wants to solve an optimization problem, but
- the inputs to this problem are the private information of self-interested agents.

The center must design a mechanism that

- solves the optimization problem while
- inducing the agents to act as the mechanism designer wishes (ideally revealing their information truthfully).

AMD vs. Classical Economic Mechanism Design

AMD considers computational constraints to be of central importance.

AMD vs. Classical Economic Mechanism Design

AMD considers computational constraints to be of central importance.

- Mechanisms that cannot be efficiently implemented in polynomial time are not considered to be viable solutions to a mechanism design problem.

AMD vs. Classical Economic Mechanism Design

AMD considers computational constraints to be of central importance.

- Mechanisms that cannot be efficiently implemented in polynomial time are not considered to be viable solutions to a mechanism design problem.
- Analytic tools of theoretical computer science, such as worst case analysis and approximation ratios, are employed.

Famous and Widespread Example

Auctions

Auctions

Auction: Any protocol that allows agents to indicate their interest in one or more resources, and uses these indications of interest to determine both the allocation of resources and a set of payments by the agents is an auction.

Auctions are important for many computational settings that would not normally be thought of as auctions and that might not even use money as the basis of payments

- E.g. the sharing of computational power in a grid computer

Different Types of Auctions

- Single good:
  - English auction
  - Dutch auction
  - First-price sealed-bid auction
  - Second-price sealed-bid (a.k.a. Vickrey) auction
  - ...
- Multiunit auctions
- Combinatorial auctions
- Double auctions

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders
- Each bidder i has a private valuation (or, willingness to pay or type) θ_i for the item

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders
- Each bidder i has a private valuation (or, willingness to pay or type) θ_i for the item
- Our bidder utility model is quasilinear utility model
  - If i loses, and has to pay p_i, her utility is $-p_i$.
    - In auctions where only winners pay, i's utility is 0.
  - If i wins at a price p_i, his/her utility is $u_i(\theta_i) = \theta_i - p_i$

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders
- Each bidder i has a private valuation (or, willingness to pay or type) θ_i for the item
- Our bidder utility model is quasilinear utility model
  - If i loses, and has to pay p_i, her utility is $-p_i$.
    - In auctions where only winners pay, i's utility is 0.
  - If i wins at a price p_i, his/her utility is $u_i(\theta_i) = \theta_i - p_i$
- We are assuming:
  - **independent private value model**, i.e. a bidder's valuation does not depend on other bidders' valuations.
  - bidders cannot collude.

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders
- Each bidder i has a private valuation (or, willingness to pay or type) θ_i for the item
- Our bidder utility model is quasilinear utility model
  - If i loses, and has to pay p_i, her utility is $-p_i$.
    - In auctions where only winners pay, i's utility is 0.
  - If i wins at a price p_i, his/her utility is $u_i(\theta_i) = \theta_i - p_i$
- We are assuming:
  - **independent private value model**, i.e. a bidder's valuation does not depend on other bidders' valuations.
  - bidders cannot collude.

Who should win the antique book at what price?

Single-Item Auctions: Setting

- A seller with a single item, such as an antique book
- n strategic bidders
- Each bidder i has a private valuation (or, willingness to pay or type) θ_i for the item
- Our bidder utility model is quasilinear utility model
  - If i loses, and has to pay p_i, her utility is $-p_i$.
    - In auctions where only winners pay, i's utility is 0.
  - If i wins at a price p_i, his/her utility is $u_i(\theta_i) = \theta_i - p_i$
- We are assuming:
  - **independent private value model**, i.e. a bidder's valuation does not depend on other bidders' valuations.
  - bidders cannot collude.

Who should win the antique book at what price?

We need an allocation rule + a payment rule

Quiz question 1: What auction is this?

- A seller is selling an antique book.
- Bidders: students present in the virtual classroom.
- You are asked to write down your bid on a piece of paper.

Rule (Allocation Rule)

The item is allocated to the bidder with the highest bid.

Rule (Payment Rule)

The winner is to pay the seller an amount equal to his/her bid.

Quiz question 2: What auction is this?

- A seller is selling an antique book.
- Bidders: students present in the virtual classroom.
- You are asked to write down your bid on a piece of paper.

Rule (Allocation Rule)

The item is allocated to the bidder with the highest bid.

Rule (Payment Rule)

The winner is to pay the seller an amount equal to the second highest bid.

Quiz question 3: What auction is this?

- A seller is selling an antique book.
- Bidders: students present in the virtual classroom.
- Auctioneer starts the bidding at some “reservation price”.
- Bidders then shout out ascending prices.
- The auction is terminated once bidders stop shouting.

Rule (Allocation Rule)

The item is allocated to the bidder who shouted the last bid (the highest bid).

Rule (Payment Rule)

The winner is to pay the seller an amount equal to his/her bid.

What sort/type of a game is a single-item auction?

Vickrey Auction + Bidders = a Game

- Let b_i denote the bid placed by bidder i
- Let $b = (b_1, b_2, \ldots, b_n)$ denote the bid profile of all bidders.

Vickrey Auction + Bidders = a Game

- Let b_i denote the bid placed by bidder i
- Let $b = (b_1, b_2, \ldots, b_n)$ denote the bid profile of all bidders.
- The set of **actions** available to each bidder is all possible bids that s/he can place
  - virtually any non-negative real,
  - unless there are rules in the auction, e.g. "only place integer bids", or "don't place a bid less than £3" (reserve price)

Vickrey Auction + Bidders = a Game

- Let b_i denote the bid placed by bidder i
- Let $b = (b_1, b_2, \ldots, b_n)$ denote the bid profile of all bidders.
- The set of **actions** available to each bidder is all possible bids that s/he can place
  - virtually any non-negative real,
  - unless there are rules in the auction, e.g. "only place integer bids", or "don't place a bid less than £3" (reserve price)
- Let $u_i(b)$ denote the utility of bidder i given bid profile b.
- The **payoff (utility)** of each bidder i depends on the outcome (allocation + payment), which in turn depends on b_i and the bids placed by the other bidders

Vickrey Auction + Bidders = a Game

- Let b_i denote the bid placed by bidder i
- Let $b = (b_1, b_2, \ldots, b_n)$ denote the bid profile of all bidders.
- The set of **actions** available to each bidder is all possible bids that s/he can place
  - virtually any non-negative real,
  - unless there are rules in the auction, e.g. "only place integer bids", or "don't place a bid less than £3" (reserve price)
- Let $u_i(b)$ denote the utility of bidder i given bid profile b.
- The **payoff (utility)** of each bidder i depends on the outcome (allocation + payment), which in turn depends on b_i and the bids placed by the other bidders
- The game induced by the Vickrey auction (in fact, any auction) is a Bayesian game.

Vickrey Auction + Bidders = a Bayesian Game

A (very simple) **toy example**:

- We have two bidders A & B and one item to sell.
- The value of each bidder for the item is an integer $\in \{0, 1, 2\}$.
- The actions available to bidders are declaring one of these three values: $0, 1, 2$.

Vickrey Auction + Bidders = a Bayesian Game

A (very simple) **toy example**:

- We have two bidders A & B and one item to sell.
- The value of each bidder for the item is an integer $\in \{0, 1, 2\}$.
- The actions available to bidders are declaring one of these three values: $0, 1, 2$.

Vickrey auction:

- The highest bid wins and pays the second-highest bid.
- If both bidders bid the same value, then choose a tie-breaking rule, e.g.: **A wins**, or B wins, or neither win (item is unallocated).

Vickrey Auction + Bidders = a Bayesian Game

A (very simple) **toy example**:

- We have two bidders A & B and one item to sell.
- The value of each bidder for the item is an integer $\in \{0, 1, 2\}$.
- The actions available to bidders are declaring one of these three values: $0, 1, 2$.

Vickrey auction:

- The highest bid wins and pays the second-highest bid.
- If both bidders bid the same value, then choose a tie-breaking rule, e.g.: **A wins**, or B wins, or neither win (item is unallocated).
- The loser's utility is zero
- The winner's utility is: winner's value - loser's bid

Vickrey Auction + Bidders = a Bayesian Game, contd.

Normal Form (a.k.a. Strategic-form) Game

A tuple (N, A, u) where

- $N = \{1, \ldots, n\}$ is a finite set of agents.
- $A = A_1 \times \ldots \times A_n$, where A_i is a finite set of actions (i.e. pure strategies) available to agent i.
- $u = (u_1, \ldots, u_n)$, where $u_i : A \mapsto \mathbb{R}$ is the utility (a.k.a. payoff) function for player i.

Attention: In Enrico M recordings (COMP6203), S_i is used to denote the set of pure strategies available to player i. In this module, I use

- S_i to refer to the set of all strategies (pure and mixed) available to agent i, and use s_i to denote a (mixed) strategy of agent i, and
- A_i to denote the set of actions (or, pure strategies) available to agent i.

Bayesian Game

A tuple (N, A, Θ, p, u) where

- $N = \{1, \ldots, n\}$ is a finite set of agents
- $A = A_1 \times \ldots \times A_n$, where A_i is the set of actions available to agent i
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$ where Θ_i is the type space of player i
- $p : \Theta \mapsto [0, 1]$ is a common-prior probability distribution on Θ
- $u = (u_1, \ldots, u_n)$, where $u_i : A \times \Theta \mapsto \mathbb{R}$ is the utility function for player i.

Bayesian Game v.s. Normal-form Game

Bayesian game is a tuple (N, A, Θ, p, u) where

- $N = \{1, \ldots, n\}$ is a finite set of agents
- $A = A_1 \times \ldots \times A_n$, where A_i is the set of actions available to agent i
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$ where Θ_i is the type space of player i
- $p : \Theta \mapsto [0, 1]$ is a common-prior probability distribution on Θ
- $u = (u_1, \ldots, u_n)$, where $u_i : A \times \Theta \mapsto \mathbb{R}$ is the utility function for player i.

Bayesian Game v.s. Normal-form Game

Bayesian game is a tuple (N, A, Θ, p, u) where

- $N = \{1, \ldots, n\}$ is a finite set of agents
- $A = A_1 \times \ldots \times A_n$, where A_i is the set of actions available to agent i
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$ where Θ_i is the type space of player i
- $p : \Theta \mapsto [0, 1]$ is a common-prior probability distribution on Θ
- $u = (u_1, \ldots, u_n)$, where $u_i : A \times \Theta \mapsto \mathbb{R}$ is the utility function for player i.

Relation to normal-form games:

- The types of the agents determine which normal-form game they are playing.
- Agents don't know the type of the other agents, only p is known.
- Based on p, each agent can assign a probability to what game s/he is playing.

Bayesian Game with **Strict Incomplete Information**

is a tuple (N, A, Θ, u) where

- $N = \{1, \ldots, n\}$ is a finite set of agents
- $A = A_1 \times \ldots \times A_n$, where A_i is the set of actions available to agent i
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$ where Θ_i is the type space of player i
- $p : \Theta \mapsto [0, 1]$ is a common-prior probability distribution on Θ
- $u = (u_1, \ldots, u_n)$, where $u_i : A \times \Theta \mapsto \mathbb{R}$ is the utility function for player i.

Sometimes p is not known. That is we have no probabilistic information in the model.

Bayesian Game Setting

A tuple (N, O, Θ, p, u)

- $N = \{1, \ldots, n\}$ is a finite set of agents
- O is a set of outcomes
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$ is a set of possible joint type vector
- p is a common-prior probability distribution on Θ
- $u = (u_1, \ldots, u_n)$, where $u_i : O \times \Theta \mapsto \mathbb{R}$ is the utility function for player i.

The key difference with Bayesian Game is that the Bayesian Game Setting does **not include actions** for the agents, and instead defines the utility function over the **set of possible outcomes**.

Further reading/watching

- Read "Badminton and the science of Rule Making", 2012 Huffington post by Jason Hartline and Robert Kleinberg
- Watch Tim Roughgarden's lecture video Introductory lecture on algorithmic game theory
- Read Tim Roughgarden's lecture notes on Mechanism Design Basics

Further reading/watching

For a thorough introduction to Bayesian Games:

- Read MAS chapter 6.3
- Watch Game Theory I - Week 6 (Bayesian Games)

For further introduction to Mechanism Design

- Read MAS chapters 10.1, 10.2, 10.3 (we haven't covered some of the material in these sections, of which we will cover some in future lectures)
- Read AGT Chapters 9.1, 9.2, 9.3.1, 9.3.2., 9.4.1, 9.4.2 (note that MAS and AGT sometimes use different notations and definitions for the same concepts)
- Watch Game Theory II - Week 2 (Mechanism Design)

Books

- Twenty Lectures on Algorithmic Game Theory, by Tim Roughgarden
- Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations by Yoav Shoham and Kevin Leyton-Brown
  - From now on we will refer to this book as **MAS**
- Algorithmic Game Theory, edited by Noam Nisan, Tim Roughgarden, Eva Tardos,Vijay V. Vazirani
  - From now on we will refer to this book as **AGT**