Book Introduction by the Authors

Invited by

Kazuo Iwama

iwama@kuis.kyoto-u.ac.jp
Bulletin Editor
Kyoto University, Japan

TWENTY LECTURES ON ALGORITHMIC GAME THEORY

Tim Roughgarden*

1 Introduction

Computer science and economics have engaged in a productive conversation 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. The standard reference in the field [3] is aimed at researchers rather than students and autodidacts, and it predates the many important results that have appeared over the past ten years.

My book *Twenty Lectures on Algorithmic Game Theory* [5] grew out of my lecture notes for a course that I taught at Stanford five times between 2004 and 2013.[1] 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. I assume no background in game theory or economics, nor can the book substitute for a

*Department of Computer Science, Stanford University, 474 Gates Building, 353 Serra Mall, Stanford, CA 94305. Email: `tim@cs.stanford.edu`. Research supported in part by NSF award CCF-1524062.

[1] See `http://timroughgarden.org/notes.html` for lecture notes for this and many other courses.

traditional book on these subjects. My book is far from encyclopedic, but fortunately there are excellent existing books and books in preparation on many of the omitted topics [1, 2, 3, 4, 6].

2 Brief Overview

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. These lectures cover the Vickrey auction and the VCG mechanism, algorithmic mechanism design, Myerson's theory of revenue-maximizing auctions, and 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. Specific topics include selfish routing, network cost-sharing games and the price of stability, potential games, and smoothness arguments. Finally, Lectures 16–20 describe positive and negative results for the computation of equilibria, both by distributed learning algorithms and by computationally efficient centralized algorithms. These lectures discuss best-response dynamics, no-regret algorithms, and $\mathcal{PLS}$- and $\mathcal{PPAD}$-completeness.

3 Top 10 List

The following "top 10 list" provides additional details about the book's contents.

1. *The second-price single-item auction (Lecture 2).* Our first example of an "ideal" auction, which is dominant-strategy incentive compatible (DSIC), welfare maximizing, and computationally efficient. Single-item auctions already show how small design changes, such as a first-price vs. a second-price payment rule, can have major ramifications for participant behavior.

2. *Myerson's lemma (Lectures 3–5).* For single-parameter problems, DSIC mechanism design reduces to monotone allocation rule design. Applications include ideal sponsored search auctions, polynomial-time approximately optimal knapsack auctions, and the reduction of expected revenue maximization with respect to a valuation distribution to expected virtual welfare maximization.

3. *The Bulow-Klemperer theorem (Lecture 6).* In a single-item auction, adding an extra bidder is as good as knowing the underlying distribution and running an optimal auction. This result, along with the prophet inequality, is

an important clue that simple and prior-independent auctions can be almost as good as optimal ones.

4. *The VCG mechanism (Lecture 7–8).* Charging participants their externalities yields a DSIC welfare-maximizing mechanism, even in very general settings. The VCG mechanism is impractical in many real-world applications, including wireless spectrum auctions, which motivates simpler and indirect auction formats like simultaneous ascending auctions.

5. *Mechanism design without money (Lectures 9–10).* Many of the most elegant and widely deployed mechanisms do not use payments. Examples include the Top Trading Cycle mechanism, mechanisms for kidney exchange, and the Gale-Shapley stable matching mechanism.

6. *Selfish routing (Lectures 11–12).* Worst-case selfish routing networks are always simple, with Pigou-like networks maximizing the price of anarchy (POA). The POA of selfish routing is therefore large only when network cost functions are highly nonlinear, corroborating empirical evidence that network over-provisioning leads to good network performance.

7. *Robust POA Bounds (Lecture 14).* All of the proofs of POA bounds in these lectures are "smoothness arguments." As such, they apply to relatively permissive and tractable equilibrium concepts like correlated and coarse correlated equilibria.

8. *Potential games (Lectures 13 and 16).* In many classes of games, including routing, location, and network cost-sharing games, players are inadvertently striving to optimize a potential function. Every potential game has at least one pure Nash equilibrium and best-response dynamics always converges. Potential functions are also useful for proving POA-type bounds.

9. *No-regret algorithms (Lectures 17–18).* No-regret algorithms exist, including simple ones with optimal regret bounds, like the multiplicative weights algorithm. If each agent of a repeatedly played game uses a no-regret or no-swap-regret algorithm to choose her mixed strategies, then the time-averaged history of joint play converges to the sets of coarse correlated equilibria or correlated equilibria, respectively. These two equilibrium concepts are computationally tractable, as are mixed Nash equilibria in two-player zero-sum games.

10. *Complexity of equilibrium computation (Lectures 19–20).* The problem of computing a Nash equilibrium appears computationally intractable in general. $\mathcal{PLS}$-completeness and $\mathcal{PPAD}$-completeness are analogs of $\mathcal{NP}$-

completeness tailored to provide evidence of intractability for pure and mixed equilibrium computation problems, respectively.

References

[1] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[2] J. D. Hartline. Mechanism design and approximation. Book in preparation, 2017.

[3] N. Nisan, T. Roughgarden, É. Tardos, and V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[4] D. C. Parkes and S. Seuken. Economics and computation. Book in preparation, 2017.

[5] T. Roughgarden. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press, 2016.

[6] Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2009.