EC 149 Introduction to Algorithmic Economics

updated: March 28, 2023

Lectures:
Monday & Wednesday 10:30 am — noon, 104 Annenberg

Instructor and TAs:

Fedor Sandomirskiy,
fsandomi@caltech.edu (don't hesitate to reach out with any questions!)
office Baxter 238
office hours: Monday 4 pm-5 pm, Wednesday 2 pm-3 pm (send me an email if these intervals do not work for you)

Sumit Goel,
sgoel@caltech.edu
office: Baxter 218 (subject to change)
office hours: TBD

Eileen Li,
emli@caltech.edu
office: TBD
office hours: TBD

About:
The course is an introduction to an active research area at the intersection of Econ and CS sometimes referred to as algorithmic economics. It deals with multi-agent systems, where agents differ in their preferences and may behave strategically to improve the outcome for themselves. We will discuss applications to voting, crowd-sourcing, rating design, revenue-maximization and auctions, allocation of resources without money, and matching markets. We may also touch on topics such as fairness in ML, differential privacy, and payment design for blockchain.

Algorithmic economics can be roughly divided into algorithmic game theory (rules of interaction are given and we ask how the system behaves) and mechanism design (rules are to be designed). As there is a full-fledged course on game theory per se (PS/EC 172), we will focus on the design perspective and try to avoid intersections as much as possible. The broad topic of matching markets will be touched on briefly; an extensive discussion they deserve is given in EC 117.

Goals:
The course will teach you the basic building blocks and insights used by economists to design multi-agent systems. You will learn the language and the model needed to formulate design goals, tools to achieve them, and some workarounds allowing you to find a reasonable solution if the goals are incompatible.

By the end of the course, you will be able to follow more than half[1] of talks at the top conferences at the intersection of economics and computation such as the annual ACM Conference on Economics and Computation (2022 edition) or start your own research on one of the cutting-edge topics.

Prerequisites:
The course has no prerequisites, in particular, prior knowledge of game theory is not needed. However, it is assumed that you are comfortable with mathematical proofs and know calculus and algebra basics (derivatives, integration, and vectors). Familiarity with the basics of probability theory (random variables, distributions, independence, the Bayes formula) will be useful but all the concepts will be explained in class if needed.

Grading:
There will be weekly or bi-weekly homework, one midterm, and one final exam. The midterm covers the first half of the course material, and the final covers the second half.

Let h, m, f be the fraction of the maximal number of points earned for homework, for the midterm, and for the final, respectively. To get “A”, the score

$$s = 0.5h + 0.25m + 0.25f$$

[1] To follow the other 30%, you need a game-theory course. The remaining 5-10% nobody understands :-)

must be at least 0.8; for "B", s must be at least 0.6. For a passing grade, students must attempt and hand in all the homework, and to earn at least 25% of the total number of points in the midterm and in the final ($\min\{m, f\} \geq 0.25$).

Students should form groups to work on home assignments. Each student is expected to first attempt all the problems individually, and then meet as a group to discuss and arrive at a consensus solution for each problem. Finally, each group should hand in a single common solution to the problem set. A group can have a size of 1, 2, or 3.

Readings:
No books are required. All the essential material will be contained in lecture notes. You can find notes from the 2022 edition: here. Updated notes will be distributed weekly.

You can also find the following sources useful (all freely available online):

- Game Theory, Alive by Anna Karlin and Yuval Peres, in my opinion, is the best introduction to both game theory and mechanism design for readers with math or CS background.

- Handbook of computational social choice edited by Felix Brandt, Vincent Conitzer, Ulle Endriss, Jerome Lang, and Ariel Procaccia
  Topics: voting (Section 2), judgment aggregation (Section 17), fair division (Section 13).

- Handbook of Algorithmic Game Theory edited by Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani
  Topics: competitive equilibrium (Chapter 5), social choice (Chapters 9,10)

- Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations by Yoav Shoham and Kevin Leyton-Brown, Cambridge University Press, 2009
  Topics: voting (Section 9), VCG (Section 10), auctions (Section 11).

I also recommend amazing lecture notes by Tim Roughgarden's "Incentives in Computer Science" from where I borrowed quite a few ideas. Some inspiration came from courses by Ariel Procaccia; slides are available on his website.

Outline:

- **Voting and social choice.** Normative approach. May's theorem. The irrationality of majority: the Condorcet paradox and judgment

aggregation. Condorcet consistency. Ranking aggregation and Arrow's theorem. Manipulations, the Gibbard–Satterthwaite theorem. Escaping impossibilities via domain restriction. Single-peaked domain: aggregating expert's opinions on a line, median VS mean.
Optional: Gil Kalai's proof of Arrow's theorem via Fourier transform. The complexity of manipulation. Crowdsourcing: noisy votes as ground truth estimators. Peer-review mechanisms.

- **Auctions.** Resource allocation with money: quasi-linear domain. First-price and second-price auctions. Welfare. VCG mechanism, sponsored search auctions. Combinatorial auctions. Computational complexity and bidding language complexity as design concerns.
  Optional: Dynamic mechanisms: clock auctions. Payment design for blockchain.

- **Revenue-maximization.** Revenue-equivalence theorem, reserve prices and revenue-maximizing auctions, the Bulow-Klemperer theorem (proofs via examples).
  Optional: Menu mechanisms and bundling. All-pay auctions and crowdsourcing contests.

- **Fairness as design concern.** Cake-cutting. Envy-freeness as equal choice opportunities. Pseudo-markets and the Eisenberg-Gale theorem. Dominant resource fairness and allocation of computational resources in cloud computing. Indivisible goods, EF1 and EFX.
  Optional: Participatory budgeting and donations channeling. Fairness in classification. Differential privacy. Fair rent division and cake-cutting with connected pieces via Sperner's lemma.

- **Two-sided markets without money.** Stability. Gale-Shapley algorithm. Manipulations. Practical design concerns.
  Optional: Lattice structure, and typical properties of large markets (the stark effect of competition).

- **Selective information provision as a design tool.** Bayesian persuasion. Social learning and herding. Implications for platform markets and recommendation systems design.