Optimizing in a Strategic World: An Introduction to Algorithmic Game Theory

Anna R. Karlin
University of Washington

An Example
Classical Optimization Problem: Maximum Weighted Matching
Input: Weighted Bipartite Graph
Output: Matching that maximizes the sum of matched edge weights.

Example Application
Selling advertising slots
- A search engine has advertising slots for sale
- Advertisers are willing to pay different amounts to have their ad shown in a particular slot

Optimal Search Engine Revenue = maximum weighted matching
Private Values

- Algorithm must solicit values
- Advertisers may lie to get a better deal

What if all advertisers speculate?

Big Picture

Many problems where input is private data of agents who will act selfishly to promote best interests

- Resource allocation
- Routing and congestion control
- Electronic commerce

Fundamental Question:
How do we optimize in a strategic world?

Use ideas from game theory and economics.
Outline of talk

- Background in game theory
- Points of contact between computer science and game theory
  - Computational economics
  - Selfish routing and the price of anarchy
- Mechanism design = “incentive engineering”
  - Primary application: ad auctions

Game Theory

- Game: players, strategies, payoff functions
- Given a set of players, each with a set of strategies, each motivated by self-interest, what constitutes rational behavior?

### Chicken

<table>
<thead>
<tr>
<th></th>
<th>Swerve</th>
<th>Straight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swerve</td>
<td>(0,0)</td>
<td>(-1,+1)</td>
</tr>
<tr>
<td>Straight</td>
<td>(+1,1)</td>
<td>(-5,-5)</td>
</tr>
</tbody>
</table>

### Matching Pennies

<table>
<thead>
<tr>
<th></th>
<th>Swerve</th>
<th>Straight</th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swerve</td>
<td>(0,0)</td>
<td>(-1,+1)</td>
<td>(+1,-1)</td>
<td>(-1,+1)</td>
</tr>
<tr>
<td>Straight</td>
<td>(+1,-1)</td>
<td>(-5,-5)</td>
<td>(-1,1)</td>
<td>(+1,-1)</td>
</tr>
</tbody>
</table>

- Fundamental Concepts: various equilibria
  - Dominant strategies: each player has a well-defined best strategy no matter what any of the other players do.
  - Nash equilibrium: each player’s strategy is best response to opponents’ strategies.
Game Theory

- John von Neumann: Every zero-sum game has a mixed Nash equilibrium.
- John Nash: Every finite game has a mixed Nash equilibrium.

### Chicken

<table>
<thead>
<tr>
<th></th>
<th>Swerve</th>
<th>Straight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swerve</td>
<td>(0,0)</td>
<td>(-1,1)</td>
</tr>
<tr>
<td>Straight</td>
<td>(+1,-1)</td>
<td>(-5,-5)</td>
</tr>
</tbody>
</table>

### Matching Pennies

<table>
<thead>
<tr>
<th></th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heads</td>
<td>(+1,-1)</td>
<td>(-1,1)</td>
</tr>
<tr>
<td>Tails</td>
<td>(-1,1)</td>
<td>(+1,-1)</td>
</tr>
</tbody>
</table>

Outline of talk

- Background in game theory
- Points of contact between computer science and game theory
  - Computational economics
  - Selfish routing and the price of anarchy
  - Mechanism design = "incentive engineering"
    - Primary application: ad auctions

Computational Economics

- Many existential equilibrium results:
  - Nash equilibria \([\text{von Neumann}] \text{[Nash]}\)
  - Market equilibria \([\text{Arrow}, \text{Debreu}]\) \(\text{another Nobel prize}\)
  - Until recently, not much algorithmic theory.

- The big question: When can the participants of a game quickly converge to an equilibrium? When can a centralized or better yet decentralized algorithm quickly compute an equilibrium?

- We care because equilibria are predictions of behavior.
Computational Economics

- Many existential equilibrium results: Nash equilibria, Market equilibria, etc.
- The big question: When can the participants of a game quickly converge to an equilibrium?

What we know:
- Strong theoretical evidence that Nash equilibria cannot be computed efficiently
- On the other hand, several significant positive results regarding market equilibria
- Important CS contribution to the discussion on equilibrium notions in economics.

Outline of talk

- Background in game theory
- Points of contact between computer science and game theory
  - Computational economics
  - Selfish routing and the price of anarchy
- Mechanism design = "incentive engineering"
  - Primary application: ad auctions

Price of Anarchy

- Explores games that arise "in the wild", such as in Internet settings.
- Tries to understand the impact of selfish behavior on society by comparing the overall performance attained in equilibrium when players behave selfishly to the performance that could be attained if decisions were made by a centralized authority.

Modeling Network Traffic using Game Theory

- Model network as directed graph.
- We assume network users are selfish — in equilibrium each user will choose a route that minimizes their travel time, given what everyone else is doing.
- In this example, equal balance of traffic is a Nash equilibrium.

Edge labels: latency as function of fraction of traffic
**Braess’s Paradox**
- Small changes can lead to counterintuitive behavior.
- Example: Government builds a new, very fast highway.

**Price of Anarchy [KP]**
- Example of larger phenomenon: selfishness can be bad for society. How bad?

\[
\text{Price of Anarchy} = \frac{\text{Avg latency at selfish equilibrium}}{\text{Avg latency at social optimum}} = \frac{4}{3}
\]

**Outline of talk**
- Background in game theory
- Points of contact between computer science and game theory
  - Computational economics
  - Selfish routing and the price of anarchy
- Mechanism design = "incentive engineering"
  - Primary application: ad auctions

**Price of Anarchy**
- How bad can selfishness be for society?

\[
\text{Price of Anarchy} = \frac{\text{Avg latency at selfish equilibrium}}{\text{Avg latency at social optimum}}
\]

is always at most \(\frac{4}{3}\)
Mechanism Design
Also called inverse game theory

Given desired goals (e.g. maximizing profit or social welfare),
design the game in such a clever way that rational
players, motivated solely by self interest, will end
up achieving the designer's goals.

Examples:
- Designing a system of tolls on roads (pricing link
  usage in computer networks).
- Selling advertisement slots on a search engine.

Online Advertising

• Multi-billion dollar industry and growing rapidly.

• One of the primary forms:
  - Sponsored search advertising: Advertisers bid to
    have their ads shown alongside search results in
    response to a searcher's query.

A Revolution in Advertising

• Highly targeted

• Only pay when ad is clicked on

• Control how much spend via daily budget.

• Can take into account demographics, geography,
  behavior, etc.
How sponsored search results get there

- Advertisers submit bids for keywords of interest (like "airline tickets") => at all times, search engine has database of standing bids.

- When user searches for a keyword => some advertisers bidding on that keyword are chosen and their advertisements appear in the sponsored search slots.

- User clicks on displayed ad => advertiser pays the search engine some amount.

Optimization issues for search engine

- Which advertisers to allocate which slots to
- What prices to charge

Let's start with a very special case:
- A single ad slot

Selling a single slot (or item)...

Goal: design auction to maximize profit

- Common way: English auction (i.e. ascending-bid auction).
- Not convenient in real-time.

Run a sealed-bid auction.
- Each bidder submits a bid.

Most natural allocation and pricing:
- Highest bidder wins and pays what they bid.

First-price auction

- What will happen?
- To answer need to understand payoff structure.

Standard bidder model:
- Each bidder has a private value $v_i$ for the item.
- Player utility: $[v_i - \text{price}]$ if wins, 0 otherwise
- Each player submits a bid $b_i$ for the item

Problems with first-price auction
- Players will "shade" their bids.
- But they have to speculate about values/bids of others => figuring out how to bid is complicated.
- Outcome unpredictable.

Is this (first-price auction) a good choice?
The Vickrey 2nd price auction.

- Alternative auction design:
  - Players submit bids.
  - The player that submitted the highest bid wins, but only pays the second highest bid submitted.
  - (68, 100, 2) player submitting 100 wins at price 68

- Key property: it is in each player’s best interest to bid their true value, no matter what the other players do!

Vickrey 2nd Price Auction

Key property: it is in each player’s best interest to bid their true value, no matter what!

Remarks:

- Players don’t need to think about how to bid, because it’s in their best interest to tell the truth!

- Outcome is same as that of English auction!

- The item will go to the player that values it the most!

- It has been proven* that, in equilibrium, seller profit using 1st and 2nd price auction is the same.
  * under modest assumptions

Sponsored Search Advertising:

- auction for a single keyword

Advertisement Slots

Advertisers bids

$0.5

$0.7

$1.1

$0.2

$0.4

$1.3

$1.0

$0.5

$1.2

Values

$0.5

$0.7

$1.3

$1.0

$0.5

bids
The mechanism we just saw

GSP (generalized second price):

- If there are $k$ slots for advertisements, $k+1$ highest bidders are selected and ranked by expected revenue to search engine.
- On each click, corresponding bidder pays search engine the minimum amount needed to maintain its slot.

What do we know about GSP, the game?

- It’s a game in which the players are the advertisers.
  - Advertisers are bidding to maximize their utility
- Unlike Vickrey 2nd price auction, it is not in each player’s best interest to bid truthfully
- Has continuum of (pure) Nash equilibria.
- These NE achieve varying tradeoffs between social welfare (total happiness) and search engine profit.

What about search engine profit?

- In GSP, each winning bidder pays the bid of the next highest bidder.
- Why not just charge them what they bid?
- One answer: massively unstable

Sponsored search auctions

We saw just one tiny piece of the game.

Many interesting open questions

- How to design for this highly dynamic and complex problem?
**Take-home messages**

- When contemplating a change to a system that is used by “strategic users”, one needs to understand implications!

- Rationality assumptions of GT increasingly relevant for CS games because participants are programs programmed to behave rationally.

- However, these programs are computationally limited...

**Conclusions**

- Online advertising is just one corner in an exciting subfield of computer science called *algorithmic game theory*.

- Recommended reading: book on this topic edited by Nisan, Roughgarden, Tardos, Vazirani

Thank you for your attention!