TWENTY LECTURES IN ALGORITHMIC GAME THEORY

TIM ROUGHGARDEN

Twenty Lectures on Algorithmic Game Theory

Computer science and economics have engaged in a lively interaction over the past 15 years, resulting in the new field of algorithmic game theory. Many problems central to modern computer science, ranging from resource allocation in large networks to online advertising, involve interactions between multiple self-interested parties. Economics and game theory offer a host of useful models and definitions to reason about such problems. The flow of ideas also travels in the other direction, and concepts from computer science are increasingly important in economics.

This book grew out of the author's Stanford course on algorithmic game theory, and aims to give students and other newcomers a quick and accessible introduction to many of the most important concepts in the field. The book also includes case studies on online advertising, wireless spectrum auctions, kidney exchange, and network management.

Tim Roughgarden is an Associate Professor of Computer Science at Stanford University. For his research in algorithmic game theory, he has been awarded the ACM Grace Murray Hopper Award, the Presidential Early Career Award for Scientists and Engineers (PECASE), the Kalai Prize in Game Theory and Computer Science, the Social Choice and Welfare Prize, the Mathematical Programming Society's Tucker Prize, and the EATCS-SIGACT Gödel Prize. He wrote the book *Selfish Routing and the Price of Anarchy* (2005) and coedited the book *Algorithmic Game Theory* (2007).

Twenty Lectures on Algorithmic Game Theory

Tim Roughgarden
Stanford University, California

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom
One Liberty Plaza, 20th Floor, New York, NY 10006, USA
477 Williamstown Road, Port Melbourne, VIC 3207, Australia
4843/24, 2nd Floor, Ansari Road, Daryaganj, Delhi - 110002, India
79 Anson Road, #06-04/06, Singapore 079906

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning, and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781107172661
10.1017/9781316779309

First published 2016

Printed in the United States of America

A catalogue record for this publication is available from the British Library.

Library of Congress Cataloging in Publication Data
Names: Roughgarden, Tim.
Title: Twenty lectures on algorithmic game theory / Tim Roughgarden, Stanford University, California.
Description: Cambridge : Cambridge University Press, 2016. | Includes bibliographical references and index.
Identifiers: LCCN 2016028351 | ISBN 9781107172661 (hardback : alk. paper)
Subjects: LCSH: Game theory. | Algorithms.
Classification: LCC QA269 .R68 2016 | DDC 519.3–dc23
LC record available at https://lccn.loc.gov/2016028351

ISBN 978-1-107-17266-1 Hardback
ISBN 978-1-316-62479-1 Paperback

To Emma

Contents

Preface

Computer science and economics have engaged in a lively interaction over the past 15 years, resulting in a new field called *algorithmic game theory* or alternatively *economics and computation.* Many problems central to modern computer science, ranging from resource allocation in large networks to online advertising, fundamentally involve interactions between multiple self-interested parties. Economics and game theory offer a host of useful models and definitions to reason about such problems. The flow of ideas also travels in the other direction, as recent research in computer science complements the traditional economic literature in several ways. For example, computer science offers a focus on and a language to discuss computational complexity; has popularized the widespread use of approximation bounds to reason about models where exact solutions are unrealistic or unknowable; and proposes several alternatives to Bayesian or average-case analysis that encourage robust solutions to economic design problems.

This book grew out of my lecture notes for my course "Algorithmic Game Theory," which I taught at Stanford five times between 2004 and 2013. The course aims to give students a quick and accessible introduction to many of the most important concepts in the field, with representative models and results chosen to illustrate broader themes. This book has the same goal, and I have stayed close to the structure and spirit of my classroom lectures. Brevity necessitates omitting several important topics, including Bayesian mechanism design, compact game representations, computational social choice, contest design, cooperative game theory, incentives in cryptocurrencies and networked systems, market equilibria, prediction markets, privacy, reputation systems, and social computing. Many of these areas are covered in the books by Brandt et al. (2016), Hartline (2016), Nisan et al. (2007), Parkes and Seuken (2016), Shoham and Leyton-Brown (2009), and Vojnović (2016).

Reading the first paragraph of every lecture provides a quick sense of the book's narrative, and the "top 10 list" on pages 299–300 summarizes the key results in the book. In addition, each lecture includes an "Upshot" section that highlights its main points. After the introductory lecture, the book is loosely organized into three parts. Lectures 2–10 cover several aspects of "mechanism design"—the science of rule-making—including case studies in online advertising, wireless spectrum auctions, and kidney exchange. Lectures 11–15 outline the theory of the "price of anarchy"—approximation guarantees for equilibria of games found "in the wild," such as large networks with competing users. Lectures 16–20 describe positive and negative results for the computation of equilibria, both by distributed learning algorithms and by computationally efficient centralized algorithms. The second and third parts can be read independently of the first part. The third part depends only on Lecture 13, with the exceptions that Sections 16.2–16.3 depend on Section 12.4 and Section 16.4 on Lecture 14. The starred sections are the more technical ones, and they can be omitted on a first reading.

I assume that the reader has a certain amount of mathematical maturity, and Lectures 4, 19, and 20 assume familiarity with polynomial-time algorithms and $\mathcal{NP}$-completeness. I assume no background in game theory or economics, nor can this book substitute for a traditional book on these subjects. At Stanford, the course is attended by advanced undergraduates, masters students, and first-year PhD students from many different fields, including computer science, economics, electrical engineering, operations research, and mathematics.

Every lecture concludes with brief bibliographic notes, exercises, and problems. Most of the exercises fill in or reinforce the lecture material. The problems are more difficult, and often take the reader step-by-step through recent research results. Hints to exercises and problems that are marked with an "*(H)*" appear at the end of the book.

Videos of my classroom lectures in the most recent (2013) offering of the course have been uploaded to YouTube and can be accessed through my home page (`www.timroughgarden.org`). Lecture notes and videos on several other topics in theoretical computer science are also available there.

I am grateful to all of the Stanford students who took my course,

which has benefited from their many excellent questions and comments. I am especially indebted to my teaching assistants: Peerapong Dhangwatnotai, Kostas Kollias, Okke Schrijvers, Mukund Sundararajan, and Sergei Vassilvitskii. Kostas and Okke helped prepare several of the figures in this book. I thank Yannai Gonczarowski, Warut Suksompong, and Inbal Talgam-Cohen for particularly detailed feedback on an earlier draft of this book, and Lauren Cowles, Michal Feldman, Vasilis Gkatzelis, Weiwei Jiang, Yishay Mansour, Michael Ostrovsky, Shay Palachy, and Rakesh Vohra for many helpful comments. The cover art is by Max Greenleaf Miller. The writing of this book was supported in part by NSF awards CCF-1215965 and CCF-1524062.

I always appreciate suggestions and corrections from readers.

Stanford University
Stanford, California

Tim Roughgarden
June 2016

Lecture 1

Introduction and Examples

This book has three parts, each with its own overarching goal. Lectures 2–10 develop tools for designing systems with strategic participants that have good performance guarantees. The goal of Lectures 11–15 is to understand when selfish behavior is largely benign. Lectures 16–20 study if and how strategic players reach an equilibrium of a game. The three sections of this lecture offer motivating examples for the three parts of the book.

1.1 The Science of Rule-Making

We begin with a cautionary tale. In 2012, the Olympics were held in London. One of the biggest scandals of the event concerned, of all sports, women's badminton. The scandal did not involve any failed drug tests, but rather a failed tournament design that did not carefully consider *incentives*.

The tournament design used is familiar from World Cup soccer. There are four groups (A, B, C, D) of four teams each. The tournament has two phases. In the first "round-robin" phase, each team plays the other three teams in its group, and does not play teams in other groups. The top two teams from each group advance to the second phase, while the bottom two teams from each group are eliminated. In the second phase, the remaining eight teams play a standard "knockout" tournament. There are four quarterfinals, with the losers eliminated, followed by two semifinals, with the losers playing an extra match to decide the bronze medal. The winner of the final gets the gold medal, the loser the silver.

The incentives of participants and of the Olympic Committee and fans are not necessarily aligned in such a tournament. What does a team want? To get as prestigious a medal as possible. What does the Olympic Committee want? They didn't seem to think carefully about this question, but in hindsight it is clear that they wanted every team to try their best to win every match. Why would a team ever want to lose a match? Indeed, in the knockout phase of the tournament, where losing leads to instant elimination, it is clear that winning is always better than losing.

To understand the incentive issues, we need to explain how the eight winners from the round-robin phase are paired up in the quarterfinals (Figure 1.1). The team with the best record from group A plays the second-best team from group C in the first quarterfinal, and similarly with the best team from group C and the second-best team from group A in the third quarterfinal. The top two teams from groups B and D are paired up analogously in the second and fourth quarterfinals. The dominoes started to fall when, on the last day of round-robin competition, there was a shocking upset: the Danish team of Pedersen and Juhl (PJ) beat the Chinese team of Tian and Zhao (TZ), and as a result PJ won group D with TZ coming in second. Both teams advanced to the knockout stage of the tournament.

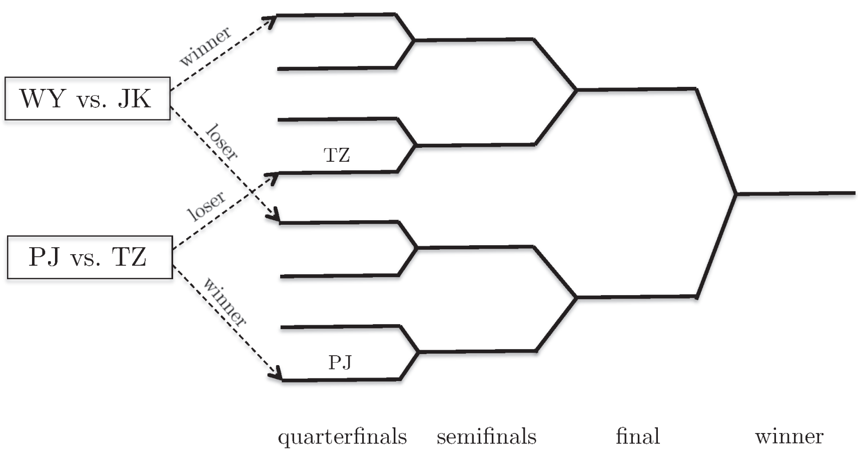


Figure 1.1: The women's badminton tournament at the 2012 Olympics. Both WY and JK preferred to play TZ in as late a round as possible.

The first controversial match involved another team from China, Wang and Yu (WY), and the South Korean team of Jung and Kim (JK). Both teams had a 2-0 record in group A play. Thus, both were headed for the knockout stage, with the winner and loser of this match the top and second-best team from the group, respectively. Here was the issue: the group A winner would likely meet the fearsome TZ team in the semifinals of the knockout stage, where a loss means a bronze medal at best, while the second-best team in group A would not face TZ until the final, with a silver medal guaranteed. Both the WY and JK teams found the difference between these two scenarios significant enough to try to deliberately lose the match![1] This unappealing spectacle led to scandal, derision, and, ultimately, the disqualification of the WY and JK teams.[2] Two group C teams, one from Indonesia and a second team from South Korea, were disqualified for similar reasons.

The point is that, in systems with strategic participants, *the rules matter*. Poorly designed systems suffer from unexpected and undesirable results. The burden lies on the system designer to anticipate strategic behavior, not on the participants to behave against their own interests. We can't blame the badminton players for optimizing their own medal placement.

There is a well-developed science of rule-making, the field of *mechanism design*. The goal in this field is to design rules so that strategic behavior by participants leads to a desirable outcome. Killer applications of mechanism design that we discuss in detail include Internet search auctions, wireless spectrum auctions, the matching of medical residents to hospitals, and kidney exchanges.

Lectures 2–10 cover some of the basics of the traditional economic approach to mechanism design, along with several complementary contributions from computer science that focus on computational efficiency, approximate optimality, and robust guarantees.

1.2 When Is Selfish Behavior Near-Optimal?

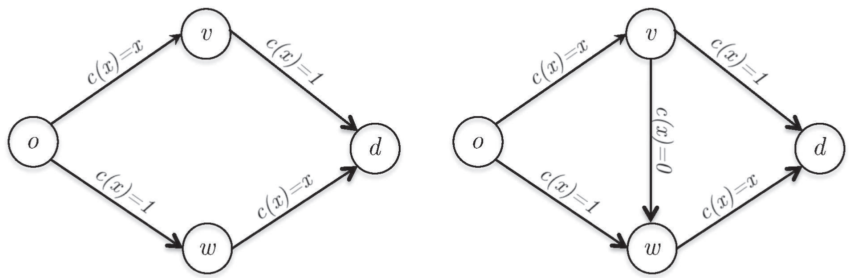
1.2.1 Braess's Paradox

Sometimes you don't have the luxury of designing the rules of a game from scratch, and instead want to understand a game that occurs

[1] In hindsight, it seems justified that the teams feared the Chinese team TZ far more than the Danish team PJ: PJ were knocked out in the quarterfinals, while TZ won the gold medal.

[2] If you're having trouble imagining what a badminton match looks like when both teams are trying to lose, by all means track down the video on YouTube.

"in the wild." For a motivating example, consider *Braess's paradox* (Figure 1.2). There is an origin o, a destination d, and a fixed number of drivers commuting from o to d. For the moment, assume that there are two non-interfering routes from o to d, each comprising one long wide road and one short narrow road (Figure 1.2(a)). The travel time on a long wide road is one hour, no matter how much traffic uses it, while the travel time in hours on a short narrow road equals the fraction of traffic that uses it. This is indicated in Figure 1.2(a) by the edge labels "$c(x) = 1$" and "$c(x) = x$," respectively. The combined travel time in hours of the two edges in one of these routes is $1 + x$, where x is the fraction of the traffic that uses the route. Since the routes are identical, traffic should split evenly between them. In this case, all drivers arrive at d an hour and a half after their departure from o.



(a) Initial network (b) Augmented network

Figure 1.2: Braess's paradox. Each edge is labeled with a function that describes the travel time as a function of the fraction of the traffic that uses the edge. After the addition of the (v, w) edge, the price of anarchy is $4/3$.

Suppose we try to improve commute times by installing a teleportation device that allows drivers to travel instantly from v to w (Figure 1.2(b)). How will the drivers react?

We cannot expect the previous traffic pattern to persist in the new network. The travel time along the new route $o \to v \to w \to d$ is never worse than that along the two original paths, and it is strictly less whenever some traffic fails to use it. We therefore expect all drivers to deviate to the new route. Because of the ensuing heavy congestion on the edges (o, v) and (w, d), all of these drivers now experience *two* hours of travel time from o to d. Braess's paradox thus shows that the intuitively helpful action of adding a new superfast link can negatively impact all of the traffic!

Braess's paradox also demonstrates that selfish routing does not minimize the commute time of drivers—in the network with the teleportation device, an altruistic dictator could assign routes to traffic to improve everyone's commute time by 25%. We define the *price of anarchy (POA)* as the ratio between the system performance with strategic players and the best-possible system performance. For the network in Figure 1.2(b), the POA is $\frac{2}{3/2} = \frac{4}{3}$.

The POA is close to 1 under reasonable conditions in a remarkably wide range of application domains, including network routing, scheduling, resource allocation, and auctions. In such cases, selfish behavior leads to a near-optimal outcome. For example, Lecture 12 proves that modest over-provisioning of network capacity guarantees that the POA of selfish routing is close to 1.

1.2.2 Strings and Springs

Braess's paradox is not just about traffic networks. For example, it has an analog in mechanical networks of strings and springs. In the device pictured in Figure 1.3, one end of a spring is attached to a fixed support and the other end to a string. A second identical spring is hung from the free end of the string and carries a heavy weight. Finally, strings are connected, with a tiny bit of slack, from the support to the upper end of the second spring and from the lower end of the first spring to the weight. Assuming that the springs are ideally elastic, the stretched length of a spring is a linear function of the force applied to it. We can therefore view the network of strings and springs as a traffic network, where force corresponds to traffic and physical distance corresponds to travel time.

With a suitable choice of string and spring lengths and spring constants, the equilibrium position of this mechanical network is described by Figure 1.3(a). Perhaps unbelievably, severing the taut string causes the weight to *rise*, as shown in Figure 1.3(b)! To explain this curiosity, note that the two springs are initially connected in series, so each bears the full weight and is stretched out to a certain length. After cutting the taut string, the two springs carry the weight in parallel. Each spring now carries only half of the weight, and accordingly is stretched to only half of its previous length. The

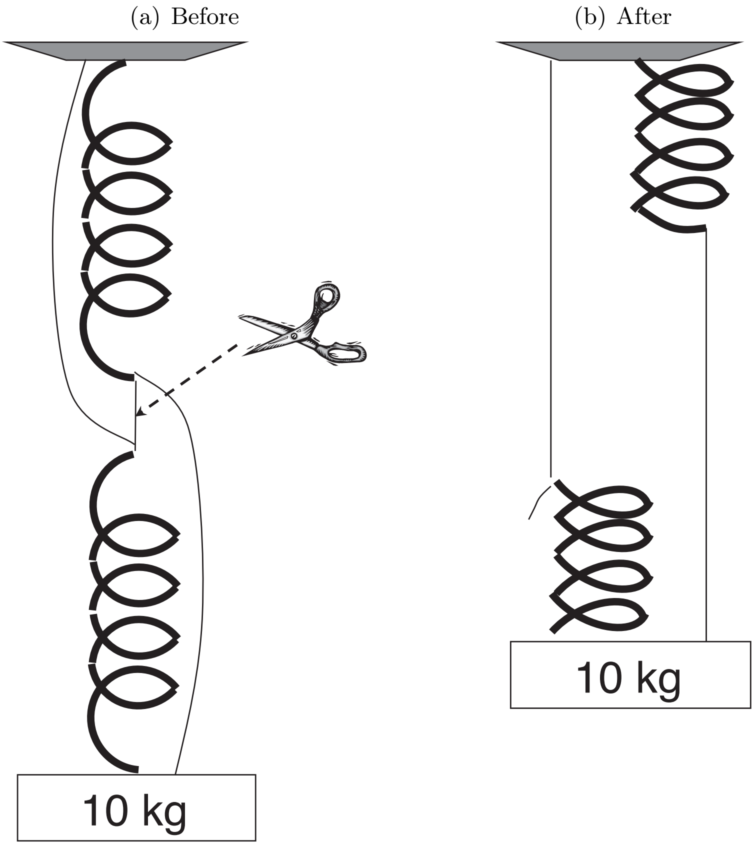


Figure 1.3: Strings and springs. Severing a taut string lifts a heavy weight.

rise in the weight is the same as the decrease in the commute time achieved by removing the teleporter from the network in Figure 1.2(b) to obtain the network in Figure 1.2(a).

1.3 Can Strategic Players Learn an Equilibrium?

Some games are easy to play. For example, in the second network of Braess's paradox (Figure 1.2(b)), using the teleporter is a no-brainer—it is the best route, no matter what other drivers do.

In most games, however, the best action to play depends on what the other players do. Rock-Paper-Scissors, rendered below in "bimatrix" form, is a canonical example.

	Rock	Paper	Scissors
Rock	$0,0$	$-1,1$	$1,-1$
Paper	$1,-1$	$0,0$	$-1,1$
Scissors	$-1,1$	$1,-1$	$0,0$

One player chooses a row and the other a column. The numbers in the corresponding matrix entry are the payoffs for the row and column player, respectively. More generally, a two-player game is specified by a finite strategy set for each player, and a payoff to each player for every pair of strategies that the players might choose.

Informally, an equilibrium is a steady state of a system where each participant, assuming everything else stays the same, wants to remain as is. There is certainly no "deterministic equilibrium" in the Rock-Paper-Scissors game: whatever the current state, at least one player can benefit from a unilateral deviation. For example, the outcome (Rock, Paper) cannot be an equilibrium, since the row player wants to switch and play Scissors.

When playing Rock-Paper-Scissors, it appears as if your opponent is randomizing over her three strategies. Such a probability distribution over strategies is called a *mixed* strategy. If both players randomize uniformly in Rock-Paper-Scissors, then neither player can increase her expected payoff via a unilateral deviation (all such deviations yield an expected payoff of zero). A pair of probability distributions with this property is a *(mixed-strategy) Nash equilibrium.*

Remarkably, allowing randomization, *every* game has at least one Nash equilibrium.

Theorem 1.1 (Nash's Theorem) *Every finite two-player game has a Nash equilibrium.*

Nash's theorem holds more generally in games with any finite number of players (Lecture 20).

Can a Nash equilibrium be computed efficiently, either by an algorithm or by strategic players themselves? In zero-sum games like Rock-Paper-Scissors, where the payoff pair in each entry sums to zero, this can be done via linear programming or, if a small amount of error can be tolerated, via simple iterative learning algorithms (Lecture 18). These algorithmic results give credence to the Nash equilibrium concept as a good prediction of behavior in zero-sum games.

In non-zero-sum two-player games, however, recent results indicate that there is no computationally efficient algorithm for computing a Nash equilibrium (Lecture 20). Interestingly, the standard argument for computational intractability, "$\mathcal{NP}$-hardness," does not seem to apply to the problem. In this sense, the problem of computing a Nash equilibrium of a two-player game is a rare example of a natural problem exhibiting intermediate computational difficulty.

Many interpretations of an equilibrium concept involve someone—the participants or a designer—determining an equilibrium. If all parties are boundedly rational, then an equilibrium can be interpreted as a credible prediction only if it can be computed with reasonable effort. Computational intractability thus casts doubt on the predictive power of an equilibrium concept. Intractability is certainly not the first stone to be thrown at the Nash equilibrium concept. For example, games can have multiple Nash equilibria, and this non-uniqueness diminishes the predictive power of the concept. Nonetheless, the intractability critique is an important one, and it is most naturally formalized using concepts from computer science. It also provides novel motivation for studying computationally tractable equilibrium concepts such as correlated and coarse correlated equilibria (Lectures 13, 17, and 18).

The Upshot

☆ The women's badminton scandal at the 2012 Olympics was caused by a misalignment of the goal of the teams and that of the Olympic Committee.

☆ The burden lies on the system designer to anticipate strategic behavior, not on the participants to behave against their own interests.

☆ Braess's paradox shows that adding a superfast link to a network can negatively impact all of the traffic. Analogously, cutting a taut string in a network of strings and springs can cause a heavy weight to rise.

- ✰ The price of anarchy (POA) is the ratio between the system performance with strategic players and the best-possible system performance. When the POA is close to 1, selfish behavior is largely benign.

- ✰ A game is specified by a set of players, a strategy set for each player, and a payoff to each player in each outcome.

- ✰ In a Nash equilibrium, no player can increase her expected payoff by a unilateral deviation. Nash's theorem states that every finite game has at least one Nash equilibrium in mixed (i.e., randomized) strategies.

- ✰ The problem of computing a Nash equilibrium of a two-player game is a rare example of a natural problem exhibiting intermediate computational difficulty.

Notes

Hartline and Kleinberg (2012) relate the 2012 Olympic women's badminton scandal to mechanism design. Braess's paradox is from Braess (1968), and the strings and springs interpretation is from Cohen and Horowitz (1991). There are several physical demonstrations of Braess's paradox on YouTube. See Roughgarden (2006) and the references therein for numerous generalizations of Braess's paradox. Koutsoupias and Papadimitriou (1999) define the price of anarchy. Theorem 1.1 is from Nash (1950). The idea that markets implicitly compute a solution to a significant computational problem goes back at least to Adam Smith's "invisible hand" (Smith, 1776). Rabin (1957) is an early discussion of the conflict between bounded rationality and certain game-theoretic equilibrium concepts.

Exercises

Exercise 1.1 Give at least two suggestions for how to modify the Olympic badminton tournament format to reduce or eliminate the incentive for a team to intentionally lose a match.

Exercise 1.2 Watch the scene from the movie *A Beautiful Mind* that purports to explain what a Nash equilibrium is. (It's easy to find on YouTube.) The scenario described is most easily modeled as a game with four players (the men), each with the same five actions (the women). Explain why the solution proposed by the John Nash character is not a Nash equilibrium.

Exercise 1.3 Prove that there is a unique (mixed-strategy) Nash equilibrium in the Rock-Paper-Scissors game.

Problems

Problem 1.1 Identify a real-world system in which the goals of some of the participants and the designer are fundamentally misaligned, leading to manipulative behavior by the participants. A "system" could be, for example, a Web site, a competition, or a political process. Propose how to improve the system to mitigate the incentive problems. Your answer should include:

(a) A description of the system, detailed enough that you can express clearly the incentive problems and your solutions for them.

(b) Anecdotal or demonstrated evidence that participants are gaming the system in undesirable ways.

(c) A convincing argument why your proposed changes would reduce or eliminate the strategic behavior that you identified.

Problem 1.2 Can you produce a better video demonstration of Braess's paradox than those currently on YouTube? Possible dimensions for improvement include the magnitude of the weight's rise, production values, and dramatic content.

Lecture 2

Mechanism Design Basics

With this lecture we begin our formal study of mechanism design, the science of rule-making. This lecture introduces an important and canonical example of a mechanism design problem, the design of single-item auctions, and develops some mechanism design basics in this relatively simple setting. Later lectures extend the lessons learned to more complex applications.

Section 2.1 defines a model of single-item auctions, including the quasilinear utility model for bidders. After quickly formalizing sealed-bid auctions in Section 2.2 and mentioning first-price auctions in Section 2.3, in Section 2.4 we introduce second-price (a.k.a. Vickrey) auctions and establish their basic properties. Section 2.5 formalizes what we want in an auction: strong incentive guarantees, strong performance guarantees, and computational efficiency. Section 2.6 presents a case study on sponsored search auctions for selling online advertising, a "killer application" of auction theory.

2.1 Single-Item Auctions

We start our discussion of mechanism design with *single-item auctions*. Recall our overarching goal in this part of the course.

Course Goal 1 Understand how to design systems with strategic participants that have good performance guarantees.

Consider a seller with a single item, such as a slightly antiquated smartphone. This is the setup in a typical eBay auction, for example. There is some number n of (strategic!) bidders who are potentially interested in buying the item.

We want to reason about bidder behavior in various auction formats. To do this, we need a model of what a bidder wants. The first key assumption is that each bidder i has a nonnegative *valuation* v_i—her maximum willingness-to-pay for the item being sold. Thus bidder i wants to acquire the item as cheaply as possible, provided the selling price is at most v_i. Another important assumption is that this valuation is *private*, meaning it is unknown to the seller and to the other bidders.

Our bidder utility model, called the *quasilinear utility model*, is the following. If a bidder i loses an auction, her utility is 0. If the bidder wins at a price p, her utility is $v_i - p$. This is arguably the simplest natural utility model, and it is the one we focus on in these lectures.

2.2 Sealed-Bid Auctions

For the most part, we focus on a simple class of auction formats: *sealed-bid auctions*. Here's what happens:

1. Each bidder i privately communicates a bid b_i to the seller—in a sealed envelope, if you like.
2. The seller decides who gets the item (if anyone).
3. The seller decides on a selling price.

There is an obvious way to implement the second step—give the item to the highest bidder. This is the only selection rule that we consider in this lecture.[1]

There are multiple reasonable ways to implement the third step, and the choice of implementation significantly affects bidder behavior. For example, suppose we try to be altruistic and charge the winning bidder nothing. This idea backfires badly, with the auction devolving into a game of "who can name the highest number?"

2.3 First-Price Auctions

In a *first-price auction*, the winning bidder pays her bid. Such auctions are common in practice.

[1] When we study revenue maximization in Lectures 5 and 6, we'll see why other winner selection rules are important.

First-price auctions are hard to reason about. First, as a participant, it's hard to figure out how to bid. Second, as a seller or auction designer, it's hard to predict what will happen. To drive this point home, imagine participating in the following first-price auction. Your valuation (in dollars) for the item for sale is the number of your birth month plus the day of your birth. Thus, your valuation is somewhere between 2 (for January 1) and 43 (for December 31). Suppose there is exactly one other bidder (drawn at random from the world) whose valuation is determined in the same way. What bid would you submit to maximize your expected utility? Would it help to know your opponent's birthday? Would your answer change if you knew there were two other bidders in the auction rather than one?[2]

2.4 Second-Price Auctions and Dominant Strategies

We now focus on a different single-item auction, also common in practice, which is much easier to reason about. What happens when you win an eBay auction? If you bid \$100 and win, do you pay \$100? Not necessarily: eBay uses a "proxy bidder" that increases your bid on your behalf until your maximum bid is reached, or until you are the highest bidder, whichever comes first. For example, if the highest other bid is only \$90, then you only pay \$90 (plus a small increment) rather than your maximum bid of \$100. *If you win an eBay auction, the sale price is essentially the highest other bid—the second highest overall.*

A *second-price* or *Vickrey* auction is a sealed-bid auction in which the highest bidder wins and pays a price equal to the second-highest bid. To state the most important property of second-price auctions, we define a *dominant strategy* as a strategy (i.e., a bid) that is guaranteed to maximize a bidder's utility, no matter what the other bidders do.

Proposition 2.1 (Incentives in Second-Price Auctions) *In a second-price auction, every bidder i has a dominant strategy: set the bid b_i equal to her private valuation v_i.*

Proposition 2.1 implies that second-price auctions are particularly easy to participate in. When selecting a bid, a bidder doesn't need

[2] For more on the theory of first-price auctions, see Problem 5.3.

to reason about the other bidders in any way—how many there are, what their valuations are, whether or not they bid truthfully, etc. This is completely different from a first-price auction, where it never makes sense to bid one's valuation—this guarantees zero utility—and the optimal amount to underbid depends on the bids of the other bidders.

Proof of Proposition 2.1: Fix an arbitrary bidder i, valuation v_i, and the bids $\mathbf{b}_{-i}$ of the other bidders. Here $\mathbf{b}_{-i}$ means the vector $\mathbf{b}$ of all bids, but with the ith component removed.[3] We need to show that bidder i's utility is maximized by setting $b_i = v_i$.

Let $B = \max_{j \neq i} b_j$ denote the highest bid by some other bidder. What's special about a second-price auction is that, even though there are an infinite number of bids that i could make, only two distinct outcomes can result. If $b_i < B$, then i loses and receives utility 0. If $b_i \geq B$, then i wins at price B and receives utility $v_i - B$.[4]

We conclude by considering two cases. First, if $v_i < B$, the maximum utility that bidder i can obtain is $\max\{0, v_i - B\} = 0$, and it achieves this by bidding truthfully (and losing). Second, if $v_i \geq B$, the maximum utility that bidder i can obtain is $\max\{0, v_i - B\} = v_i - B$, and it achieves this by bidding truthfully (and winning). ■

Another important property is that a truthful bidder—meaning one that bids her true valuation—never regrets participating in a second-price auction.

Proposition 2.2 (Nonnegative Utility) *In a second-price auction, every truthful bidder is guaranteed nonnegative utility.*

Proof: Losers receive utility 0. If a bidder i is the winner, then her utility is $v_i - p$, where p is the second-highest bid. Since i is the winner (and hence the highest bidder) and bid her true valuation, $p \leq v_i$ and hence $v_i - p \geq 0$. ■

Exercises 2.1–2.5 ask you to explore further properties of and variations on second-price auctions. For example, truthful bidding is the *unique* dominant strategy for a bidder in a second-price auction.

[3] This may be wonky notation, but it's good to get used to it.

[4] We're assuming here that ties are broken in favor of bidder i. You should check that Proposition 2.1 holds no matter how ties are broken.

2.5 Ideal Auctions

Second-price single-item auctions are "ideal" in that they enjoy three quite different and desirable properties. We formalize the first of these in the following definition.

Definition 2.3 (Dominant-Strategy Incentive Compatible) An auction is *dominant-strategy incentive compatible (DSIC)* if truthful bidding is always a dominant strategy for every bidder and if truthful bidders always obtain nonnegative utility.[5]

Define the *social welfare* of an outcome of a single-item auction by

$$\sum_{i=1}^{n} v_i x_i,$$

where x_i is 1 if i wins and 0 if i loses. Because there is only one item, we have the feasibility constraint that $\sum_{i=1}^{n} x_i \leq 1$. Thus, the social welfare is just the valuation of the winner, or 0 if there is no winner.[6] An auction is *welfare maximizing* if, when bids are truthful, the auction outcome has the maximum possible social welfare. The next theorem follows from Proposition 2.1, Proposition 2.2, and the definition of second-price auctions.

Theorem 2.4 (Second-Price Auctions Are Ideal) *A second-price single-item auction satisfies the following:*

(1) ***[strong incentive guarantees]*** *It is a DSIC auction.*

(2) ***[strong performance guarantees]*** *It is welfare maximizing.*

(3) ***[computational efficiency]*** *It can implemented in time polynomial (indeed, linear) in the size of the input, meaning the number of bits necessary to represent the numbers* $v_1, \ldots, v_n$.

[5] The condition that truthful bidders obtain nonnegative utility is traditionally considered a separate requirement, called *individual rationality* or *voluntary participation*. To minimize terminology in these lectures, we fold this constraint into the DSIC condition, unless otherwise noted.

[6] The sale price does not appear in the definition of the social welfare of an outcome. We think of the seller as an agent whose utility is the revenue she earns; her utility then cancels out the utility lost by the auction winner from paying for the item.

All three properties are important. From a bidder's perspective, the DSIC property makes it particularly easy to choose a bid, and levels the playing field between sophisticated and unsophisticated bidders. From the perspective of the seller or auction designer, the DSIC property makes it much easier to reason about the auction's outcome. Note that *any* prediction of an auction's outcome has to be predicated on assumptions about how bidders behave. In a DSIC auction, the only assumption is that a bidder with an obvious dominant strategy will play it. Behavioral assumptions don't get much weaker than that.[7]

The DSIC property is great when you can get it, but we also want more. For example, an auction that gives the item away for free to a random bidder is DSIC, but it makes no effort to identify which bidders actually want the item. The welfare maximization property states something rather amazing: even though the bidder valuations are a priori unknown to the seller, the auction nevertheless identifies the bidder with the highest valuation! (Provided bids are truthful, a reasonable assumption in light of the DSIC property.) That is, a second-price auction solves the social welfare maximization problem as well as if all of the bidders' valuations were known in advance.

Computational efficiency is important because, to have potential practical utility, an auction should run in a reasonable amount of time. For example, auctions for online advertising, like those in Section 2.6, generally need to run in real time.

Section 2.6 and Lectures 3–4 strive for ideal auctions, in the sense of Theorem 2.4, for applications more complex than single-item auctions.

2.6 Case Study: Sponsored Search Auctions

2.6.1 Background

A Web search results page comprises a list of organic search results—deemed relevant to your query by an algorithm like PageRank—and a list of sponsored links, which have been paid for by advertisers. (Go do a Web search now to remind yourself, preferably on a valuable keyword like "mortgage" or "attorney.") Every time you type a

[7]Non-DSIC auctions are also important; see Section 4.3 for a detailed discussion.

search query into a search engine, an auction is run in real time to decide which advertisers' links are shown, how these links are arranged visually, and what the advertisers are charged. It is impossible to overstate how important such *sponsored search auctions* have been to the Internet economy. Here's one jaw-dropping statistic: around 2006, sponsored search auctions generated roughly 98% of Google's revenue. While online advertising is now sold in many different ways, sponsored search auctions continue to generate tens of billions of dollars of revenue every year.

2.6.2 The Basic Model of Sponsored Search Auctions

We discuss next a simplistic but useful and influential model of sponsored search auctions. The items for sale are k "slots" for sponsored links on a search results page. The bidders are the advertisers who have a standing bid on the keyword that was searched on. For example, Volvo and Subaru might be bidders on the keyword "station wagon," while Nikon and Canon might be bidders on the keyword "camera." Such auctions are more complex than single-item auctions in two ways. First, there are generally multiple items for sale (i.e., $k > 1$). Second, these items are not identical. For example, if ads are displayed as an ordered list, then higher slots in the list are more valuable than lower ones, since people generally scan the list from top to bottom.

We quantify the difference between different slots using *click-through rates (CTRs).* The CTR α_j of a slot j represents the probability that the end user clicks on this slot. Ordering the slots from top to bottom, we make the reasonable assumption that $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_k$. For simplicity, we also make the unreasonable assumption that the CTR of a slot is independent of its occupant. Everything we'll say about sponsored search auctions extends to the more general and realistic model in which each advertiser i has a "quality score" β_i (the higher the better) and the CTR of advertiser i in slot j is the product $\beta_i \alpha_j$ (e.g., Exercise 3.4).

We assume that an advertiser is not interested in an impression (i.e., being displayed on a page) per se, but rather has a private valuation v_i for each *click* on her link. Hence, the expected value derived by advertiser i from slot j is $v_i \alpha_j$.

2.6.3 What We Want

Is there an ideal sponsored search auction? Our desiderata are:

(1) DSIC. That is, truthful bidding should be a dominant strategy, and never leads to negative utility.

(2) Social welfare maximization. That is, the assignment of bidders to slots should maximize $\sum_{i=1}^{n} v_i x_i$, where x_i now denotes the CTR of the slot to which i is assigned (or 0 if i is not assigned to a slot). Each slot can only be assigned to one bidder, and each bidder gets only one slot.

(3) Computational efficiency. The running time should be polynomial (or even near-linear) in the size of the input $v_1, \ldots, v_n$. Remember that zillions of these auctions need to be run every day!

2.6.4 Our Design Approach

What's hard about auction design problems is that we have to design jointly two things: the choice of who wins what, and the choice of who pays what. Even in single-item auctions, it is not enough to make the "correct" choice to the first design decision (e.g., giving the item to the highest bidder)—if the payments are not just right, then strategic participants will game the system.

Happily, in many applications including sponsored search auctions, we can tackle this two-prong design problem one step at a time.

> **Step 1:** Assume, without justification, that bidders bid truthfully. Then, how should we assign bidders to slots so that the above properties (2) and (3) hold?
>
> **Step 2:** Given our answer to Step 1, how should we set selling prices so that the above property (1) holds?

If we efficiently solve both of these problems, then we have constructed an ideal auction. Step 2 ensures the DSIC property, which means that bidders will bid truthfully (provided each bidder with an obvious dominant strategy plays it). The hypothesis in Step 1 is then satisfied, so the outcome of the auction is indeed welfare-maximizing (and computable in polynomial time).

We conclude this lecture by executing Step 1 for sponsored search auctions. Given truthful bids, how should we assign bidders to slots to maximize the social welfare? Exercise 2.8 asks you to prove that the natural greedy algorithm is optimal (and computationally efficient): for $i = 1, 2, \ldots, k$, assign the ith highest bidder to the ith best slot.

Can we implement Step 2? Is there an analog of the second-price rule—sale prices that render truthful bidding a dominant strategy for every bidder? The next lecture gives an affirmative answer via Myerson's lemma, a powerful tool in mechanism design.

The Upshot

- ✰ In a single-item auction there is one seller with one item and multiple bidders with private valuations. Single-item auction design is a simple but canonical example of mechanism design.
- ✰ An auction is DSIC if truthful bidding is a dominant strategy and if truthful bidders always obtain nonnegative utility.
- ✰ An auction is welfare maximizing if, assuming truthful bids, the auction outcome always has the maximum possible social welfare.
- ✰ Second-price auctions are "ideal" in that they are DSIC, welfare maximizing, and can be implemented in polynomial time.
- ✰ Sponsored search auctions are a huge component of the Internet economy. Such auctions are more complex than single-item auctions because there are multiple slots for sale, and these slots vary in quality.
- ✰ A general two-step approach to designing ideal auctions is to first assume truthful bids and understand how to allocate items to maximize

the social welfare, and second to design selling prices that turn truthful bidding into a dominant strategy.

Notes

The concept of dominant-strategy incentive-compatibility is articulated in Hurwicz (1972). Theorem 2.4 is from Vickrey (1961), the paper that effectively founded the field of auction theory. The model of sponsored search presented in Section 2.6 is due independently to Edelman et al. (2007) and Varian (2007). The former paper contains the mentioned jaw-dropping statistic. Problem 2.1 is closely related to the secretary problem of Dynkin (1963); see also Hajiaghayi et al. (2004).

The 2007 Nobel Prize citation (Nobel Prize Committee, 2007) presents a historical overview of the development of mechanism design theory in the 1970s and 1980s. Modern introductions to the field include Börgers (2015), Diamantaras et al. (2009), and chapter 23 of Mas-Colell et al. (1995). Krishna (2010) is a good introduction to auction theory.

Exercises

Exercise 2.1 Consider a single-item auction with at least three bidders. Prove that awarding the item to the highest bidder, at a price equal to the third-highest bid, yields an auction that is *not* DSIC.

Exercise 2.2 Prove that for every false bid $b_i \neq v_i$ by a bidder in a second-price auction, there exist bids $\mathbf{b}_{-i}$ by the other bidders such that i's utility when bidding b_i is strictly less than when bidding v_i.

Exercise 2.3 Suppose there are k identical copies of an item and $n > k$ bidders. Suppose also that each bidder can receive at most one item. What is the analog of the second-price auction? Prove that your auction is DSIC.