Research Statement

Yogeshwer Sharma

The world around us is *uncertain* and *uncoordinated*. For algorithm designers, this is not a very pleasant situation to be in. Many algorithms we know well assume that the input to the problem is accurate and complete. In the world of Internet and autonomy that we live in today, this assumption seems to hold in fewer and fewer scenarios, ranging from finding information in a distributed system, to deciding what advertisements to place on a search engine. Hence, we need to solve problems when the input is only partially available, and/or it is held by self-interested rational agents. I work on *designing and analyzing algorithms* for these settings and examining whether the lack of certainty or coordination forces us to settle for (provably) worse solutions.

My research lies in *decision-making with partial information* (input not available completely), *game theory* (input held by self-interested agents), and *approximation algorithms* (finding optimal solution is provably hard). The main thrust of my research has been to come up with expressive frameworks that model the applications more faithfully, and design and analyze algorithms for resulting models.

Decision making under uncertainty and lack of coordination

An extensively investigated problem for studying the uncertainty in the environment is the *multi-armed bandit* (MAB) problem. Although it was originally motivated by design of clinical trials in medicine [7], due to its wide applicability in areas ranging from collaborative decision systems [9] to placing ads on webpages [8], it has turned out to be of immense practical value in statistics, computer science, and related fields. In the MAB problem, the algorithm (decision maker) has a choice of K arms (options) in each of the T sequential time rounds. Each arm gives a certain reward in every time round. The algorithm decides to pull one arm (choose one option) in each time round, observing the rewards associated with its choices. The goal is to maximize the cumulative reward collected in T rounds. The performance of the algorithm is measured in terms of its *regret*: the difference between the reward obtainable by always following the *single best arm* (the *benchmark*) and the reward obtained by the algorithm. There are well-known efficient algorithms which can achieve regret that grows only sublinearly in T.

Despite its applicability, there are certain limitations of the basic model described above. For example, it is implicit in the algorithms that the number of arms is small, all arms are always available, their rewards can be observed accurately, the decision maker is risk-neutral and so on.

The focus of my research in this area has been to notice the applications of the MAB problem where these assumptions don't hold, and relax the assumptions in order to model the application scenarios more faithfully. The two important scenarios that I have applied this approach to are (1) online sponsored search auctions, where the arms are strategic advertisers, and the algorithm cannot observe the true rewards, and (2) many cases of interest in computer systems where not all options are always available.

(1) Strategic arms As mentioned before, one important assumption in the MAB problem is that the algorithm can observe the *accurate* reward for the chosen arms. Although this is often the case, there are many important exceptions. Consider, for example, sponsored search auctions, where a search engine needs to decide what advertisement (ad) to show in an ad slot besides the search results, in order to maximize the social welfare. When a specific ad is shown and clicked on, the search engine doesn't know precisely how much "reward" it got, since the reward derived from the click goes to the *strategic* advertiser, not to the search engine.

When we model sponsored search auctions as an MAB problem by identifying ad i with arm i that yields expected reward $c_i v_i$ (c_i: *click through rate*, v_i: *value per click*), the arms have effectively become strategic agents, because v_i is the value that ad i derives when clicked on, which only ad i knows about and the algorithm cannot directly observe it. Ads might have an incentive to either understate or overstate their value per click. We therefore need to look for equilibrium behavior of strategic ads. We focus on the equilibrium where every ad gets the most benefit by telling the true value per click to the search engine. How do we design the algorithm such that truthtelling is indeed an equilibrium?

In [1], we consider the *truthful multi-armed bandit* problem: a strategic version of the classical MAB problem, which models the sponsored search auction scenario described above. We model the auction as a mechanism design problem, in which each agent (ad) i bids a value b_i as a proxy for its true value per click v_i (b_i may not be equal to v_i). The allocation algorithm then allocates ads (to the ad slot) for T time rounds. At the end, it charges payments p_i from agent i, in return of showing its ad (called the charging scheme). The pair (allocation algorithm, charging scheme) is called the mechanism. It is *truthful* if each agent derives as much utility from bidding her true value, as she can derive from bidding any false value, where the utility of agent i is defined as the number of clicks she got times her value per click minus the payment she paid. We aim to find a truthful mechanism which minimizes the regret (defined as in the classical MAB problem).

Can we find as good mechanisms (in terms of low regret) for the truthful MAB problem as there are algorithms for the classical MAB problem? We investigate this question in [1], and indeed prove rigorously that the structure of truthful mechanisms for truthful MAB problem is very restrictive; in particular, they cannot simultaneously explore arms and exploit an empirically good arm. As a result, these mechanisms end up performing much worse (sometimes exponentially worse) than the algorithms for the classical MAB problem. More quantitatively, any truthful mechanism must suffer a regret of $\Omega(K^{1/3}T^{2/3})$, while best algorithms for classical MAB problem are known to achieve regret $\tilde{O}(K^{1/2}T^{1/2})$, where K is the number of arms. This work also suggests that a truthful mechanism sometimes must show ads without charging anything, which is somewhat counterintuitive.

(2) Unavailable arms Another implicit assumption in the MAB problem is the availability of arms in every round. Consider the problem of picking a good node from which to download a file in a peer-to-peer network (in general, the problem of using a resource in a distributed system. We can model this as an MAB problem by identifying nodes with arms. However, the assumption that arms are always available is clearly violated. In a realistic setting, nodes enter and leave constantly, they are down for maintenance, certain parts of the network are unreachable, and there is no guarantee that any node is always available. In [2], we consider the *sleeping bandit problem* where the arms are arbitrarily unavailable in time rounds. As no single arm is always available anymore, we suggest an extension of the "single-best arm benchmark", called the *ordering benchmark*, which orders the K arms according to the best possible order (out of $K!$ feasible orders), and always picks the first available arm in the chosen order. The ordering benchmark enjoys the following nice properties: (a) it reduces to the usual single-best arm benchmark when all arms are available, (b) it is natural since this is how people seem to pick one option from the set of available options, and (c) it gives the optimal strategy when the algorithm knows the distribution of arms' rewards.

When the rewards for arms in each round are independent samples from arm-dependent distributions, we provide an efficient algorithm which suffers a regret that is best possible within a constant factor of what is achievable. The lower bound is a particular novelty in this work, since it holds uniformly over time, rather than only in the limit of time horizon approaching infinity. We also give results in different models of feedback (partial versus full feedback) and different models of rewards (stochastic versus adversarial rewards). For all these variants, we provide algorithms that achieve regret that almost or exactly match the best achievable. This paper received the *Best Student Paper Award* at COLT 2008, and was invited to the special issue of Machine Learning Journal.

Future directions

The research area exploring the design of algorithms in situations where information is not available centrally is an exciting one, and has a lot of potential, in my opinion. With the widespread use of distributed systems, and genuine concerns over privacy of individuals, this design space is going to get even richer. I plan to continue to explore this space. Here are a few concrete directions.

Price of truthfulness Our results in [1] can be thought of as answering the question: how much worse does the *achievable regret* of an online problem become if the rewards are held by self-interested agents? A related and fascinating question is: how much worse does the *achievable approximation factor* of a problem become if its input is held by self-interested agents? Many results establishing the bounds on the power of truthful mechanisms (see e.g., [10]) first characterize the structure of such mechanisms (proving they must be "affine-maximizers") and then bound the power of affine-maximizers. I am working on *characterizing problem domains* where any truthful mechanism has to be an affine-maximizer. By this characterization, we get simpler proofs of many results, and hope to get insights into what conditions on the problem domains might lead to a general separation result between power of truthful mechanisms and general algorithms.

There are several results that prove separation between power of truthful mechanisms and general algorithms for restricted classes of mechanisms (see [13] for scheduling problems). I am very interested in strengthening these results to arbitrary mechanisms and working on such separation results for various other problem domains.

More realistic models for sponsored search There are many interesting questions about sponsored search that my research in [1] has raised, and I want to pursue them. For example, in [1], we consider an "ex-post" definition of truthfulness, which requires that bidding truthfully be a dominant strategy for every outcome of clicks. But sometimes, this might be too strong a requirement. A natural relaxation is that bidding truthfully be a dominant strategy *in expectation* over the clicks. Do truthful mechanism still suffer worse regret than general algorithms? It will be satisfying to answer this question in the affirmative or negative. What if don't insist on exactly truthful, but only *approximately truthful* mechanisms? Also, it is also natural to consider a *different notion of equilibrium* (e.g., Nash equilibrium)? Can such relaxed mechanisms perform as well as general algorithms? Another limitation of our work is that it does not take into account *click-through-rate estimates* of ads and *similarity information* among ads, information that a search engine is able to estimate (albeit it is hard to model). I would like to investigate how to model these distributional assumptions in a tractable way and get results that can be directly applied to the real world settings.

Risk averse MAB Another implicit assumption in the MAB problem is that the decision maker is *risk-neutral*, meaning that it only cares about the expected reward (and not about, e.g., the variance). This assumption somewhat breaks down when we apply this model to the financial market setting for example. A related drawback is that the outcome might not be "diverse": our optimal solution (the benchmark) of always picking the same best arm is not diverse for example. In other words, there is no provision for "covering the bases". I want to look into these issues where the agents are not risk-neutral or when the solution required needs to be diverse (for example, when the reward of an arm diminishes if it is played very often).

Beyond MAB And finally, MAB problem is just one way to deal with the uncertainty in the data. *Online algorithms* and *stochastic optimization* are others, to name some. I plan to investigate how game-theoretic situations affect the online and stochastic models, and how they can be modified when strategic agents are interacting with them, instead of being given an honest input.

Other areas of interest

En route to developing my research agenda above, I have also explored various areas in the theoretical computer science, and investigated problems in approximation algorithms and game theory, which I describe next.

(1) Network design Network design is the general area of designing low-cost networks with specified properties. A classical problem in network design is the prize-collecting Steiner forest problem, where the input is a graph $G = (V, E)$, edge costs $c : E \to \mathbb{R}$, k pairs (s_i, t_i) of vertices with associated penalties π_i. The goal is to find a subset of edges so as to minimize the sum of costs of the chosen subset plus the penalties of disconnected pairs. The motivation comes from k pairs of people wanting to connect by phone, who are willing to pay π_i amount for the connection, and the phone company designing the network to minimize the edge costs plus the forgone profit from disconnected clients.

We observe in [4] that the penalty of pair i could depend on which other pairs are disconnected: for example, if all my friends are disconnected, then I would be willing to pay less compared to when my friends are connected. This intuition is well captured by a submodular penalty function. For submodular penalties, we are able to give constant (2.54) factor approximation algorithms using two different well known techniques: the primal-dual approach and LP rounding.

In [3], unlike previous works, we model general connectivity—each connection required a specified number of times—and give a constant factor approximation algorithm based on Jain's iterated rounding.

(2) Improvement over the Nash equilibrium Envision a network administrator who sets two prices for her service: high price payers get to choose their routes in the network; low price payers are routed by the administrator to minimize average delay. How much traffic she should try to control (and how much profit she should forgo) in order to make the average delay in the network attractive to its customers? In [5], we consider the problem of finding the minimum amount of flow that the administrator must control in order to improve the quality of the resulting equilibrium over the Nash equilibrium. For parallel link networks with linear delay functions, we are able to find a simple expression for this quantity, which complements an earlier result in the literature that finds the minimum quantity to control in order to reach the optimal solution.

(3) Message-ferries Imagine a calamity-hit area where the communication infrastucture has been destroyed, or is seriously impaired. Rescue workers working in small isolated pockets need to collect information to get a global picture of the situation. One common way for this is by message-ferrying, where unmanned aircrafts traverse the network geography and carry messages from senders to receivers. A central problem here is to design the routes for the message-ferries to minimize "latency". In [6], we consider two common measures of latency—average latency, and maximum latency suffered by any packet—and provide very simple constant factor approximation algorithms to minimize both measures. Simulations results in the paper demonstrate the efficacy of the simple algorithm.

Future work Network design is a fundamental problem in algorithm design which is widely applicable. I plan to pursue an agenda of devising expressive and tractable models, analyzing them, and empirically evaluating them. The examples would of course be driven from applications, but here are some.

All or nothing connectivity In [3], we assume that if a pair has fewer than requested number of connections, the penalty incurred grows slowly as a function of number of disconnections. But sometimes, it is natural to consider "all-or-nothing" requirements. For such case, the progress has been slow because of the discontinuous nature of the penalty. I plan to pursue this in my research.

Bidirected cut relaxation for Steiner tree An intriguing question that I have thought about is whether the bidirected cut relaxation (a linear programming relaxation for the Steiner tree problem) is any better than the usual linear programming relaxation. After much work, we don't know the answer for general graphs. I want to apply an approach similar to [12]—enumerating extreme point solutions for small instances and computing the integrality gaps empirically—to see if we can get insight into the structure of the bidirected cut polytope.

Conclusions With the emergence of Internet (as a platform for strategic agents) and large scale social networks, more expressive models of learning in game theoretic settings are going to be more and more applicable in the years to come, and the tools from areas such as learning theory, machine learning, game theory and statistics are going to be of central importance. I believe that my skills, combined with my passion for learning, are going to help me become a successful researcher amongst the challenges of future. Through these challenges, I look forward to contributing to computer science in particular and society in general.

References

[1] Moshe Babaioff, Yogeshwer Sharma, and Aleksandrs Slivkins. Characterizing truthful multi-armed bandit mechanisms. *ACM Conference on Electronic Commerce* (EC), pages 79–88. ACM, 2009. Full version on arXiv http://arxiv.org/abs/0812.2291.

[2] Robert D. Kleinberg, Alex Niculescu-Mizil, and Yogeshwer Sharma. Regret bounds for sleeping experts and bandits. To appear in COLT'08 special issue for the *Machine Learning Journal*. Preliminary version in *Conference on Learning Theory* (COLT), 2008, pages 425–436.

[3] Chandrashekhar Nagarajan, Yogeshwer Sharma, and David P. Williamson. Approximation algorithms for prize-collecting network design problems with general connectivity requirements. *Approximation and Online Algorithms* (WAOA), pages 174–187. Springer, 2008.

[4] Yogeshwer Sharma, Chaitanya Swamy, and David P. Williamson. Approximation algorithms for prize collecting forest problems with submodular penalty functions. *ACM-SIAM Symposium on Discrete Algorithms* (SODA), pages 1275–1284. SIAM, 2007.

[5] Yogeshwer Sharma and David P. Williamson. Stackelberg thresholds in network routing games or the value of altruism. *Games and Economic Behavior*, 67(1):174–190, September 2009. Also appeared in *ACM Conference on Electronic Commerce* (EC) 2007, pages 93–102. Invited to Special Issue of *Games and Economic Behavior*.

[6] Lukas Kroc, Yogeshwer Sharma, Mahesh Balakrishnan, Ken Birman. An analytical approach to designing message ferry routes. Manuscript available online at http://www.cs.cornell.edu/~mahesh/pizzas-full.pdf, 2008.

[7] Donald A. Berry and Larry M. Pearson. Optimal designs for two-stage clinical trials with dichotomous responses. *Statistics in Medicine*, 4:487–508, 1985.

[8] Omid Madani and Dennis DeCoste. Contextual recommender problems. *UBDM*, 2005.

[9] Robert Kleinberg. Online decision problems with large strategy sets. Ph. D. Thesis, MIT, 2005.

[10] Christos H. Papadimitriou, Michael Schapira, Yaron Singer. On the Hardness of Being Truthful. FOCS 2008: 250–259.

[11] Sridhar Rajagopalan, Vijay V. Vazirani. On the Bidirected Cut Relaxation for the Metric Steiner Tree Problem. SODA 1999: 742–751.

[12] Boyd, S., Elliott-Magwood, P. The Structure of the Extreme Points of the Subtour Elimination Polytope of the STSP. Technical Report TR-2007-09, SITE, University of Ottawa, Ottawa, Canada.

[13] Itai Ashlagi, Shahar Dobzinski, Ron Lavi. An optimal lower bound for anonymous scheduling mechanisms. *ACM Conference on Electronic Commerce* (EC) 2009: 169–176.