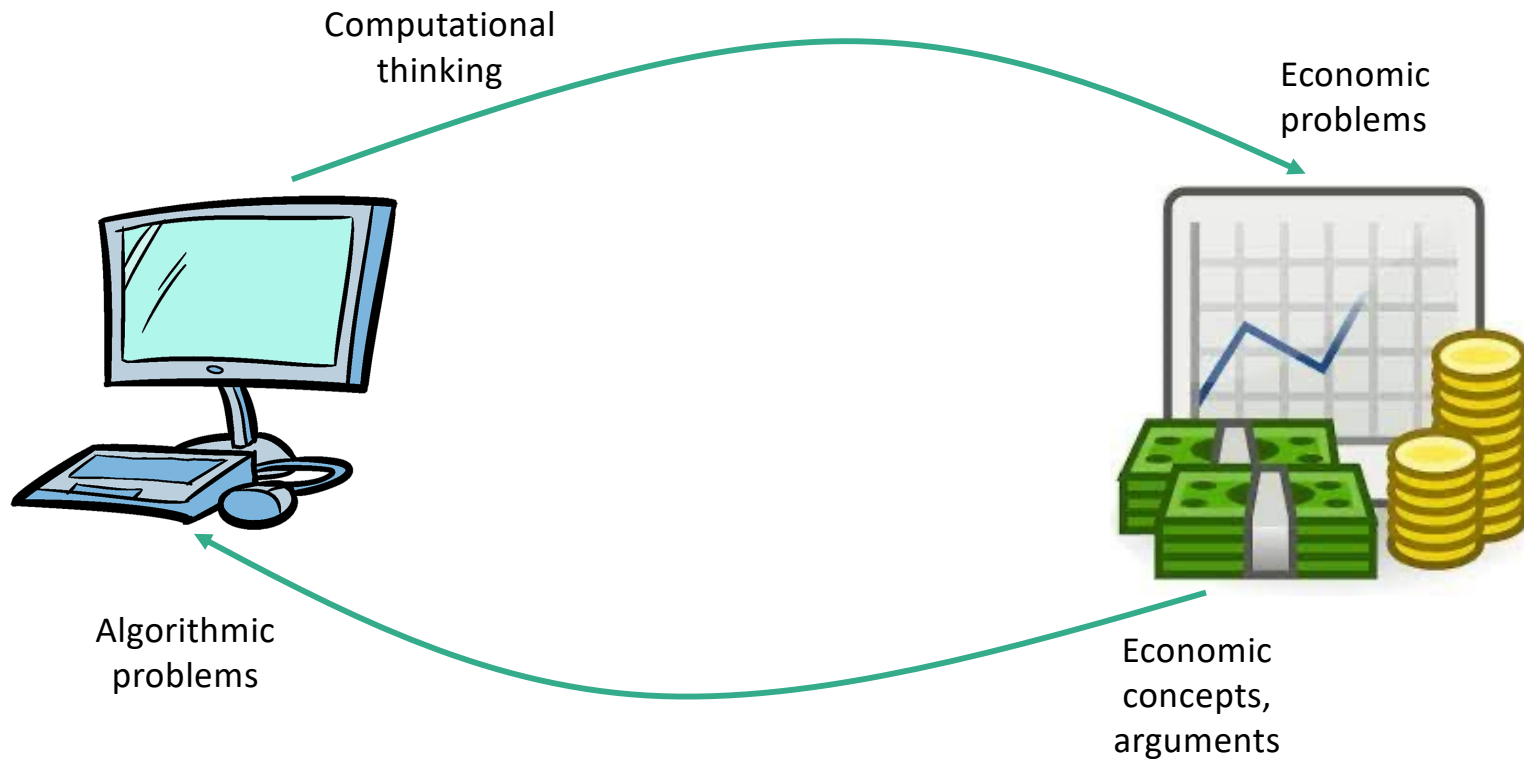


DS 574: Algorithmic Mechanism Design

PROFESSOR KIRA GOLDNER

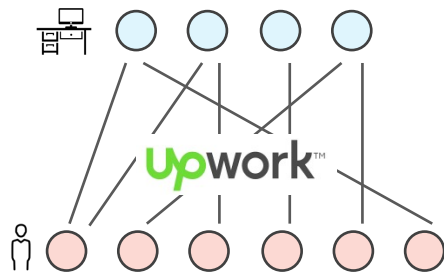
What is “EconCS”?

Also referred to as:
Algorithmic Game Theory (AGT)

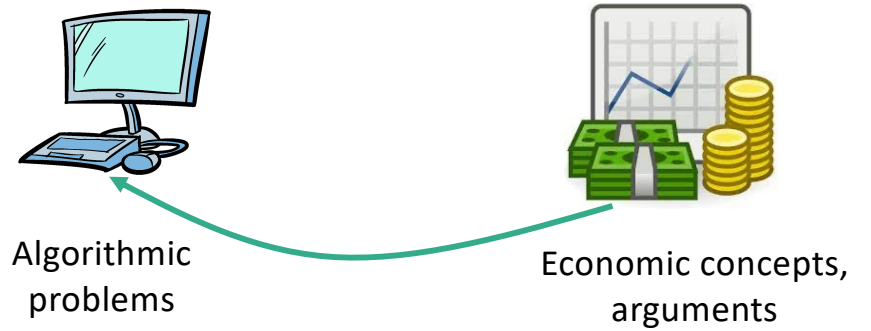
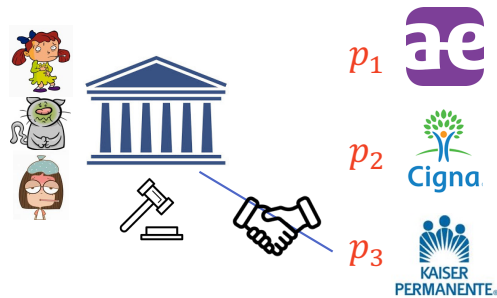


Econ → CS

Online Labor Markets

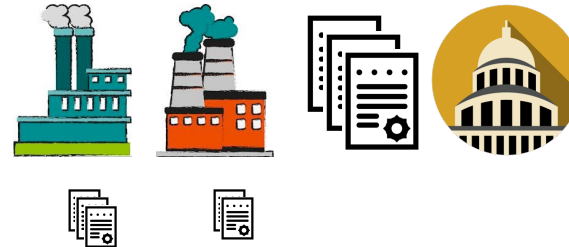


Health Insurance



- The systems interact with **strategic individuals**.
- We must **design** them to be **robust** to **strategic behavior**.

Carbon Emissions



Econ → CS



Algorithmic problems

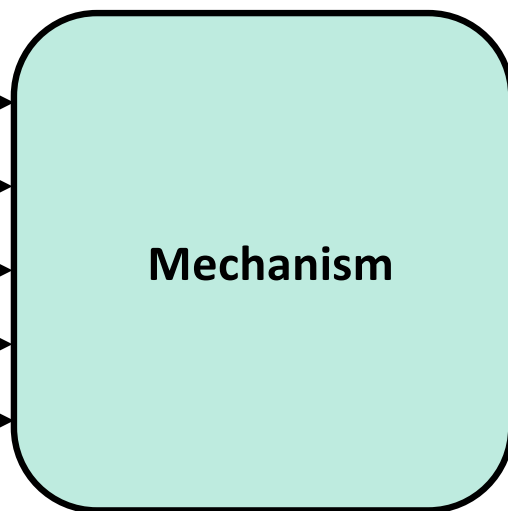


Economic concepts, arguments

Input:
Data reported by
strategic agents.



Objective: Maximize
buyer's value



Output:

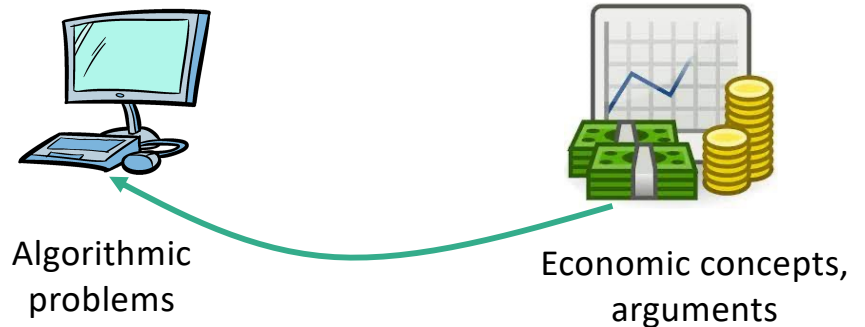
-who gets what
-who pays (gets paid) what



Use **game theory** to reason about **incentives** within the **algorithm** so that we can **guarantee** (approximate) optimality.

Econ → CS

Elegant **proofs** using an economic lens:



Maximum weight matching [Demange Gale Sotomayor '86]

- LHS runs ascending auction “bidding” on RHS until perfecting matching achieved.

Online bipartite matching [Karp Vazirani Vazirani '90]

- Algorithm: Randomly permute RHS. LHS arrives and takes first available item in LHS according to permutation.
- Prove this using elegant random price argument. [Eden Feldman Fiat Segal '21]

CS → Econ



1 item



- Simple.
- Easy to compute.
- Only one real option.

[Myerson '81]

\$5: $\Pr[\text{apple}] = 1$

Computational thinking



Economic problems



Open Problem: What optimal mechanisms can we characterize beyond 1 item?

2 items



- Uncountably infinite options. [Manelli Vincent '07, Daskalakis Deckelbaum Tzamos '15]
- Intractable to compute. [Daskalakis Deckelbaum Tzamos '13]
- We still know very little about how to do this.

\$5.89: $(\Pr[\text{apple}] = .60, \Pr[\text{orange}] = .29)$

CS → Econ



1 item



easy
[Mye'81]

FedEx



explicit
[FGKK'16]

Multi-Bidder FedEx
[WZ '21]

Approximate FedEx
[SSW '18]

Budgets
\$5, \$10, \$12 budgets
[DW '17]

Computational thinking



Economic problems



2 items



intractable, chaos
[DDT'13,'15; MV'07]

Single-Minded
[DGSSW '20]

Multi-Unit Pricing
1,2,3-cap for documents
[DHP '17, DGSSW '20]

Coordinated Valuations
Wifi, +TV, +Cable [w/ g(v)]
[DGSSW '20]

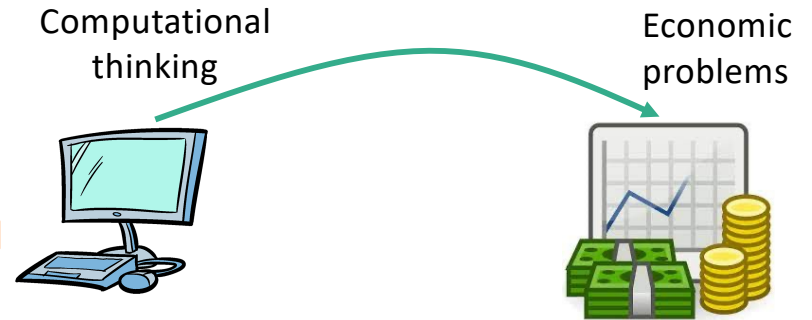
CS → Econ

2 items



- Uncountably infinite options. [Manelli Vincent '07, Daskalakis Deckelbaum Tzamos '15]
- Intractable to compute. [Daskalakis Deckelbaum Tzamos '13]
- We still know very little about how to do this.

\$5.89: (Pr[🍏]=.60, Pr[🍊]=.29)



Simple Mechanisms



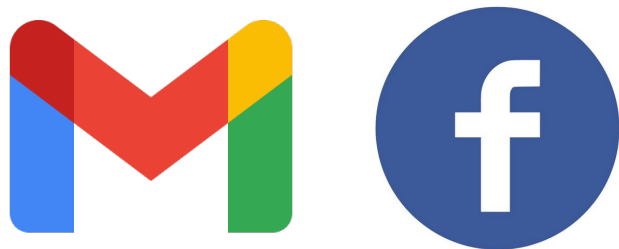
(Lack of) information



Robustness

Why is this important to learn about?

Mechanism Design and Society



Computationally Efficient:

- To design.
- To run.
- To strategize within.



Mechanism Design and Society

Settings where:

- Allocations are a mess.
- There are perverse incentives.

Computationally Efficient:

- To design.
- To run.
- To strategize within.

Health Insurance



What should you expect to learn?

- Mechanism Design basics (welfare, revenue, environments)
 - Similar to other MD/EconCS courses. Probably the only part that is.
- Mechanism Design for Social Good
- Robustness
- New frontiers (two-sided markets, interdependent values, fairness)
- LP Duality applied to mechanism design

Where can you go after this course?

Research in related fields:

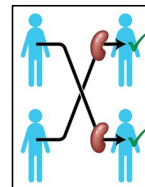
- EconCS (from CS)
- Operations Research (IE or Business)
- Microeconomic theory
- Some interdisciplinary split!

Add incentives or an economics perspective to your research:

- Privacy for strategic agents
- Learning with strategic agents

Related industries:

- Platform economics
- Allocation systems in welfare or industry
- Legal regulation (when is regulation better than markets?)

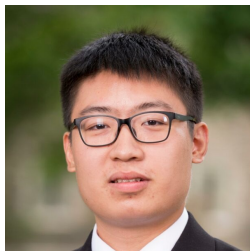


Logistics

Teaching Staff

Instructor: Prof. Kira Goldner
Email: goldner@bu.edu
OH: Tues 3-4PM & by appointment
Office Location: CCDS 1339

TF: Peiran Xiao
Email: pxiao@bu.edu
OH: TBD
Location: TBD



Time of Class

Class is scheduled 1:30-3:15pm.

Typically, class will occupy 1:30-2:45pm.

Because I have some travel scheduled, sometimes class will be canceled and made up by extending other classes until 3:15pm. You will be notified in advance (via the class website).

On those Tuesdays, office hours will occur 3:30-4:30pm (instead of 3-4pm).

Class Resources

Course website: <https://www.kiragoldner.com/teaching/DS574/>

- Lecture notes, links to everything

Piazza (access code AMD):

- Questions and answers; **alternative for email**
- I am a human who does not live inside the computer!

Gradescope (entry code ZZV4DV):

- Turn in assignments and view grades



Sign up for these if you have not already! (Links on... the course website!)

This is a theoretical problem-solving class

No programming assignments! Evaluation based on problem sets and project.

Prerequisites:

- A first proofs class that's Discrete-Math-esque (DS 121, CS 131, MA 293, ...)
- Undergrad algorithms (DS 320, CS 330, ...)—algorithmic reasoning, runtime and complexity notions
- Intro probability (MA 581)—know r.v.s and compute their moments
- Mathematical maturity

Not expected:

- Any background in game theory/incentives/economics.

Evaluation

Homework (45%)

- Collaborative problem sets ~every other week.

Mechanism Design for Social Good problem formulation (15%)

- Formulate a problem and defend why the question is important both for the domain and within mechanism design. Identify a domain expert for potential collaboration.

Class participation (5%)

- In class and via Piazza (asking and answering questions) gets 100% here.

Final Project (35%)

- Investigate a research question not covered in class—read papers and write a survey OR do original research. Write up and presentation.

Homework Policies

- Expect to spend at least 10 hours per assignment.
- **Late policy:** You have 4 late days, max 2 per assignment (integer numbers used only). No exceptions.
- Type up homework with **LaTeX**.
- Turn in via **gradescope**. Due at 11:59pm on the date assigned.

- **Regrades:** Requests within 7 days, only via gradescope, with explanation/argument. Only for **incorrect** grading (not insufficient credit). If you request a regrade, the whole assignment/exam may be regraded, and your score may go up or down.

Collaboration Policy

Collaboration is encouraged!!!

- You may work with up to two classmates on an assignment. **List your collaborators' names on your assignment. (E.g., Collaborators: None.)**
- Good rough rule: Nobody should leave the room with anything written down. If you really understand, you should be able to reconstruct it on your own.
- You may **not** use the internet on homework problems. You may use course materials and the recommended readings from textbooks.

I believe **strongly** in learning over evaluation, learning via collaboration, and academic integrity. Please adhere to BU's academic conduct policy.

Class Etiquette

This is an interactive class taught via worksheets—**bring your worksheet to class** (available on the course website).

I strive toward an accessible and equitable classroom for all students.

- Raise your hand.
- Be conscious of how often you participate (in class and in collaboration).
 - Don't talk over others, leave room for other voices if you speak up a lot, and speak up more if you do not.

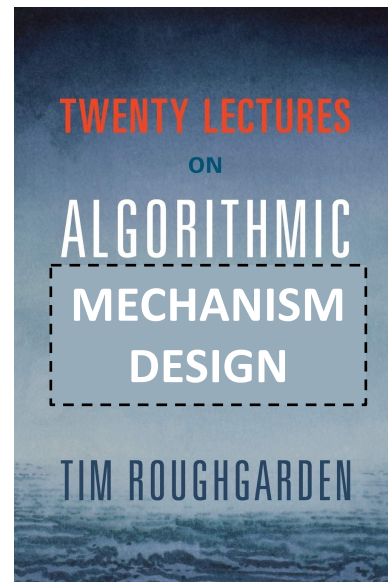
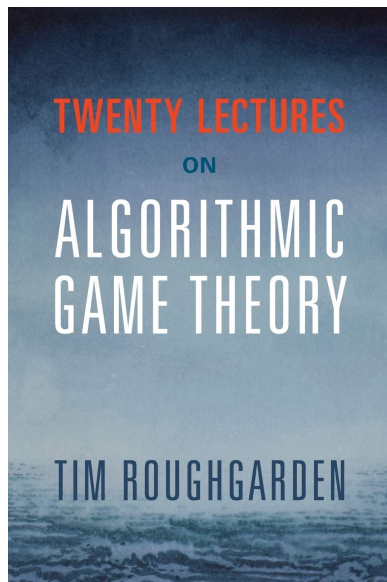
But also

- Ask questions!!!!!!

Best advice I ever got was to just ask and not wait to fill in gaps myself later.

Book

There is no required textbook, and the lecture notes will be self contained. But many of the topics we are covering are well covered in standard algorithms textbooks; some lectures are adapted from Tim Roughgarden's lecture notes.



arXiv



Introductions!

- Name
- Department + Year
- Why are you taking this class?
- Somewhere you've been that you think no one else here has been

Let's get started!
