Table of Contents

ACM SIGecom Exchanges, Vol. 18, No. 1, July 2020

Editors-in-Chief: Inbal Talgam-Cohen and S. Matthew Weinberg

Information Director: Yannai A. Gonczarowski

ACM Staff: Irene Frawley

Editors' Introduction

INBAL TALGAM-COHEN
The Technion - Israel Institute of Technology
and
S. MATTHEW WEINBERG
Princeton University

This issue of SIGecom Exchanges contains a letter from the new SIGecom Diversity and Inclusion Chair, three research letters, two brief book announcements, and a workshop summary.

Rediet Abebe contributed a short introduction from her new role as the inaugural SIGecom Diversity and Inclusion Chair.

Drew Fudenberg and Annie Liang contributed a letter on machine learning for evaluating and improving theories, based on their AER 19 paper presented in highlights beyond EC 2019. Shuchi Chawla, Yifeng Teng, and Christos Tzamos contributed a letter on buy-many mechanisms, based on their EC 2019 and EC 2020 papers. Sepehr Assadi and Sahil Singla contributed a letter on improved truthful mechanisms for combinatorial auctions with submodular bidders, based on their FOCS 19 paper.

Alex Slivkins contributed a short announcement for his recent book "Introduction to Multi-Armed Bandits." Michael Kearns and Aaron Roth contributed a short announcement for their recent book "The Ethical Algorithm."

Kira Goldner contributed a brief summary of the recent YoungEC workshop.

Finally, we'd like to remind the readers of Vincent Conitzer's puzzle from Issue 17.2 (`https://www.sigecom.org/exchanges/volume_17/2/PUZZLE.pdf`). In addition to attempting the puzzle yourself, you might encourage students (including undergraduate) to take a stab at it. In particular, note that the author is excited to receive any submissions, including those which solve only special cases.

We would like to especially thank all of the authors for finding time to write high-quality pieces under the difficult circumstances of the past several months.

We hope you, the reader, will enjoy this issue.

Author's address: italgam@cs.technion.ac.il; smweinberg@princeton.edu

Diversity, Equity, and Inclusion in Economics and Computation

REDIET ABEBE
Harvard Society of Fellows

SIGecom embraces diversity, fosters mutual understanding and respect, and recognizes the inherent dignity of every person and group [sig 2020]. Over the years, the SIG has taken many measures towards these objectives, including establishing SafeEC, creating a virtual conference format designed to ease participation across time zones, and launching a Global Outreach program to increase participation of low- and middle-income countries [Immorlica et al. 2019].

But much work remains. E.g., participation in the Economics and Computation (EC) conference has yet to reach gender parity. Individuals from historically disenfranchised groups – such as Black researchers – remain under-represented. And there are surely many stark issues faced by other groups, e.g., individuals with disabilities, that have gone unnoticed.

The impact of these issues goes well beyond our research community. The Economics and Computation community has made great strides in advancing mathematical foundations, e.g., matching markets, and improving applications in many domains, e.g., ride-sharing. But there is now increased awareness around opportunities and risks that techniques from this research interface present – of the *necessity* for having diverse and inclusive research communities to ensure that advances are to the benefit of all of society and that risks do not fall disproportionately on disadvantaged groups.

Recognizing this obligation and opportunity to improve diversity and inclusion, the SIG has created a new Diversity and Inclusion chair position. During my term in this role, I will work to achieve the following key goals:

(1) Design and implement an annual community survey to measure and assess the current landscape with regards to representation, equity, and inclusion of disadvantaged communities.
(2) Improve collection of data regarding authors of paper submissions and accepted papers at EC, of program committee members, and of conference organizers.
(3) Evaluate impact of current activities towards improving representation and inclusion and identify new opportunities at EC or in the SIG virtual network.
(4) Synthesize these findings and present recommendations to the SIG executives.

I look forward to working with the SIG executive, officers, and the community towards each of these goals. I encourage you all to reach out if you have concerns, ideas for activities, or suggestions for resources that may be needed.

REFERENCES

2020. ACM Special Interest Group on Economics and Computation. `http://www.sigecom.org/`.

Author's address: rabebe@fas.harvard.edu

Immorlica, N., Kominers, S. D., and Ligett, K. 2019. Letter from sigecom executive committee. *SIGecom Exch. 17,* 2, 2–3.

Machine Learning for Evaluating and Improving Theories

DREW FUDENBERG
MIT
and
ANNIE LIANG
University of Pennsylvania

We summarize our recent work that uses machine learning techniques as a complement to theoretical modeling, rather than a substitute for it. The key concepts are those of the *completeness* and *restrictiveness* of a model. A theory's completeness is how much it improves predictions over a naive baseline, relative to how much improvement is possible. When a theory is relatively incomplete, machine learning algorithms can help reveal regularities that the theory doesn't capture, and thus lead to the construction of theories that make more accurate predictions. Restrictiveness measures a theory's ability to match arbitrary hypothetical data: A very unrestrictive theory will be complete on almost any data, so the fact that it is complete on the actual data is not very instructive. We algorithmically quantify restrictiveness by measuring how well the theory approximates randomly generated behaviors. Finally, we propose "algorithmic experimental design" as a method to help select which experiments to run.

Categories and Subject Descriptors: J.4 [**Social and Behavioral Sciences**]: Economics

General Terms: Economics, Experimentation, Measurement, Theory

Additional Key Words and Phrases: machine learning, economic theory, modeling, prediction

1. INTRODUCTION

This survey summarizes our recent and ongoing work [Fudenberg and Liang 2019; Fudenberg et al. 2019; Fudenberg et al. 2020] on how to use machine learning techniques to evaluate and then improve theories. Black-box algorithms can generate better predictions than parametric theories, but direct application of these methods generally does not yield an improved understanding into the behavior of interest. We demonstrate how black-box algorithms can nevertheless contribute to this latter objective when used as a complement to traditional modeling.

In Section 2, we define the "completeness" of a theory to be the fraction of *achievable* prediction that it attains, benchmarked against the performance of a fully nonparametric black box. We show by example how studying cases where a machine learning algorithm predicts well, but the theory does not, can allow us to identify new regularities that the theory has not yet captured. A theory that is very complete for prediction of the actual data captures most of the important regularities in the observed behavior. But if a theory can approximate *most* patterns of behavior, then its ability to fit the actual data doesn't speak to its relevance. In Section 3, we quantify the "restrictiveness" of a model by measuring how well it

Authors' addresses: `drewf@mit.edu`, `anliang@upenn.edu`

approximates arbitrary behaviors. We illustrate our ideas with two classic prediction problems from experimental economics—predicting certainty equivalents for binary lotteries and predicting initial play in matrix games—and evaluate models in these domains from the dual perspectives of completeness and restrictiveness.

Both of our proposed measures depend on the domain of "test cases" used to evaluate the model, which is generally a choice variable of the experimenter the collecting data. In Section 4, we show how algorithms can help in designing new instances for data collection, for example finding cases in which a given theory is likely to fail. Our work here shows that machine learning techniques are useful not only for identifying structure in given data, but can also be useful to experimenters in figuring out what new data to acquire.

1.1 Prediction Problems

Let x be an observable *feature vector* taking values in a set X, and let y be an *outcome* of interest taking values in Y. An analyst observes pairs $z_i = (x_i, y_i)$, where each x_i takes on a finite set of values that were selected by the analyst, e.g. which games or lotteries to use in a laboratory experiment. We call any function $f : X \to Y$ a *predictive mapping* or simply *mapping*. We are interested in parametric models $\mathcal{F}_\Theta = \{f_\theta\}_{\theta \in \Theta}$, where Θ is a finite-dimensional, closed, and compact set and f is continuous in θ.

We evaluate predictions with a *loss function*, $\ell : Y \times Y \to \mathbb{R}$, where $\ell(y', y)$ is the error assigned to prediction of y' when the realized outcome is y. The commonly used loss functions *mean-squared error* and *classification loss* correspond to $\ell(y', y) = (y' - y)^2$ and $\ell(y', y) = \mathbb{1}(y' \neq y)$.

Definition 1.1. Let P denote the joint distribution of (x, y). The *(expected) prediction error for model* f is the expected error on a new test case: $\mathcal{E}_P(f) = \mathbb{E}_P[\ell(f(x), y)]$. The *prediction error for a model* $\mathcal{F}_\Theta$ is $\mathcal{E}_P(f^*_\Theta)$, where

$$f^*_\Theta = \arg\min_{f \in \mathcal{F}_\Theta} \mathcal{E}_P(f)$$

is the error-minimizing prediction rule from $\mathcal{F}_\Theta$.

Typically, the distribution P is not known, so these quantities need be estimated from the data. For example, to evaluate the error $\mathcal{E}_P(f^*_\Theta)$ for a model $\mathcal{F}_\Theta$, we might estimate its economic parameter θ from training data, and test the trained mapping f_θ on new observations. We put aside details of estimation for this survey, and refer interested readers to our papers.

1.2 Examples

We illustrate our methodologies using two examples from the economics literature.

Example 1 *Risk Preferences.* We consider the problem of predicting *certainty equivalents* for lotteries, i.e. the certain payment that an individual considers equivalent to the lottery's random payment. We use a data set from Bruhin et al. [2010] of the reported certainty equivalents (across different subjects) for a set of 25 binary lotteries over positive prizes. The feature space X is the set of 25 unique tuples $x = (\overline{z}, \underline{z}, p)$ describing the binary lotteries, where $\overline{z} > \underline{z} \geq 0$ are the two prizes, and p is the probability of $\overline{z}$. The outcome to be predicted is a *given* subject's certainty equivalent for a given lottery, so $Y = \mathbb{R}$. We use mean-squared error as the loss function, so the optimal prediction is the average certainty equivalent in the data.

The economic model that we evaluate is the three-parameter version of *Cumulative Prospect Theory* suggested by Goldstein and Einhorn [1987] and Lattimore et al. [1992]. The parameter vector here is $\theta = (\alpha, \delta, \gamma)$, and the associated model is $f_\theta(\overline{z}, \underline{z}, p) = w(p)v(\overline{z}) + (1 - w(p))v(\underline{z})$, where $w(p) = (\delta p^\gamma) / (\delta p^\gamma + (1-p)^\gamma)$ with $\delta \geq 0$ and $\gamma \geq 0$ is a nonlinear probability weighting function, and $v(z) = z^\alpha$ with $\alpha \geq 0$ is a value function for money.

Example 2 *Predicting Play in Games.* Our second example is predicting how people will play the first time they encounter a new simultaneous-move game. We use a data set of play in 3×3 normal-form games constructed by Wright and Leyton-Brown [2014] from six previous papers. The feature space X is the set of 86 unique payoff matrices $x \in \mathbb{R}^{18}$. The outcome to be predicted is the action that is chosen by the row player in a given instance of play, so $Y = \{a_1, a_2, a_3\}$. We use the misclassification rate as our loss function, so the optimal prediction is the modal action.

The economic model that we evaluate is the *Poisson Cognitive Hierarchy Model* (PCHM), which supposes that there is a distribution over players of differing levels of sophistication: The *level-0* player randomizes uniformly over his available actions, while the *level-1* player best responds to level-0 play [Stahl and Wilson 1994; 1995; Nagel 1995]. Camerer et al. [2004] defines the play of level-k players, $k \geq 2$, to be the best response to a perceived distribution over (lower) opponent levels, which is a Poisson distribution with rate parameter τ (truncated at k and re-normalized). The parameter τ is the only free parameter in this model.

2. COMPLETENESS

In Fudenberg et al. [2019], we define the "completeness" of a model as the amount that it improves predictions over a naive rule, compared to the best achievable improvement given the available features. We normalize in this way because in many cases there is residual variation in the outcome y after conditioning on the features x, and so perfect prediction is not achievable by any mapping that makes predictions using the feature set X. The mapping from X to Y that minimizes prediction error is

$$f^*(x) = \underset{y' \in Y}{\arg\min}\, \mathbb{E}_P \left[\ell(y', y) \mid x\right]. \tag{1}$$

For example, if the outcome y is real-valued, and the loss function is mean-squared error, then f^* assigns to each feature vector x its conditional mean.

To interpret the prediction error of a model, it is useful to distinguish between two sources of error. The **irreducible error** in the prediction problem is the error $\mathcal{E}_P(f^*) = \mathbb{E}_P\left[\ell(f^*(x), y)\right]$ of the ideal rule on a new test observation. This is a bound on how well any mapping could perform. In addition, there can be error due to the specification of the class: If $\mathcal{F}_\Theta$ leaves out an important regularity, then the prediction error of the best mapping from this class, $\mathcal{E}_P(f^*_\Theta)$ may be substantially higher than the irreducible error, $\mathcal{E}_P(f^*)$.

These two sources of prediction error have very different implications for how to generate better predictions. If the model's prediction error is substantially higher than the irreducible error, it may be possible to identify new regularities and incorporate them into new models that improve prediction given the same feature set. Conversely, if the model's prediction error is close to the irreducible error for the current feature set, the priority should be to identify additional features that will allow for better predictions.

We define the **completeness** of a model to be the ratio of the reduction in prediction error (over a selected **naive mapping** f_n) that it achieves compared to the best possible reduction, which is to the irreducible error. We set the naïve prediction for a lottery's certainty equivalent to be its expected value, and we set the naïve prediction of initial play to be a uniform distribution over the available actions.

Definition 2.1. The **completeness of model $\mathcal{F}_\Theta$ is**

$$\frac{\mathcal{E}_P(f_n) - \mathcal{E}_P(f^*_\Theta)}{\mathcal{E}_P(f_n) - \mathcal{E}_P(f^*)}. \tag{2}$$

Table I reports completeness measures for the two economic models and the corresponding prediction tasks described in Section 1.1.

	Risk Preferences		Initial Play	
	Error	Completeness	Error	Completeness
Naive Benchmark	98.32 (4.00)	0%	0.66 (0.02)	0%
Economic Model	64.92 (4.49)	91%	0.40 (0.02)	76%
Irreducible Error	61.64 (3.00)	100%	0.32 (0.03)	100%

Table I. We report the completeness of the CPT and PCHM models in their respective prediction tasks.

We find that CPT is nearly complete, achieving 91% of the feasible reduction in prediction error, while its absolute level of prediction error is 64.92. The PCHM achieves 76% of the achievable reduction, which is good, but leaves room for improvements that capture additional regularities. We note additionally that the best PCHM model on this data set is the simpler 0-parameter *Level-1* model, which predicts the action that is a best response to uniform play.

In Fudenberg and Liang [2019], we trained a bagged decision tree algorithm to predict play in the games considered in Table I. This algorithm led to a further improvement in predictive accuracy. We then examined the 14 (out of 86) games where play was predicted correctly by our algorithm, but not by level-1/PCHM. Each of these games had an action whose average payoffs closely approximated the level-1 action, but which led to lower variation in possible payoffs. Players were more likely in the data to choose this "almost" level-1 action than the actual level-1 action.

One explanation for this behavior is that players maximize a concave function over game payoffs, as if they are risk averse. This led us to add a single parameter α

to the level-1 model, so that the prediction is the level-1 action when dollar payoffs u are transformed under $f(u) = u^{\alpha}$. The performance of this model, called level-1(α), weakly improved upon the decision tree ensemble, which shows that atheoretical prediction rules fit by machine learning algorithms can help researchers discover interpretable and portable extensions of existing models.

3. RESTRICTIVENESS

The high completeness of CPT and level-1(α) suggest that these models capture many of the regularities in the data. But because each of these models has free parameters that are chosen to maximize fit, one explanation for the high completeness measures is simply that these models are flexible enough to accommodate any pattern of behavior.[1] We would thus like to distinguish high completeness because a model includes most of the functions from X to Y from high completeness because the model includes the "right" regularities, namely those that are observed in actual data. In Fudenberg et al. [2020], we propose an algorithmic method for quantifying the restrictiveness of a model, which allows us to separate these cases.

Our strategy is to generate random mappings $f : X \to Y$ from a set $\mathcal{F}_{\mathcal{M}}$ of "permissible mappings"—for example, all mappings of certainty equivalents that are consistent with the property that people prefer more money to less—and evaluate how well these mappings can be approximated using the model $\mathcal{F}_{\Theta}$. The more mappings from $\mathcal{F}_{\mathcal{M}}$ that can be approximated by a model, the less restrictive that class is. To operationalize our measure, we define restrictiveness relative to a distribution μ on $\mathcal{F}_{\mathcal{M}}$ chosen by the analyst, where we interpret μ as the analyst's prior over the space of mappings. (One natural option would be a uniform prior.)

Formally, for any two mappings f and f', define $d(f, f') = \mathbb{E}_{P_X}(l(f(x), f'(x))$ to be the (average) distance between their outcomes, where P_X is the marginal distribution over the feature space. If f' describes the actual relationship between the features x and the outcome y, and the distance between f and f' is large, then predictions using the mapping f will (in expectation) lead to large errors. Further define $d(\mathcal{F}_{\Theta}, f) = \inf_{f' \in \mathcal{F}_{\Theta}} d(f', f)$ to be the distance between f and the closest mapping in $\mathcal{F}_{\Theta}$, so that $d(\mathcal{F}_{\Theta}, f)/d(f_n, f)$ is a **normalized distance** between $\mathcal{F}_{\Theta}$ and f, relative to the naive prediction rule introduced in Section 2. The models that we study nest the associated naive rule, so $d(\mathcal{F}_{\Theta}, f) \leq d(f_n, f)$. Thus the normalized distance lies between 0 and 1 on any prediction problem.

The restrictiveness of model $\mathcal{F}_{\Theta}$ is then defined to be the average normalized distance between random mappings f (drawn according to distribution μ on $\mathcal{F}_{\mathcal{M}}$) and the model $\mathcal{F}_{\Theta}$.

Definition 3.1. The *restrictiveness* of model $\mathcal{F}_{\Theta}$ is $r := \mathbb{E}_{\mu}\left[\frac{d(\mathcal{F}_{\Theta}, f)}{d(f_n, f)}\right]$.

Larger r corresponds to a more restrictive model: If $r = 1$, then the model fails to improve upon the naive mapping for most maps f, which implies that $\mathcal{F}_{\Theta}$ is very restrictive. If $r = 0$, then $\mathcal{F}_{\Theta}$ includes all mappings from the permissible set $\mathcal{F}_{\mathcal{M}}$, so it is completely unrestrictive. In Figure 1, we report a histogram of normalized

[1] Although there are "representation theorems" that characterize which data are consistent with a general CPT specification, the empirical content for the 3-parameter functional form is not known, and the same is true for the PCHM.

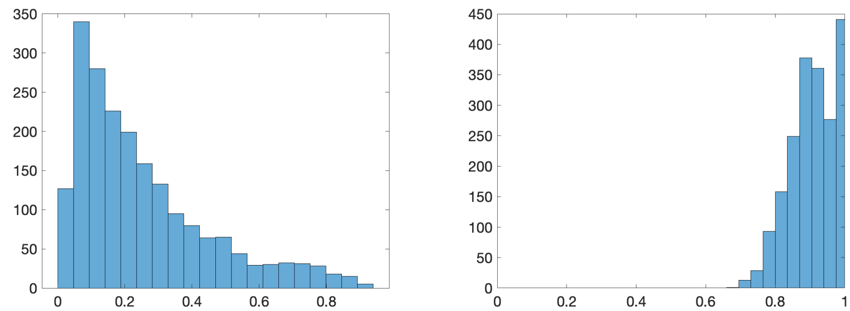

Fig. 1. *Left:* Distribution of normalized distances between CPT and random mappings; *Right:* Distribution of normalized distances between level-1(α) and random mappings.

distances between CPT and level-1(α) and 2000 random mappings of lotteries or games to certainty equivalents or modal actions respectively.

The estimated restrictiveness of CPT is 0.25, so in expectation CPT approximates a randomly selected mapping four times as well as the naive mapping does. In contrast, the restrictiveness of level-1(α) is 0.91, meaning that the level-1(α) model barely improves upon a naive mapping for approximating random mappings between games and initial play.

Since the level-1(α) model is a substantially more restrictive theory than CPT, its high completeness is suggestive that it more precisely captures the observed regularities.

4. ALGORITHMIC EXPERIMENTAL DESIGN

Our completeness and restrictiveness measures both depend on the underlying marginal distribution P_X over the feature space. Although we expect the conditional distribution $P(y \mid s)$ to be a fixed distribution describing the dependence of the outcome on the specified set of features, the marginal distribution on X is a choice variable for the experimenter. For example, we used a data set of certainty equivalents for the set of binary lotteries selected by Bruhin et al. [2010], and we used observations of initial play in 3×3 matrix games that had been chosen by different teams of authors with different purposes in mind. It isn't feasible, however, to run experiments on all lotteries or 3×3 games. The idea of *algorithmic experimental design* is to use machine learning to determine which test cases in X would be most informative.

In Fudenberg and Liang [2019], we used this approach to select which 3×3 games to include in a new experiment. Our goal was to identify games where behavior was likely to depart from the level-1(α) model, as this data could then allow us to discover further regularities in play. We trained a machine learning algorithm to predict the frequency of the level-1(α) action, and then selected games that achieved low predicted frequencies according to this algorithm. This approach is related in spirit to adversarial machine learning [Huang et al. 2011] and generative adversarial networks [Goodfellow et al. 2014] in that we are generating instances to trick the level-1(α) model, although our goal is to design new instances for *data* *collection* instead of refining predictions for a given data set.

We experimentally elicited play on these "algorithmically-generated" games on the platform Mechanical Turk, and found that the frequency of level-1(α) play is indeed low in these games. In keeping with our desire for interpretable conclusions, we did not simply look for the best black-box algorithm on our new data set. Instead, we developed a hybrid approach: We identified two models, each of which fit some of the data reasonably well, and trained a decision tree to predict which model would perform better on which games. This hybrid model outperformed its two constituent models, and studying the optimal assignment of games to models shed light on when the level-1(α) model is outperformed by an equally simple alternative model.

5. CONCLUSION

As we have shown, machine learning and associated algorithmic techniques can aid in the improvement of economic theories. When theories are incomplete, machine learning can help researchers identify regularities that are not captured by existing models and then develop new theories that predict better. Conversely, when a theory is highly complete, algorithmic techniques can show whether this is simply due to the theory's ability to fit any possible data, or whether the good fit results from the theory describing behaviors in the real world. Finally, machine learning can be used to guide researchers in choosing which experiments to run.

REFERENCES

Bruhin, A., Fehr-Duda, H., and Epper, T. 2010. Risk and rationality: Uncovering heterogeneity in probability distortion. *Econometrica 78,* 4, 1375–1412.

Camerer, C. F., Ho, T.-H., and Chong, J.-K. 2004. A cognitive hierarchy model of games. *The Quarterly Journal of Economics 119,* 3, 861–898.

Fudenberg, D., Gao, W., and Liang, A. 2020. Quantifying the restrictiveness of theories. Working Paper; available at `http://economics.mit.edu/faculty/drewf/working`.

Fudenberg, D., Kleinberg, J., Liang, A., and Mullainathan, S. 2019. Measuring the completeness of theories. Working Paper; available at `http://economics.mit.edu/faculty/drewf/working`.

Fudenberg, D. and Liang, A. 2019. Predicting and understanding initial play. *American Economic Review 109,* 12, 4112–41.

Goldstein, W. M. and Einhorn, H. J. 1987. Expression theory and the preference reversal phenomena. *Psychological review 94,* 2, 236.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. 2014. Generative adversarial networks. *Advances in neural information processing systems*, 2672–2680.

Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B., and Tygar, J. 2011. Adversarial machine learning. *Proceedings of 4th ACM Workshop on Artificial Intelligence and Security*, 43–58.

Lattimore, P. K., Baker, J. R., and Witte, A. D. 1992. The influence of probability on risky choice: A parametric examination. *Journal of Economic Behavior & Organization 17,* 3, 315–436.

Nagel, R. 1995. Unraveling in guessing games: An experimental study. *American Economic Review 85,* 5, 1313–1326.

Stahl, D. O. and Wilson, P. W. 1994. Experimental evidence on players' models of other players. *Journal of Economic Behavior and Organization 25,* 3, 309–327.

Stahl, D. O. and Wilson, P. W. 1995. On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior 10,* 1, 218–254.

Wright, J. R. and Leyton-Brown, K. 2014. Level-0 meta-models for predicting human behavior in games. *Proceedings of the fifteenth ACM conference on Economics and computation*, 857–874.

Buy-Many Mechanisms: What Are They and Why Should You Care?

SHUCHI CHAWLA
University of Wisconsin-Madison
and
YIFENG TENG
University of Wisconsin-Madison
and
CHRISTOS TZAMOS
University of Wisconsin-Madison

Multi-item mechanisms can be very complex, offering many different bundles to the buyer that could even be randomized. Such complexity is thought to be necessary as the revenue gaps between randomized and deterministic mechanisms, or deterministic and simple mechanisms are huge even for additive valuations. Furthermore, the optimal revenue displays strange properties such as non-continuity: changing valuations by tiny multiplicative amounts can change the optimal revenue by an arbitrarily large multiplicative factor. Our work shows that these strange properties do not apply to most natural situations as they require that the mechanism overcharges the buyer for a bundle while selling individual items at much lower prices. In such cases, the buyer would prefer to break his order into smaller pieces paying a much lower price overall.

We advocate studying a new revenue benchmark, namely the optimal revenue achievable by "buy-many" mechanisms, that is much better behaved. In a buy-many mechanism, the buyer is allowed to interact with the mechanism multiple times instead of just once. We show that the optimal buy-many revenue for *any* n item setting is at most $O(\log n)$ times the revenue achievable by item pricing. Furthermore, a mechanism of finite menu-size $(n/\epsilon)^{2^{O(n)}}$ suffices to achieve $(1+\epsilon)$-approximation to the optimal buy-many revenue. Both these results are tight in a very strong sense, as any family of mechanisms with description complexity sub-doubly-exponential in n cannot achieve better than logarithmic approximation in revenue. In contrast, for buy-one mechanisms, no simple mechanism of finite menu-size can get a finite-approximation in revenue. Moreover, the revenue of buy-one mechanisms can be extremely sensitive to multiplicative changes in values, while as we show optimal buy-many mechanisms satisfy revenue continuity.

Categories and Subject Descriptors: J.4 [**Social and Behavioral Science**]: Economics

General Terms: Economics, Theory

Additional Key Words and Phrases: Multi-item Mechanisms, Buy-many Mechanisms, Menu-size Complexity, Revenue Maximization, Approximation

1. INTRODUCTION

It is well-known that revenue-optimal mechanisms can be complicated when a seller has more than one item to sell to a buyer. Even in the simplest possible setting where the buyer has additive values, the optimal mechanism may offer the buyer infinitely many options, each of which corresponds to a randomized allocation over bundles of items at a certain price. Furthermore, when the buyer's values for the

Authors' addresses: shuchi@cs.wisc.edu, yifengt@cs.wisc.edu, tzamos@wisc.edu

items are correlated, no mechanism described by a finite number of menu entries can get any finite approximation in revenue [Briest et al. 2015; Hart and Nisan 2019]. In fact, these negative results hold already when the seller has only two items to sell, exhibiting an extreme "curse of dimensionality".

These negative results are regarded as unavoidable barriers in understanding the trade-off between optimality and simplicity for revenue maximization. Consequently, the focus has shifted to understanding this trade-off when the buyer's valuation distribution has a specific structure. For example, it is known that if the buyer's values for different items are independent and the buyer's value is unit-demand [Chawla et al. 2007; Chawla et al. 2015] or additive [Hart and Nisan 2012; Li and Yao 2013; Babaioff et al. 2014], either selling each item separately or selling all items as a grand bundle gets a constant fraction of optimal revenue. Such a result can be extended to a more general setting with subadditive valuations [Rubinstein and Weinberg 2018; Chawla and Miller 2016] as well as multiple buyers [Chawla et al. 2010; Yao 2015; Chawla and Miller 2016; Cai and Zhao 2017], but continues to require some degree of independence across individual item values. A different approach by Psomas et al. [2019] studies the smoothed complexity for revenue optimal mechanism design over correlated value distributions, but arrives at similar conclusions.

Our work [Chawla et al. 2019; 2020] presents an alternate view of the complexity of revenue-optimal mechanisms for buyers with arbitrary valuations. Our thesis is that the curse of dimensionality only happens when the buyer's actions are restricted. Let us illustrate through an example adapted from Hart and Nisan [2019]. Consider a seller with two items and a single buyer with additive values. With probability 4/7, the buyer values the first item at \$1 and the second item at \$0. With probability 2/7, the buyer values the first item at \$0 and the second item at \$2. With the remaining probability, 1/7, the buyer values each item at \$4 (and the pair at \$8). The optimal mechanism for such a setting is as follows. The seller offers a menu of 4 options to the buyer: the buyer can choose to pay \$5 and get both items, or pay \$1 and get only the first item, or pay \$2 and get only the second item, or pay \$0 and get nothing. Observe that in such an optimal mechanism, the buyer offers deterministic pricing that is superadditive over the items. In fact, all of the examples in the literature achieving large gaps between optimal mechanisms and simple mechanisms exhibit similar kinds of superadditivity in prices. However, to correctly implement such mechanisms with superadditive price, the seller must make sure that the buyer only interacts with the mechanism once. In the above example, a buyer that values each item \$4 would purchase both items and pay \$5 if he is only allowed to participate once. If we allow the buyer to interact with the mechanism for multiple times, he would first purchase the first item at a price of \$1, then purchase the second item at a price of \$2. Thus the revenue of the mechanism decreases when the buyer is allowed to participate in the mechanism more than once.

We advocate the study of revenue maximization under the **buy-many constraint**, which allows the buyer to interact with the mechanism multiple times and effectively disallows the seller from charging super-additive prices as in the above example. By the Taxation Principle, any single-buyer mechanism can be described by a menu of possible outcomes, each of which is a lottery that assigns a price to a randomized allocation. In a **buy-many** mechanism the buyer is allowed to adaptively purchase any multi-set of menu options. Such a definition is in contrast to the traditional **buy-one** mechanisms, where the buyer is only allowed to interact with the mechanism once.

The buy-many constraint is a natural property that most real-world mechanisms satisfy. Most classes of simple mechanisms studied in the literature, such as item pricing, grand bundle pricing, two-part tariffs, etc. also satisfy this constraint. Furthermore, as we show below, the optimal buy-many revenue exhibits many nice properties – bounded menu size, continuity, and approximability by simple mechanisms – that the optimal unconstrained (buy-one) revenue doesn't. As such we believe that the optimal buy-many revenue is worthy of further study as well as a good benchmark for revenue maximization.

In this note we describe the basic properties of the optimal buy-many revenue as well as simplicity versus optimality results under the buy-many constraint.

2. PROPERTIES OF BUY-MANY MECHANISMS

2.1 Revenue Approximation via Simple Mechanisms

It is known that an infinite revenue gap exists between optimal mechanisms and simple mechanisms when the buyer's values for different items are arbitrarily correlated:

THEOREM 2.1. [Hart and Nisan 2019] *There exists a distribution $\mathcal{D}$ over additive values on 2 items, such that for any finite menu size $m < \infty$,*

$$\frac{\textsc{Rev}(\mathcal{D})}{\textsc{Rev}_m(\mathcal{D})} = \infty.$$

Here $\textsc{Rev}_m(\mathcal{D})$ denotes the maximum revenue achievable by mechanisms with menu size at most m.

We notice that the large-revenue-gap examples established by Briest et al. and Hart and Nisan are no longer valid since their optimal revenue is achieved by superadditive pricings, which are not buy-many. On the other hand, Babaioff et al. [2018] observed that there exists a revenue gap between the optimal buy-one mechanism and the optimal buy-many mechanism even for additive buyers with independent item values. The first question we want to ask is whether the revenue gap between optimal mechanisms and simple mechanisms can still be large when the optimal mechanism satisfies the buy-many constraint. Our first result shows that it cannot.

THEOREM 2.2. *For any distribution $\mathcal{D}$ over arbitrary valuation functions,*

$$\frac{\textsc{BuyManyRev}(\mathcal{D})}{\textsc{Srev}(\mathcal{D})} = O(\log n).$$

Here $\textsc{BuyManyRev}(\mathcal{D})$ denotes the optimal revenue obtained by any buy-many mechanism; $\textsc{Srev}(\mathcal{D})$ denotes the optimal revenue obtained by item pricing. This theorem generalizes a previous result by Briest et al. [2015], who established the same revenue gap for a unit-demand buyer. We emphasize that the theorem requires no assumption over the distribution, except that the valuation function should be monotone non-decreasing in the set of items allocated: allocating extra items will never lower the buyer's value.

Can we do better than $O(\log n)$ in the approximation ratio? We show that the above result is tight in a very strong sense, even if we are allowed to use much more complex mechanisms.

THEOREM 2.3. *There exists a distribution over XOS valuation functions for which no mechanism with description complexity at most* $2^{2^{o(n^{1/4})}}$ *can obtain a* $o(\log n)$ *fraction of the optimal revenue obtained by buy-many mechanisms.*

Observe that item pricing can be described using $O(n \log R)$ bits when the values are integers in the range $[1, R]$. Our construction sets $R = \text{poly}(n)$. This implies that to obtain $O(\log n)$-approximation in revenue, we only need a mechanism with description complexity almost linear in n. However, if we want to do even slightly better asymptotically, we need a mechanism that has to be described by doubly-exponential in n number of bits.

2.2 Menu-size Complexity for Near-optimal Mechanisms

The menu size of a mechanism, defined as the number of different outcomes the seller offers to the buyer, was first introduced by Hart and Nisan [2019]. It has been studied extensively in literature as a measure of complexity for single-buyer mechanisms (see, e.g., [Babaioff et al. 2017; Gonczarowski 2018; Kothari et al. 2019]). It has a direct correspondence with the communication complexity of the interaction protocol between the buyer and the seller.

For the unconstrained setting, it is known that even for selling two items to an additive buyer, the menu-size complexity of the optimal buy-one mechanism can be infinite [Hart and Nisan 2019]. The same is true for any mechanism that achieves a bounded approximation to the optimal revenue. Positive results for getting near-optimal revenue using mechanisms with finite menu-size complexity have only been established for buyers with subadditive valuation functions and independent values for each item [Babaioff et al. 2017; Kothari et al. 2019].

For buy-many mechanisms, we define their menu-size complexity to be similar to the "additive menu size" introduced by Hart and Nisan [2019], which corresponds to the number of "basic" options the buy-many mechanism offers. An example in [Daskalakis et al. 2017] shows that the optimal buy-many mechanism may still have an infinite menu size. Can we approximate the optimal revenue obtained by buy-many mechanisms with bounded menu-size? We give an affirmative answer to this question in the following theorem.

THEOREM 2.4. *For any distribution* $\mathcal{D}$ *over arbitrary valuation functions, there exists a buy-many mechanism* $\mathcal{M}$ *generated by* $(n/\epsilon)^{2^{O(n)}}$ *menu entries, such that* $\text{REV}_{\mathcal{D}}(\mathcal{M}) \geq (1-\epsilon)\text{BUYMANYREV}(\mathcal{D})$.

The above theorem shows that a mechanism with menu-size complexity doubly-exponential in n can get near-optimal revenue obtained by any buy-many mechanism. Like Theorem 2.2, this theorem does not make any assumption on the value distribution and allows it to be arbitrarily correlated without structure. By Theorem 2.3, the doubly-exponential dependency on n is also tight, since no mechanism with sub-doubly-exponential description complexity can get $o(\log n)$-approximation in revenue.

2.3 Revenue Continuity of Optimal Mechanisms

We are interested in understanding the extent to which the optimal revenue changes if the value distribution is perturbed slightly. This extent depends, of course, on the manner in which the value distribution is perturbed. It is known [Daskalakis and Weinberg 2012; Rubinstein and Weinberg 2018; Gonczarowski and Weinberg 2018; Kothari et al. 2019; Brustle et al. 2019]), for example, that when each valuation function in the support of the distribution gets perturbed additively by some small $\epsilon > 0$, the optimal revenue does not change too much multiplicatively. However, when the buyer's values are unbounded, it is natural to ask what happens if the valuation function is perturbed multiplicatively, since such a perturbation is scale-invariant.

Formally, let $\mathcal{D}$ be a distribution over valuation functions, and let $\mathcal{D}'$ be another distribution obtained by taking each valuation function in the support of $\mathcal{D}'$ and changing each component of this function multiplicatively by some factor in $[1-\epsilon, 1+\epsilon]$ for some small $\epsilon > 0$. Can we then show that $\textsc{Rev}_{\mathcal{D}'} \geq (1-\epsilon')\textsc{Rev}_{\mathcal{D}}$ where ϵ' goes to 0 as ϵ goes to 0? We call such a property *revenue continuity.*

While revenue continuity is inherently interesting, it also has practical implications. Continuity implies that revenue estimates established on the basis of market analysis will be robust to errors in estimating demand. Furthermore, the accuracy of these estimates will improve directly with a reduction in measurement error. From an algorithmic standpoint, revenue continuity allows discretizing the values to their most significant digits through a sufficiently fine multiplicative grid. This is possible to do without a significant drop in revenue.

A surprising result based on an example by Psomas et al. [2019] shows that optimal revenue obtained by buy-one mechanisms does not exhibit revenue continuity, even for additive buyers. It is even possible that the optimal revenue is infinite before the perturbation, but finite afterward. However, such revenue discontinuity does not happen to buy-many mechanisms. We state the result in the following theorem.

Theorem 2.5. *Let $\mathcal{D}$ be a distribution over arbitrary valuation functions, and $\mathcal{D}'$ a $(1 \pm \epsilon)$-multiplicative-perturbation of $\mathcal{D}$. Then*

$$\textsc{BuyManyRev}(\mathcal{D}') \geq (1 - \mathit{poly}(n, \epsilon))\,\textsc{BuyManyRev}(\mathcal{D}).$$

The above theorem implies that in sharp contrast to the unconstrained setting, buy-many mechanisms always satisfy revenue continuity, while again we do not need to have an assumption over the value distribution. The polynomial dependency on n is necessary even when the buyer is unit-demand.

3. CONCLUSION AND OPEN QUESTIONS

In this note, we advocate studying the revenue achievable by buy-many mechanisms as an alternative benchmark for the study of revenue-maximizing multi-item mechanisms. The optimal buy-many revenue has many nice properties that the unconstrained optimal buy-one revenue lacks for general value distributions: simple mechanisms such as item pricing can get a good approximation to the optimal buy-many revenue; mechanisms of bounded menu-size can give a near-optimal approximation; the optimal buy-many revenue satisfies revenue-continuity.

There are several interesting directions for future work. One direction is to investigate other revenue maximization problems using buy-many revenue as the benchmark. For example, for special cases such as an additive buyer with independent item values, is there an efficient approximation scheme computing the optimal buy-many revenue? Can we find improved constant factors approximations using simple mechanisms for other special cases? Another direction is to investigate similar revenue benchmarks for multi-buyer limited supply settings. How should the definition of buy-many mechanisms be extended to those settings?

REFERENCES

Babaioff, M., Gonczarowski, Y. A., and Nisan, N. 2017. The menu-size complexity of revenue approximation. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. 869–877.

Babaioff, M., Immorlica, N., Lucier, B., and Weinberg, S. M. 2014. A simple and approximately optimal mechanism for an additive buyer. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 21–30.

Babaioff, M., Nisan, N., and Rubinstein, A. 2018. Optimal deterministic mechanisms for an additive buyer. In *Proceedings of the 2018 ACM Conference on Economics and Computation*. ACM, 429–429.

Briest, P., Chawla, S., Kleinberg, R., and Weinberg, S. M. 2015. Pricing lotteries. *Journal of Economic Theory 156*, 144–174.

Brustle, J., Cai, Y., and Daskalakis, C. 2019. Multi-item mechanisms without item-independence: Learnability via robustness. *arXiv preprint arXiv:1911.02146*.

Cai, Y. and Zhao, M. 2017. Simple mechanisms for subadditive buyers via duality. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2017. 170–183.

Chawla, S., Hartline, J. D., and Kleinberg, R. 2007. Algorithmic pricing via virtual valuations. In *Proceedings of the 8th ACM conference on Electronic commerce*. ACM, 243–251.

Chawla, S., Hartline, J. D., Malec, D. L., and Sivan, B. 2010. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*. ACM, 311–320.

Chawla, S., Malec, D., and Sivan, B. 2015. The power of randomness in bayesian optimal mechanism design. *Games and Economic Behavior 91*, 297–317.

Chawla, S. and Miller, J. B. 2016. Mechanism design for subadditive agents via an ex ante relaxation. In *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 579–596.

Chawla, S., Teng, Y., and Tzamos, C. 2019. Buy-many mechanisms are not much better than item pricing. In *Proceedings of the 2019 ACM Conference on Economics and Computation*. 237–238.

Chawla, S., Teng, Y., and Tzamos, C. 2020. Menu-size complexity and revenue continuity of buy-many mechanisms. *arXiv preprint arXiv:2003.10636*.

Daskalakis, C., Deckelbaum, A., and Tzamos, C. 2017. Strong duality for a multiple-good monopolist. *Econometrica 85,* 3, 735–767.

Daskalakis, C. and Weinberg, S. M. 2012. Symmetries and optimal multi-dimensional mechanism design. In *Proceedings of the 13th ACM Conference on Electronic Commerce*. 370–387.

Gonczarowski, Y. A. 2018. Bounding the menu-size of approximately optimal auctions via optimal-transport duality. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2018. 123–131.

Gonczarowski, Y. A. and Weinberg, S. M. 2018. The sample complexity of up-to-ε multi-dimensional revenue maximization. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 416–426.

Hart, S. and Nisan, N. 2012. Approximate revenue maximization with multiple items. In *ACM Conf. on Electronic Commerce*. 656.

Hart, S. and Nisan, N. 2019. Selling multiple correlated goods: Revenue maximization and menu-size complexity. *Journal of Economic Theory 183*, 991–1029.

Kothari, P., Singla, S., Mohan, D., Schvartzman, A., and Weinberg, S. M. 2019. Approximation schemes for a unit-demand buyer with independent items via symmetries. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 220–232.

Li, X. and Yao, A. C.-C. 2013. On revenue maximization for selling multiple independently distributed items. *Proceedings of the National Academy of Sciences 110,* 28, 11232–11237.

Psomas, A., Schvartzman, A., and Weinberg, S. M. 2019. Smoothed analysis of multi-item auctions with correlated values. In *Proceedings of the 2019 ACM Conference on Economics and Computation*. ACM, 417–418.

Rubinstein, A. and Weinberg, S. M. 2018. Simple mechanisms for a subadditive buyer and applications to revenue monotonicity. *ACM Transactions on Economics and Computation (TEAC) 6,* 3-4, 19.

Yao, A. C.-C. 2015. An n-to-1 bidder reduction for multi-item auctions and its applications. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 92–109.

Improved Truthful Mechanisms for Combinatorial Auctions with Submodular Bidders

SEPEHR ASSADI
Rutgers University
and
SAHIL SINGLA
Princeton University and Institute for Advanced Study

A longstanding open problem in Algorithmic Mechanism Design is to design computationally-efficient truthful mechanisms for (approximately) maximizing welfare in combinatorial auctions with submodular bidders. The first such mechanism was obtained by Dobzinski, Nisan, and Schapira [STOC'06] who gave an $O(\log^2 m)$-approximation where m is the number of items. This problem has been studied extensively since, culminating in an $O(\sqrt{\log m})$-approximation mechanism by Dobzinski [STOC'16].

We present a computationally-efficient truthful mechanism with approximation ratio that improves upon the state-of-the-art by almost an exponential factor. In particular, our mechanism achieves an $O((\log \log m)^3)$-approximation in expectation, uses only $O(n)$ demand queries, and has universal truthfulness guarantee.

Categories and Subject Descriptors: F.2 [**Theory of computation**]: Mechanism Design

General Terms: Algorithms, Economics, Theory

Additional Key Words and Phrases: Combinatorial Auctions, Truthful Mechanisms

1. INTRODUCTION

A fundamental problem at the intersection of Computer Science and Economics is to allocate resources among strategic bidders. In combinatorial auctions we want to allocate interrelated resources. They arise naturally in applications like spectrum auctions where a bidder's valuation for a bundle of items need not be the sum of item values, and instead be a set function such as a submodular function.

A "paradigmatic" [Dobzinski et al. 2006; Abraham et al. 2012; Fotakis et al. 2017], "central" [Mu'alem and Nisan 2008; Dughmi and Vondrák 2011], and "arguably the most important" [Dobzinski 2007] problem in Algorithmic Mechanism Design is to design mechanisms for combinatorial auctions that are both *truthful* and *computationally-efficient* (see §2 for formal definitions). At the root of this problem is an inherent clash between computational-efficiency and truthfulness. On one hand, the celebrated VCG mechanism of [Vickrey 1961; Clarke 1971; Groves et al. 1973] is a truthful mechanism that returns the welfare maximizing allocation. Alas, this mechanism requires finding the welfare maximizing allocation exactly, which is not possible in poly(m, n) time for most valuations. On the other hand, from an algorithmic point of view, constant factor approximation algorithms exist for many interesting classes of valuations, but they are no longer truthful.

Authors' addresses: `sepehr.assadi@rutgers.edu`, `singla@cs.princeton.edu`

A particular case of this problem that has received significant attention is when the valuation functions of all the bidders are *submodular* functions as they capture the notion of diminishing return. There is no poly-time algorithm for finding the optimal allocation of submodular bidders [Mirrokni et al. 2008; Feige and Vondrák 2010; Dobzinski and Vondrák 2013] and thus VCG mechanism is not computationally-efficient here. On the other hand, by using only value queries, a simple greedy algorithm can achieve a 2-approximation [Lehmann et al. 2006] and this can be further improved to $(\frac{e}{e-1})$-approximation [Vondrák 2008], and even slightly better using demand queries [Feige and Vondrák 2006]. This leads to one of the earliest and the most basic questions in Algorithmic Mechanism Design:

> *How closely can the approximation ratio of* ***truthful*** *mechanisms for submodular bidders match what is possible from a* ***purely algorithmic*** *point of view that ignores the strategic behavior of the bidders?*

Already more than a decade ago, Dobzinski, Nisan, and Schapira [Dobzinski et al. 2005] gave the first non-trivial answer to this question by designing an $O(\sqrt{m})$-approximation mechanism. This approximation ratio was soon after exponentially improved by the same authors [Dobzinski et al. 2006] to $O(\log^2 m)$, which in turn was improved to $O(\log m \cdot \log\log m)$ by [Dobzinski 2007], and then to $O(\log m)$ by [Krysta and Vöcking 2012]. Breaking this logarithmic barrier remained elusive until the recent $O(\sqrt{\log m})$ approximation breakthrough of [Dobzinski 2016a].

Our Contribution. Our main contribution is an almost *exponential* factor improvement over the $\Theta(\sqrt{\log m})$ approximation mechanism of [Dobzinski 2016a].

THEOREM 1. *There exists a randomized and universally truthful mechanism for combinatorial auctions with submodular valuations that achieves an* $O((\log\log m)^3)$ *approximation to the social welfare in expectation using polynomial number of value and demand queries.*

We shall note that our mechanism (as well as all previous ones in [Dobzinski et al. 2006; Dobzinski 2007; Krysta and Vöcking 2012; Dobzinski 2016a]) actually works for the much broader class of *XOS* valuations (see §2 for definition). Our result reduces the gap between the approximation ratio of truthful mechanisms vs algorithms for submodular and XOS bidders significantly, namely, from poly$(\log(m))$ in prior work to poly$(\log\log(m))$.

Further Related Work. The gap between the approximation ratio of truthful mechanisms and general algorithms has been studied from numerous angles in the literature. It is known that algorithms that use only poly(m, n) many value queries, or are poly-time in the input representation (for succinctly representable valuations) can only achieve a $m^{\Omega(1)}$-approximation [Papadimitriou et al. 2008; Dobzinski 2011; Dughmi and Vondrák 2011; Dobzinski and Vondrák 2012; 2012; Daniely et al. 2015] (the latter assuming RP $\neq$ NP).

Nevertheless, these results do not apply to mechanisms that are allowed other natural types of queries, e.g., demand queries. This has led the researchers to study the communication complexity of this problem that can capture arbitrary queries to valuations [Nisan 2000; Blumrosen and Nisan 2002; Dobzinski et al. 2005; Nisan and

Segal 2006; Dobzinski and Vondrák 2013; Dobzinski et al. 2014; Dobzinski 2016b; Assadi 2017; Braverman et al. 2018; Ezra et al. 2018]. Only recently, [Assadi et al. 2020] proved the first separation between the communication complexity of truthful mechanisms and general algorithms. Another relevant recent paper is of [Cai et al. 2020] who show how to run the mechanisms in this paper using only value queries (i.e., no demand queries) under some mild relaxations of truthfulness.

2. BACKGROUND AND COMPARISON TO PRIOR WORK

Notation and Preliminaries. We denote by N the set of n bidders and by M the set of m items. We use bold-face letters to denote vectors of prices and capital letters for allocations. For a price vector $\boldsymbol{p}$ and a set of items $M' \subseteq M$, we define $\boldsymbol{p}(M') := \sum_{j \in M'} p_j$. is an allocation A' consisting of $A_i \cap M'$ for every $i \in N'$.

We make the standard assumption that valuation v_i of each bidder i is normalized, i.e., $v_i(\emptyset) = 0$, and monotone, i.e., $v_i(S) \leq v_i(T)$ for every $S \subseteq T \subseteq M$. We are interested in the case when bidders valuations are *submodular* and hence capture the notion of "diminishing marginal utility" of items for bidders. A valuation v is submodular iff $v(S \cup T) + v(S \cap T) \leq v(S) + v(T)$ for any $S, T \subseteq M$. Submodular functions are a strict subset of *XOS* valuations also known as *fractionally additive* valuations [Feige 2006] defined as follows. A valuation a is additive iff $a(S) = \sum_{j \in S} a(\{j\})$ for every bundle S. A valuation function v is XOS iff there exists t additive valuations $\{a_1, \ldots, a_t\}$ such that $v(S) = \max_{r \in [t]} a_r(S)$ for every $S \subseteq M$. If $a \in \arg\max_{r \in [t]} a_r(S)$, then a is called a *maximizing clause* for S and $a(\{j\})$ is a *supporting price* of item j in this maximizing clause. We say that an allocation $A = (A_1, \ldots, A_n)$ of items to n bidders with XOS valuation is *supported* by prices $\boldsymbol{q} = (q_1, \ldots, q_m)$ iff each q_j is a supporting price for item j in the maximizing clause of the bidder i to whom j is allocated, i.e., $j \in A_i$.

We are going to make the following simplifying assumption. This assumption is natural and can be lifted entirely; see [Assadi and Singla 2019, Section 6].

Assumption 1. *We assume there exists two non-negative numbers* $\psi_{\min} \leq \psi_{\max}$ *such that* (i) $\Psi := \psi_{\max}/\psi_{\min}$ *is bounded by some fixed* $\text{poly}(m)$*, and* (ii) *for every valuation, supporting price of any item for any clause belongs to* $\{0\} \cup [\psi_{\min} : \psi_{\max}]$*. We further assume that the mechanism is given* $\psi_{\min}$ *and* $\psi_{\max}$ *as input.*

Combinatorial Auctions. In a combinatorial auction, m items are to be allocated between n bidders. Each bidder i has a valuation function v_i that describes their value $v_i(S)$ for every bundle S of items. The goal is to design a mechanism that finds an allocation A of items that maximizes the *social welfare*, which is defined as $\mathsf{val}(A) := \sum_i v_i(A_i)$ where A_i is the bundle allocated to bidder i.

For a mechanism to be feasible, it needs to be *computationally-efficient*, i.e., run in $\text{poly}(m, n)$ time given query access to valuation functions. The most standard queries considered in this context are (i) value queries: "Given a bundle S, what is the value of $v(S)$?", and (ii) demand queries: "Given a vector of prices on items $\boldsymbol{p}$, what is the "most demanded" bundle, i.e., a bundle $S \in \arg\max_{S'} \{v(S') - \boldsymbol{p}(S')\}$?". At the same time, mechanisms should also take into account the strategic behavior of the bidders. A mechanism in which the dominant strategy of each bidder is to reveal their true valuation in response to given queries is called *truthful*. For randomized mechanisms, we consider *universally truthful* mechanisms which are distributions over truthful mechanisms (this is a stronger guarantee than truthful-in-expectation considered in, e.g., [Lavi and Swamy 2005] and [Dughmi et al. 2011]).

Fixed-Price Auctions (FPAs). All previous attempts for designing truthful and computationally-efficient mechanisms for this problem [Dobzinski et al. 2006; Dobzinski 2007; Krysta and Vöcking 2012; Dobzinski 2016a], at their core, relied on the following key observation: to design truthful mechanisms for submodular or XOS bidders, "all" we need is to find "good" estimates of the *supporting prices* in an optimal allocation; the rest can be handled by a simple *fixed-price auction* using these prices. In the following, we first define fixed-price auctions and then formalize this observation in Lemma 1.

For an *ordered* set N of bidders, M of items, and a price vector $\boldsymbol{p}$, we define $\texttt{FixedPriceAuction}(N, M, \boldsymbol{p})$ as follows: Iterate over the bidders i of the ordered set N in the given order and allocate $A_i \in \arg\max_{S \subseteq M}\{v_i(S) - \boldsymbol{p}(S)\}$ to bidder i and update $M \leftarrow M \setminus A_i$.

It is easy to see that FixedPriceAuction can be implemented using one demand query per bidder. It is truthful because bidders have no influence on the prices.

The following lemma gives a key property of this auction.Variants of this lemma have already appeared in the literature, e.g., in [Dobzinski et al. 2006; Dobzinski 2007; Feldman et al. 2015; Dobzinski 2016a; Ehsani et al. 2018].

LEMMA 1. *Let* $A := \texttt{FixedPriceAuction}(N, M, \boldsymbol{p})$ *and* $\delta < 1/2$. *Suppose O is an allocation with supporting prices $\boldsymbol{q}$ and M^* is the set of items j with* $\delta \cdot q_j \leq p_j < \frac{1}{2} \cdot q_j$. *Then,* $\mathsf{val}(A) \geq \delta \cdot \boldsymbol{q}(M^*)$.

Intuitively, Lemma 1 says that if we run a FPA with "approximately correct" prices for a subset of items (compared to their supporting prices in an optimal allocation), the resulting welfare would be almost as good as the optimal welfare obtained from this subset of items.

Let us now show how can one use FPAs and Lemma 1 to obtain a very simple $O(\log m)$ approximation truthful mechanism. Given Assumption 1, consider a FPA that draws a random integer $k \in \left[0, \lceil \log_2 \Psi \rceil\right]$ and sets a price $\boldsymbol{p}$ of $2^k \cdot \psi_{\min}$ on every item. By Lemma 1, we know that the welfare of this auction is at least a constant fraction of the sum of supporting prices around $2^k \cdot \psi_{\min}$. Since $\log_2(\frac{\psi_{\max}}{\psi_{\min}}) = O(\log m)$, all non-zero supporting prices have at least $1/\Omega(\log m)$ chance to appear, which gives an $O(\log m)$ approximation.

Our Techniques and Comparison to Prior Work. Our mechanism also uses FPAs but departs from prior work in the following key conceptual way. Instead of learning a coarse-grained "statistics" about the prices, say, the range they should belong to [Dobzinski et al. 2006; Dobzinski 2007], and using these statistics to "guess" a small number of good prices (e.g., $O(1)$ prices in [Dobzinski et al. 2006; Dobzinski 2007], and $O(\sqrt{\log m})$ in [Dobzinski 2016a]), we strive to "learn" the entire price vector of items in a fine-grained way (at least for a large fraction of items). This fine-grained view allows us to get much more accurate prices and ultimately leads to the exponential improvement.

A cornerstone of our approach is a "learning process" which starts with a simple guess of item prices and *iteratively* refine this guess. Each iteration of this process involves running *several* FPAs with the prices learned so far and use the resulting allocations to further refine our learned prices. The key to the analysis of this mechanism is the "Learnable-Or-Allocatable Lemma": Roughly speaking, we prove that in each iteration, we can either refine our learned prices significantly (Learnable), or the FPA with the currently learned prices already gets a high-welfare allocation (Allocatable). Thus, after a *few* iterations, the resulting prices have been refined enough to allow for a high-welfare allocation.

One ingredient in the proof of this lemma is an interesting property of FPAs that stems from their greedy nature: if we run a FPA with a *random ordering* of bidders, either we obtain a high-welfare allocation or we sell almost all items (most likely to wrong bidders). Such a property was first proved (in a similar but not identical form) by [Dobzinski 2016a] and is closely related to other similar results about greedy algorithms for maximum matching [Konrad et al. 2012], matroid intersection [Guruganesh and Singla 2017], and constrained submodular maximization [Norouzi-Fard et al. 2018].

3. THE HIGH-LEVEL OVERVIEW

We now give a streamlined overview of our main mechanism in Theorem 1. We describe our mechanism using three parameters $\alpha := \Theta(1)$, $\beta := O(\log\log m)$, and $\gamma := \Theta(\alpha\beta)$. Let $O = (O_1, \ldots, O_n)$ be an optimal allocation with welfare OPT and $\boldsymbol{q} = (q_1, \ldots, q_m)$ be its supporting prices (obviously, O and $\boldsymbol{q}$ are unknown). For now, let us assume that every q_j belongs to $\left\{1, \gamma, \gamma^2, \ldots, \gamma^K\right\}$, for some $K = O(\log m)$ by Assumption 1 (and hence prices are roughly poly(m) large).

The crux of our mechanism is to "learn" $\boldsymbol{q}$, namely, find another price vector $\boldsymbol{p}$ such that for some subset $C \subseteq M$ with $\boldsymbol{q}(C) \approx \mathsf{val}(O)$, $\boldsymbol{p}$ *point-wise* γ-approximates $\boldsymbol{q}$ for items in C (i.e., within a multiplicative factor of γ). Having learned such prices, we can run a FPA with prices $\boldsymbol{p}$, and by Lemma 1, obtain an allocation with welfare $\approx \gamma \cdot \mathsf{val}(O)$.

In order to obtain the price vector $\boldsymbol{p}$, we start with a rough guess $\boldsymbol{p}^{(1)}$ for what prices should be (say, all ones), and update our guess over (at most) β *iterations*. In each iteration $i \in [\beta]$, we use the prices $\boldsymbol{p}^{(i)}$ learned so far to find α new price vectors $\boldsymbol{p}^{(i)}_1, \ldots, \boldsymbol{p}^{(i)}_\alpha$, and "explore" for *each* item $j \in M$ which of these α vectors best represents its price in $\boldsymbol{q}$, and then assign that price to item j in $\boldsymbol{p}^{(i+1)}$. We continue this for β iterations until we converge to the desired price vector $\boldsymbol{p} := \boldsymbol{p}^{(\beta+1)}$, or we decide along the way that the prices learned so far are already "good enough". There are three main questions to answer here: (i) how to choose which prices to explore in each iteration, (ii) how to explore a new price for each item, and finally (iii) how to implement all this in a truthful (and computationally-efficient) manner.

Part (i) **– which prices to explore.** This question can be best answered from the perspective of a single item $j \in M$. Originally, we set $p^{(1)}_j \in \boldsymbol{p}^{(1)}$ to be 1, and so with our assumption that $q_j \in \left\{1, \gamma, \ldots, \gamma^K\right\}$, price $p^{(i)}_j$ will (γ^K)-approximate $q_j \in \boldsymbol{q}$. We want $p^{(2)}_j$ to $(\gamma^{K/\alpha})$-approximate q_j in the next iteration. Thus, we simply need to check for every $\ell \in \{0, \ldots, \alpha - 1\}$, whether $q_j \geq \gamma^{\ell \cdot K/\alpha}$ or not (using part (ii) below). By picking the largest ℓ^* for which this is true, we can get a $(\gamma^{K/\alpha})$-approximation to q_j. As such, for each item, there are only α choices of prices that we need to explore next, which allows us to devise price vectors $\boldsymbol{p}_1^{(1)}, \ldots, \boldsymbol{p}_\alpha^{(1)}$ accordingly. We repeat the same idea for later iterations as well, maintaining that in iteration i, price $p_j^{(i)} \in \boldsymbol{p}^{(i)}$ will $(\gamma^{K/\alpha^{i-1}})$-approximate q_j, and use α prices as before in $\boldsymbol{p}_1^{(i)}, \ldots, \boldsymbol{p}_\alpha^{(i)}$ to update this to a (γ^{K/α^i})-approximation for the next iteration. This way, after $\beta = O(\log\log m)$ iterations, we obtain $p_j^{(\beta+1)}$ that γ-approximates q_j as desired. See Figure 1 for an illustration.

In the above discussion, we talked about an item j as if its price is learned correctly throughout (i.e., $p_j^{(i)}$ is $(\gamma^{K/\alpha^{i-1}})$-approximating q_j for all $i \in [\beta+1]$). Our mechanism cannot guarantee this property for every item (but rather for most of them). Moreover, we are also not able to decide which items have been correctly priced, so we simply treat all items as being priced correctly in the mechanism and perform the above process for them. This means that for some items, their price may have been learned incorrectly in some iteration; so we conservatively ignore their contribution from now on in the analysis – a key part of our analysis is to show that this does not hurt the performance of the mechanism by much.

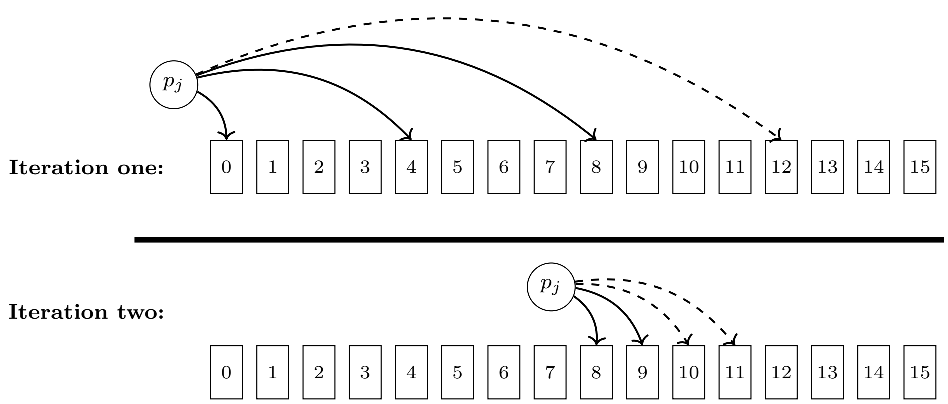

Fig. 1. A trajectory of the prices of a single item in the mechanism. Here, $\alpha = 4$ (number of auctions per iteration) and $\beta = 2$ (number of iterations). Each block i corresponds to price γ^i. Arrows correspond to the price of this item in the FPA; a solid arrow means the item was sold at that price, while a dashed arrow means it was not. The learned price of this item is γ^9.

Part (ii) **– how to explore a new price.** For this part, we build on a key idea from Dobzinski [2016a] in using FPAs themselves as a "proxy" for determining correctness of a guess for item prices. The idea is as follows: suppose we run a FPA with prices $\boldsymbol{p}_\ell^{(i)}$ for $\ell \in [\alpha]$ that we want to explore in an iteration i. As these prices may be very far from $\boldsymbol{q}$ yet, there is no guarantee that this auction returns a high-welfare allocation. However, if we choose the ordering of bidders *randomly*, then the *only way* this auction does not succeed in outputting a high-welfare allocation is because it sold *almost all* the items at the current prices (most likely to wrong bidders). Hence, an item getting sold in a certain FPA is a "good indicator" that its price in $\boldsymbol{q}$ is at least as high as the price used in this FPA. Such an idea was used in Dobzinski [2016a] to narrow down the range of item prices from $O(\log m)$ values to $O(\sqrt{\log m})$, which in turn allows the mechanism to simply guess a correct price for each item and achieves an $O(\sqrt{\log m})$-approximation.

We take this idea to the next step to obtain our Learnable-Or-Allocatable Lemma. Roughly speaking, we show that in each iteration i, starting from the set $C^{(i)}$ of correctly priced items, either one of the α auctions for exploring prices will lead to an $O(\beta^2)$-approximate allocation, or after this iteration, we will manage to further refine the prices of almost all items in $C^{(i)}$. I.e., we obtain a set $C^{(i+1)}$ with $\boldsymbol{q}(C^{(i+1)}) \approx \boldsymbol{q}(C^{(i)})$ and with $\boldsymbol{p}^{(i+1)}$ approximating prices $\boldsymbol{q}$ for $C^{(i+1)}$ much more accurately than $\boldsymbol{p}^{(i)}$ (as described in part (i)). Hence, either during one of the iterations there is an auction that gives us an $O(\beta^2)$-approximation, or we eventually end up with $\boldsymbol{p}^{(\beta+1)}$ that point-wise γ-approximates $\boldsymbol{q}$ for a large set of items $C^{(\beta+1)}$. Therefore, by ensuring $\boldsymbol{q}(C^{(\beta+1)}) = \Omega(\mathsf{OPT})$, a FPA with prices $\boldsymbol{p}^{(\beta+1)}$ gives a $\gamma = O(\alpha\beta)$-approximation by Lemma 1.

This outline oversimplifies many details. Let us briefly mention two here. Firstly, running FPAs only help us in not *underpricing* items for the next iteration; we also need to take care of *overpricing*. This is handled by making sure there is a *gap* of γ between different prices explored so that not many overpriced items can be sold in an auction (for the purpose of this discussion we simply assumed the existence of this gap, while in the actual mechanism we need to *create* this gap). Secondly, our mechanism has no way of determining (in a truthful way) which case of the Learnable-Or-Allocatable Lemma we are in. This means that there are $\alpha\cdot\beta$ auctions in the mechanism and any one of them may give an $O(\beta^2)$-approximation welfare. (If not, then we can learn the prices accurately and the final auction would be an $O(\alpha\beta)$-approximation.) The solution here is then to simply pick one of the $(\alpha\beta+1)$ auctions *uniformly at random* and allocate according to that. This way we succeed in finding a good auction with probability at least $1/\alpha\beta$ and hence, in expectation, we obtain an $O(\alpha\beta^3)$-approximation.

Part (iii) – how to ensure truthfulness. Recall that a FPA is truthful primarily because the responses of the bidders has no effect on the price of their allocated bundle. However, our mechanism consists of multiple FPAs and the outcomes of these auctions do influence the prices for *later* iterations. As such, to ensure truthfulness, each bidder should only participate in the auctions of a single iteration. Hence, at the beginning of the mechanism, we randomly partition the bidders into $\beta+1$ groups $N_1, \dots, N_{\beta+1}$. Then, in each iteration i, we use the bidders in group N_i for FPAs with prices $\boldsymbol{p}_1^{(i)}, \dots, \boldsymbol{p}_\alpha^{(i)}$ to learn prices $\boldsymbol{p}^{(i+1)}$, and in the final iteration we run one FPA with bidders $N_{\beta+1}$ and prices $\boldsymbol{p} = \boldsymbol{p}^{(\beta+1)}$.

This partitioning of bidders results in a key challenge: Our goal in learning the prices should actually be different from what was stated earlier. In particular, the auctions in each iteration i with bidders N_i should reveal the $\boldsymbol{q}$ prices of items allocated in O to bidders in $N_{>i} := N_{i+1}, \dots, N_{\beta+1}$, *as opposed to* bidders in N_i. This is because we are no longer able to allocate any item to bidders in $N_1, \dots, N_i$. We handle this also by our Learnable-Or-Allocatable Lemma. Instead of learning the set $C^{(i+1)}$ with $\boldsymbol{q}(C^{(i+1)}) \approx \boldsymbol{q}(C^{(i)})$, we have a more refined statement in which the LHS is replaced with $\boldsymbol{q}$ of items allocated *only* to bidders in $N_{>i}$.

4. CONCLUDING REMARKS AND OPEN PROBLEMS

In this letter, we reported on our work in [Assadi and Singla 2019] in designing a computationally-efficient and universally truthful mechanism for combinatorial auctions with submodular (even XOS) bidders with $O((\log\log m)^3)$-approximation.

The obvious question left open at this point is whether this gap can be improved further. We do not believe that our $O((\log\log m)^3)$ approximation is the best possible (in fact, our bounds can be slightly improved already; see [Assadi and Singla 2019]). On the other hand, the limit of our approach seems to be an $\Omega(\log\log m)$ approximation. Can one improve the approximation ratio all the way down to a constant? We shall note that even improving the approximation ratio of our mechanism down to $O(\log\log m)$ seems challenging.

REFERENCES

Abraham, I., Babaioff, M., Dughmi, S., and Roughgarden, T. 2012. Combinatorial auctions with restricted complements. In *Proceedings of EC*.

Assadi, S. 2017. Combinatorial auctions do need modest interaction. In *Proceedings of EC*.

Assadi, S., Khandeparkar, H., Saxena, R., and Weinberg, M. 2020. Separating the communication complexity of truthful and non-truthful combinatorial auctions. In *Proceedings of STOC*.

Assadi, S. and Singla, S. 2019. Improved truthful mechanisms for combinatorial auctions with submodular bidders. In *Proceedings of FOCS*.

Blumrosen, L. and Nisan, N. 2002. Auctions with severely bounded communication. In *Proceedings of FOCS*.

Braverman, M., Mao, J., and Weinberg, S. M. 2018. On simultaneous two-player combinatorial auctions. In *Proceedings of SODA*.

Cai, L., Thomas, C., and Weinberg, S. M. 2020. Implementation in advised strategies: Welfare guarantees from posted-price mechanisms when demand queries are np-hard. In *Proceedings of ITCS*.

Clarke, E. H. 1971. Multipart pricing of public goods. *Public choice 11,* 1, 17–33.

Daniely, A., Schapira, M., and Shahaf, G. 2015. Inapproximability of truthful mechanisms via generalizations of the VC dimension. In *Proceedings of STOC*.

Dobzinski, S. 2007. Two randomized mechanisms for combinatorial auctions. In *Proceedings of APPROX and RANDOM*. 89–103.

Dobzinski, S. 2011. An impossibility result for truthful combinatorial auctions with submodular valuations. In *Proceedings of STOC*.

Dobzinski, S. 2016a. Breaking the logarithmic barrier for truthful combinatorial auctions with submodular bidders. In *Proceedings of STOC*.

Dobzinski, S. 2016b. Computational efficiency requires simple taxation. In *Proceedings of FOCS*.

Dobzinski, S., Nisan, N., and Oren, S. 2014. Economic efficiency requires interaction. In *Proceedings of STOC*.

Dobzinski, S., Nisan, N., and Schapira, M. 2005. Approximation algorithms for combinatorial auctions with complement-free bidders. In *Proceedings of STOC*.

Dobzinski, S., Nisan, N., and Schapira, M. 2006. Truthful randomized mechanisms for combinatorial auctions. In *Proceedings of STOC*.

Dobzinski, S. and Vondrák, J. 2012. The computational complexity of truthfulness in combinatorial auctions. In *Proceedings of EC*.

Dobzinski, S. and Vondrák, J. 2013. Communication complexity of combinatorial auctions with submodular valuations. In *Proceedings of SODA*.

Dughmi, S., Roughgarden, T., and Yan, Q. 2011. From convex optimization to randomized mechanisms: toward optimal combinatorial auctions. In *Proceedings of STOC*. ACM.

Dughmi, S. and Vondrák, J. 2011. Limitations of randomized mechanisms for combinatorial auctions. In *Proceedings of FOCS*.

Ehsani, S., Hajiaghayi, M., Kesselheim, T., and Singla, S. 2018. Prophet secretary for combinatorial auctions and matroids. In *Proceedings of SODA*.

Ezra, T., Feldman, M., Neyman, E., Talgam-Cohen, I., and Weinberg, S. M. 2018. Complement-free couples must communicate: A hardness result for two-player combinatorial auctions. *CoRR abs/1811.09871*.

Feige, U. 2006. On maximizing welfare when utility functions are subadditive. In *Proceedings of STOC*.

Feige, U. and Vondrák, J. 2006. Approximation algorithms for allocation problems: Improving the factor of 1 - 1/e. In *Proceedings of FOCS*.

Feige, U. and Vondrák, J. 2010. The submodular welfare problem with demand queries. *Theory of Computing 6,* 1, 247–290.

Feldman, M., Gravin, N., and Lucier, B. 2015. Combinatorial auctions via posted prices. In *Proceedings of SODA*.

Fotakis, D., Krysta, P., and Ventre, C. 2017. Combinatorial auctions without money. *Algorithmica 77,* 3, 756–785.

Groves, T. et al. 1973. Incentives in teams. *Econometrica 41,* 4, 617–631.

Guruganesh, G. P. and Singla, S. 2017. Online matroid intersection: Beating half for random arrival. In *Proceedings of IPCO*. Springer.

Konrad, C., Magniez, F., and Mathieu, C. 2012. Maximum matching in semi-streaming with few passes. In *Proceedings of APPROX and RANDOM*.

Krysta, P. and Vöcking, B. 2012. Online mechanism design (randomized rounding on the fly). In *Proceedings of ICALP*. Springer.

Lavi, R. and Swamy, C. 2005. Truthful and near-optimal mechanism design via linear programming. In *Proceedings of FOCS*.

Lehmann, B., Lehmann, D. J., and Nisan, N. 2006. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior 55,* 2, 270–296.

Mirrokni, V. S., Schapira, M., and Vondrák, J. 2008. Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions. In *Proceedings of EC*.

Mu'alem, A. and Nisan, N. 2008. Truthful approximation mechanisms for restricted combinatorial auctions. *Games and Economic Behavior 64,* 2, 612–631.

Nisan, N. 2000. Bidding and allocation in combinatorial auctions. In *Proceedings of EC*.

Nisan, N. and Segal, I. 2006. The communication requirements of efficient allocations and supporting prices. *J. Economic Theory 129,* 1, 192–224.

Norouzi-Fard, A., Tarnawski, J., Mitrovic, S., Zandieh, A., Mousavifar, A., and Svensson, O. 2018. Beyond 1/2-approximation for submodular maximization on massive data streams. In *Proceedings of ICML*.

Papadimitriou, C. H., Schapira, M., and Singer, Y. 2008. On the hardness of being truthful. In *Proceedings of FOCS*.

Vickrey, W. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance 16,* 1, 8–37.

Vondrák, J. 2008. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of STOC*.

Book Announcement: "Introduction to Multi-Armed Bandits"

ALEKSANDRS SLIVKINS
Microsoft Research NYC

"Introduction to multi-armed bandits" is a broad and accessible textbook which emphasizes connections to economics and operations research.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems; F.1.2 [**Computation by Abstract Devices**]: Modes of Computation—*Online computation*; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms: Algorithms, Economics, Theory

I am pleased to announce "Introduction to multi-armed bandits" [Slivkins 2019], a broad and accessible textbook recently published in the "Foundations and Trends in Machine Learning" series. In addition to the machine learning issues, the book covers connections to economics and operations research. The book is geared towards a wide audience interested in machine learning, and may be of particular interest to the "economics and computation" community.

Multi-armed bandits is a simple but very powerful framework for algorithms that make decisions over time under uncertainty. In the basic version, an algorithm repeatedly chooses from a fixed set of actions (a.k.a., *arms*), and receives a reward for the chosen action. The reward comes from a fixed, action-specific distribution that is not known to the algorithm. Crucially, no "counterfactual" feedback is received: the algorithm does not know what would have happened had a different action been chosen. Hence, we have a tension between *exploring* different arms in order to acquire information and *exploiting* this information to maximize rewards, a.k.a. *exploration-exploitation tradeoff*.

This tradeoff is essential in many application scenarios. The term "multi-armed bandits" comes from a fictitious gambling scenario with multiple slot machines, a.k.a. one-armed bandits, which look identical to the gambler but may have different payout distributions. The original motivation comes from the design of medical trials: how to quickly phase out under-performing treatments without compromising statistical validity? Modern applications include web search (what are the best result for a given search query), content optimization (*e.g.,* what are the best news articles for a given user), online advertisement (what are the best ads to display for a given ad opportunity), and many others.

Aleksandrs Slivkins. "Introduction to multi-armed bandits".
Foundations and Trends in Machine Learning, vol. 12, pp. 1-286, November 2019.
At *Now Publishers*: `https://www.nowpublishers.com/article/Details/MAL-068`.
On Arxiv (in plain format): `https://arxiv.org/abs/1904.07272`.

Author's address: `slivkins@microsoft.com`,
`https://www.microsoft.com/en-us/research/people/slivkins/`.

I am particularly interested in how exploration-exploitation tradeoff plays out in "game-theoretic" environments, when the algorithm interacts with self-interested behavior. In retail, how to optimize prices and product assortments? In online advertisement, how to optimize parameters of an ad auction and how to allocate ads so as not to skew the advertisers' bidding behavior? In online labor markets, how to control quality and match workers to tasks, and which prices or contracts to offer? For online platforms that collect ratings and reviews, how to incentivize users to explore when they usually prefer to exploit? If two bandit algorithms play a repeated game against each other, does this game converge to an equilibrium, in which sense, and how fast? What if online platforms (*e.g.,* search engines) learn from interactions with users *and* compete with one another for the said users?

An enormous, multi-dimensional body of work has accumulated since the 1950-ies, with a big surge of interest in the past two decades. While the basic model described above captures something fundamental about the application scenarios, it can be extended – and made more realistic – in many different directions. To wit, what if the reward distributions can change over time? What if there is a huge number of actions, with some structure that can help us navigate the action space? What if more feedback is available to the algorithm, before and/or after each action is chosen? What if the algorithm consumes resources and operates under budget constraints thereon? What if the algorithm interacts with self-interested parties and needs to be compatible with their incentives? These questions, and their numerous refinements, prompted many distinct lines of work.

How to present all this work, let alone make it accessible? My approach is to present a broad picture, favoring fundamental ideas and elementary, teachable proofs over the strongest possible results. Each chapter handles one big direction in the problem space, covers the first-order concepts and results on a technical level, and provides a detailed literature review for further exploration. Most chapters conclude with exercises (which often introduce pertinent results that do not quite fit into the main technical narrative.) The book is teachable by design: each chapter corresponds to one week of my class. While some exposure to probability and statistics would help, a standard undergraduate course on algorithms should suffice for background.

I believe that multi-armed bandits are both deeply theoretical and deeply practical. Aside from all the math, I try to be careful and explicit about motivation. Our models might not capture the full complexity of application scenarios, and that's OK. Instead, the point is to capture some essential features present in many motivating examples. Practical aspects are discussed in considerable detail, based on a system for contextual bandits developed at Microsoft Research.[1]

For the "economics and computation" audience, the value proposition is as follows. Two chapters in my book (out of 11 total) cover connections to game theory and mechanism design, two more are directly motivated by dynamic pricing and similar problems, and the remaining chapters provide the necessary background. Economic aspects of multi-armed bandits is an increasingly popular subject, with ACM EC being the "conference home" for much of it.

[1] `https://www.microsoft.com/en-us/research/project/real-world-reinforcement-learning`. Also, see `https://www.microsoft.com/en-us/research/project/multi-world-testing-mwt`.

On a more personal note, this was some project. Once I sent my lecture notes to the publisher for the initial review, I thought it would take me a month or two to finish the book. As it happened, it took me *two and a half years*, with many interruptions in between ... I am planning a minor revision — some time after the pandemic is over, my son is back to school, and his parents are back to sanity — so, please send me bug reports and any other feedback.

P.S. Several books, published over the years, provide an in-depth treatment of various specific aspects of multi-armed bandits [Berry and Fristedt 1985; Cesa-Bianchi and Lugosi 2006; Gittins et al. 2011; Bubeck and Cesa-Bianchi 2012; Hazan 2015; Russo et al. 2018]. My book provides a more uniform, textbook-like treatment of the subject.

An upcoming book on bandits by Lattimore and Szepesvári [2020, preprint] provides a deeper treatment for a number of topics, and omits a few others. Evolving simultaneously and independently over the past 3 years, their book and mine reflect the authors' somewhat differing tastes and presentation styles, and, I believe, are complementary to one another.

REFERENCES

Berry, D. A. and Fristedt, B. 1985. *Bandit problems: sequential allocation of experiments.* Springer, Heidelberg, Germany.

Bubeck, S. and Cesa-Bianchi, N. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning 5,* 1, 1–122. Published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1204.5721`.

Cesa-Bianchi, N. and Lugosi, G. 2006. *Prediction, learning, and games.* Cambridge University Press, Cambridge, UK.

Gittins, J., Glazebrook, K., and Weber, R. 2011. *Multi-Armed Bandit Allocation Indices*, 2nd ed. John Wiley & Sons, Hoboken, NJ, USA. The first edition, single-authored by John Gittins, has been published in 1989.

Hazan, E. 2015. Introduction to Online Convex Optimization. *Foundations and Trends® in Optimization 2,* 3-4, 157–325. Published with *Now Publishers* (Boston, MA, USA). Also available at https://arxiv.org/abs/1909.05207.

Lattimore, T. and Szepesvári, C. 2020. *Bandit Algorithms.* Cambridge University Press, Cambridge, UK. Preprint, to be published in 2020. Versions available at `https://banditalgs.com/` since 2018.

Russo, D., Roy, B. V., Kazerouni, A., Osband, I., and Wen, Z. 2018. A tutorial on thompson sampling. *Foundations and Trends in Machine Learning 11,* 1, 1–96. Published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1707.02038`.

Slivkins, A. 2019. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning 12,* 1-2 (Nov.), 1–286. Published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1904.07272`.

Ethical Algorithm Design

MICHAEL KEARNS
University of Pennsylvania
and
AARON ROTH
University of Pennsylvania

In this letter, we summarize the research agenda that we survey in our recent book *The Ethical Algorithm*, which is intended for a general, nontechnical audience. At a high level, this research agenda proposes *formalizing* the ethical and social values that we want our algorithms to maintain — values including privacy, fairness, and explainability — and then to embed these social values directly into our algorithms as part of their design. This broad research area is most mature in the area of privacy, specifically *differential privacy*. It is off to a good start in emerging areas like algorithmic fairness, and seems promising for more nebulous goals like explainability, if only we can find the right definitions. Most work in this area to date analyzes algorithms as isolated components, but game-theoretic and economic analysis will become increasingly important as we try and study the effects of algorithmic interventions in larger sociotechnical systems.

Categories and Subject Descriptors: F.0 [**Theory of Computation**]: General

General Terms: Fairness, Privacy, Explainability, Game Theory

Additional Key Words and Phrases: Fairness, Privacy, Game Theory

In the past decade, as machine learning has become more capable, it has also been deployed in increasingly consequential domains. Trained machine learning models are now used to automatically make credit and lending decisions [Koren 2016], to inform hiring and compensation decisions [Miller 2015], to inform bail and parole decisions [Angwin et al. 2016], and to help target healthcare resources [Obermeyer et al. 2019]. It is therefore not surprising that there has been rising concern over the potential for these technologies to violate basic social norms like fairness, privacy, transparency, and accountability [O'Neil 2016; Eubanks 2018; Schneier 2015; Benjamin 2019]. After all, when we have humans in important decision-making pipelines, we expect them to respect these basic social values. There is no reason we shouldn't also expect this of algorithmic decision-making.

How do we go about making sure that algorithms (often predictive models trained via machine learning) are *privacy preserving* or *fair*? An important thesis of our book [Kearns and Roth 2019] is that although it is necessary for the software engineers and scientists who are designing and deploying algorithms to themselves be ethical, it is far from sufficient. The vast majority of documented instances of algorithmic bias, for example, do not seem to be the result of any ill intentions on the part of the designers, but were instead the *unanticipated consequences* of applying the standard tools and principles of machine learning. This standard framework can run afoul in a number of ways. For example, because machine learning is data-

Authors' addresses: `mkearns@cis.upenn.edu`, `aaroth@cis.upenn.edu`

intensive, it can be tempting to use datasets of convenience — those with large numbers of data points consisting of cheaply and easily collected measurements. Such datasets often involve proxies for what we actually want to predict, instead of the real underlying values: arrests as a proxy for crime in criminal risk prediction applications [Berk et al. 2018], or medical costs as a proxy for severity of disease in healthcare applications [Obermeyer et al. 2019]. These proxies can reflect existing human and structural biases, which will in turn be mimicked in the learned model. For example, Black citizens being policed at higher rates may lead to artificially elevated arrest rates as a function of criminal activity, and having less access to healthcare can result in artificially low health costs as a function of disease severity. There is no reason to believe such biases in the data will be "unlearned" when applying machine learning.

Another basic problem is that high-dimensional model optimization often results in "corner solutions" that are difficult to understand, and cannot be expected to satisfy any property that you didn't explicitly ask for (like some notion of fairness or privacy). We can be confident of little about a trained model other than that it will perform well according to the objective function that was optimized for, which is usually some narrow and myopic proxy for aggregate error or profit. This manifests itself frequently. For example, standard machine learning methods often lead models to "memorize" training data points in ways that let others recover them (an unanticipated privacy violation) [Shokri et al. 2017], and to have higher error on minority populations (an unanticipated fairness violation [Angwin et al. 2016]). In fact, optimizing average error naturally leads to models that disadvantage minorities, simply because (by definition) minorities contribute less to the overall error. Thus whenever an algorithm cannot simultaneously fit two different subsets of the population optimally, it will fit the majority class at the expense of the minority class. And seemingly sensible precautions — such as hiding sensitive demographic information from the algorithm — may exacerbate this problem by forcing it to attempt to fit different populations with the same model.

If the problem is that machine learning produces models that are unpredictable once we move away from their defined objectives and constraints, the solution is to be explicit about what exactly we want from our algorithms when we ask that they be private or fair. This is a difficult task, because words like "privacy" and "fairness" can have a broad set of informal and nuanced meanings, and can mean very different things to different people. Language and law are often able to skirt such ambiguity by deferring many precise questions to human beings and courts when necessary. But constraining the sorts of optimization procedures used in machine learning requires mathematical rigor and definitions. Coming up with definitions that capture at least some parts of the intuitive concepts is difficult.

But formalization is not impossible. A success story in this broad research agenda is differential privacy [Dwork et al. 2016], which is the focus of the first chapter of our book. Differential privacy does not encompass everything that is meant by the word "privacy", as no single formal definition could. But it does capture much of what one might mean by privacy in statistical computations, and provides a language in which to discuss different kinds of privacy guarantees. Informally, differential privacy is a guarantee of plausible deniability: there should be no statistical procedure that can determine (substantially) better than random guessing whether a particular individual's data was used in a computation or not. Thus (almost) nothing can be learned about them from the output of a computation that could not have been learned absent their data. This provides a way to disentangle the "secrets" of a particular person (information that can only be learned by examining a person's data, which differential privacy protects) from information that is implicit in population-level correlations (which differential privacy does not protect). Because it is a formal guarantee, we can design algorithms that *provably* provide differential privacy — and we can study the inevitable tradeoffs that arise between privacy and accuracy. Moreover, differential privacy is parameterized quantitatively: the words "substantially" and "almost" in our informal discussion above are precisely quantified in its formal definition. This allows us to think quantitatively about Pareto frontiers: there are different ways to trade off accuracy with privacy that we can map out using the language of differential privacy. *How* we want to make the tradeoff is a policy decision that has no universal answer — but differential privacy provides a language in which to have the debate. And in the decade and a half that has passed since its original introduction, differential privacy has gone from an object of mathematical study to a real technology that is becoming widely deployed in both industry and government.

The fairness in machine learning literature (the focus of our second chapter) is at a nascent stage in which there is no broad agreement on what the right definitions are, and our understanding of the properties of the definitions we have remains elementary [Chouldechova and Roth 2020]. Nevertheless, we can aspire to achieve for algorithmic fairness what differential privacy has achieved: to develop a formal language in which to precisely discuss different kinds of fairness guarantees, to study what is (and is not) achievable subject to these constraints, and to eventually translate these guarantees from the whiteboard to deployed technology. This won't be easy — it will be a decades-long research agenda, as it was (and remains) for data privacy. And we already know that the study of fairness will be messier than the study of privacy, in the sense that there are multiple statistical fairness constraints that one could ask for which are mutually incompatible with one another [Kleinberg et al. 2016; Chouldechova 2017]. There is also a divide between "statistical" notions of fairness that are already practical but provide limited promises to individuals, and definitions that provide stronger individual fairness guarantees [Dwork et al. 2012; Joseph et al. 2016] but which have various barriers to realistic implementation. One promising line of work aims to find definitions that can bridge this gap in various ways, being achievable without needing to make unrealistic assumptions on the data, but still corresponding to a promise to individuals [Kearns et al. 2018; Hebert-Johnson et al. 2018; Gillen et al. 2018; Dwork et al. 2019; Sharifi-Malvajerdi et al. 2019; Kim et al. 2018; Yona and Rothblum 2018; Jung et al. 2019; Ilvento 2019].

Another shortcoming of the aforementioned science is that it tends to focus myopically on isolated machine learning problems, whereas fairness issues generally do not arise in a vacuum around a single algorithm, but rather as part of larger sociotechnical systems in which algorithms are deployed. Data is generated by *people*, and the decisions of algorithms have effects in the world that change human incentives and behavior. Understanding how algorithms affect larger-scale societal dynamics and equilibrium is important to understanding fairness more broadly, and is the place where algorithmic game theory (the topic of our third chapter) has broad potential to be useful. There is already some early work studying the effects of various fairness constraints in dynamic settings (both game theoretic and not) [Jabbari et al. 2017; Hu and Chen 2018; Liu et al. 2018; Kannan et al. 2019; Liu et al. 2020; Jung et al. 2020], but much remains to be done here.

While the study of algorithmic fairness may be a decade or so behind data privacy, it is off to a promising start in the sense that it is at least already grappling with *definitions*. Another major theme of our book is that precise definitions are a crucial prerequisite to progress, especially when trying to pin down nebulous social values in ways that can be embedded in code. How can we make progress on accomplishing our goals if we cannot yet enunciate what our goals are? A lesson from both algorithmic privacy and algorithmic fairness is that the process of working through definitions carefully can make clear why seemingly sensible heuristics often fail. Attempting to "anonymize" data is doomed to failure because of the prevalance of external sources of information; instead we need to directly constrain what can be inferred from outputs (as differential privacy does). Hiding demographic information from machine learning algorithms fails — both because it doesn't work (it is distressingly easy to predict most demographic attributes from seemingly innocuous traits) — and because it can actually exacerbate error disparities in algorithms by forcing them to fit different distributions with the same model. Instead we must directly constrain the outcomes of the algorithm (which most current fairness definitions do). In our fifth chapter we briefly discuss important parts of this research agenda — like "explainability" and "interpretability" — that so far seem still to be awaiting the right definitions. We might be primed to make rapid progress on these important goals if only we can figure out what they should mean.

Our book is designed to be approachable to a general, non-technical reader, but we hope that it will also be informative and enjoyable for researchers interested in the intersections of machine learning, algorithms, game theory and related topics. While it is written without equations, we attempt to communicate key ideas in each research area without undue dilution. We expect that most readers will find at least some topics that are new to them: differential privacy (Chapter 1), fairness in machine learning (Chapter 2), algorithmic game theory (Chapter 3), adaptive data analysis (Chapter 4), and important topics that have yet to be formalized in convincing ways — such as transparency, explainability, accountability, and AI safety (Chapter 5).

REFERENCES

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. 2016. Machine bias. *Propublica*.

Benjamin, R. 2019. *Race after technology: Abolitionist tools for the new jim code.* John Wiley & Sons.

Berk, R., Heidari, H., Jabbari, S., Kearns, M., and Roth, A. 2018. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 0049124118782533.

Chouldechova, A. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data 5,* 2, 153–163.

Chouldechova, A. and Roth, A. 2020. A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM 63,* 5, 82–89.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*. 214–226.

Dwork, C., Kim, M. P., Reingold, O., Rothblum, G. N., and Yona, G. 2019. Learning from outcomes: Evidence-based rankings. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 106–125.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. 2016. Calibrating noise to sensitivity in private data analysis. *Journal of Privacy and Confidentiality 7,* 3, 17–51.

Eubanks, V. 2018. *Automating inequality: How high-tech tools profile, police, and punish the poor*. St. Martin's Press.

Gillen, S., Jung, C., Kearns, M., and Roth, A. 2018. Online learning with an unknown fairness metric. In *Advances in Neural Information Processing Systems*. 2600–2609.

Hebert-Johnson, U., Kim, M., Reingold, O., and Rothblum, G. 2018. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*. 1939–1948.

Hu, L. and Chen, Y. 2018. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*. 1389–1398.

Ilvento, C. 2019. Metric learning for individual fairness. *arXiv preprint arXiv:1906.00250*.

Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., and Roth, A. 2017. Fairness in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1617–1626.

Joseph, M., Kearns, M., Morgenstern, J. H., and Roth, A. 2016. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*. 325–333.

Jung, C., Kannan, S., Lee, C., Pai, M. M., Roth, A., and Vohra, R. 2020. Fair prediction with endogenous behavior. In *The Twenty-First ACM Conference on Economics and Computation*.

Jung, C., Kearns, M., Neel, S., Roth, A., Stapleton, L., and Wu, Z. S. 2019. Eliciting and enforcing subjective individual fairness. *arXiv preprint arXiv:1905.10660*.

Kannan, S., Roth, A., and Ziani, J. 2019. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 240–248.

Kearns, M., Neel, S., Roth, A., and Wu, Z. S. 2018. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*. 2564–2572.

Kearns, M. and Roth, A. 2019. *The Ethical Algorithm: The Science of Socially Aware Algorithm Design*. Oxford University Press.

Kim, M. P., Reingold, O., and Rothblum, G. N. 2018. Fairness through computationally-bounded awareness. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. 4847–4857.

Kleinberg, J., Mullainathan, S., and Raghavan, M. 2016. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*.

Koren, J. R. 2016. What does that web search say about your credit? *Los Angeles Times*. Retrieved 9/15/2016.

Liu, L. T., Dean, S., Rolf, E., Simchowitz, M., and Hardt, M. 2018. Delayed impact of fair machine learning. *arXiv preprint arXiv:1803.04383*.

Liu, L. T., Wilson, A., Haghtalab, N., Kalai, A. T., Borgs, C., and Chayes, J. 2020. The disparate equilibria of algorithmic decision making when individuals invest rationally. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 381–391.

Miller, C. C. 2015. Can an algorithm hire better than a human? *The New York Times*. Retrieved 4/28/2016.

Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. 2019. Dissecting racial bias in an algorithm used to manage the health of populations. *Science 366,* 6464, 447–453.

O'Neil, C. 2016. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books.

Schneier, B. 2015. *Data and Goliath: The hidden battles to collect your data and control your world*. WW Norton & Company.

Sharifi-Malvajerdi, S., Kearns, M. J., and Roth, A. 2019. Average individual fairness: Algorithms, generalization and experiments. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds. 8240–8249.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 3–18.

Yona, G. and Rothblum, G. 2018. Probably approximately metric-fair learning. In *International Conference on Machine Learning*. 5680–5688.

Report on YoungEC 19

KIRA GOLDNER
Columbia University

We present a report on the second Young Research Workshop on Economics and Computation (YoungEC 19) held in Tel Aviv, Israel on December 31, 2019–January 2, 2020, highlighting the aspects that make this event unique from other workshops in the community.

Categories and Subject Descriptors: [**Theory of computation**]: Algorithmic game theory and mechanism design; J.4 [**Computer Applications**]: Social and Behavioral Sciences—*Economics*; I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms: Algorithms, Economics, Theory

Additional Key Words and Phrases: Workshops

1. INTRODUCTION

In the spring of 2019, organizers Michal Feldman and Noam Nisan sent out a call to researchers for recommendations of brilliant young researchers—advanced graduate students and postdocs—from computer science, economics, operations research, game theory, and related areas. On December 31, twenty-seven such junior researchers, broadly diverse by research area, university, and geographic area, arrived at Tel Aviv University in Israel for the 2019 Young Researcher Workshop on Economics and Computation (YoungEC 19). The three-day workshop was centered around the junior researchers, including 25 minute slots for each participant to present their work, five 45 minute keynotes by senior researchers, and numerous breaks and activities planned for interaction among participants—all of which were also attended by keynote speakers and other faculty who were eager to engage with the young researchers. Generous funding from the European Research Council made it possible to cover almost all expenses for the young researchers, removing barriers to attendance.

2. THE YOUNG RESEARCHER EXPERIENCE

Each of the 27 junior researchers were given a 25 minute slot to present their research, giving them a chance to introduce themselves and their work, and consequently receive feedback and initiate research discussions. The young researchers included: Omer Ben-Porat (Technion IE&M), Ben Berger (Tel Aviv U CS), Arpita Biswas (IISc Bangalore), Lee Cohen (Tel Aviv U CS), Andrés Cristi (Universidad de Chile OR), Yuan Deng (Duke CS), Alon Eden (Harvard CS), Tomer Ezra (Tel Aviv U CS), Kira Goldner (Columbia CS), Yannai Gonczarowski (Microsoft Research), Zi Yang Kang (Stanford GSB), Ron Kupfer (Hebrew U CS), Bar Light (Stanford GSB), Giorgio Martini (Microsoft Economics), Divya Mohan (Princeton CS), Ellen Muir (Stanford Economics), Neil Newman (UBC CS), Gali Noti (Hebrew U CS), Rebecca Reiffenhauser (Sapienza U Rome CS), Steffen Schuldenzucker (U Zurich

Author's address: kgoldner@cs.columbia.edu

Informatics), Ariel Schvartzman (Princeton CS), Elisheva S. Shamash (Technion IE&M), Weiran Shen (CMU Software Research), Segev Shlomov (Technion IE&M), Yixin Tao (NYU CS), Yifeng Teng (UW-Madison CS), Ellen Vitercik (CMU CS), and Shuran Zheng (Harvard CS).

As participants learned about one another's work from these talks, they followed up with questions or discussions at breaks, using the opportunities both to get feedback and make further connections to their own research. Even the bus rides to and from the conference—including those at 8 a.m.—were full of research dialogues between seatmates. Neil Newman, a third-year student in computer science at the University of British Columbia echoed these sentiments, "It was exciting to hear what problems others were focusing on and to gain feedback on my own work." Shuran Zheng, a third-year student in computer science at Harvard, also commented on how the presentation experience and feedback helped: "I was more nervous because it was a less familiar audience, but it went well. I also got some very good suggestions about presentation skills."

Having such a diverse group that broadly represents the field of Economics and Computation allowed the participants to gain exposure to other aspects of EconCS that they may not typically learn about in their own institutions and to get up-to-date on recent exciting work. Shuran mentioned this aspect as well, "I think it's really an exciting workshop that gathers the most cutting-edge works in the field of algorithmic game theory as well as economic theory." She also noted that compared to mostly seeing more AI-related EconCS work at her own university, "It has been very nice to see what people are working on these days. Overall, I think these talks really expanded my knowledge about the theoretical side of economics and computation and the economics world." Ellen Muir, a third-year student from Stanford Economics echoed this, saying "There were a lot of really high-quality talks presented by young researchers and as an economist, it was great to get a better idea of what computer scientists are currently working on in fields like game theory, mechanism design and auction theory. I also really enjoy these smaller, more intimate workshops where there are plenty of opportunities to talk to other speakers and receive feedback during the many coffee breaks." The senior researchers also agreed, with Steve Tadelis noting, "It was great to have a mix of computer scientists and economists sharing different approaches and answering a variety of broadly related questions with different tools and techniques. Many of the young scholars have ambitious and creative research agendas!"

The format and focus of the small workshop on young researchers allowed the participants to really get to know each other and form a cohort as up-and-coming researchers in EconCS. Andrés Cristi, a second-year in Operations Research at Universidad de Chile, pointed this out, mentioning that, "I was very excited while giving my talk, because I felt I was speaking to the new generation of researchers in the area. It was a really motivating environment, to see what other students from different places were doing, but also with a relaxed and familiar atmosphere so we could chat about our work and share with young and senior researchers." Neil Newman agreed, "My favourite aspect was the workshop's small size and focus on speakers in similar career stages (other advanced graduate students and postdocs) as connections were valuable and easy to make."

3. SENIOR PRESENCE AT THE WORKSHOP

YoungEC 19 featured five invited speakers—three of whom were flown in from outside of Israel—who participated in the whole YoungEC workshop. They gave 45 minute keynote talks, but also attended the talks of the juniors researchers and engaged in conversations with them over the week. In addition, many local faculty and senior industry researchers, and those in town, attended YoungEC as well.

The five keynotes were delivered by Nina Balcan (Carnegie Melon CS), Liane Lewin-Eytan (Alexa Shopping at Amazon), Yishay Mansour (Tel Aviv University CS), Paul Milgrom (Stanford Economics), and Steve Tadelis (UC Berkeley Haas School of Business). Steve opened the workshop with his talk "Raising the Bar: Certification Thresholds and Market Outcomes," which discussed an empirical study of certification thresholds on eBay and their impact on buyer behavior. Paul then surprised attendees of all fields by presenting work on necessary conditions for good approximations in a particular setting: "Investment Incentives in Near-Optimal Mechanisms." ("Even Paul Milgrom endorses approximation!" became a common joke in later talks.) On the second day, Liane's talk, entitled "Alexa, Why Do People Buy Seemingly Irrelevant Items in Voice Product Search?" touched on a study as to when and why people engage with objectively irrelevant search results. On the final day, Nina spoke on "Machine Learning for Mechanism Design," highlighting methods for providing sample complexity bounds, as well as many other applications of these methods. Yishay presented additional learning work, "Exploration, Exploitation and Incentives," in which he discussed multiple settings where strategic agents interact with a bandit a designer optimizes for, and approaches for each setting.

The keynotes covered a diversity of topics, fields, and touched on both theoretical and applied work. The young researchers were thus able to get a sampling of what the senior participants were working on and find research connections. Shuran Zheng commented, "All the invited talks were very inspiring. It was very impressive to see how Paul Milgrom controlled the time of the talk exactly as he wanted. Some of the students' talks are more difficult for me to understand, mainly because I'm not familiar with the topics. As for my own talk, [...] I got some very nice feedback and questions about the work, including some comments from Moshe Babaioff, who is one of the authors of the paper I followed up."

4. BENEFITS OF LOCATING IN TEL AVIV

The fact that YoungEC 19 took place in Tel Aviv, Israel had many benefits for the young researchers and the other attendees. The area surrounding Tel Aviv is arguably one of the most academically-dense regions, particularly for EconCS, with all of the top five universities of Israel within an hour. Hence many faculty, grad students, and even prospective students of these universities were able to attend YoungEC.

One such attendee was Tomer Manket, who had recently finished his masters and was in the process of determining if, where, and in what topic he would pursue a PhD. After the workshop, he said, "The YoungEC 19 was the first AGT conference I've ever attended, and I'm very glad I had this opportunity. I love the combination of theoretical research and its real-world applications. The exposure to the great young researchers inspired me to consider this direction in my future studies."

YoungEC was colocated with TAU Theory-Fest, a six-day conference at Tel Aviv University hosting senior plenary speakers broadly within theoretical computer science, including Christos Papadimitriou and Vijay Vazirani. The Theory-Fest also included six sub-workshops, of which YoungEC was one. This enabled the many Theory-Fest participants to attend some of the YoungEC workshop, and for the YoungEC participants to take advantage of TAU Theory-Fest.

The workshop's location included other benefits as well—an incredible workshop dinner with Israeli food, the opportunity to celebrate New Year's Eve in Tel Aviv, and a tour of historic sites in Jerusalem the day after the workshop. The tour included sites of importance to various religions, with the history of how control has passed from different group to different group over time. Shuran commented, "The trip to Jerusalem was a wonderful bonus. My knowledge about the history was a total blank before that." Neil agreed about the benefits of the location, "Highlights (aside from the great talks) included a group tour of the old city of Jerusalem and sampling Israeli cuisine (shawarma, falafel, sabich, café hafuch, etc.)" It's likely that all attendees would agree that the workshop contained perfect talks, opportunities for interaction, and coffee.