Computers are supposed to be simple things, the executors of our commands. Trust in their circuits enables abstractions, and allows us to manage a computation at a much higher level than the gates processing our instructions.

But sometimes a computation involves or affects agents who can strategize. In such instances, the output of the computation can affect the input — how the computation works can affect how people act, or what information they submit as input to the computation. The way Google runs their page-ranking algorithm affects how site designers build websites. When Yelp asks for reviews, restaurants can change how they serve and incentivize customers to leave reviews. In such settings, the computer is more complicated than a collection of circuits: it involves all of the agents making strategic decisions in accordance with their own incentives rather than their instructions. When Google alters their page-ranking algorithm, they are not only programming servers in their own datacenters, they are programming all of the websites and all of the designers who will strategically react.

I call these strategic computations, and I want to understand them. One model of strategic computation is a mechanism: an algorithm where strategic agents possess the inputs to the algorithm and must be incentivized to reveal their information. I focus on non-truthful mechanisms where the strategic agents are not incentivized to reveal their actual information, but just to reveal something about their information. Non-truthful mechanisms are harder to analyze theoretically than truthful mechanisms, but are much more common in practice.

The first-price auction is one example of a non-truthful mechanism. In a (sealed-bid) first-price auction, bidders report bids; the highest bidder wins and pays his bid. The bid from the bidder is not exactly how much they value the item, but a strategically chosen amount based on how they think other bidders will bid.

I aim to develop and use the theory of non-truthful mechanism design as a means to understand strategic computing. I focus on mechanisms for resource allocation problems — auctions — because they offer a simplified setting to explore the issues of incentives, and because resource allocation problems are often at the heart of strategic computing problems.

Our current theoretical tools and models of non-truthful mechanisms fall short of usefulness as a broader theory of strategic computing in two ways I am working to address. First, many of our current techniques for analysis of non-truthful mechanisms rely on analytically solving for equilibrium, which is generally impossible in all but the simplest of settings. I have been working to come up with tools and analyses for non-truthful mechanisms that do not rely on knowing the equilibrium. Second, our current theories of non-truthful mechanism design rely strongly on precise assumptions of the decision-making behavior of the agents: that they are risk-neutral and always choose their optimal action. I am working in this light to make our understanding of mechanisms more robust to the exact risk-attitudes and exact decision-making behavior of strategic agents.

Tools for analysis of non-truthful mechanisms
Classical auction theory focuses on truthful auctions because they are theoretically elegant: the equilibria are easy to compute for designers and there is no need for com-
complicated strategic behavior from the agents. Despite this elegance, most auctions used in the real world are non-truthful [Ausubel and Milgrom 2006]. Non-truthful auctions generally lack equilibria that are easy to solve for analytically\footnote{One example of the challenge of solving for equilibrium: [Vickrey 1961] asked whether or not there was an analytical form to the equilibrium of a first-price auction with two bidders and values drawn from asymmetric, uniform distributions. The analytical form was finally found 50 years later by Kaplan and Zamir [2012].}, which impedes many theoretical analyses. To build up theories of non-truthful mechanism design, we need more tools for analyzing mechanisms without relying on the characterization of equilibrium.

Recent work on the smoothness framework in auctions [Roughgarden 2012; Syrgkanis and Tardos 2013] offers a hopeful approach: analyzing how agents respond to the bids of others, no matter whether in equilibrium or not can lead to robust theoretical guarantees for welfare without needing to solve for the equilibrium.

However, for general strategic computations incentives are not always aligned as nicely as they are for welfare in auctions. Welfare is the sum of all agents’ utilities, and so it naturally aligns the incentives of bidders and the designer. To provide useful building blocks for general strategic computations, we need to understand settings where incentives are not so perfectly aligned.

My recent work with Sam Taggart and Jason Hartline [2014] generalized the smoothness framework approach to the objective of revenue in auctions. When an auctioneer wants to maximize revenue, the incentives of the auctioneer are no longer so nicely aligned with the incentives of the bidders: the auctioneer wants to make as much money as possible while the bidders want to pay as little money as possible. Our work used the optimal auction theory of [Myerson 1981] to measure the alignment of the objectives of the bidders against the objective of the designer. When the objectives of bidder and designer are sufficiently aligned, then the bidders can be trusted to participate in the auction; when they are not aligned, the auctioneer must find a way to mitigate their impact. In a first-price auction with a few light regularity assumptions, we found implementing a reserve price is sufficient to eliminate the impact of agents with misaligned incentives.

The pattern we used is one that has much broader applicability than just revenue in auctions. Precisely measuring the alignment of the agents’ incentives against the designer’s objective and engineering the system to remove agents with misaligned incentives should be a fundamental tool for the analysis and engineering of general strategic computations. In my future work, I want to abstract this result as much as possible and understand for what objectives and what settings we can simplify the problem of analysis down to analyzing objective alignment.

**Decision Models**

To understand robust principles of the engineering of strategic computing, we need theories that allows us to account for differences in decision making behavior. A common assumption in auction theory and mechanism design is that agents are risk-neutral and choose exactly their utility maximizing bid. Most people do not make decisions this way: they are risk-averse, or they are loss averse, or they make reference dependent decisions [Kahneman and Tversky 1979].

Risk-aversion is one example of a behavioral complication that can have a large effect on auctions. Bidders in first-price auctions tend to bid much higher [Holt and Laury 2005] than they would have if they were risk-neutral: they are okay trading off happiness when they win for an increased chance of winning. Thus the revenue in a first-price auction can be much more than with risk-neutral bidders; similarly,
the optimal auction can achieve much more revenue. The optimal auction becomes much more complicated, and is generated through optimal control and many differential equations [Matthews 1983; Maskin and Riley 1984]. Yet the results from these auctions are not robust; they rely on knowing exactly how risk-averse a bidder is and exactly what their utility function is. Any changes to these and the optimality of the auction may disappear.

As a result, these optimal auctions are not used. Rather, people use auctions similar to the first-price auctions because they seem to work well. My work with Jason Hartline and Hu Fu [2013] offers an explanation: for a stylized model of risk-attitudes, the revenue of the first-price auction is always a constant fraction of the optimal revenue.

I want to understand this behavior more broadly — does the first-price auction behave well for all risk-attitudes, and when agents have differing risk-attitudes? What aspects of the first-price auction’s good behavior can we generalize to other strategic computations?

One primary limit of analysis of risk-averse mechanisms has been the aforementioned challenge of solving analytically for equilibrium. Present works have been limited to the far too symmetric settings for which we can solve for equilibrium: when all agents share the same risk-attitude as well as values for the good drawn from the same distribution.

To break beyond symmetric settings with easy-to-compute equilibria, we will need to leverage the tools discussed earlier for analyzing equilibrium without solving for it. Notably, the approach of measuring the alignment of agents’ and designer’s objectives against one another will be very helpful when dealing with different risk-attitudes or general differences in decision-making behavior. When our proofs focus so heavily on just decision making behavior, changes in decision making behavior are at the forefront and easily exposed for theoretical analysis. Risk-aversion is just a change in the bidders’ incentives that must be analyzed against the objective of the designer. I am very hopeful that these techniques will lead to a robust theoretical understanding of non-truthful auctions with risk-averse bidders.

Behavioral Economics offers a number of other explanations for the behavior of economic agents: they can be loss-averse; they can make reference-dependent decisions [Kahneman and Tversky 1979; Kszegi and Rabin 2006]; and sometimes they make decisions based on heuristics, or only choose approximately the best action. I think that laying the groundwork of approximation results for risk-averse agents will allow us to observe how the model changes as we change the behavior model of our bidders, and offer a glimpse of what will happen as we consider even more realistic models of decision making behavior.

**Conclusion**

I want to understand strategic computation. Non-truthful mechanisms offer a simplified setting with many of the critical components of general strategic settings: strategic behavior, agents with different objectives, and a need to learn about the underlying models of behavior of the agents.

Each of the research directions I have outlined will be essential to further our understanding of strategic computation. Each builds on the others: understanding how to analyze complicated equilibria lead to how to infer things about complicated equilibria and vice-versa. By combining these directions, I hope to develop more useful theories of mechanism design and bring non-truthful mechanism design closer to a general theory of strategic computing.
REFERENCES