Mechanism Design and Analysis Using Simulation-Based Game Models

by

Yevgeniy Vorobeychik

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in The University of Michigan
2008

Doctoral Committee:
Professor Michael P. Wellman, Chair
Professor Edmund H. Durfee
Associate Professor Satinder Singh Baveja
Associate Professor Emre Ozdenoren

I dedicate this work to my wife, Polina, and my daughter, Avital.

ACKNOWLEDGMENTS

There is little doubt that my dissertation would never have been completed, had it not been for the support of a great number of people, including family, friends, and faculty mentors. I am very greatful to my thesis advisor, Michael Wellman, for having guided me along these past six years with timely advice, encouragement, and invaluable ideas that have had a direct and profound influence on my work. I also want to thank my committee members, Satinder Singh, Ed Durfee, and Emre Ozdenoren. My numerous adventures into each of their offices had invariably left me rejuvenated and brimming with new ideas.

All of my published work to date is co-authored, and I am grateful to all of the co-contributors. Prominent among these are Chris Kiekintveld, my perennial officemate—before and after the move to our wonderful new building—who has been both a great collegue and a great friend, Daniel Reeves, who has been not only a collegue and a friend, but also a mentor, particularly in my days as a fledgling researcher, and Patrick Jordan, whose intellectual insight is often sharp and always invaluable. I am grateful to my collegues at Yahoo! Research, Jenn Wortman, John Langford, Lihong Li, Yiling Chen, and David Pennock for the fantastic time I spend working with them during my internship, and to Isaac Porche for his tutelage during my summer tenure at RAND Corporation.

Finally, I am grateful to my family for having supported me in my lengthy doctoral endeavor, particularly to my wife, Polina, who had been with me through the highs and the lows, and who gave me a most precious gift—a beautiful daughter Avital.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure

LIST OF APPENDICES

APPENDIX

ABSTRACT

As agent technology matures, it becomes easier to envision electronic marketplaces teeming with autonomous agents. Since agents are explicitly programmed to optimally compete in these marketplaces (within bounds of computational tractability), and markets themselves are designed with specific objectives in mind, tools are necessary for systematic analyses of strategic interactions among autonomous agents. While traditional game-theoretic approaches to the analysis of multi-agent systems can provide much insight, they are often inadequate, as they rely heavily on analytic tractability of the problem at hand; however, even mildly realistic models of electronic marketplaces contain enough complexity to render a fully analytic approach hopeless.

To address questions not amenable to traditional theoretical approaches, I develop methods that allow systematic computational analysis of game-theoretic models in which the players' payoff functions are represented using simulations (i.e., simulation-based games). I develop a globally convergent algorithm for Nash equilibrium approximation in infinite simulation-based games, which I instantiate in the context of infinite games of incomplete information. Additionally, I use statistical learning techniques to improve the quality of Nash equilibrium approximation based on data collected from a game simulator. I also derive probabilistic confidence bounds and present convergence results about solutions of finite games modeled using simulations. The former allow an analyst to make statistically-founded statements about results based on game-theoretic simulations, while the latter provide formal justification for approximating game-theoretic solutions using simulation experiments. To address the broader mechanism design problem, I introduce an iterative algorithm for search in the design space, which requires a *game solver* as a subroutine. *As a result, I enable computational mechanism design using simulation-based models of games by availing the designer of a set of solution tools geared specifically towards games modeled using simulations.*

I apply the developed computational techniques to analyze strategic procurement and answer design questions in a supply-chain simulation, as well as to analyze dynamic bidding strategies in sponsored search auctions. Indeed, the techniques I develop have broad potential applicability beyond electronic marketplaces: they are geared towards any system that features competing strategic players who respond to incentives in a way that can be reasonably predicted via a game-theoretic analysis.

CHAPTER 1

Introduction

I focus in this thesis on the problem of *mechanism design*. Broadly speaking, mechanism design is the problem of manipulating a system to achieve some desired end. In a sense, such a general definition seems almost meaningless: for example, we may now class an act of screwing in a light bulb under this rubric. The essence of mechanism design, however, is not in achieving the objective per se, but in the *ability to predict* the consequences of design actions on outcomes—and, thus, on the designer's objective. When designing and building a physical system (say, a car), predictability is governed by physical laws, albeit usually too complex to apply with complete precision, but in any case largely predictable. But a designer of economic systems—the problem which is the focus of my thesis—must predict the behavior of economic agents, who are, more often than not, people. No matter how complex physical laws may be that govern physical systems, they seem a far cry from the complexity of human behavior.

Faced with such an unenviable problem, the Economists gave birth to a new being, a *homo economicus*—an economic man. A homo economicus is millennia ahead of a homo sapiens. He is perfectly rational in all endeavors, completely unemotional, has perfect hindsight, and has an uncanny ability to instantaneously replicate himself and thereby parallelize computation. With this new creation, a new world of possibilities opens up, as economic systems can now be reduced to mathematics. The problem of predicting the behavior of economic agents becomes, thus, the problem of predicting the activity of utility-maximizing players.

The assumption that economic agents are rational has naturally drawn considerable criticism over the years, both of philosophical and empirical nature. As computational agent technology matures, however, it becomes easier to envision electronic marketplaces teeming with *autonomous computational agents*. The environment thereby produced would be closer than ever to an idealized Economic system, since computational agents are explicitly programmed for rationality (within bounds of computational tractability and problem complexity considerations). Naturally, since agents are designed to (nearly) optimally compete in these marketplaces, and markets themselves are designed with some objectives in mind, tools are necessary for systematic analysis of strategic interactions among autonomous agents. For example, when autonomous agents (or a mix of agents and people) compete as a part of a supply chain, we may be interested in determining the effect of these strategic interactions on the availability and price of inputs and outputs, individual agent profitability, and overall efficiency. As a result, economic analysis is becoming increasingly prevalent in Computer Science.

In analyzing economic systems, especially those composed primarily of autonomous computational agents, the complexities must typically be distilled into a stylized model, with the hope that the resulting model captures the key components of the system and is nevertheless amenable to analytics. Increasingly, the boundaries of analytic tractability are being pushed, particularly as the systems, as well as the participating agents, are being more precisely engineered, and as the computational barriers that had once rendered complex mechanisms impractical are now coming down en masse.[1] The advantage of the increasingly precise models of Economic micro-systems, such as complex (e.g., combinatorial) auctions, is the ability to model these quite closely in a stylized way. However, the complexity of most practical mechanism design problems typically renders them analytically intractable, even though they may often be precisely modeled.

The impediments to analytic approaches come in many forms. One common difficulty is that the design objectives, while often amenable to a relatively precise specification, are just as often unique to the particular problem at hand. As such, each problem instance

[1] Consider, for example, combinatorial auctions, which had in the past been shunned because of the complexity of the winner determination problem, but have now become ubiquitous (in academic literature, as well as practice) [Cramton *et al.*, 2006].

is unique, and it may well be difficult (and misleading) to relate it to past theoretical literature.

Another difficulty that is frequently faced is that standard stylized models of mechanism design problems treat them in complete isolation from their context. Context, however, is often of critical importance, both to designing a mechanism and to its ultimate participants. For example, the FCC auction of radio spectrum licenses involves participants (bidders) who may be competitors outside of the realm of the auction. As such, the interests of bidders in specific licenses may be governed not only by their prospective uses of these, but also by future competitive considerations. This poses a difficulty both because it suggests that each setting may need to be treated as unique, and because the models which accurately describe the resulting design setting must become complex and lose analytic tractability.

A final difficulty I would like to highlight is more technical than conceptual. As it turns out, much of the theoretical treatment of mechanism design relies on the *revelation principle*, which states that the outcome of any conceivable mechanism can be replicated by a mechanism which is *incentive compatible*, that is, under which participants have no reason but to report their preferences to the designer truthfully. An implicit assumption under this sweeping principle is that there do not exist arbitrary restrictions of the set of mechanisms which the designer can consider. What this may mean in practice is that the designer has freedom to propose any mechanism he chooses, without regard to whatever mechanisms are already in place. At times, complete design overhaul is feasible and may even be necessary; often, however, the powers that be are far more receptive to mechanism (or policy) proposals which require relatively minimal change, perhaps only in the values of several parameters. Such restriction of mechanism design to parameter tweaking does, however, in general invalidate the revelation principle and calls for alternative methods.

In my thesis, I address analytically intractable mechanism design problems. To do this, I develop a set of methods that allow systematic computational analysis of *simulation-based representations* of strategic interactions among multiple self-interested agents. On one level, I am eager to answer questions about outcomes of game-theoretic interactions given a particular setting—for example, given a specification of a supply chain. On another level, I introduce a framework for designing mechanisms that induce incentives among participating agents that are aligned with a particular specified objective. My techniques actually have broad potential applicability beyond economic systems or electronic marketplaces: they are geared towards any system which features competing strategic entities (businesses, people, autonomous agents) that respond to incentives in a way that can be reasonably predicted via a game-theoretic analysis.

I begin below by providing a high-level model of the mechanism design problem which provides structure for most of this work.

1.1 A High-Level Model

I model the mechanism design process as a two-stage, one-shot game in which the *designer* first chooses the mechanism, that is, the rules of the strategic setting, and the *players* (i.e, those that get to ultimately use the mechanism) are fully aware of the rules once they are in place. I depict this setup in Figure 1.1. The implication of this model is

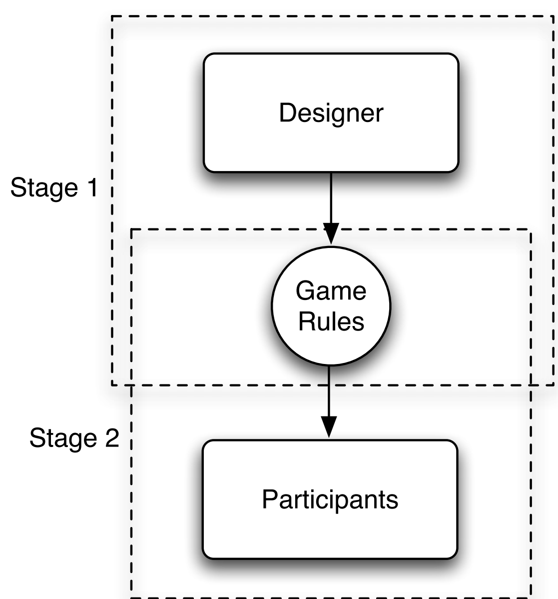


Figure 1.1: An informal diagram of the two-stage mechanism design model.

that the mechanism design problem can in principle be solved by backwards induction:

first, compute *predictions* of agent behavior for every possible mechanism choice, and second, select the mechanism that yields the best value of the designer's objective, *given* the predictions of play. In practice it will generally be infeasible to use backwards induction in the manner I just described for several reasons. First, I will deal with settings in which the design space is large or infinite. Second, even when the set of possible mechanisms is small, evaluating each requires solving a game to obtain predictions of play (e.g., Nash equilibria), a task well-recognized to be very challenging in general.

The spirit of backwards induction can still be salvaged, however, in the form of an iterative algorithm which repeatedly generates candidate mechanisms and evaluates each by solving the induced game (Figure 1.2). After a sequence of such iterative operations, we can select a mechanism which yields the best evaluation from all that have been tried.

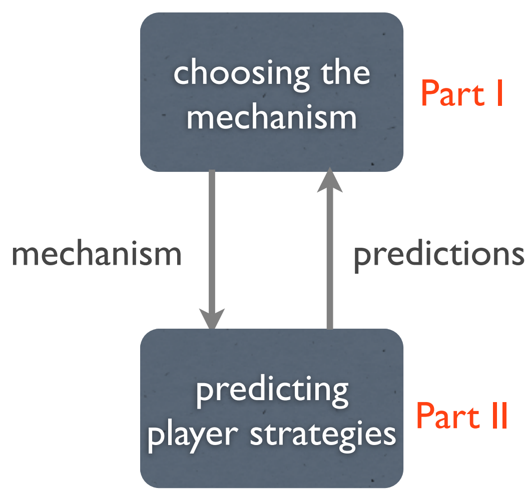


Figure 1.2: An iterative approach to mechanism design.

By treating the mechanism design process as an iterative evaluation algorithm, I effectively break the problem down into two subproblems:

1. solving the game induced by a particular mechanism choice (i.e., predicting outcomes of strategic interactions between mechanism participants), and

2. approximating or estimating an optimal mechanism when a game solver (i.e., a tool which yields predictions for any game in a class of interest) is available.

The subproblems above roughly correspond to the two parts of the backwards induction process. In this work, I provide a series of methods which address both of these problems. Specifically, in Part I, I discuss the second problem, whereas Part II provides methods to tackle the first.

1.2 Contributions

My contributions in this thesis are in the form of methods for simulation-based game-theoretic analysis and mechanism design, asymptotic convergence results, and empirical evaluations, as well as applications to particular strategic scenarios and mechanism design problems of interest.

One of my central contributions is a framework for performing general mechanism design on constrained (parametrized) design spaces. This framework is rooted in the two-stage mechanism design model I described above and involves an iterative mechanism exploration and evaluation process. I instantiate the framework in the setting of mechanism design for infinite games of incomplete information and present a heuristic randomized search algorithm (borrowed from the black-box stochastic optimization domain). I also derive some relevant probabilistic confidence bounds and provide a series of examples that demonstrate the efficacy of the approach.

On the front of approximating solutions to games in which payoff functions of the players are represented using simulations, my contributions fall into several categories. First, I offer a theoretical treatment of (small) finite games derived from noisy simulations. In this context, I provide theoretical convergence guarantees, as well as probabilistic confidence bounds for several game-theoretic solution concepts. I also show how to extend the confidence bounds to infinite games in some limited settings. Additionally, aside from relatively direct methods for estimating Nash equilibria in small finite games, I present an alternative method which incorporates information about the noise in the payoff estimates. Furthermore, I present algorithms for estimating Nash equilibria in infinite games, particularly in infinite games of incomplete information. One algorithm I present is provably convergent (in probability) to a Nash equilibrium (if one exists in the corresponding restricted strategy space). Other algorithms are not necessarily convergent. I also provide an extensive empirical analysis of the relative efficacy of the alternative algorithms in this context. My final algorithmic contribution is in using regression learning techniques for estimating payoff functions or game-theoretic regret functions based on simulation data. While I provide no theoretical guarantees about learning effectiveness in this domain, I do demonstrate empirically that the approach can be extremely effective, and, additionally, provide some empirical guidance in selecting the learning target depending on the particulars of the problem at hand.

Besides my broad methodological contributions, I explore several problems of mechanism design and game-theoretic analysis that are complex enough to lend themselves only to simulation-based methods. One such problem involves a supply-chain simulation, in which I provide evidence of a simulation design flaw through a systematic simulation analysis and suggest that this flaw *cannot* be fixed by an appropriate parameter setting. In another setting, that of dynamic bidding in sponsored search auctions,[2] I use simulations to study the relative stability properties (in the game-theoretic sense) of various dynamic bidding strategies in a restricted class.

1.3 Overview of Thesis

In this thesis, I address the two problems fundamental to the simulation-based mechanism design setting that I outlined above. The first of these, the problem of designing nearly optimal mechanisms under the assumption that we can solve games induced by them, is addressed in Part I. Chapter 4 describes the general methods for mechanism design using simulation-based games and applies some of these in the context of a supply-chain simulation. In Chapter 5 I describe a general framework for mechanism design on constrained design spaces for infinite games of incomplete information. I then apply the framework to several applications, one of which is analytically quite complex and unlikely to be solved optimally. Most of the applications provide ways to analytically

[2] Sponsored search auctions are the auctions run by the web search engines for placement of advertisements as a sidebar on pages generated by search keywords.

verify the solutions that the automated framework provides.

The problem of determining solutions to a game induced by a particular mechanism choice is the subject of Part II. In Chapter 7 I describe a series of methods for solving and analyzing finite games specified using simulations. In that chapter, I derive probabilistic bounds in order to assess how close the solution estimates based on simulation-based games are to actual solutions. To my knowledge, this is the first analytic derivation of such bounds for the setting of games with noisy payoffs. Additionally, I describe an application of the methods to analysis of a supply-chain simulation. Chapter 8 suggests using statistical learning techniques to improve Nash equilibrium estimates based on data collected from game simulations. Chapter 9 describes a series of methods for guiding simulations of infinite games and forming (a) estimates of best responses and (b) estimates of Nash equilibrium solutions. The effectiveness of all the methods is analyzed empirically in the context of single-item auctions and in a small combinatorial auction. In Chapter 10 I provide a detailed description of the application of finite-game methods to analysis of dynamic bidding in sponsored search auctions.

In Chapter 11 I attempt to go beyond Nash and approximate Nash equilibria by introducing the idea of belief distributions of play. There, I investigate several ways in which distributions of play can be heuristically formed and provide some probabilistic bounds for the associated solution concepts.

My final set of contributions is in providing some limited information about convergence, both for simulation-based games and for simulation-based mechanism design. The former analysis is provided in Chapter 7. The convergence of mechanism design based on simulations is addressed in Chapter 12, both when predictions of play are formed using (approximate) Nash equilibria and when they use certain kinds of belief distributions of play.

CHAPTER 2

Game-Theoretic Preliminaries

IN WHICH *I describe fundamental concepts from game theory and introduce some notation.*

This chapter is devoted entirely to introducing some basic game theoretic concepts and notation. It is by no means an exhaustive treatment of game theory or, particularly, of game-theoretic solutions: I focus exclusively on common solution concepts that are of most direct relevance to the current work. For an in-depth treatment of game theory, I refer an interested reader to any one of the plethora of texts, for example, the well-known books by Fudenberg and Tirole [1991] and Osborne and Rubinstein [1994].

2.1 Games in Normal Form

2.1.1 Notation

A generic normal-form (strategic form) game is formally expressed as

$$[I, \{R_i\}, \{u_i(r)\}],$$

where I refers to the set of players and $m = |I|$ is the number of players. R_i is the set of strategies available to player $i \in I$. Commonly, a distinction between *pure strategies* and *mixed strategies* is made, where pure strategies are some underlying deterministic choices by the players, while mixed strategies are probability distributions over the pure strategy space. Often, pure strategies are also called *actions*, and I will at times denote these sets of actions by A_i to indicate explicitly that atomic one-shot choices are meant. Much of this work is at the level of abstraction at which the distinction between pure and mixed strategies is not of much fundamental importance. More important is that the strategies can be restricted in arbitrary ways—for example, to pure strategies. R_i is thus used to denote any set of agent strategies, with or without particular restrictions. Whenever relevant, the specifics of this strategy set will either be clarified or will be clear from context. I denote the set of joint strategies of all players by $R = R_1 \times \cdots \times R_m$.

The utility function, $u_i(r) : R \to \mathbb{R}$ defines the payoff of player i when players jointly play $r = (r_1, \ldots, r_m)$, where each player's strategy r_i is selected from his strategy set, R_i. If the utility function is specified over pure strategies, it can be extended to mixed strategies by taking the expectation with respect to the corresponding distributions (and assuming a von Neumann-Morgenstern utility). That is, if r denotes a mixed strategy profile, $u_i(r)$ is:

$$u_i(r) = \int_A u_i(a) dr_1(a_1) \cdots dr_m(a_m).$$

When the set of pure strategies is finite, I let $r_i(a_i)$ denote the probability of playing a_i under r_i, with $u_i(r)$ defined by

$$u_i(r) = \sum_{a \in A} [r_1(a_1) \cdots r_m(a_m)] u_i(a).$$

It is often convenient to refer to the strategy of player i separately from that of the remaining players. To accommodate this, I use r_{-i} to denote the joint strategy of all players other than i.

Note that the normal form is a completely general representation of games, even though it appears to be defined in a one-shot fashion.[1] Specifically, if the game is actually dynamic and contains uncertainty, we can convert it to normal form by defining strategies to be functions of histories (including the history of nature's moves to incorpo-

[1]However, we may lose representational structure and, as a result, some important solution concepts defined on alternative game representations will no longer be captured in normal form.

rate uncertainty).

In this study I devote much attention to games that exhibit symmetry with respect to payoffs.

Definition 2.1 *A game* $[I, \{R_i\}, \{u_i(r)\}]$ *is* symmetric *if* $\forall i, j \in I$,

- $R_i = R_j$, *and*
- $u_i(r_i, r_{-i}) = u_j(r_j, r_{-j})$ *whenever* $r_i = r_j$ *and* $r_{-i} = r_{-j}$.

Symmetric games have relatively compact descriptions and may present associated computational advantages [Cheng *et al.*, 2004]. Given a symmetric game, we may focus on the subclass of symmetric equilibria, which are arguably most natural [Kreps, 1990], and avoid the need to coordinate on roles.[2] In fairly general settings, symmetric games do possess symmetric equilibria [Nash, 1951; Cheng *et al.*, 2004].

2.1.2 Basic Normal-Form Solution Concepts

In one-shot normal-form games, players make decisions about their strategies simultaneously and accrue payoffs, upon which the game ends. Faced with a one-shot game, an agent would ideally play its best strategy given those played by the other agents. A configuration where all agents play strategies that are best responses to the others constitutes a *Nash equilibrium*.

Definition 2.2 *A strategy profile* $r = (r_1, \ldots, r_m)$ *constitutes a* Nash equilibrium *of game* $[I, \{R_i\}, \{u_i(r)\}]$ *if for every* $i \in I$, $r'_i \in R_i$,

$$u_i(r_i, r_{-i}) \geq u_i(r'_i, r_{-i}).$$

While as a notion of stable strategic outcomes Nash equilibrium is quite compelling, it has been criticized on a number of grounds. One of the criticisms is that it requires a high level of rationality from players: each player must be able to actually compute a

[2] Contention may arise when there are disparities among payoffs in asymmetric equilibrium. Even for symmetric equilibria, coordination issues may still be present with respect to equilibrium selection.

best response. A weaker solution concept, one used also to approximate Nash equilibria, is an ϵ-Nash equilibrium.

Definition 2.3 *A strategy profile* $r = (r_1, \ldots, r_m)$ *constitutes an* ϵ-Nash equilibrium *of game* $[I, \{R_i\}, \{u_i(r)\}]$ *if for every* $i \in I$, $r_i' \in R_i$,

$$u_i(r_i, r_{-i}) + \epsilon \geq u_i(r_i', r_{-i}).$$

Observe that ϵ-Nash equilibrium generalizes Nash equilibrium: a Nash equilibrium is just an ϵ-Nash equilibrium with $\epsilon = 0$. Notationally, it is helpful to relate these concepts using game theoretic regret, which can be then used also as a measure of strategic stability. Thus, I define the *regret* of a profile r to be:

$$\epsilon(r) = \max_{i \in I} \max_{r_i' \in R_i} [u_i(r_i', r_{-i}) - u_i(r)].$$

In words, the regret of a profile is the most any player can gain by unilaterally deviating.

Another commonly used concept is *dominant strategies* and the *dominant strategy equilibrium* solution concept that arises as a result. It turns out that the definition of dominant strategies in game theory and mechanism design is slightly different. I first present the game-theoretic definitions of *weakly dominant* and *strictly dominant* strategies, and then provide the definition used in mechanism design.

Definition 2.4 *A strategy* $r_i \in R_i$ *is called* strictly dominant *if for every* $r_{-i} \in R_{-i}$,

$$u_i(r_i, r_{-i}) > u_i(r_i', r_{-i}) \; \forall \, r_i' \in R_i, r_i' \neq r_i.$$

A strategy $r_i \in R_i$ *is called* weakly dominant *if for every* $r_{-i} \in R_{-i}$

$$u_i(r_i, r_{-i}) \geq u_i(r_i', r_{-i}) \; \forall \, r_i' \in R_i$$

and there is $r_{-i}^* \in R_{-i}$ *such that*

$$u_i(r_i, r_{-i}^*) > u_i(r_i', r_{-i}^*) \; \forall \, r_i' \in R_i, r_i' \neq r_i.$$

Note that *weakly dominant* strategies as defined above are actually strictly better than any alternative for at least one joint strategic choice by other players. This may be somewhat counterintuitive given the term "weak", as one would perhaps expect it to imply that such strategies need never be strictly better than any such profile. The latter idea is behind the term *dominant* as it is commonly used in the mechanism design literature and also referred to as *very weakly dominant*.

Definition 2.5 *A strategy $r_i \in R_i$ is called* very weakly dominant *(or simply* dominant *in the mechanism design context) if for every $r_{-i} \in R_{-i}$*

$$u_i(r_i, r_{-i}) \geq u_i(r'_i, r_{-i}) \; \forall \; r'_i \in R_i.$$

Since this work has mechanism design as its primary focus, I will henceforth use the term *dominant strategy* in the last sense defined.

Given the notion of dominant strategies, the corresponding equilibrium is composed of dominant strategies used by all players.

Definition 2.6 *A profile $r \in R$ is a* dominant strategy equilibrium *if every player i plays a* dominant strategy*, that is, for every $i \in I$ and for every $r_{-i} \in R_{-i}$*

$$u_i(r_i, r_{-i}) \geq u_i(r'_i, r_{-i}) \; \forall \; r'_i \in R_i.$$

Note that a dominant strategy equilibrium is trivially a Nash equilibrium. While theoretically a very appealing solution concept, its main flaw is that unlike Nash equilibria, dominant strategy equilibria rarely exist in games.

2.2 Games of Incomplete Information and Auctions

I denote *one-shot games of incomplete information* (alternatively, *Bayesian games*) by $[I, \{A_i\}, \{T_i\}, F(\cdot), \{u_i(r, t)\}]$. A_i is the set of actions available to player $i \in I$, and $A = A_1, \times, A_m$ is the joint action space. T_i is the set of types (private information) of player i, with $T = T_1 \times \cdots \times T_m$ representing the joint type space. A one-shot game of incomplete information is said to be *infinite* if either A or T are infinite. Since I presume that a player knows his type prior to taking an action, but does not know types of others, I allow him to condition his action on his own type. Thus, I define a strategy of a player i to be a function $s_i : T_i \to \mathbb{R}$, and use $s(t)$ to denote the vector $(s_1(t_1), \dots, s_m(t_m))$. Let S_i be the set of all strategies s_i and define $S = S_1 \times \cdots \times S_m$. $F(\cdot)$ is the distribution over the joint type space. I denote the payoff (utility) function of each player i by $u_i : A \times T \to \mathbb{R}$, where $u_i(a_i, a_{-i}, t_i, t_{-i})$ indicates the payoff to player i with type t_i for playing action $a_i \in A_i$ when the remaining players with joint types t_{-i} play a_{-i}. Given a strategy profile $s \in S$, the expected payoff of player i is $\tilde{u}_i(s) = E_t[u_i(s(t), t)]$.

Note that the one-shot game of incomplete information can be represented in normal form by $[I, \{S_i\}, \{\tilde{u}_i(\cdot)\}]$, and the corresponding solution concepts then apply directly.[3] Particularly, the Nash equilibrium of the resulting normal form game is well-defined and is commonly referred to as the *Bayes-Nash equilibrium* of the underlying game of incomplete information. Similarly, we can define $\epsilon(s)$ with respect to strategy profiles $s \in S$ based on the deviations of each player within S_i.

A practically important class of one-shot games is *auctions*. In the case of one-shot auctions, the bidder actions are their bids b_i. The set of all actions is the set of all possible bids. For example, in single-item auctions, the set of all bids is just $\mathbb{R}$ if we allow for players to bid negative amounts (negative bids may imply agreements to be paid to obtain the item, which may be the case if the item is undesirable). The remaining aspects of the game are defined by the auction rules, which determine how the item is allocated (i.e., how the winner of the auction is determined), as well as payment (e.g., how much the winner has to pay for the item he won), and the joint distribution of player types.

[3] This definition can be further generalized to allow for mixed strategies, that is, probability distributions over strategies in S. However, I focus primarily on approximating pure strategy equilibria below, and, indeed, in much of my discussion the distinction is blurred.

2.3 Games in Extensive Form

2.3.1 A Formal Description

In this section I present the *extensive form*, which is a common model of dynamic games, that is, games in which players take actions over time. For example, a game in which one player moves in period one, while a second player moves in period two, after observing the action of the first, can be naturally represented in extensive form. Indeed, as I mentioned above, it is possible to represent such games in normal form also: the actions of all players would be *policies*, that is, actions conditional on observed history. In the example I just presented, the set of actions of the first player are identical in the normal and extensive form representation, since he has observed nothing prior to making his move. The second player, however, has observed the action of the first—let us call it a_1. His strategy, s_2, may then be different depending on which specific action a_1 he has observed. Thus, $s_2(a_1)$ will be a function of the first player's action. More generally, the extensive form provides a natural representation of the games which have such dynamic character by explicitly capturing the dynamic structure of the game, that is, by representing explicitly the order in which moves are made and what information is available at every point in the game to the decision-makers.

In this work, whenever I deal with extensive form games, I restrict attention to those that possess *perfect information*, that is, in which all actions made by players in any particular round are revealed in the round that follows. Formally, an extensive form game of perfect information is defined as a tuple

$$[I, H, P(\cdot), \{u_i(\cdot)\}],$$

where

1. I is a set of players
2. H a set of histories (of play) such that
   (a) The empty history $\emptyset$ is in H

(b) If $(a^k)_{k=1}^K \in H$ and $L < K$, then $(a^k)_{k=1}^L \in H$ (if a sequence of actions is viable, it must be that its subsequence is also viable)

(c) If any infinite sequence (a^k) satisfies $(a^k)_{k=1}^L \in H$ for every $L > 0$, then $(a^k) \in H$

3. P a function that assigns to each non-terminal history a subset of players.[4] Formally, suppose that Z is a terminal history (that is, no action can follow $z \in Z$). Then $\forall h \in H \setminus Z, P(h) \subset I$

4. $u_i(\cdot)$ is a payoff function for each player defined over the terminal nodes $z \in Z$

The tuple $[I, H, P(\cdot)]$ is called the *game form*, since it specifies all the aspects of the game but the payoffs. For any non-terminal history h, a set of joint actions available to all players in $P(h)$ is denoted $A(h)$ (or $A(h) = \times_{i \in P(h)} A_i(h)$). A (pure) *strategy* of a player i in the extensive form game is a function which assigns a (legal) action to every non-terminal history, that is $s_i(h) \subset A_i(h), h \in H \setminus Z$ (if $j \notin P(h)$, $A_j(h) = \emptyset$). A mixed strategy is a probability distribution over pure strategies.

A Nash equilibrium of the extensive form game is a profile of strategies s in which each player's strategy $s_i(h)$ is a best response to the profile $s_{-i}(h)$. While a very strong concept already, it suffers from certain theoretical flaws when the extensive form structure of the game is known. Specifically, there may be Nash equilibria which involve *non-credible threats* by some players. Informally, a non-credible threat by a player is a commitment to some decision at a future time which will not be optimal for the player once that decision point is actually reached.

To describe the flaw and propose a solution more formally, I start with defining a *subgame* of the extensive form game, which intuitively is a game which begins after a sequence of actions has already taken place. Let the notation $(a, b) \in H$ denote a sequence of actions a followed by a sequence of actions b.

[4] Technically, this is not quite a game of perfect information as it is typically defined, since I am allowing subsets of players to act concurrently and, thus, some of the players will not be able to observe the actions of others. However, all the actions become observable in the very next stage of the game. The reason for allowing sets of players rather than single players to play in some histories h is that it lends itself more naturally to repeated games without having to introduce an entire host of additional notation which would pertain to general games with imperfect information.

Definition 2.7 *A* subgame *of the extensive form game* $[I, H, P(\cdot), \{u_i(\cdot)\}]$ *that follows the history* $h \in H$ *is defined by*

$$\Gamma_E(h) = [I, H|_h, P|_h, \{u_i(\cdot)\}],$$

where $H|_h$ *is the set of sequences* h' *for which we have* $(h, h') \in H$ *and* $P|_h$ *is defined by* $P|_h(h') = P(h, h')$.

Given a subgame $\Gamma_E(h)$, a strategy $s_i(h')$ will naturally induce a strategy on a subgame, $s_i(h')|_h$. Now, the formal issue with the concept of Nash equilibrium as defined above is that even while it may be that a strategy profile s constitutes a Nash equilibrium of the extensive form game, the profile it induces in some subgame h may not itself be a Nash equilibrium in that subgame. The solution to this is to define a *subgame perfect equilibrium*, which guarantees that every subgame (and, consequently, the entire game) is in a Nash equilibrium under the strategy profile s.

Definition 2.8 *A* subgame perfect equilibrium *of a game* $[I, H, P(\cdot), \{u_i(\cdot)\}]$ *is a strategy profile* s *such that for every player* $i \in I$ *and for every history* $h \in H \setminus Z$ *for which* $i \in P(h)$, $s|_h$ *is a Nash equilibrium of* $\Gamma_E(h)$.

While the Nash equilibrium is, perhaps, the most common solution concept for normal-form games, a subgame perfect equilibrium is most useful in extensive form games with perfect information.

2.3.2 Infinitely Repeated Games with Complete and Perfect Information

The infinitely repeated game model divides time into an infinite number of discrete stages and presumes that at each stage players interact strategically in a one-shot fashion (that is, no one agent can observe actions of others until the next stage). Naturally, all players care not just about the payoffs they receive in one stage, but all the payoffs in the subsequent stages of the dynamic interaction. We assume that their total utility from playing the repeated game is a discounted sum of stage game utilities.

Formally, a repeated game can be described by the tuple $[I, \{R_i\}, u_i(r), \gamma_i]$, where I, R_i and $u_i(r)$ are as before, and γ_i is the amount by which each player discounts utility at each stage. That is, if we let $h = \{r_1, r_2, \ldots, r_i, \ldots\}, r_j \in R$ be an infinite sequence of choices (a terminal history) by players indexed by the chronological sequence of stages, then

$$U_i(h) = \sum_{t=1}^{\infty} \gamma_i^{t-1} u_i(r_t).$$

A non-terminal history in this model would be some finite sequence of joint player choices, $h_k = \{r_1, r_2, \ldots, r_k\}$, with k indicating the number of stages played.

Define a stage-k *subgame* of a repeated game as a restricted repeated game which begins at stage k rather than at stage 1. It would be characterized by the definition of $\Gamma_E(h_k)$ above. The solution concept that I will use for infinitely repeated games is the *subgame perfect Nash equilibrium*, which obtains when the players have no incentive to deviate from their sequence of strategic choices in any stage k given the history of equilibrium play until that stage, h_k.

Part I

Will the Optimal Mechanism Please Stand Up?

CHAPTER 3

Mechanism Design: Classical and Computational Approaches

IN WHICH *I discuss mechanism design concepts and literature.*

In this chapter I describe classical social choice theory and mechanism design, and then present some more recent computational mechanism design approaches.

3.1 Social Choice Theory

Social choice theory concerns itself with two closely related questions: how individual preferences can be aggregated to form a coherent social preference relation, and how individual preferences can be translated into social choices. A canonical example of preference aggregation is voting. For example, suppose that we have two presidential candidates. A straightforward preference aggregation technique would have all voters submit their votes for the candidate they prefer, and the majority of votes determines the president. As we will see below, this majority voting scheme "works" in general when there are two candidates, but cannot be effectively implemented (I will be precise about what this means below) when the number of candidates is above two.

Suppose that I is the set of agents (with $m = |I|$) and O the set of outcomes. Each agent $i \in I$ is assumed to have a rational preference relation, $\succ_i$ over O. Define $\mathcal{R}$ to be the set of all possible preferences over O.[1] I denote a profile (that is, an ordered collection) of all agent preferences by $\succ$ and let $\succ_{-i}$ denote the profile of preferences of all agents other than i. Let $\mathcal{A} \subset \mathcal{R}^m$ be some restriction on the set of possible preference profiles the agents may have and define the *social welfare functional (SWF)*, that is, the function which aggregates individual preferences into a social preference over O, to be $F : \mathcal{A} \rightarrow \mathcal{R}$. Let F_p indicate that F yields a strict social preference.

I now define some desirable properties of a *SWF*. The first of these is that whenever all agents strictly prefer o to o', so should F.

Definition 3.1 *F is* Paretian *if $\forall o, o' \in O$, $\succ \in \mathcal{A}$,*

$$o \succ_i o' \forall i \Longrightarrow o F_p(\succ) o'.$$

The second of these is the property that the social preference should not depend on irrelevant outcomes.

Definition 3.2 *F satisfies* independence of irrelevant outcomes (IIO) *if $\forall o, o' \in O$ and $\forall \succ, \succ' \in \mathcal{A}$ such that whenever*

$$o \succ_i o' \Longleftrightarrow o \succ'_i o' \text{ and } o' \succ_i o \Longleftrightarrow o' \succ'_i o$$

we have $oF(\succ)o' \Longleftrightarrow oF(\succ')o'$ and $o'F(\succ)o \Longleftrightarrow o'F(\succ')o$.

The next definition is actually a highly undesirable property of social welfare functionals:

[1] To simplify the discussion, I assume that all preferences with respect to distinct outcomes in O are strict.

it affirms that social preferences are consistently in line with preferences of one of the agents.

Definition 3.3 *F is* dictatorial *if $\exists h \in I$ such that $\forall o, o' \in O$ and $\forall \succ \in \mathcal{A}$, $o \succ_h o' \Longrightarrow oF_p(\succ)o'$.*

One of the most famous results in social choice theory is *Arrow's Impossibility Theorem*, which asserts that there are fundamental limitations in constructing *Paretian* and *IIO* social welfare functionals for all possible preferences over three or more outcomes.

Theorem 3.4 (Arrow's Impossibility Theorem) *Let $|O| > 2$ and $\mathcal{A} = \mathcal{R}^m$. Then every* SWF *that is* Paretian *and* IIA *is* dictatorial*.*

This result in effect suggests that as long as we consider the *Paretian* and *IIA* conditions important, then we must either restrict the class of preferences that the *SWF* can support, or loosen the notion of rationality in F.[2] It is some consolation that the impossibility only obtains when there are at least three outcomes in O. With two outcomes, preferences can be aggregated effectively by weighted majority voting schemes.

If we suppose that a social preference F over O is defined, it is clear how social choice may then be made: a policy-maker should select the most preferred outcome with respect to F. The question that may arise, however, is whether we can short circuit the process entirely if we need only to make an effective social choice for every preference profile. This, indeed, is the second question of social choice theory. To begin, let me define the *social choice function (SCF)* to be $f : \mathcal{A} \to O$. In words, *SCF* selects an outcome for every preference profile in the restricted preference space $\mathcal{A}$. As above, I now present several desirable properties of social choice functions. The first is that it choose a Pareto optimal outcome for every preference profile.

Definition 3.5 *f is* weakly Paretian *if $\exists o, o' \in O$ such that $o \succ_i o' \forall i \Longrightarrow f(\succ) \neq o'$.*

The next desirable property is a form of monotonicity of f: we want to ensure that an outcome o cannot drop from being chosen by f unless it becomes undesirable by at least one agent.

[2]For example, one widely considered restriction (at this level of abstraction, in any case) has been to *single-peaked preferences*. For a more detailed discussion, see Mas-Colell *et al.* [1995].

Definition 3.6 *f is* monotonic *if* $\forall \succ, \succ' \in \mathcal{A}$, *if* $f(\succ) = o$ *and* $o \succ_i o' \implies o \succ'_i o'$, *then* $f(\succ') = o$.

The final definition describes the undesirable dictatorial property similarly to the above.

Definition 3.7 *f is* dictatorial *if* $\exists h \in I$ *such that* $\forall \succ \in \mathcal{A}, f(\succ) \in \{o \in O | o \succ_h o' \forall o' \in O\}$.

That is, there is a dictator agent h such that f never makes a choice which is suboptimal for h. The next theorem presents the impossibility result regarding *weakly Paretian* and *monotonic* social choice functions akin to Arrow's Impossibility Theorem.

Theorem 3.8 *Let* $|X| > 2$ *and* $\mathcal{A} = \mathcal{R}^m$. *Then every* weakly Paretian *and* monotonic SCF *is dictatorial.*

This theorem is a corollary of Arrow's Impossibility Theorem. It affirms that we cannot sidestep the basic impossibility of aggregating preferences in a "reasonable" way, even if we ultimately only care about the choices.

While social choice theory does not address the question of incentives, it provides a bound on how much can be achieved in mechanism design: naturally, if the impossibilities obtain when the designer can determine agent preferences, they certainly will when he must elicit these. To formalize the connection, I first define what it means for agents to have an incentive to misrepresent their preferences.

Definition 3.9 *f satisfies* no-incentive-to-misrepresent (NIM) *if* $\forall h \in I,\ \succ_h \in \mathcal{R},\ \succ \in \mathcal{R}^m$,

$$f(\succ_h, \succ_{-h}) \succ_h f(\succ'_h, \succ_{-h}).$$

The following theorem states that it is essentially impossible to simultaneously ensure *NIM* and Pareto optimality.

Theorem 3.10 *Let* $|X| > 2$, $\mathcal{A} = \mathcal{R}^m$. *Then, any f which is* weakly Paretian *and satisfies* NIM *is dictatorial.*

Notice that the *NIM* condition may be somewhat reminiscent of the notion of dominant strategy equilibrium. This is no coincidence: in the next section I will build towards a similar impossibility result, known as the Gibbard-Satterthwaite theorem.

3.2 Mechanism Design Under Incomplete Information

3.2.1 Setup

Until now the discussion has been in terms of abstract preferences over the set of outcomes O. The mechanism design theory, however, generally focuses on settings in which agent utility functions are available. Thus, besides, the preferences $\succ_i$ for all players $i \in I$, the players are also characterized by utility functions $u_i(\cdot)$. In the case of incomplete information, the player utility functions are private information. Formally, let T_i be the set of possible *types* of player i and T be the set of type profiles of all players. The utility function for i is $u_i : O \times T_i \to \mathbb{R}$; thus, $u_i(o, t_i)$ assigns a real numbered utility value designating player i's utility for outcome $o \in O$ when i's type is $t_i \in T_i$. Furthermore, let $F_i(t_i)$ be the (commonly known) distribution of i's types. Finally, let $\mathcal{A}$ be the set of all preference profiles generated by T. In this setting, the *social choice function* is redefined to be $f : T \to O$, assigning an outcome for every profile of player types. One desirable property of a *SCF* is that it is *ex post efficient*, defined as follows.

Definition 3.11 *f is* ex post efficient (EPE) *if $\forall t \in T, o \in O$, either*

$$\forall i \in I, u_i(f(t), t_i) \geq u_i(o, t_i)$$

or

$$\exists i \in I s.t. u_i(f(t), t_i) > u_i(o, t_i).$$

That is, for any type profile $t \in T$, f selects a Pareto optimal outcome with respect to the player utilities.

In formalizing the problem of mechanism design and implementation, one needs to define precisely what is meant by a *mechanism*.

Definition 3.12 *A* mechanism $[R = \{R_1, \ldots, R_m\}, M]$ *is a collection of strategy sets R_i and an outcome function $M : R \to O$.*

Observe that a fixed mechanism defines a game of incomplete information between players in I. To see this, let $\tilde{u}_i(r, t_i) = u_i(M(r), t_i)$ and define the Bayesian game to be

$$\Gamma_{[R,M]} = [I, R, \{T_i\}, \{F_i(\cdot)\}, \{\tilde{u}_i(\cdot)\}].$$

A special important class of mechanisms, *direct* mechanisms, limits the players to strategies which are in effect declarations of their types.

Definition 3.13 *A* direct *mechanism has* $R_i = T_i$ *for all agents* $i \in I$.

Much of classical mechanism design has as its focus the goal of achieving desirable social choice functions as an equilibrium of the induced Bayesian game. Most commonly, two equilibrium solution concepts are considered: dominant strategy equilibrium and Bayes-Nash equilibrium. I first address the results, both positive and negative, when implementation in dominant strategies is desired, and later when Bayes-Nash equilibrium implementation would suffice.

3.2.2 Implementation in Dominant Strategies

To begin, recall the definition of a *dominant strategy equilibrium* from Chapter 2, slightly modified to fit the notation of this section and specific to one-shot games of incomplete information:

Definition 3.14 *A profile* $s(t) \in S$ *is a* dominant strategy equilibrium *if every player* i *plays a* dominant strategy*, that is, for every* $i \in I, t_i \in T_i$ *and for every* $a_{-i} \in A_{-i}$

$$u_i(M(s_i(t_i), a_{-i}), t_i) \geq u_i(M(a_i', a_{-i}), t_i) \; \forall \; a_i' \in A_i,$$

where $S = S_1 \times \cdots \times S_m$*, with* S_i *the set of all functions* $s : T_i \to A_i$*.*

The notion of *implementation in dominant strategies* then requires that the mechanism achieves a social choice function in *some* dominant strategy equilibrium.[3]

[3]But not necessarily in *all* dominant strategy equilibria. The notion of implementation that I focus on here is *weak implementation*. Under *strong implementation*, which I discuss below, all equilibria have to implement the desired social choice function.

Definition 3.15 *The mechanism* $[A, M]$ *implements* f *in dominant strategies if there exists a dominant strategy equilibrium* $s^*(t) = \{s_1(t_1), \ldots, s_m(t_m)\}$ *such that* $M(s^*(t)) = f(t)$ *for all* $t \in T$*.*

While at the level of abstraction that I have taken thus far, the mechanism design problem seems generally quite difficult, a well-known "trick" substantially simplifies it. This "trick", known as the *revelation principle*, suggests that the designer need only consider the space of direct mechanisms which are *dominant strategy incentive compatible* or *strategyproof*, that is, in which the agents reveal their types truthfully.

Definition 3.16 *A mechanism* $[T, M]$ *is* dominant strategy incentive compatible (DSIC) *or* strategyproof *if* $s(t) = t$ *is a dominant strategy equilibrium.*

Theorem 3.17 (The Revelation Principle in Dominant Strategies) *Suppose that there exists a mechanism* $[A, M]$ *which implements* f *in dominant strategies. Then* f *can be implemented by a* direct strategyproof *mechanism.*

Proof. Let $[A, M]$ implement f in dominant strategies and let $s^*(t)$ be the corresponding dominant strategy equilibrium. This means that $M(s^*(t)) = f(t)$ and, consequently, for all players $i \in I$, $u_i(M(s^*(t)), t_i) = u_i(f(t), t_i)$. The mechanism $[T, f]$, thus, also implements f in dominant strategies. □

The proof of this theorem seems trivial, and, in a sense, it is. The depth of the result stems from the fact that f may have any desirable properties, and as long as these properties are attainable at all, they can be replicated by a direct truthful mechanism—the designer simply reproduces the outcomes which the players would have chosen by their strategic responses.

The revelation principle has become an almost universal theoretical tool for mechanism design. It can be used to derive positive results, since any mechanism $[T, M]$ which is strategyproof and implements f cannot be improved upon by considering a more general problem. The converse is that if f cannot be implemented by a strategyproof mechanism, it cannot be implemented by any mechanism.

The first result, one of the most famous general results in classical mechanism design, is a strong negative statement about the frontier of possibilities in dominant strategy implementation. It is known as the Gibbard-Satterthwaite Theorem.

Theorem 3.18 (Gibbard-Satterthwaite Theorem) *Suppose that $|O| > 2$ and finite, suppose that $\mathcal{A} = \mathcal{R}^m$, and let $f(T) = O$. Then f is implementable in dominant strategies if and only if it is dictatorial.*

This result has the flavor of several impossibility theorems above, and it is indeed, a consequence of these. It implies that in order to have any useful positive results in dominant strategy implementation, we must of necessity restrict the set of preferences that agents may have. The most famous such restriction is to assume that utility functions of all agents are quasilinear in the numeraire commodity (money). This restriction gives rise to a family of *Vickrey-Clarke-Groves* or *VCG* mechanisms, which I describe in the next section.

Vickrey-Clarke-Groves Mechanisms

Vickrey-Clarke-Groves or *VCG* mechanisms are, perhaps, the most famous class of mechanisms for both practical and theoretical reasons. Theoretically, this is a principled restriction on the set of agent preferences that yields powerful positive results. Practically, the restriction on agent utilities applies in the common models of two important domains: the design of auctions and the decision to fund a public project.

The idea behind *VCG* is remarkably ingenious in its simplicity: pay each player the amount of his effect on social utility; in doing so, you align the incentives of all players with social welfare. Formally, let O denote the finite set of non-payment outcomes. For example, this could be the set of possible allocations of the goods or the set of project choices. Let p_i denote the transfer (payment) to player i. The utility of player i is then

$$u_i((o, p_i), t_i) = v_i(o, t_i) + p_i,$$

where (o, p_i) is what used to be entirely abstracted into the notion of "outcome", and $v_i(o, t_i)$ is the player's utility with respect to non-monetary outcomes. I refer to $v_i(o, t_i)$

as player i's *value* function. Since the utility function is quasilinear in payment p_i, it clearly is a strong restriction on the class of preferences, and, consequently, the Gibbard-Satterthwaite impossibility result need not apply here. The power of this restriction comes from the ability to affect each player's utility by imposing transfers which may depend on their reported types.

In the domain of quasilinear preferences, the social choice function f takes the form $f(t) = (o(t), p(t))$, with $p(t) = \{p_1(t), \dots, p_m(t)\}$. Ex post efficiency with respect to the choice $o(t)$, in turn, can be redefined as follows.

Definition 3.19 *The outcome choice function* (o^*) *is* ex post efficient *if for every* $t \in T$

$$o^*(t) \in \arg\max_o \sum_{i \in I} v_i(o, t_i).$$

In words, an ex post efficient $o(t)$ maximizes the sum of the players' values for all profiles of types. I now define the general *VCG* mechanism.

Definition 3.20 *The* VCG *mechanism is defined by the outcome function* $o^*(t)$ *in Definition 3.19 and the following payment rule:*

$$p_i(t) = \sum_{j \neq i} v_j(o^*(t), t_j) + h_i(t_{-i}).$$

The following theorem is a central positive result in classical mechanism design.

Theorem 3.21 VCG *implements an ex post efficient outcome function and is DSIC.*

Proof. The fact that it implements an *EPE* social choice function is clear from the definition of *VCG*, so it is only left to show that the implementation is, indeed, strategyproof. To see this, consider the utility function of player i when he reports t_i' as his type:

$$u_i(o^*(t_i', t_{-i}), p_i^*(t_i', t_{-i}), t_i) = v_i(o^*(t_i', t_{-i}), t_i) + \sum_{j \neq i} v_j(o^*(t_i', t_{-i}), t_j) + h_i(t_{-i}) =$$

$$= \sum_{j \in I} v_j(o^*(t_i', t_{-i}), t_j) + h_i(t_{-i}).$$

Let t_{-i} be any report by agents other than i. First, note that $h_i(t_{-i})$ does not depend on the agent's report t'_i. Furthermore, the social utility $\sum_{j \in I} v_j(o^*(t'_i, t_{-i}), t_j)$ is maximized by definition with respect to the profile of types (t_i, t_{-i}) when i reports his type truthfully. As such, the agent's utility is exactly aligned with social utility and he should report t_i no matter what the reports by other agents are. □

Perhaps the best-known *VCG* payment scheme, introduced by Clarke [1971], is

$$p_i(t) = \sum_{j \neq i} \left[v_j(o^*(t), t_j) - v_j(o^*_{-j}(t_{-j}), t_j) \right],$$

which amounts to the effect that agent i has on social welfare. This is known also as a *pivotal* mechanism, because the payment of agent i is non-zero only if the agent is pivotal, that is, if his presence has an effect on the choice of outcome $o^*(t)$.

While the results in the quasilinear domain seem very positive thus far, note that full efficiency of f would require also that no money would go to waste. Formally, f would need to satisfy the following *budget balance condition*:

$$\sum_{i \in I} p_i(t) = 0 \;\; \forall t \in T.$$

When this requirement is added, the situation becomes somewhat bleak, as the following impossibility theorem, due to Green and Laffont [1979], suggests.

Theorem 3.22 *Let $\mathcal{F}$ be the set of all possible value functions, that is, $\mathcal{F} = \{v : O \to \mathbb{R}\}$. Suppose that for each agent i, $\{v_i(o, t_i) | t_i \in T_i\} = \mathcal{F}$.*[4] *Then there is no strategyproof social choice function $f = (o(t), p(t))$ such that $o(t)$ is ex post efficient and $p(t)$ is budget balanced.*

3.2.3 Implementation in Bayes-Nash Equilibrium

The notion of dominant strategy implementation is rife with negative results in large part due to the fact that dominant strategies are not easy to come by. Most games that are

[4] In words, the set of all possible value functions is generated by value functions over the type space T_i.

studied, for example, do not have dominant strategy equilibria. On the other hand, Nash equilibria are guaranteed to exist in every finite game, as well as in most of the interesting classes of infinite games. Presumably, then, we can hope for more positive results if we consider achieving social choice functions in a Bayes-Nash equilibrium.

As before, I begin by adapting the definition of a Bayes-Nash equilibrium to the current notation.

Definition 3.23 *A profile* $s(t) \in S$ *is a* Bayes-Nash equilibrium *if for every* $i \in I$,

$$E_t u_i(M(s_i(t_i), s_{-i}(t_{-i})), t_i) \geq E_t u_i(M(a_i', s_{-i}(t_{-i})), t_i) \; \forall \, a_i' \in A_i, \; a.e. \; t_i \in T_i,$$

where $S = S_1 \times \cdots \times S_m$*, with* S_i *the set of all functions* $s : T_i \to A_i$*.*

Just as in the case of dominant strategy implementation, we can appeal to the revelation principle (this time, the Bayes-Nash equilibrium form of it) to substantially simplify the design problem. In this case, the principle states that if there is some arbitrary mechanism $[R, M]$ which implements the social choice function in a Bayes-Nash equilibrium, then it can also be implemented by a direct mechanism in which revealing the true type is a Bayes-Nash equilibrium. The latter property is known as *Bayes-Nash incentive compatibility*.

Definition 3.24 *A mechanism* $[T, M]$ *is* Bayes-Nash incentive compatible (BNIC) *if the strategy profile* $s(t) = t$ *is a Bayes-Nash equilibrium.*

As I mentioned, since *BNIC* is a substantially weaker restriction than *DSIC*, it should be easier to implement social choice functions given this solution concept. This is, indeed, the case, as manifested by the *expected externality mechanism* in the setting with quasilinear preferences [d'Aspremont and Gérard-Varet, 1979].

Definition 3.25 *The* expected externality mechanism *is defined by* $o^*(t)$ *given by Definition 3.19 and the following payment rule* $p(t)$*:*

$$p_i(t) = \sum_{j \neq i} \left[E_{t_{-i}} v_j(o^*(t_i, t_{-i}), t_j) - \frac{1}{m-1} E_{t_{-j}} \left[\sum_{k \neq j} v_k(o^*(t_j, t_{-j}), t_k) \right] \right].$$

Theorem 3.26 *Suppose that the types t_i are independently distributed for all i. Then the* expected externality mechanism *yields an ex post efficient $o^*(t)$ and is budget balanced.*

3.2.4 Participation Constraints

A now standard constraint in mechanism design problems has become to ensure that the players have an incentive to participate in the resulting mechanism. This requirement is quite natural: it is unlikely, for example, that any auction mechanism in which players lose money on average would ever be adopted. If a government levies too high a tax on its citizens, the first response may be to emigrate, while the next response would likely be a revolt. The typical requirement has in mind that the players will only participate in the mechanism if their utility from it is no less than that from the next best option. The strongest such constraint, most commonly used when the designer is interested in dominant strategy implementation, is *ex post individual rationality*.

Definition 3.27 *The mechanism $[R, M]$ is* ex post individually rational (EPIR) *if for all players $i \in I$ and all type profiles $t \in T$,*

$$u_i(M(t), t_i) \geq c_i(t_i),$$

where $c_i(t_i)$ is the opportunity cost of player i with type t_i.

A weaker form of the participation constraint, used generally in Bayes-Nash implementation problems, is *ex interim individual rationality*.

Definition 3.28 *The mechanism $[R, M]$ is* ex interim individually rational (EIIR) *if for all players $i \in I$ and all types $t_i \in T_i$,*

$$E_{t_{-i}}[u_i(M(t), t_i)] \geq c_i(t_i),$$

where $c_i(t_i)$ is the opportunity cost of player i with type t_i.

As the next section demonstrates, the participation constraints darken the heretofore rosy picture of Bayes-Nash implementation possibilities.

3.2.5 Bilateral Trade

Suppose there are two players, s (seller) and b (buyer). The seller, naturally, has some good for sale and values this good at t_s, whereas the buyer has value t_b for the same good. We would like the following to take place: if $t_s < t_b$, the buyer should purchase the good from the seller, paying a price p_b to the seller. Otherwise, no exchange should take place. Let's assume that the preferences are quasilinear in price and O is the set of all possible allocations of the good (that is, whether the buyer or the seller get the good). This procedure defines an *ex post efficient* social choice function, since the good always goes to the one with higher value, and it is clearly budget balanced. Now, suppose that participation is not required, that is, the mechanism must be *EIIR* for both the buyer and the seller. The question is, can an efficient social choice function be implemented as a Bayes-Nash equilibrium? The answer depends on the distribution of buyer and seller types.

Suppose that $t_s \in [\underline{t}_s, \overline{t}_s]$ and $t_b \in [\underline{t}_b, \overline{t}_b]$ and consider the case when $\overline{t}_s < \underline{t}_b$, that is, the buyer of necessity values the good more than the seller. The following mechanism is trivially *BNIC*, *EIIR*, and is *ex post efficient*: give the good to the buyer, who pays $p_b = \frac{1}{2}(\overline{t}_s - \underline{t}_b)$ to the seller. Indeed, any price between the highest seller and lowest buyer valuations would suffice. If the case is reversed and the seller always values the good more than the buyer, no exchange should ever occur. The most interesting case is when $[\underline{t}_s, \overline{t}_s] \cap [\underline{t}_b, \overline{t}_b] \neq \emptyset$. In this case, it turns out, it is impossible to implement an *ex post efficient* f as a Bayes-Nash equilibrium and at the same time satisfy the *EIIR* constraint.

Theorem 3.29 (Myerson-Satterthwaite Theorem) *Suppose that both the buyer and the seller are risk neutral, their values are drawn independently, and* $[\underline{t}_s, \overline{t}_s] \cap [\underline{t}_b, \overline{t}_b] \neq \emptyset$. *Then there is no* BNIC *and* EIIR *social choice function that implements an ex post efficient allocation and is budget balanced.*

3.2.6 Weak vs. Strong Implementation

Until now the discussion focused entirely on implementing social choice functions in *some* equilibrium, the notion referred to as *weak implementation*. Thus, if there is a multiplicity, the designer is allowed to assume that the participants will play the desirable equilibrium. This is commonly justified by suggesting that the designer presents the desirable equilibrium to the players, thereby ensuring common knowledge. The players are then presumed to follow the desirable equilibrium, since it is a mutual best response.

The idea of multiplicity of equilibria may nevertheless still leave a designer uncomfortable in real situations, when the common knowledge assumption may be relatively unsafe, even if the designer attempts to suggest an equilibrium to the players. Rather than relying on players to play the desirable equilibrium, the designer may opt for *strong implementation*, which ensures that the social choice function is implemented in *every* equilibrium.

Definition 3.30 *The mechanism* $[R, M]$ strongly implements *the social choice function* f *if every equilibrium* $r^*(t)$ *of the game induced by* $[R, M]$ *is such that* $M(r^*(t)) = f(t)$.[5]

3.3 Implementation vs. Optimality

The discussion on mechanism design has thus far focused on implementing particular desirable social choice functions (or, rather, social choice functions with desirable properties). A market or mechanism designer seems in practice unlikely to take such a limited view, particularly since he may often run into the impossibility results. Rather, a more relevant question may be how to attain the best possible mechanism, given some objective. While this question can be phrased in the terminology of the implementation theory, the theory itself has relatively little to say about it in general. Implementation theory is also not a very natural language to pose such problems.

An alternative and, arguably, more natural way to ask mechanism design questions is by posing the problem as constrained optimization, as was done, for example, by Myerson [1981]. Let $[R, M]$ be a mechanism as above and denote the designer's objective function by $W(r(M(t), t), M(t), t)$, where $r(M(t), t)$ is the profile of equilibrium strate-

[5]Perhaps a more practical notion of strong implementation may be that every equilibrium implements the specific desirable properties of a social choice functions, but different equilibria may implement different such functions f.

gies of the game induced by $[R, M]$. The general mechanism design problem would be

$$\max_{M:T \to O} E_t W(r(M(t), t), M(t), t).$$

It is easy to see that this formulation can incorporate social choice functions by using the following objective function:

$$W(r(M(t), t), M(t), t) = \mathbf{I}\{r(M(t), t) = f(t) \forall t \in T\}.$$

Similarly, it is not difficult to account for both weak and strong implementation, with weak implementation taking the maximum value of the objective with respect to the set of all solutions and strong implementation taking the minimum.

3.4 Auction Design

Auction theory is one of the most practically important applications of mechanism design. While auctions have been used from time immemorial, the formal game-theoretic study of the subject was first undertaken by Vickrey [1961]. There are various ways one may think of what generally constitutes an auction, some more and some less restrictive. A relatively restrictive notion is defined by Krishna [2002]: an *auction* is a universal mechanism for selling goods in that (a) it does not depend on the specifics of a particular set of goods for sale, and (b) the identity of bidders plays no role. Myerson [1981], on the other hand, takes a broader view of auctions and defines them as any mechanism for selling a good (or a set of goods) in which the potential buyers submit bids for the good(s) and the seller determines payments and allocation based on their bids. In what follows, I take the broader perspective on what it means to be an auction, mainly for the unity of exposition rather than any philosophical qualm about Vijay Krishna's definition. The review of the auction theory literature that I provide below is very cursory, omitting scores of important domains and results. Krishna [2002] is a wonderful text which provides a reasonably broad and readable overview of the subject; indeed, much of my discussion below is taken from it.

3.4.1 Single Item Auctions

I begin the discussion with the simplest and best understood class of auctions: single item auctions. Specifically, there is a seller that offers one indivisible good for sale. For convenience, assume that it is publicly known that the seller values the good at 0, whereas the buyer valuations v_i for the good are distributed on $[0, V]$ according to distributions $F_i(\cdot)$. Let v be the vector of valuations of all buyers. I assume that each buyer's utility is zero if he does not obtain the good and $u_i(v, p_i) = \bar{u}_i(v) - p_i$ if he does.[6] For the remainder of this section, I assume a pure private value model, that is, each bidder's utility depends only on his value.

I study two commonly considered design problems in auction design: maximizing revenue and maximizing welfare. The former objective function is formally described by

$$W(\cdot) = \sum_{i \in I} p_i(v) o_i(v),$$

where $o_i(v)$ is the probability that player i obtains the good. The latter is

$$W(\cdot) = \sum_{i \in I} v_i o_i(v).$$

Welfare Optimal Auctions

Auctions that maximize welfare are very much in the spirit of the general Economic pursuit of global welfare. It is worthy of note, however, that as the auctions are only one small part of the world, optimal welfare in auctions need not imply global welfare: it may well be that an auction that allocates the item to the highest bidder will as a result yield a monopoly in a related market. Nevertheless, the goal of allocating the item to the highest bidder seems quite sensible in many instances.

The fact that allocating the item to the highest bidder is, indeed, welfare optimal based on the objective defined above is not difficult to see. Suppose that values v_i have no ties. If there is any weight on $V' < \max_i v_i$ (that is, positive probability of allocating the good

[6]This utility function assumes that the bidders are risk neutral. See Krishna [2002] for a discussion of what happens when players are risk averse.

to V'), moving a little bit of this weight towards the highest value clearly increases the sum by linearity.

Once we recognize that a welfare optimal (alternatively, *efficient*) auction should allocate the item to the player with the highest valuation, it suffices to have the following three conditions to obtain efficiency:

1. the bids $b_i(v_i)$ are strictly increasing in v_i
2. the bids are symmetric, that is, $\forall i \in I, b_i(v_i) = b(v_i)$, where $b(\cdot)$ is the symmetric bid function
3. the auction allocates the item to the highest bidder

If the first two conditions hold, clearly the highest bidder has the highest valuation, and, therefore, efficiency is obtained by allocating the item to the highest bidder.

If the bidders are ex ante symmetric, the conditions above seem quite reasonable, and, indeed, efficiency is not difficult to achieve. Consider, for example, the case of bidders with independent and identically distributed private values, and let's look at two standard one-shot sealed-bid auction mechanisms. In the first, known as the *first-price sealed-bid auction (FPSB)*, the winner of the auction pays his bid. In the second, known as the *second-price sealed-bid auction* or *Vickrey auction*, the winner pays the second highest bid. In both of these, the winner is the player with the highest bid.

In the first-price auction, the following symmetric bidding strategy is a Bayes-Nash equilibrium [Krishna, 2002]:

$$b(v_i) = E[Y_1|Y_1 < v_i],$$

where Y_1 is the first order statistic of the values of $m-1$ players. As long as the distributions of valuations are strictly increasing in v_i, it is not difficult to see that so is the bidding strategy. In the special case when the values are uniformly distributed, the corresponding equilibrium strategies are

$$b(v_i) = \frac{m-1}{m} v_i,$$

so the bids are linear in player values.

The matter is much simpler with the second-price auction: bidding the true value is a dominant strategy [Vickrey, 1961]. To see this, fix the profile of all bids other than i at b_{-i}. Naturally, the bidder has nothing to gain by bidding $b > v_i$, since by doing so he increases his chances of winning the good precisely when he would also obtain a negative surplus. On the other hand, he also stands to gain nothing by bidding lower, since he may not obtain the good precisely when he would obtain a positive surplus from winning it. Indeed, the Vickrey auction is strategyproof even when valuations are not symmetric for precisely the same reason and is, thus, efficient in this case also. Bidding in the first-price auction under asymmetric valuations, however, yields asymmetric equilibrium strategies and, as a consequence, the auction may be inefficient [Krishna, 2002].

Revenue Considerations

It may well be argued that while efficiency is a worthy goal, it is secondary if the auction designer is the seller himself, or is someone whose incentives align with those of the sellers (e.g., if the designer receives commissions from auction sales). The primary goal would be to maximize revenue. One of the first general treatments of the problem is due to Myerson [1981]. His setting is that of independent private values, which may be asymmetrically distributed. A key element of his optimal auctions is the ranking of bidders by their *virtual valuations* rather than actual valuations (which he can always obtain w.l.o.g. because of the revelation principle).

Definition 3.31 *A* virtual valuation *of a bidder i is defined to be*

$$\phi_i(v_i) = v_i - \frac{1 - F_i(v_i)}{f_i(v_i)},$$

that is, it is the bidder's valuation less his reverse hazard rate.

If virtual valuations of all bidders are increasing, Myerson's mechanism ranks the bidders by their virtual, rather than actual, valuations.[7] The following theorem completely describes his mechanism, which he proves to yield optimal revenue to the auctioneer.

[7]Myerson also demonstrate how to solve the problem in the general case by "ironing".

Theorem 3.32 *The mechanism* $[o(v), p(v)]$ *such that*

$$o_i(v) > 0 \text{ iff } \phi_i(v_i) = \max_{j \in I} \phi_j(v_j) \text{ and } \phi_i(v_i) \geq 0.$$

and

$$p_i(v) = o_i(v)v_i - \int_0^{v_i} o_i(z_i, v_{-i})dz_i.$$

maximizes auctioneer revenue among all BNIC *auctions (and, consequently, among all auctions by the revelation principle).*

Note that this mechanism is not efficient, since it ranks the bidders by their virtual, rather than actual, valuations. Even with the assumption that virtual valuation are increasing in actual valuations, they may be asymmetric and thereby cause the bidders with lower values to win the auction. Indeed, even if we assume that the distributions of bidder valuations are symmetric, there may be an inefficiency simply due to the fact that the designer will choose to keep the good when virtual valuations are negative. Since the designer values that good at 0, this is inefficient. To consider a simple example of the "Myerson" optimal auction, suppose that the bidders' valuations are symmetric and uniform on [0,1]. The optimal auction would allocate the item to the highest bidder, but only when the highest bid is above a reserve price of $\frac{1}{2}$.

Besides his famous result, Myerson also presents a famous *revenue equivalence principle*.

Theorem 3.33 (Revenue Equivalence [Myerson, 1981]) *Any two incentive compatible auctions with the same allocation rule yield (essentially) the same revenue.*[8]

Similar results have been obtain in comparing first- and second-price auctions under symmetrically distributed valuations.

Theorem 3.34 *With independently and identically distributed private values, the expected revenue in a first-price auction is the same as the expected revenue in a second-price auction.*

[8] Specifically, the payments are equivalent up to a constant.

When valuations are asymmetric, however, the revenue comparison between the first- and second-price auction formats is ambiguous [Krishna, 2002].

3.4.2 Multi-Unit Auctions

Unlike single-unit auctions, there are relatively few general results in auctions in which multiple units of the same good are for sale. One exception is the extension of the Vickrey auction to this setting. Indeed, note that the Vickrey auction is actually a special case of the *VCG* mechanism I discussed above, and its extension to multi-unit auctions will be a specialization of *VCG* to this setting.

To formalize the setting, suppose that there are K units of a good for sale. Each player has a private vector of values, $v_i = (v_i^1, \ldots, v_i^K)$, such that each v_i^k indicates his marginal value for the kth unit of the good. The player submits a vector of bids b_i with an analogous interpretation. In order to attain efficiency, the auctioneer needs to allocate the good to the bidders with the K highest marginal valuations. If the values can be elicited truthfully, then, efficient allocation can be achieved. A more general result is the subject of the following theorem.

Theorem 3.35 *An equilibrium of a multi-unit auction is efficient if and only if there exists a bid function $b(v_i^k)$ such that for every player i and unit k*

$$b_i^k(v_i) = b(v_i^k).$$

Recall that the Clarke pricing scheme in the *VCG* auction would charge each player the amount of the externality he introduces. This amount in the case of multi-unit auctions is the sum of values (bids) of the k_i losers, where k_i is the number of units bidder i wins. Formally, let b_{-i} be the sorted vector of bids of all players other than i. The payment of bidder i is then

$$p_i = \sum_{k=1}^{k_i} b_{-i}^{K-k_i+k}.$$

Two other sealed-bid multi-unit auction mechanisms are commonly considered: a discriminatory auction, in which each player pays the amount of his bids, and a uniform-price auction, in which each player pays the $(K + 1)$st highest bid (that is, the highest losing bid). All these allocate the units to the K highest bids. The uniform-price auction is known to exhibit the phenomenon of *demand reduction*, that is, bidders have an incentive to submit bids below their valuations. However, the bid for the first item is truthful, and the incentive for demand reduction increases with the number of items demanded. As a result, the equilibrium bids in this auction are not symmetric and, hence, the auction is inefficient. Similarly, discriminatory auctions yield in general asymmetric bids and are, thus, inefficient, although Ausubel and Cramton [2002] provide an example of conditions under which the discriminatory auction results in an efficient allocation. It is worth noting, however, that inefficiencies do not arise as a result of the availability of multiple units for sale, but from multi-unit demand: when bidders each demand just one unit, the efficiency results are the same as in the single-unit domain.

In terms of revenue, little can be said definitively in the case of multi-unit auctions: the basic results are that revenue ranking is ambiguous between the three auction formats [Krishna, 2002].

3.4.3 Combinatorial Auctions

Combinatorial auctions are a generalization of multi-unit auctions to the case when non-identical goods are for sale. Suppose that there is a set G of goods for sale and for simplicity (and without really any loss in generality) let's assume that each good $g \in G$ is indivisible. Each buyer i has a valuation $v_i(X)$ for every subset $X \subseteq G$ of goods. This value function is assumed to have a basic monotonicity property that for every $X \subseteq Y$, $v_i(X) \leq v_i(Y)$. I also assume that we are dealing with a private values setting. The allocation rule $o(b)$ now assigns a subset of items in G to all buyers based on the reported bid functions $b_i(X)$. Specifically, let $o_i(b)$ be the subset of goods allocated to player i. Naturally, in this setting the allocation has to have the property that no good is assigned to two different buyers, that is

$$o_i(b) \cap o_j(b) = \emptyset \ \forall\, i \neq j. \tag{3.1}$$

The literature has nearly nothing to say about revenue optimal combinatorial auctions, with Ledyard [2007] being one notable exception, solving the problem when bidders are single-minded (that is, each bidder only values one subset of goods). As for efficiency, the key result in sealed-bid combinatorial auctions is a basic application of the *VCG* mechanism. Let

$$W(\cdot, I) = \sum_{i \in I} v_i(o_i(v))$$

be the welfare objective function for a set of players I and define

$$o^* = \arg\max_o W(\cdot, I),$$

and

$$o^*_{-i} = \arg\max_o W(\cdot, I \setminus i).$$

Both o^* and o^*_{-i} are the solutions to these optimization programs subject to the constraint in Equation 3.1. The *VCG* scheme would select $o^*(v)$ which maximizes $W(\cdot, I)$ and the following (Clarke) payment scheme:

$$p_i(v) = \sum_{j \neq i} \left[v_j(o^*_{-i}(v_{-i})) - v_j(o^*(v)) \right].$$

As I showed above, *VCG* is strategyproof and, hence, efficient.

There has recently been a considerable proliferation of results, both theoretical and practical, in the area of combinatorial auction design and implementation. Much of the work focuses on *open* auctions, that is, auctions in which bids are submitted through a series of rounds rather than in a one-shot fashion. Cramton *et al.* [2006] is a recent text which provides a great overview of the literature.

3.5 Computational Approaches to Mechanism Design

3.5.1 Automated Mechanism Design (AMD)

The Problem

Perhaps most akin to my motivation and approach is the idea of *automated mechanism design* developed in a series of papers by Conitzer and Sandholm ([Conitzer and Sandholm, 2002, 2003, 2004a,b; Sandholm *et al.*, 2007]). While most theoretical treatment of mechanism design considers fairly standard designer objectives, such as welfare or revenue, Conitzer and Sandholm propose constructing a general optimization program that can take as input an arbitrary objective defined on a finite outcome space.

The basic problem setup for automated mechanism design on discrete domains consists of the following elements (assuming throughout for notational convenience that the players are ex-ante symmetric, that is, the type distributions and utility functions are symmetric; note that the extension to the general case is direct, albeit with more notational baggage):

- Outcome space, $O = \{o_1, \ldots, o_K\}$
- Type space, $T = \{t_1, \ldots, t_L\}$
- Probability distribution over player types, $P : T \to [0, 1]$ with $\sum_{t \in T} P(t) = 1$ (assume that types are independently distributed)
- The designer's objective function, $W : T^m \times O \to \mathbb{R}$
- Symmetric utility functions of players, $u : T \times O \to \mathbb{R}$

Let m be the number of players, $t \in T^k$ for $k \leq m$ (that is, the type profile of a subset of players), and let I_t be the subset of players playing t. I overload the function $P(\cdot)$ by defining $P(t) = \prod_{i \in I_t} P(t_i)$.

A *mechanism* in this setting is a function which maps joint reports of player types to outcomes, that is, $M : T^m \to O$. We can also consider a *randomized mechanism*, which simply maps type reports to probability distributions over outcomes, $M : T^m \times O \to [0,1]$ with $\sum_{o \in O} M(t,o) = 1$ for every $t \in T^m$.

By focusing on the space of player strategies consisting exclusively of their types, the above setup takes advantage of the revelation principle. To invoke it effectively, however, requires the corresponding constraint that players are actually incentivized to report their *true* types. In the AMD setup, two possibilities can be considered: Bayes Nash incentive compatibility (BNIC) and dominant strategy incentive compatibility (DSIC). Under BNIC, each player maximizes his expected utility with respect to the joint type distribution of others by reporting his true type. Under DSIC, truthful reporting is a dominant strategy equilibrium. The definition below captures both of these notions formally in the notation of this section.

Definition 3.36 *A mechanism M is* Bayes-Nash incentive compatible *if for every $i \in I$ and $t_i \in T$*

$$E_{t_{-i}}\left[\sum_{o \in O} u_i(t_i,o) M(t_i,t_{-i},o)\right] \geq E_{t_{-i}}\left[\sum_{o \in O} u_i(t_i,o) M(t_i',t_{-i},o)\right] \ \forall\, t_i' \in T.$$

A mechanism M is dominant strategy incentive compatible *if for every $i \in I$ and $t_i \in T$*

$$\sum_{o \in O} u_i(t_i,o) M(t_i,t_{-i},o) \geq \sum_{o \in O} u_i(t_i,o) M(t_i',t_{-i},o) \ \forall\, t_i' \in T, t_{-i} \in T^{m-1}.$$

Another standard constraint in classical mechanism design is *individual rationality* (IR), which captures the intuition that no player can be expected to participate in a mechanism that yields him a net loss relative to his opportunity cost (see Section 3.2.4 above). Just like incentive compatibility, individual rationality comes in two varieties: *ex interim individual rationality* (EIIR), in which players take expected utility with respect to the distribution of other players' types, and *ex post individual rationality* (EPIR), which provides guarantees no matter what the types of other players are realized.

Definition 3.37 *A mechanism M is* ex interim individually rational *if for every $i \in I$ and*

$t_i \in T$

$$E_{t_{-i}}\left[\sum_{o \in O} u_i(t_i, o) M(t_i, t_{-i}, o)\right] \geq 0.$$

A mechanism M is ex post individually rational *if for every $i \in I$, $t_i \in T$, and $t_{-i} \in T^{m-1}$*

$$\sum_{o \in O} u_i(t_i, o) M(t_i, t_{-i}, o) \geq 0.$$

Given the two types of IC and IR constraints, two mechanism design optimization programs can be defined. The first, which would utilize the BNIC and EIIR constraints, I call *Bayesian mechanism design* (BMD), while the second (stronger) program, which would utilize DSIC and EPIR constraints, I call *dominant strategy mechanism design* (DSMD).

Let me first define a program for randomized BMD:

$$\begin{aligned}
&\max_{M \in \mathbb{R}^{L^m \times K}} \sum_{t \in T^m} \sum_{o \in O} W(t, o) M(t, o) P(t) \\
&\text{s.t.} \\
&\text{IR}: \quad \sum_{t_{-i}} \sum_{o \in O} u(t_i, o) M(t_i, t_{-i}, o) P(t_{-i}) \geq 0, \ \forall i \in I, t_i \in T, \\
&\text{IC}: \quad \sum_{t_{-i}} \sum_{o \in O} u(t_i, o)[M(t_i, t_{-i}, o) - M(t'_i, t_{-i}, o)] P(t_{-i}) \geq 0 \\
&\qquad\qquad \forall i \in I, t_i \in T, t'_i \in T, \\
&\sum_{o \in O} M(t, o) = 1 \quad \forall t; M(t, o) \geq 0 \, \forall t \in T^m, o \in O,
\end{aligned}$$

where the last constraints ensure that the resulting mechanism is a valid distribution over the outcome space for every reported vector of types. The program for randomized DSMD can be similarly defined:

$$\max_{M \in \mathbb{R}^{L^m \times K}} \sum_{t \in T^m} \sum_{o \in O} W(t,o) M(t,o) P(t)$$

$$\text{s.t.}$$

$$\text{IR}: \quad \sum_{o \in O} u(t_i, o) M(t_i, t_{-i}, o) \geq 0 \quad \forall i \in I, t_i \in T, t_{-i} \in T^{m-1}$$

$$\text{IC}: \quad \sum_{o \in O} u(t_i, o)[M(t_i, t_{-i}, o) - M(t'_i, t_{-i}, o)] \geq 0$$

$$\forall i \in I, t_i, t'_i \in T, t_{-i} \in T^{m-1}$$

$$\sum_{o \in O} M(t,o) = 1 \quad \forall t \in T^m, M(t,o) \geq \forall t \in T^m, o \in O.$$

We can see by inspection that both of these are linear programs with a number of constraints which is polynomial in the problem size if the number of players is constant. Thus, they can be solved in polynomial time. In principle, the case of deterministic mechanism design is not very different: the program is as above, with the exception that the mechanism design variables are restricted to be binary. This restriction, however, is crucial, since it now yields an integer program. Indeed, the problem can be shown to be NP-Hard.

We can note (following Conitzer and Sandholm [2004b]) that allowing for payments does not actually affect the structure of the BMD and DSMD programs, nor their complexity. The reason is that, while the true outcome space becomes infinite, if we impose structure on the problem such that both the designer objective and player utilities are linear in payments, the objectives and the affected constraints all remain linear.

Several early small applications of automated mechanism design are described in Conitzer and Sandholm [2003], with demonstrated success in all of these. Furthermore, a scalability study shows that the LP becomes very large and run time increases dramatically with even relatively few types per player and when the number of players grows above seven.

Algorithms for the Single-Agent Setting

All the theoretical complexity of deterministic mechanism design outlined above still obtains when there is but a single agent. Conversely, perhaps development of an effective algorithm for the single-agent setting would also suggest effective approaches when there is an arbitrary number of agents. To this end, Conitzer and Sandholm [2004a] develop two algorithms for search in the space of outcome subsets, one based on branch-and-bound and the other based on *IDA**. They showed that these tend to outperform the CPLEX implementation of the integer programs above.

3.5.2 Partial Revelation of Preferences

One of the greatest drawbacks of classical mechanism design and automated mechanism design as described thus far is the complexity of the space of joint agent types. In most realistic settings, one cannot expect an agent to determine, let alone reveal, his preferences for every possible outcome. For example, in a combinatorial auction, it seems unlikely that bidders will attempt to determine their value for every package that they can possibly win if the items number hundreds or thousands. A common approach to tackle the type space complexity issue is through the use of multi-stage mechanisms [Sandholm *et al.*, 2007; Parkes, 2006]. A possible alternative suggested by Hyafil and Boutilier [2007a] is to partition the type space and restrict the space of player strategies to elements of this partition. In this setting, the authors show how to achieve approximate dominant strategy implementation. They develop a series of algorithms for automated mechanism design under this approximate solution concept [Hyafil and Boutilier, 2007b].

Formally, a *direct partial revelation mechanism* is defined as follows:

Definition 3.38 (Partial Revelation Mechanisms [Hyafil and Boutilier, 2007a]) *A* direct partial revelation mechanism (PRM) *is a mechanism in which the action set A_i of players is a set of partial types Θ_i such that $\theta_i \in \Theta_i$ is a set of types, that is, $\theta_i \subseteq T_i$.*

These mechanisms have a corresponding definition of (approximate) incentive compatibility.

Definition 3.39 *A* PRM *is* incentive compatible (IC) *if it induces equilibrium strategies r^* such that $t_i \in r_i^*(t_i)$.*

The notion of individual rationality can be applied to the partial revelation framework directly.

It is easy to note that ex-post efficiency is, in general, impossible to achieve when only partial types are revealed: we may simply construct settings in which different socially optimal choices are required for different joint type profiles in the same partial type profiles. As a result, Hyafil and Boutilier [2007b] demonstrate that reasonable social welfare properties cannot be achieved in the partial revelation setting, neither under dominant strategy, nor under Bayes-Nash implementation. On the positive side, they introduce the notion of regret-based PRMs, which select an allocation minimizing the maximum efficiency loss with respect to other possible allocations and types in the reported partial type vector. Upon extending the *VCG* (Clarke) payments to the partial revelation framework, the authors demonstrate that a uniform bound on optimal regret translates into a nearly efficient mechanism, implemented in nearly dominant strategies and ensuring near-ex-post individual rationality. The following is the formal statement of this result.

Theorem 3.40 *Let the regret of the optimal regret-based partial revelation mechanism with partial Clarke payments be uniformly (i.e., for all $\theta \in \Theta$) bounded by ϵ. Then the mechanism is ϵ-efficient, ϵ-dominant incentive compatible, and ex-post ϵ-individually rational.*

Hyafil and Boutilier [2007b] apply the PRM framework to automated mechanism design, placing an arbitrary design objective $W(\cdot)$ in place of efficiency, but retaining the idea of regret-based design. They provide a series of algorithms, both for heuristically partitioning the space of player types and for automatically designing mechanisms in the partial revelation setting.

3.5.3 Incremental Mechanism Design

Yet another approach to automated mechanism design was introduced by Conitzer and Sandholm [2007]. The idea was to begin with a manipulable mechanism and incrementally improve it to produce (at least asymptotically) a strategyproof mechanism. The authors demonstrate the efficacy of the general approach and its instantiation in particular settings, and provide results about worst-case complexity of finding a utility-improving manipulation. Another, somewhat complementary, work proposes an algorithm for passively verifying that a mechanism is strategyproof based on its input-output behavior [Kang and Parkes, 2006].

3.5.4 Evolutionary Approaches

An alternative approach to automatically generating mechanisms, particularly as applied to an extremely complex setting of continuous double auctions, is to co-evolve the mechanism together with player strategies [Cliff, 2002a,b; Phelps *et al.*, 2002]. These approaches use some notion of social utility and agent payoffs (not necessarily in the classical sense) as fitness criteria. An alternative to co-evolution, explored in Phelps *et al.* [2003], was to optimize a well-defined welfare function of the designer using genetic programming. In this work, the authors used a common learning strategy for all agents and defined an outcome of a game induced by a mechanism parameter as the outcome of joint agent learning.

3.5.5 The Metalearning Approach

Recently, Pardoe *et al.* [2006] introduced a metalearning approach for adaptively designing auction mechanisms based on empirical bidder behavior. The mechanisms designed thereby are actually learning rules that adapt to observed bidder behavior, and the metalearning technique is used to determine the learning parameters.

This approach is in many ways fundamentally different from all the other mechanism design approaches discussed in this section in that it entirely ignores the issue of bidder incentives. One possible caveat is that bidders, insofar as they may bid in multiple encounters with the seller, may well attempt to jeopardize the learning process, or at least "teach" it to their own advantage. Perhaps a more realistic issue is that as long as bidders are strategic, the learning environment is non-stationary, and convergence need not ever be attained (indeed, the learning rules may produce poor mechanisms on average in such an environment, and metalearning may be of very limited value).

CHAPTER 4

Simulation-Based Mechanism Design

IN WHICH *I introduce the notion of simulation-based mechanism design and provide a high-level description of a number of methods for tackling problems of this nature. I motivate simulation-based mechanism design with a problem that arose in the context of a supply-chain simulation and demonstrate throughout how my techniques can be used to address this problem.*[1]

Simulation models have long been used in Operations Research where analytical approaches fall short [Law and Kelton, 2000]. Naturally, many simulation analysis tools have been developed that allow estimation of relevant system variables. Additionally, the field of *simulation optimization* introduced scores of techniques for estimating optimal settings of control parameters [Olafsson and Kim, 2002]. Insofar as the problem of optimization using simulations is difficult in itself, one may imagine how much more so it becomes when we pose questions about solutions to games (i.e., models of strategic interactions) that are represented as simulations.

My approach in the next several chapters is to present a series of methods for incorporating game simulation analysis into a systematic mechanism design framework. As such, I will typically take as given that simulation-based games can be "solved" (in the sense that predictions of outcomes of strategic interactions can be effectively made) and focus for the moment (almost) solely on the mechanism design aspects of the problem.

[1]Some of the material in this chapter is taken from Vorobeychik *et al.* [2006].

The problem of forming the predictions of play (that is, estimating or approximating solutions to games) based on simulations is addressed in Part II.

The remainder of this chapter provides an overview of the general ideas, some of which are treated in detail in later chapters, while others are relegated to future work. While some of the discussion that follows is relatively abstract, I will demonstrate the use of several methods described in the context of simulation-based mechanism design for a supply-chain scenario of the Trading Agent Competition (TAC/SCM) [Arunachalam and Sadeh, 2005]. I, thus, begin this chapter by describing briefly the application of interest, followed by the account of the strategic issue which is the focus of the mechanism design problem I address. Thereafter, I interleave the general methodological discussion with the specific application to the TAC/SCM setting.

4.1 Trading Agent Competition and the Supply-Chain Game

The Trading Agent Competition (TAC) is an annual venue that provides a stochastic market simulation environment in which autonomous trading agents can compete using an API provided by the designers. The first competition, held in the summer of 2000, featured a travel-shopping scenario, in which the trading agents were to act as travel agents for simulated customers [Wellman *et al.*, 2001]. Several years later, the supply-chain game was introduced. In this scenario, the trading agents act as PC manufacturers, procuring components from the simulated suppliers, while bidding on and filling the orders from simulated customers. A somewhat more detailed account of the Trading Agent Competition and the Supply-Chain game can be found in the appendix. In the meantime, I turn my attention to the strategic issues encountered by the designers of TAC/SCM and how they relate to the general simulation-based mechanism design problem.

4.1.1 The Story of Day-0 Procurement in TAC/SCM

TAC/SCM was initially introduced in 2003. Early in the seeding rounds, a particular strategic element of the game began quickly to attract considerable attention: all of the high-scoring agents were submitting very large orders for all or nearly all of the input components *on the first day of the simulation*. Once this strategy became widely recognized as yielding a considerable advantage, progressively more agents adopted it. In retrospect, this behavior, termed *aggressive day-0 procurement*, can be understood from studying the supplier pricing model: it turns out that the prices of all inputs are lowest and their availability highest on simulation day 0.[2]

The matters become somewhat subtle, however, if we consider also the PC demand process. In this first year of the TAC/SCM competition, the customer demand for finished PCs had very high variability, with a bias towards extremes, and especially towards the low demand extreme. Consequently, large early procurement carried with it substantial risk: an agent that commits to large shipments of the inputs early would have to sell manufactured products well below cost if the customer demand is very low, particularly if it is so towards the end of the game. As a result, it may seem that the overall prevalence of games with low average demand should serve to counterbalance the supply-side incentives for large early procurement. However, that was not the case in the actual tournament, and, as I demonstrate below in a more systematic analysis of the problem, the incentives are still skewed towards aggressive day-0 procurement.

In the final rounds of TAC/SCM 2003, one of the agents, Deep Maize, introduced a strategy designed to *preempt* the other agents' day-0 procurement. By requesting an extremely large quantity of a particular component, Deep Maize would prevent the supplier from making reasonable offers to the other agents,[3] at least in response to their requests on that day. The premise was that it would be sufficient to preempt only day-0 procurement, since after day 0 prices are not so especially attractive. I discuss the preemptive strategy as used by Deep Maize as well as its effect on the incentives of other

[2] Details of the supplier pricing model are provided in the appendix.

[3] More precisely, the supplier would not be able to make reasonable offers to agents whose requests it considers after seeing the request by Deep Maize.

players in Section 7.5.

Although jockeying for day-0 procurement turned out to be an interesting strategic issue in itself [Wellman *et al.*, 2005], the phenomenon detracted from other important problems, such as adapting production levels to varying demand (since component costs were already sunk), and dynamic management of production, sales, and inventory. Several participants noted that the predominance of day-0 procurement overshadowed other key research issues, such as factory scheduling [Benisch *et al.*, 2004] and optimizing bids for customer orders [Pardoe and Stone, 2004]. After the 2003 tournament, there was a general consensus in the TAC community that the rules should be changed to deter large day-0 procurement.

In response to the problem, the TAC/SCM designers adopted several rule changes intended to penalize large day-0 orders. These included modifications to supplier pricing policies and introduction of storage costs assessed on inventories of components and finished goods. Despite the changes, day-0 procurement was very high in the early rounds of the 2004 competition. In a drastic measure, the Game Master imposed a fivefold increase of storage costs midway through the tournament. Even this did not stem the tide, and day-0 procurement in the final rounds actually *increased* (by some measures) from 2003 [Kiekintveld *et al.*, 2005].

4.1.2 The TAC/SCM Design Problem

The task facing game organizers can be viewed as a problem in *mechanism design*. The designers have certain game features under their control, and a set of objectives regarding game outcomes. Unlike most academic treatments of mechanism design, the objective is a behavioral feature (moderate day-0 procurement) rather than an allocation feature like economic efficiency, and the allowed mechanisms are restricted to those judged to require only an incremental modification of the current game. Replacing the supply-chain negotiation procedures with a one-shot direct mechanism, for example, was not an option. A central motivation of this chapter, and, indeed, of this entire work, is that such operational restrictions and idiosyncratic objectives are actually quite typical of practical mechanism design settings, where they are perhaps more commonly characterized as incentive engineering problems.

The apparent difficulty in identifying rule modifications that effect moderation in day-0 procurement is quite striking. Although the designs were widely discussed, predictions for the effects of various proposals were supported primarily by intuitive arguments or at best by back-of-the-envelope calculations. Much of the difficulty, of course, is anticipating the agents' (and their developers') responses without essentially running a gaming exercise for this purpose. The episode caused us to consider whether new approaches or tools could enable more systematic analysis of design options. Standard game-theoretic and mechanism design methods are clearly relevant, although the lack of an analytic description of the game seems to be an impediment. Under the assumption that the simulator itself is the only reliable source of outcome computation, I refer to the task as *simulation-based* mechanism design.

The analysis below focuses on the setting of storage costs (taking other game modifications as fixed), since this is the most direct deterrent to early procurement adopted. My results confirm the basic intuition that incentives for day-0 purchasing decrease as storage costs rise. I also confirm that the high day-0 procurement observed in the 2004 tournament is a rational response to the setting of storage costs used. Finally, I conclude from the data that it is very unlikely that any reasonable setting of storage costs would result in acceptable levels of day-0 procurement, so a different design approach would have been required to eliminate this problem.

I describe the simulation-based design analysis methods below, interleaving a detailed application to the TAC/SCM scenario throughout. Recall that during the 2004 tournament, the designers of the supply-chain game chose to dramatically increase storage costs as a measure aimed at curbing day-0 procurement, to little avail. As a part of my analysis of TAC/SCM, I systematically explore the relationship between storage costs and the aggregate quantity of components procured on day 0 in equilibrium. In doing so, I consider several questions raised during and after the tournament. First, does increasing storage costs actually reduce day-0 procurement? Second, was the excessive day-0 procurement that was observed during the 2004 tournament rational? And third, could increasing storage costs sufficiently have reduced day-0 procurement to an "acceptable" level, and if so, what should the setting of storage costs have been? It is this third question that defines the mechanism design aspect of my analysis.[4]

4.2 What is Simulation-Based Mechanism Design?

Until now, I provided a somewhat abstract overview of a general computational setup for solving mechanism design problems. Since the title of this chapter prominently features *simulation-based mechanism design*, a natural question arises: what exactly do I mean by this term?

Informally, *simulation-based mechanism design* is a computational mechanism design approach in which the design choices θ are represented by a finite-dimensional vector (that is, $\Theta \subset \mathbb{R}^n$) and the induced games are modeled using simulations. (I defer the formal definition of simulation-based games until Section 7.1.) What I envision, indeed, is a simulation-based model of a system which exposes a set of *design* parameters and sets of strategic parameters, all of which jointly determine outcomes (and, consequently, payoffs). This is diagrammatically shown in Figure 4.1.

Note that our simulation analysis setting is more conceptually challenging than the typical Operations Research simulation setting: while both classes of simulations map parameter choices to outcomes, in our case the designer has complete control only over a subset of all parameters—the *design* parameters (i.e., θ)—and has to predict what choices (and, consequently) outcomes will result from strategic decision-making by the players. Let me qualify the last statement: certainly the designer has control over the parameters at the simulation stage; the issue is that he has to relinquish control in the implementation stage, that is, when the design choices have to be publicly announced and implemented in the actual system. Indeed, a fundamental part of my exercise is to guide how the designer can glean information about the induced game, and, consequently, about the ultimate

[4]I do not address whether and how other measures (e.g., constraining procurement directly) could have achieved design objectives. My approach takes as given some set of design options, in this case defined by the storage cost parameter. In principle my methods could be applied to a different or larger design space, though with corresponding complexity growth.

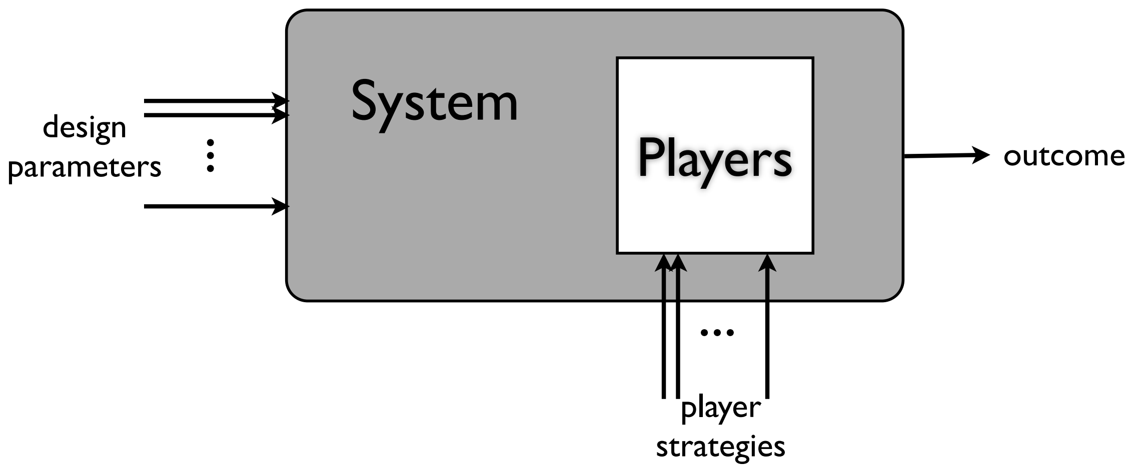


Figure 4.1: A high-level view of simulations that incorporate player choices as parameters not directly within the designer's control.

strategic choices of agents, based on simulation experiments. In the process, the designer will use simulations to gather empirical data as evidence about the game.

4.3 A General Framework for Simulation-Based Mechanism Design

Simulation-based mechanism design can be conceptualized as a two-pronged approach which involves, on one level, the outcome prediction problem and, on another level, the mechanism design (or mechanism search) problem. I begin by introducing some notation that will allow me to define the mechanism design problem precisely.

I model the strategic interactions between the designer of the mechanism and its participants as a two-stage game. The designer moves first by selecting a value θ from a set of allowable mechanism settings, Θ. All the participant agents observe the mechanism parameter θ and move simultaneously thereafter. For example, the designer could be deciding between a first-price and a second-price sealed-bid auction, with the presumption that after the choice has been made, the bidders will participate with full awareness of the auction rules.

Since the participants know the mechanism parameter, I define a game between them in the second stage as

$$\Gamma_\theta = [I, S, \{u_i(\cdot)\}],$$

that is, a one-shot game in normal form. It is well known that any game can be represented in normal form. For example, an extensive form game can be converted to "one-shot" by allowing the strategy sets to be functions of histories. Similarly, strategy sets in a Bayesian game are functions of players' private information (types).[5]

I refer to Γ_θ as the game *induced* by θ. Given an induced game, the designer would like to predict the outcomes of strategic interactions between the players in order to assess the incentive effects of his design choices. Traditionally in the game theory literature, this predictive role is played by a *solution concept*—typically, a set of Nash equilibria or approximate Nash equilibria. Indeed, most commonly mechanism design considers only a particular candidate Nash equilibrium, and frequently, one most favorable to the designer (known as *weak implementation*), although another stream has also considered designing mechanisms which are robust with respect to an entire set of equilibria (termed *strong implementation*) [Osborne and Rubinstein, 1994]. In Chapter 11, I consider a notion of *belief distributions of play* which a designer would form in predicting game-theoretic outcomes. In the meantime, however, I focus on sets of solutions in a more traditional sense and build my methods around these. Fortunately, all the methods I discuss naturally generalize to accommodate belief distributions.

The notation I use for a set of solutions to game Γ_θ is $\mathcal{N}(\theta)$, having in mind, perhaps, a set of approximate Nash equilibria. Given the set of solutions $\mathcal{N}(\theta)$, the designer's goal is to maximize a welfare function $W(\mathcal{N}(\theta), \theta)$. I assume for the moment that this function is specified by the designer.

Observe that if the designer *knew* $\mathcal{N}(\theta)$ as a function of θ, he would simply be faced with an optimization problem. This insight is actually a consequence of the application of backwards induction, which would have us find $\mathcal{N}(\theta)$ first for every θ and then compute an optimal mechanism with respect to these equilibria. If the design space were small, backwards induction applied to my two-stage mechanism design model would thus yield

[5]There is a caveat that because we lose the structure of strategies in the conversion, some solution concepts, such as subgame perfection, are also lost as a result.

an algorithm for optimal mechanism design. Indeed, if additionally the games Γ_θ featured small sets of players, strategies, and types, and deterministically specified payoff functions, I would say little more about the subject. My goal, however, is to develop mechanism design tools for settings in which it is infeasible to obtain solutions of Γ_θ for every $\theta \in \Theta$, either because the space of possible mechanisms is large, or because solving (or approximating solutions to) Γ_θ is computationally daunting. Additionally, I try to avoid making assumptions about the objective function or constraints on the design problem (or the agent type distributions, when dealing with Bayesian games).

4.4 Formalization of the TAC/SCM Design Problem

To apply simulation-based mechanism design methods to the TAC/SCM mechanism design problem, one must specify the agent strategy sets, the designer's welfare function, the mechanism parameter space, and the simulator. I restrict the agent strategies to multiples of the quantity of the day-0 requests by one of the finalists, Deep Maize, in the 2004 TAC/SCM tournament. Thus, if we let Q_{DM} denote the quantity of day-0 component requests by Deep Maize, a pure strategy a_i implies that the agent i orders $a_i Q_{DM}$ components on day 0. I further restrict the space of pure strategies to the set [0,1.5], since any strategy below 0 is illegal and strategies above 1.5 are extremely aggressive (thus unlikely to provide refuting deviations beyond those available from included strategies, and certainly not part of any desirable equilibrium). All other behavior is based on the behavior of Deep Maize and is identical for all agents. This choice can provide only an estimate of the actual tournament behavior of a "typical" agent. However, I believe that the general form of the results should be robust to changes in the full agent behavior, since I am focusing on arguably the most important strategic element of TAC/SCM 2003/2004.

I model the designer's welfare function as a threshold on the sum of day-0 purchases. Let $\phi(a) = \sum_{i=1}^{6} a_i$ be the aggregation function representing the sum of day-0 procurement of the six agents participating in a particular supply-chain game (for mixed strategy profiles s, I take expectation of ϕ with respect to the mixture). Henceforth, I use $\phi(a)$ and the phrase "total day-0 procurement" interchangeably. The designer's welfare function $W(\mathcal{N}(\theta), \theta)$ is then given by $\mathbf{I}\{\sup\{\phi^*(\theta)\} \leq \alpha\}$, where α is the maximum acceptable level of day-0 procurement, $\mathbf{I}$ is the indicator function, and ϕ^* is the set of (aggregated) equilibrium outcomes.[6] The designer selects a value θ of storage costs, expressed as an annual percentage of the baseline value of components in the inventory (charged daily), from the set $\Theta = \mathbb{R}^+$. Since the designer's decision depends only on $\phi^*(\theta)$, I present all of the results in terms of the value of the aggregation function $\phi(\cdot)$. For a given designer choice θ and strategic choices of players a, payoffs to all players are determined by the profits of the corresponding trading agents at the end of a single TAC/SCM game simulation. Since simulation runs are stochastic, we must use multiple simulation runs to estimate the expected player profits for any choices of θ and a. In the following section, I present a simulation-based game-theoretic analysis of the TAC/SCM domain and use it to enlighten the associated mechanism design problem. Thereafter, I attempt to generalize the methods applied to the TAC/SCM problem and present other techniques for general simulation-based mechanism design.

4.5 Simulation-Based Design Analysis

4.5.1 Estimating Nash Equilibria

The objective of TAC/SCM agents is to maximize profits realized over a game instance.[7] Let us fix a particular design choice θ. Then, if we fix a strategy for each agent at the beginning of the simulation and record the corresponding profits at the end, we will

[6]In my application to TAC/SCM, I focus exclusively on approximate Nash equilibria, although the techniques can be generalized to any solution concept.

[7]This actually describes an idealized TAC/SCM agent. In the actual tournament, one of the primary goals of an agent is to win, that is, to have the highest average payoff in the agent pool. While profit maximization is a substantial part of this goal, it does distort the incentives somewhat: an agent may now sacrifice some of its own profits in order to impose an even greater cost on others. There is considerable evidence from tournament experience that agents do, indeed, resort to such anti-social strategies. For example, in TAC/SCM '04, some agents used a "blocking" strategy, buying out the supply of particular components for the entire game span, so as to block future production by other players. Such a blocking strategy is clearly hurtful to the agent, since it will now acquire components that it will not need, but it hurts other agents much more in the process. While such complications are important, I ignore them for the analysis that follows. As I argue below, the presence of this strictly competitive aspect actually strengthens my conclusions.

have obtained a data point in the form $(\theta, a, U(\theta, a))$. I refer to the data set of such points as D. This data set, then, contains data only in the form of pure strategies of players and their corresponding payoffs, and, consequently, in order to formulate the designer's problem as optimization, we must first determine or approximate the set of Nash equilibria of each game Γ_θ based on the subset of D for a fixed value of θ, D_θ.

In the context of TAC/SCM, the sets of player actions a are infinite and, thus, we need methods for approximating Nash equilibria for infinite games (that is, games with infinite sets of actions). The first, discussed in considerable detail in Chapter 8, involves learning payoff functions of players and using the Nash equilibria of the resulting game as approximate Nash equilibria of the game of interest. The second attempts to approximate *sets* of Nash equilibria (or, rather, *ranges* of Nash equilibrium outcomes) based on sets of approximations with respect to the data set D_θ. The question I address first is how simulations can be guided to produce a data set of experience for a fixed choice of θ, which may then be used for Nash equilibrium estimation. A brief description of the two equilibrium estimation methods follows.

Search in Strategy Profile Space

Before we can even attempt to perform analysis of a strategic setting based on simulations, we must collect data which would enlighten such analysis. Presuming that we cannot take a sufficient number of samples (or even a single sample) for every possible strategy profile, the most natural way to collect data is by selecting a subset of all profiles uniformly randomly and performing simulations a fixed number of times for each of these. Since such a method ignores any structure that may be present in the game, it is natural to imagine that we can do better with a more directed search. One such directed search heuristic I now present. While I provide no formal or empirical justification for the method here, its efficacy is empirically confirmed in the work by Jordan *et al.* [2008]. I address the question of exploration in simulation-based games in Section 7.3.

The idea behind the search heuristic below is to explore profiles which may themselves be profitable deviations by some player from a Nash equilibrium approximation based on the data gathered thus far. To formalize it, I begin by defining the concept of a *strategic neighbor*.

Definition 4.1 *A* strategic neighbor *of a pure strategy profile a is a profile that is identical to a in all but one strategy. Define $S_{nb}(a, D)$ as the set of all strategic neighbors of a available in the data set D. Similarly, define $S_{nb}(a, \tilde{D})$ to be all strategic neighbors of a not in D. Finally, for any $a' \in S_{nb}(a, D)$ define the deviating agent as $i(a, a')$.*

Another concept fundamental to the heuristic is that of the quality of Nash equilibrium approximation of a strategy profile a with respect to the data set D.

Definition 4.2 *The regret bound (ϵ-bound), $\hat{\epsilon}$, of a pure strategy profile a is*

$$\hat{\epsilon}(a) = \max_{a' \in S_{nb}(a,D)} \max\{u_{i(a,a')}(a') - u_{i(a,a')}(a), 0\}.$$

The profile a is referred to as a candidate δ-equilibrium *for $\delta \geq \hat{\epsilon}$.*

The search method operates by exploring deviations from candidate equilibria. I refer to it as "min-regret-first search", as it selects with probability one a strategy profile $a' \in S_{nb}(a, \tilde{D})$ that has the smallest $\hat{\epsilon}$ in D. When $S_{nb}(a, \tilde{D}) = \emptyset$ (i.e., all strategic neighbors are represented in the data), a is *confirmed* as an $\hat{\epsilon}$-Nash equilibrium.

Since the TAC/SCM game with strategies that include all possible day-0 procurement choices is infinite, it is necessary to terminate min-regret-first search before exploring the entire space of strategy profiles. The termination time is determined in a somewhat ad hoc manner, based on observations about the current set of candidate equilibria.[8]

Payoff Function Approximation

Once a data set of profiles and corresponding player payoffs D_θ is generated, we can use it to estimate a sample Nash equilibrium or sets of Nash equilibria.

The first method for estimating Nash equilibria based on data in the TAC/SCM problem (which I use to estimate only a sample Nash) uses supervised learning to estimate payoff functions of mechanism participants from a data set of game experience, discussed

[8]Generally, search is terminated once the set of candidate equilibria is small enough to draw useful conclusions about the likely range of equilibrium outcomes in the game.

in some detail in Chapter 8. Once approximate or estimated payoff functions are available for all players in some convenient and compact form, the Nash equilibria may be either found analytically or approximated using numerical techniques, depending on the learning model. In what follows, I estimate only a sample Nash equilibrium using this technique, although this restriction can be removed at the expense of additional computation time.

I use the following methods for approximating payoff functions: quadratic regression (QR), locally weighted average (LWA), and locally weighted linear regression (LWLR). I also used control variates to reduce the variance of payoff estimates [Ross, 2001]. In the case of QR, I extended the quadratic model to linearly account for exogenous stochastic factors, average customer demand $\bar{Q}$, and average starting supplier capacity $\bar{S}$. For the other models, I estimated mean payoffs for each profile controlling for $\bar{Q}$ and $\bar{S}$ and learned using the data set consisting of strategy profiles and corresponding estimates of expected payoffs.

The quadratic regression model makes it possible to compute equilibria of the learned game analytically. For the other methods I applied replicator dynamics [Friedman, 1991] to a discrete approximation of the learned game. The expected total day-0 procurement in equilibrium was taken as the estimate of an equilibrium outcome.

Estimation of Nash Equilibrium Outcome Ranges

In the previous section I describe a method which I use to estimate a sample Nash equilibrium in a TAC/SCM game for a given fixed mechanism (storage cost) choice θ. I now describe an alternative estimator for this domain, which produces an estimate of a *set* of Nash equilibria. Of course, the method of learning payoff functions may be used for estimating sets of equilibria also. However, the technique I now describe allows us to take advantage of any locality structure on the equilibrium set induced by the aggregation function. I define an estimator for a set of Nash equilibria as follows.

Definition 4.3 *For a set K, define $Co(K)$ to be the convex hull of K. Let B_δ be the set of candidate δ-equilibria at level δ. I define $\hat{\phi}^*(\theta) = Co(\{\phi(a) : a \in B_\delta\})$ for a fixed δ*

to be an estimator of $\phi^*(\theta)$.

In words, the estimate of a set of equilibrium outcomes is the convex hull of all strategy profiles (or their aggregation values $\phi(a)$) with ϵ-bound below some fixed δ. This definition allows us to exploit structure arising from the aggregation function. If two profiles are close in terms of aggregation values, they may be likely to have similar ϵ-bounds. In particular, if one is an equilibrium, the other may be as well. Additionally, while sets of pure strategy Nash equilibria lie in $\mathbb{R}^m$, the aggregation function reduces its dimensionality substantially: in the TAC/SCM problem, I approximate a set of Nash equilibria by an interval. I present some theoretical support for this method of estimating the set of Nash equilibria in Section 7.6.

4.5.2 Data Generation

The data was collected by simulating TAC/SCM games on a local version of the 2004 TAC/SCM server, which has a configuration setting for the storage cost. Agent strategies in simulated games were selected from the set $\{0, 0.3, 0.6, \ldots, 1.5\}$ in order to have positive probability of generating strategic neighbors.[9] A baseline data set D_o was generated by sampling 10 randomly generated strategy profiles for each $\theta \in \{0, 50, 100, 150, 200\}$. Between 5 and 10 games were run for each profile after discarding games that had various flaws.[10] I used min-regret-first search to generate a simulated data set D_s, performing between 12 and 32 iterations for each of the above settings of θ.[11] Since the simulation cost is extremely high (a game takes nearly 1 hour to run), I was able to run a total of 2670 games over the span of more than six months. For comparison, to get the entire description of a game defined by the restricted finite joint strategy space for each value of $\theta \in \{0, 50, 100, 150, 200\}$ would have required at least 23100 games (sampling each profile 10 times).

[9] Of course, I do not restrict the Nash equilibrium estimates to stay in this discrete subset of [0,1.5].

[10] For example, if I detected that any agent failed during the game (failures included crashes, network connectivity problems, and other obvious anomalies), the game would be thrown out.

[11] An iteration entails between 5 and 10 games run for a particular strategic neighbor of a candidate profile.

4.5.3 Results

Analysis of the Baseline Data Set

I applied the three learning methods described above to the baseline data set D_o. Additionally, I generated an estimate of the Nash equilibrium correspondence, $\hat{\phi}^*(\theta)$, by applying Definition 4.3 with δ =2.5E6. The results are shown in Figure 4.2. As we can see, the correspondence $\hat{\phi}^*(\theta)$ has little predictive power based on D_o, and reveals no interesting structure about the game. In contrast, all three learning methods suggest that total day-0 procurement is a decreasing function of storage costs.

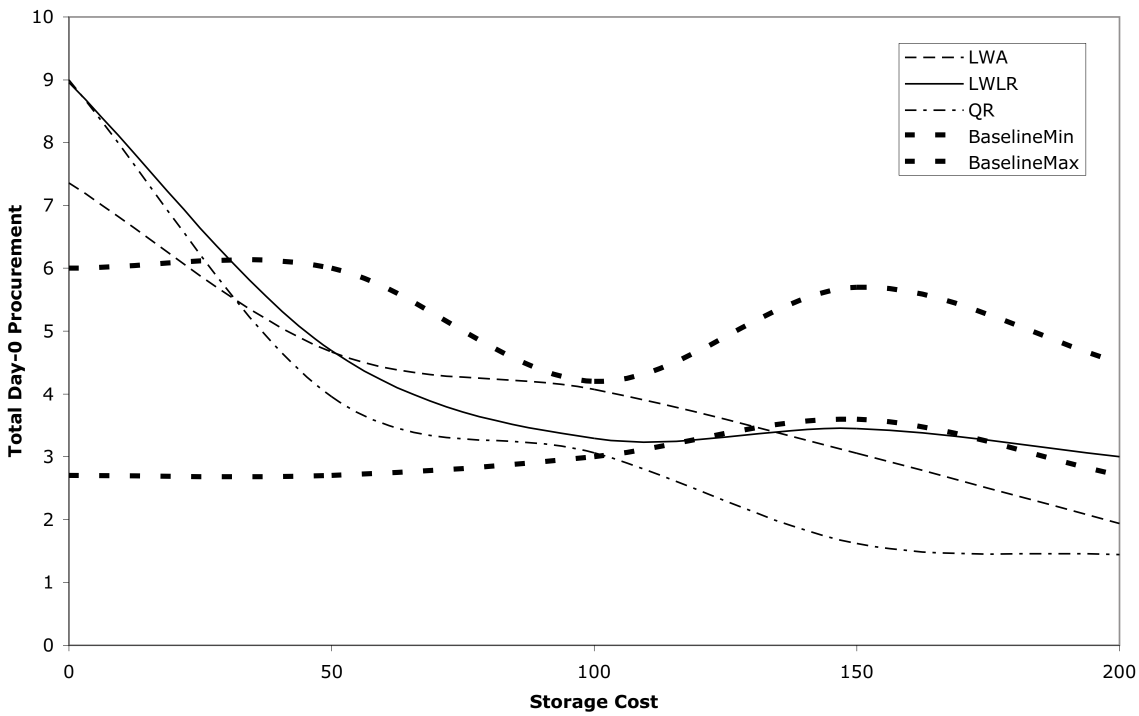


Figure 4.2: Aggregate day-0 procurement estimates based on D_o. The correspondence $\hat{\phi}^*(\theta)$ is the interval between "BaselineMin" and "BaselineMax".

Analysis of Search Data

To corroborate the initial evidence from the learning methods, I estimated $\hat{\phi}^*(\theta)$ (again, using δ =2.5E6) on the data set $D = \{D_o, D_s\}$, where D_s is data generated through the application of min-regret-first search. Figure 4.3 shows how the sets of Nash equilibrium estimates change as more iterations of min-regret-first search are performed when storage cost is 200. It is encouraging to see that the resulting correspondence appears to converge in that both upper and lower bounds become tighter as the number of iterations rises.

Figures 4.4, 4.5, 4.6, 4.7, and 4.8 show the estimates of $\hat{\phi}^*(\theta)$ for $\theta = 0, 50, 100, 150,$ and 200 respectively. The estimate of the Nash equilibrium correspondence based on Definition 4.3 applied for every value of θ in the above set is plotted together with the Nash equilibrium functions produced by applying the learning methods trained on D_o in Figure 4.9.[12] First, note that the addition of the search data narrows the range of potential equilibria substantially. Furthermore, the actual point predictions of the learning methods and those based on ϵ-bounds after search are reasonably close. Combining the evidence gathered from these two very different approaches to estimating the outcome correspondence yields a much more compelling picture of the relationship between storage costs and day-0 procurement than either method used in isolation.

This evidence supports the initial intuition that day-0 procurement should be decreasing with storage costs, since Nash equilibrium functions produced by learning, as well as both upper and lower bounds of the Nash equilibrium correspondence are decreasing in storage cost in Figure 4.9. It also confirms that high levels of day-0 procurement are a rational response to the 2004 tournament setting of average storage cost, which corresponds to $\theta = 100$. The *minimum* prediction for aggregate procurement at this level of storage costs given by any experimental methods is approximately 3. This is quite high, as it corresponds to an expected commitment of 1/3 of the total supplier capacity for the entire game. The maximum prediction is considerably higher at 4.5. In the actual 2004 competition, aggregate day-0 procurement was equivalent to 5.71 on the scale used here [Kiekintveld *et al.*, 2005]. My predictions underestimate this outcome to some degree, but show that any rational outcome was likely to have high day-0 procurement. There are several reasons why my estimates may be more conservative than the actual agent procurement behavior. One reason I already mentioned above: while I presume agents to care exclusively about profits, the actual game participants care primarily about

[12] It is unclear how meaningful the results of learning would be if D_s were added to the training data set. Indeed, the additional data may actually increase the learning variance.

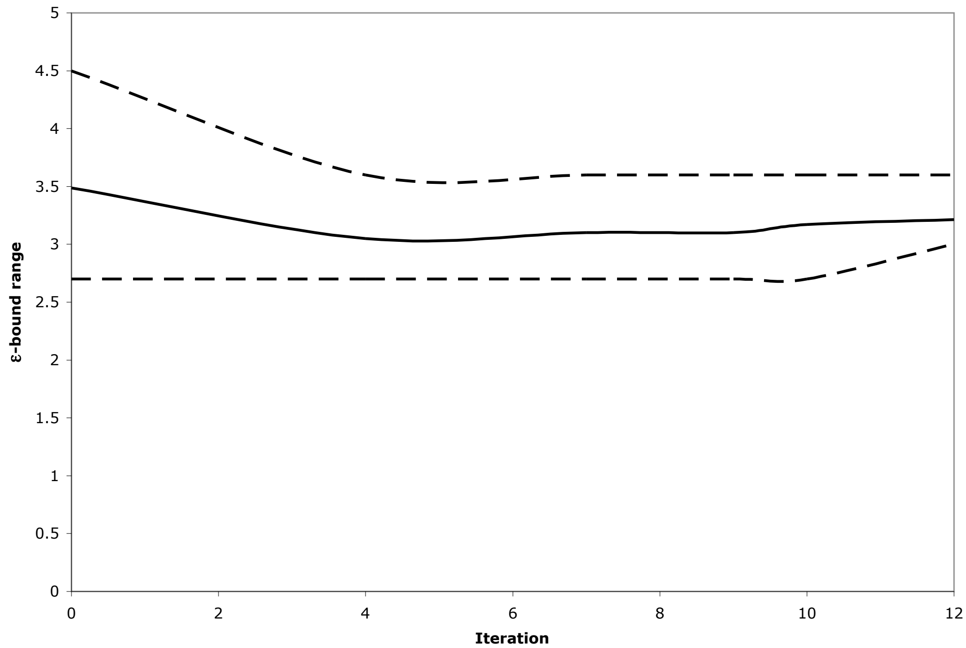


Figure 4.3: Estimates of the equilibrium ranges as the number of min-regret-first search iterations increases. Iteration is defined as a fixed number of samples taken from one selected unsampled neighbor of every current candidate Nash equilibrium. In the figure, ranges are denoted by the dashed lines, with the heavy line indicating the average.

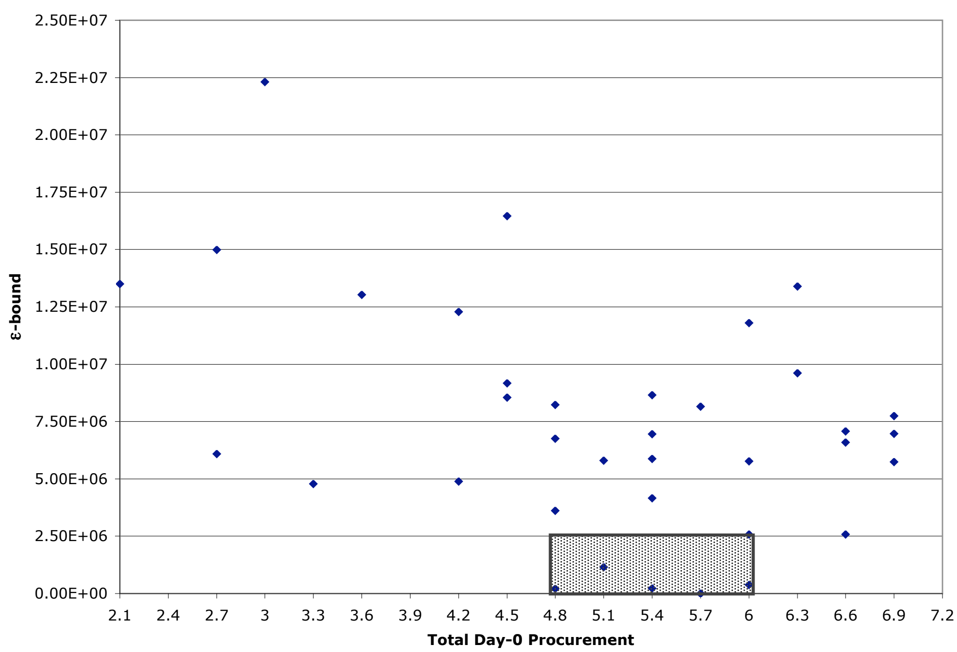


Figure 4.4: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 0$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(0)$ with δ =2.5M.

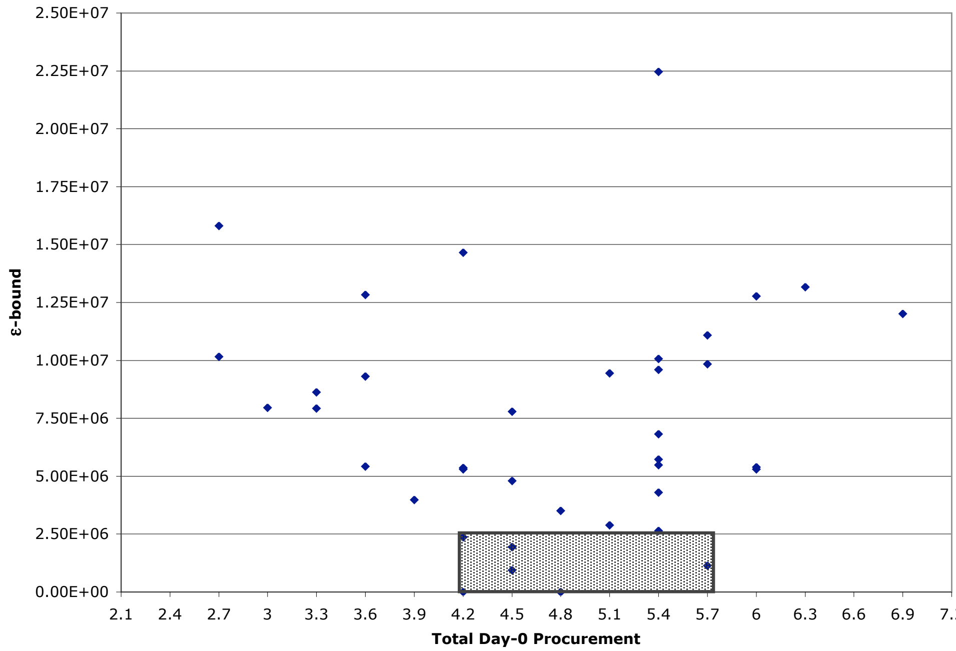


Figure 4.5: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 50$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(50)$ with δ =2.5M.

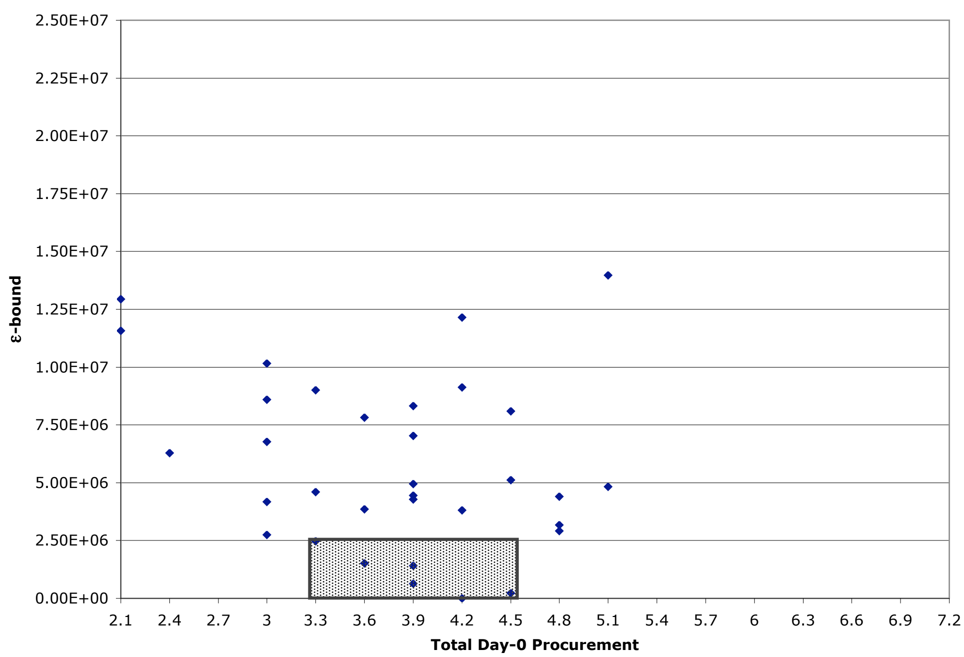


Figure 4.6: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 100$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(100)$ with δ =2.5M.

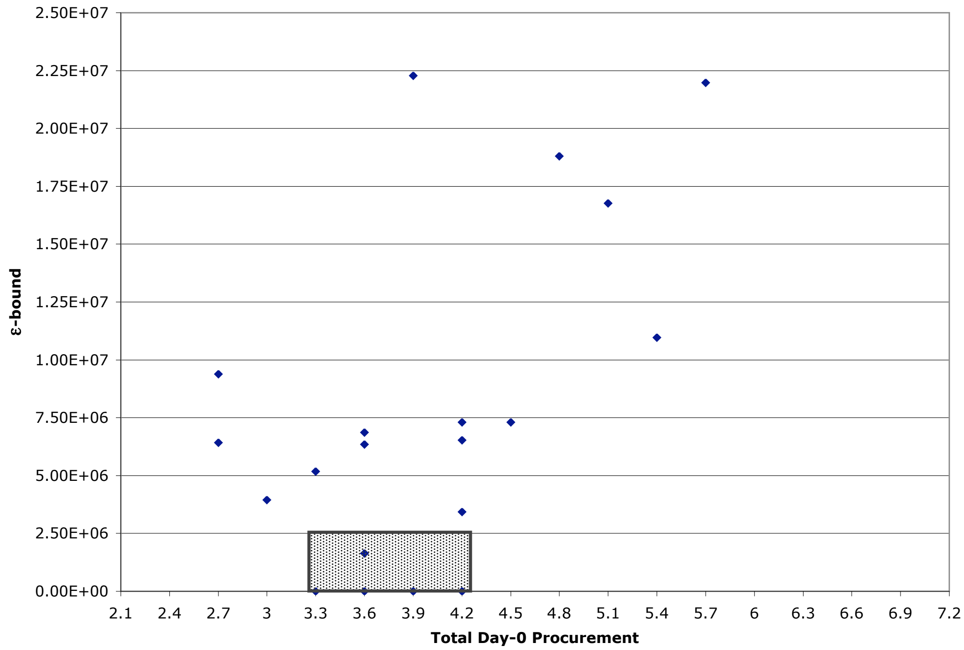


Figure 4.7: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 150$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(150)$ with δ =2.5M.

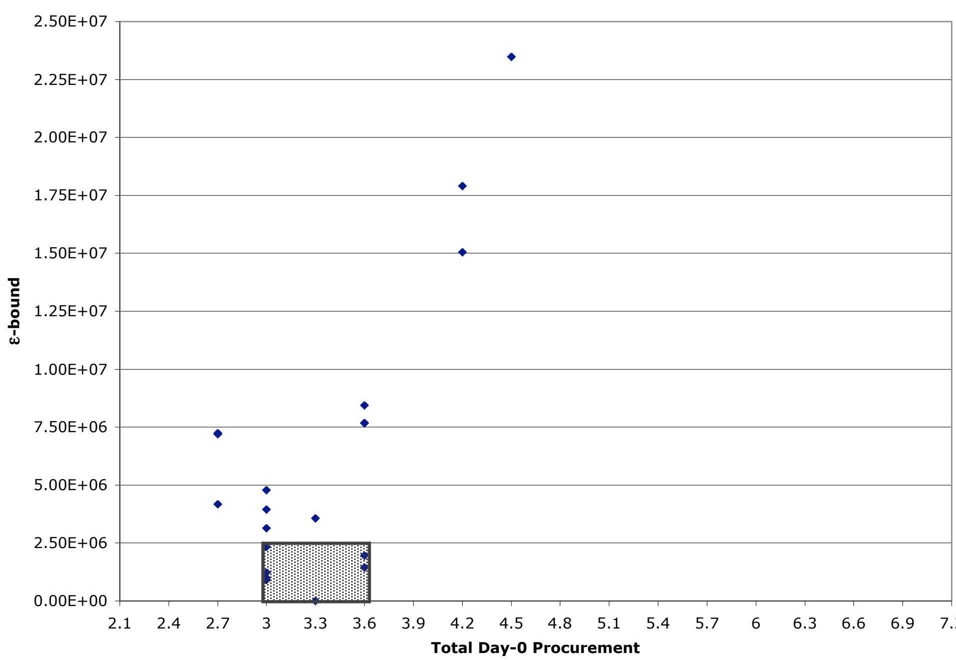


Figure 4.8: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 200$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(200)$ with δ =2.5M.

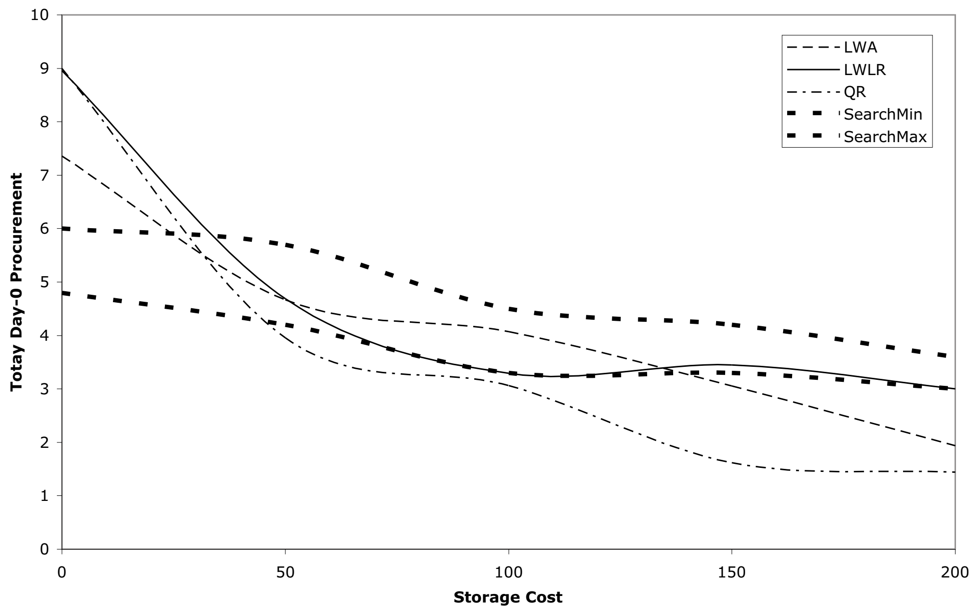


Figure 4.9: Aggregate day-0 procurement estimates based on search in strategy profile space compared to function approximation techniques trained on D_o. The correspondence $\hat{\phi}^*(\theta)$ for $D = \{D_o, D_s\}$ is the interval between "SearchMin" and "SearchMax".

winning the competition. The latter objective will at times force the agents to behave very aggressively in spite of the resulting loss in profits, as it thus imposes a heavy negative externality on other players. Another reason, somewhat less plausible to me, but nevertheless a possibility, is that the tournament results are simply due to the suboptimality of agent design: that is, it may be that the agent heuristic strategies (which are generally done based on intuition, as well as some game experience) lead them somewhat astray. In any case, the conclusions thus far, as well as those I make below, are actually strengthened if I presume that the results underestimate the actual aggregate day-0 procurement.

Extrapolating the Solution Correspondence

We have reasonably strong evidence that the outcome correspondence is decreasing. However, the ultimate goal is to be able to either set the storage cost parameter to a value that would curb day-0 procurement in equilibrium or conclude that this is not possible.

To answer this question directly, suppose that we set a conservative threshold $\alpha = 2$ on aggregate day-0 procurement.[13] Linear extrapolation of the maximum of the outcome

[13]Recall that designer's objective is to incentivize aggregate day-0 procurement that is below the thresh-

correspondence estimated from D yields $\theta = 320$.

The data for $\theta = 320$ were collected in the same way as for other storage cost settings, with 10 randomly generated profiles followed by 33 iterations of min-regret-first search. Figure 4.10 shows the detailed ϵ-bounds for all profiles in terms of their corresponding values of ϕ.

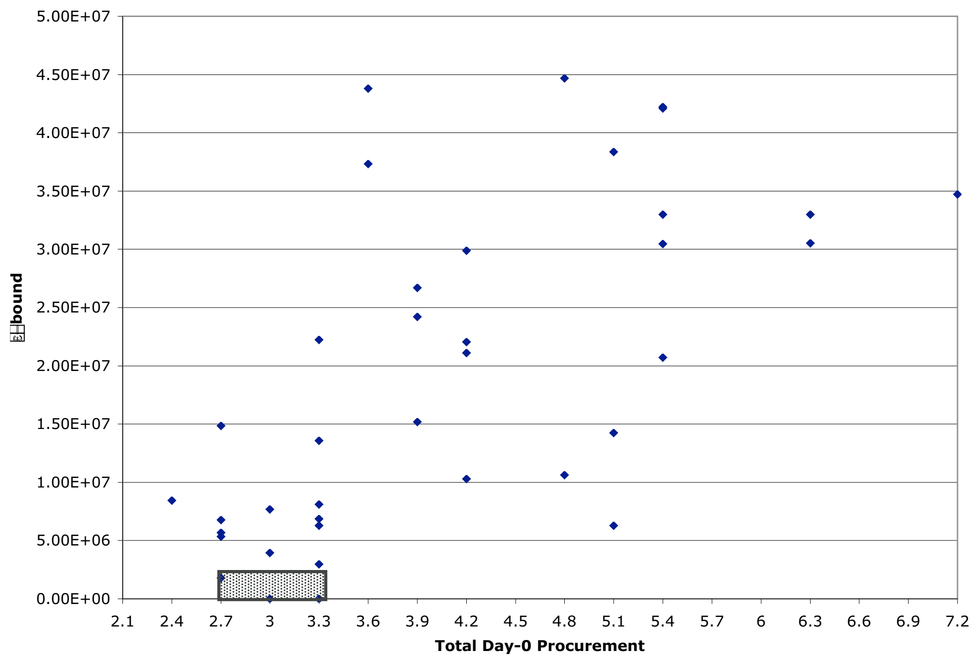


Figure 4.10: Values of $\hat{\epsilon}$ for profiles explored using search when $\theta = 320$. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$. The gray region corresponds to $\hat{\phi}^*(320)$ with δ =2.5M.

The estimated set of aggregate day-0 outcomes (the range of the x-axis of the box highlighted in gray in Figure 4.10) is very close to that for $\theta = 200$, indicating that there is little additional benefit to raising storage costs above 200. Observe, that even the lower bound of the estimated set of Nash equilibria is well above the target day-0 procurement of 2. Furthermore, payoffs to agents are almost always negative at $\theta = 320$. Consequently, increasing the costs further would be undesirable even if day-0 procurement could eventually be curbed. Since we are reasonably confident that $\phi^*(\theta)$ is decreasing in θ, we also do not expect that setting θ somewhere between 200 and 320 will achieve the desired result.

old α. The threshold here still represents a commitment of over 20% of the suppliers' capacity for the entire game on average, so in practice we would probably want the threshold to be even lower.

I conclude that it is unlikely that day-0 procurement could ever be reduced to a desirable level using any reasonable setting of the storage cost parameter. That the predictions in this work tend to underestimate tournament outcomes reinforces this conclusion. To achieve the desired reduction in day-0 procurement requires redesigning other aspects of the mechanism.

4.6 "Threshold" Design Problems

In the previous section I presented a series of simulation-based mechanism design techniques applied to the TAC/SCM setting. My current task is to abstract these methods away from the application in order to make them more generally applicable.

Recall that the TAC/SCM design problem features a particularly structured objective function. Since I was able to "compress" Nash equilibrium profiles to a one-dimensional summary and presumed that the designer was interested exclusively in driving this summary value (total day-0 procurement) below some constant α, I could define the objective function to be $W(\mathcal{N}(\theta), \theta) = \mathbf{I}\{\sup\{\phi^*(\theta)\} \leq \alpha\}$, with $\phi^*(\cdot)$ the summary or *aggregation* function. Generalizing, I define a class of *threshold* objectives to be of the form

$$W(\mathcal{N}(\theta), \theta) = \mathbf{I}\{f(\mathcal{N}(\theta)) \in B\},$$

where B is some designer-defined "target" or constraint. Since the designer is thus presumed to be indifferent between vast portions of the outcome space, the design goal is simply constraint satisfaction, that is, finding θ such that $f(\mathcal{N}(\theta)) \cap B \neq \emptyset$. A particular approach to the design problem would attempt to estimate a solution correspondence $\mathcal{N}(\theta)$, apply the transformation $f(\cdot)$, and finally determine some subset of the intersection of the resulting correspondence with B. An approach which relies on estimating the correspondence $\mathcal{N}(\theta)$ can be used more generally also, applying $W(\cdot)$ and finding (or approximating) the maximum of the result. In the case of threshold objectives, however, there is likely to be more structure in the solution correspondence than in the objective itself.

The particular threshold relevant to the TAC/SCM design problem, together with the use of an aggregation function, actually imposed even greater structure on the problem which made only two aspects of $f(\mathcal{N}(\theta))$ relevant: that the correspondence was decreasing in θ and that either its lower bound was above the threshold or its upper bound below it.

There are a number of ways we can approach the problem of estimating the correspondence $f(\mathcal{N}(\theta))$ when it has certain structure. If the game has a finite set of Nash equilibria, but infinite Θ, we may attempt to learn the correspondence. The problem is, however, quite challenging, because each $\theta \in \Theta$ in the data set may map to a set of points (since $f(\mathcal{N}(\theta))$ may often be a set rather than a point). Consequently regression methods cannot be directly applied.

If the game has continuous ranges of Nash equilibria for every θ, the problem is even more difficult, and I am not aware of any general approach in such a case that could be applied to approximating the correspondence where Θ is also continuous. In special cases, however, the problem can be amenable to relatively straightforward application of basic regression techniques. One such special case emerges in the TAC/SCM mechanism design problem, where the correspondence was a two-dimensional convex hull, which can be described by simply specifying the upper- and lower-bound functions. In this case, the upper bound and the lower bound can be learned from data, yielding an approximation to the solution correspondence. In the context of the TAC/SCM design problem, I did not pursue a formal learning approach, such as low-degree polynomial regression models, performing, rather, a relatively crude visual analysis. As an illustration, I present in Figure 4.11 the result of applying linear regression to lower and upper bounds of the correspondence. Clearly, the regression indicates that the lower and upper bounds are decreasing, although linearity is a good fit only in the range [0,200] of storage costs, since the correspondence becomes quite flat thereafter. In any case, regression does confirm one of the basic ingredients of my analysis: since a linear fit is quite good in this range, I do not expect large dips in the ranges were data is lacking.

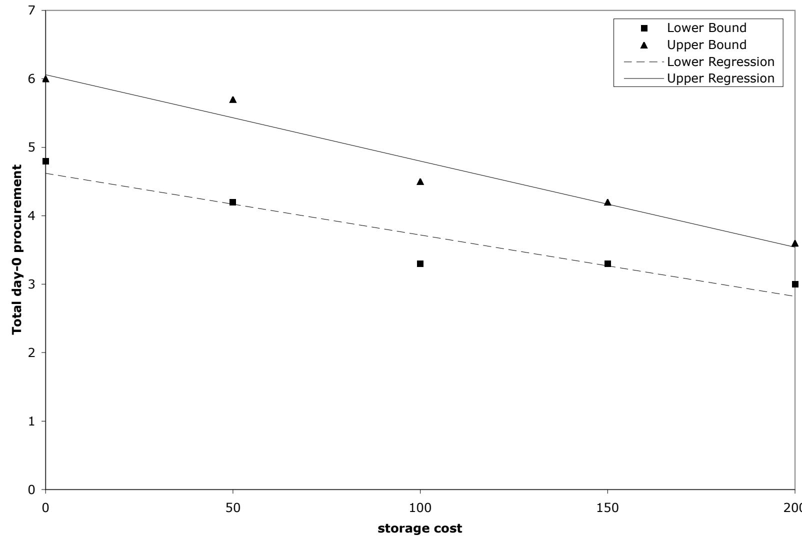


Figure 4.11: Aggregate day-0 procurement estimates based on linear regression of the lower and upper bounds of the Nash equilibrium correspondence.

4.7 General Objective Functions

While threshold objectives are an interesting special case (indeed, a relatively simple one, since we can look at these problems as those possessing a single constraint and no objective), my goal is to develop tools for analyzing problems with general objective functions and with general sets of constraints. Particularly, I want to allow the designer to specify an objective and constraints as black boxes, for example, via algorithmic descriptions.

4.7.1 Learning the Objective Function

Rather than attempting to learn a complex solution correspondence, it may often be more natural and effective to learn the objective function. To this end, we can actually apply any of a number of regression techniques directly. The caveat is that the objective function may be very complex, perhaps discontinuous as is the case with "threshold" objectives, and, consequently, not very easy to learn. The relative efficacy of learning the objective versus learning the correspondence would generally be problem dependent: sometimes we would be able to take advantage of the particular structure of the solution correspondence; other times, the objective may be more structured, or, at least, more amenable to learning approaches.

Actually, the objective in the TAC/SCM problem could very easily have been learned: for all the values of θ that were explored, the objective evaluated to 0. Hence, any simple learning approach would have yielded 0 as the value of the objective everywhere. While this would ultimately make the same prediction as was made via a set of more complex techniques, it is much less revealing: I was able to indicate also some of the reasons why the objective is unlikely to be attainable. Given such a small data set for learning, I needed to reveal as much information as possible to present compelling evidence for the ultimate conclusion. Consequently, there is another reason for meddling at the level of solution correspondence rather than the objective: when data is scarce, this gives us more insight about the problem and, perhaps, more support for the ultimate conclusions.

Returning now to the case of learning the objective functions, I have not specified precisely how the mechanism design would proceed thereafter. The natural next step is to find or approximate the maximum of the learned function. This problem may, of course, be in its own right complicated if the learned function could take arbitrary form. Indeed, an entire body of literature is dedicated to the subject of optimizing in a black-box setting [Olafsson and Kim, 2002], even when the setting is deterministic (as would be the case once it is learned). The question then arises: what do we win by learning the objective function? The answer, of course, is quite a bit. First, while black-box optimization is difficult even when function evaluations are deterministic, this setting is undoubtedly easier than its stochastic equivalent. Second, since we have a choice of regression models that we learn, we may well guide our choices to be models which can thereafter be relatively easily optimized. This idea I actually explore in a setting of learning payoff functions in games in order to estimate solutions in Chapter 8. Additionally, both of these ideas have been indirectly explored by a particular set of methods for black-box optimization, the sample-path methods [Gukan *et al.*, 1994; Chan, 1995], which suggest approximating the underlying function based on noisy evaluations, and finding the maximum based on this approximation.

The primary weakness of the idea of learning the payoff function lies in the relatively weak convergence guarantees [Gukan *et al.*, 1994]. Stronger guarantees can be made if we consider an iterative method for searching the domain of the stochastic function, which I now describe.

4.7.2 Stochastic Search

The approach to computational mechanism design I have thus far discussed takes the backwards induction algorithm quite literally: it attempts to compute solutions to the induced game for every $\theta \in \Theta$, evaluates each, and then proceed to find the θ which yielded the highest evaluation. Whenever evaluating the objective is not directly feasible for every θ—for example, in the case when Θ is infinite—the algorithm would attempt to approximate the objective function and use this approximation to obtain an optimal or approximately optimal design.

In this section, I note that backwards induction need not be taken so literally at all. What it in effect implies is that in order to evaluate the objective function (i.e., the designer's utility), we must obtain solutions to the induced game for the corresponding value of θ. As a result, we can use the following high-level procedure for finding or approximating optimal mechanisms:

1. Select a candidate mechanism, θ.
2. Find (approximate) solutions to Γ_θ.
3. Evaluate the objective and constraints given solutions to Γ_θ.
4. Repeat this procedure for a specified number of steps.
5. Return an approximately optimal design based on the resulting optimization path.

I visually represent this procedure by a diagram in Figure 4.12.

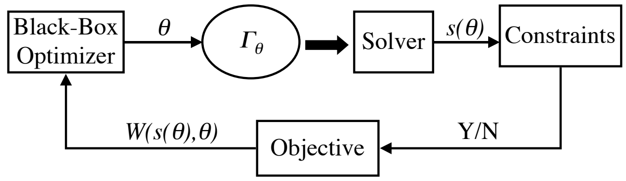


Figure 4.12: Computational mechanism design procedure based on black-box optimization.

4.7.3 Constraints

I have until this section largely ignored the issue of adding constraints to the designer's optimization problem. In the case of "threshold" objectives, the objective itself was essentially a constraint, and I presumed that the problem had no additional constraints—at least not in the first stage of the design.[14] In tackling the general problem, however, I certainly want to avoid making any assumptions about constraints, or lack thereof.

To begin, I want to make a distinction between two types of constraints in my setting: the *direct* constraints on the design choice, θ, implicit whenever I say that $\theta \in \Theta$; and the *indirect* constraints on the design choice in that they are conditional on the *outcomes* of the induced games, rather than on the value of θ per se. The reason this distinction is significant is that the former constraints can be checked prior to evaluating the objective—significantly, prior to solving the game. As such, we can use very simple methods to generate $\theta \in \Theta$, such as rejection sampling or, even simpler, evaluating the objective at this constraint to negative infinity. It seems very wasteful, however, to do either when we have to solve games in order to evaluate the constraint: since we already have to incur the considerable computational cost of solving games, it is especially important to capture as much information as possible from the process. One approach is to "move" the constraints into the objective. This can be done in an ad hoc way, for example, by introducing constant weights for the constraints.[15] More systematic methods also exist,

[14]Of course, in the application to TAC/SCM that I discussed in the preceding chapter, I introduced a "secondary" constraint that allowed me to argue for the infeasibility of achieving the ultimate objective in that setting. However, this additional constraint was introduced in a somewhat ad hoc manner, and whatever systematic analysis was done involved only the threshold objective.

[15]Additionally, one would have to decide how to measure the magnitude of constraint violation.

for example, penalty and barrier methods. In penalty methods, for example, the penalty weight is assigned to each constraint and increased over time. If one imagines a sequence of such unconstrained problems, convergence of their solutions to the solution of the original problem can be obtained [Nocedal and Wright, 2006]. Significantly, using these methods can introduce structure into the objective which would otherwise be thrown away, and may often considerably improve the optimization results. Additionally, certain classes of constraints impose additional structure on the problem. For example, individual rationality constraints can be "fixed" by the designer via fixed payments to all players in the amount of maximum violation, assuming that payments are allowed and utilities are quasilinear in payments. Consequently, if the designer is allowed to provide participation incentives to players, we can always eliminate the individual rationality constraints by mapping to a feasible (with respect to the individual rationality constraint) design choice directly.

A complementary approach to dealing with constraints in a way that exploits problem structure is via a constraint generation procedure, outlined as follows:

1. Start with an empty set of *active* constraints and a random initial θ
2. Find an (approximate) optimum θ^* to the problem using only the constraints in the *active* set
3. If no constraint has been violated, return θ^*
4. Otherwise, add the violated constraint to the set of *active* constraints and perform "threshold" optimization given this active set starting from θ^* and set the new initial θ to the feasible result
5. Repeat steps 2-4 until no constraint is violated

Clearly, since this approach involves constrained optimization as a part of the procedure, it would need to incorporate one of the above methods (e.g., a penalty method) and is, thus, complementary. However, by focusing the early search on a subset of constraints, it may effectively exploit continuity in the objective function and a subset of constraints to obtain good solutions early, and additionally exploit locality and synergy between the objective and constraints.

In much of Chapter 5, I do, nevertheless, pursue the naive approach and evaluate any failed constraint to negative infinity. Actual application of the penalty method and, possibly, of the constraint generation algorithm, I relegate to future work.

4.8 Solving Games Induced by Mechanism Choices

At the root of all of my methods for computational (or automated) mechanism design is a subroutine which computes or approximates solutions to games. The subject of computational game theory—that is, of computational techniques to solve games—has indeed received considerable attention, with GAMBIT [McKelvey *et al.*, 2005] being, perhaps, the best-known numerical toolbox for solving finite games. Above I described two techniques for estimating Nash equilibria in TAC/SCM. I devote the entire Part II to discuss the state-of-the art techniques, as well as my own contributions, particularly in the context of games specified as simulations.

4.9 Sensitivity Analysis

The remaining sections in this chapter are somewhat technical in nature and not very fundamental to the understanding of the rest of the thesis. As such, a reader interested only in obtaining a high-level grasp of my techniques may find it profitable to skip them.

Let me assume in this section that the design space is finite, that is, $\Theta = \{\theta_1, \dots, \theta_J\}$, and that we have estimates of the objective function for each of these, that is, we have $\hat{W}_1, \dots, \hat{W}_k$, along with the associated confidence intervals such that $|W_j - \hat{W}_j| \leq q_j$ with probability at least $1 - \alpha_j$. Furthermore, suppose that there are no ties between $\hat{W}_j$ and the confidence interval around the maximal objective value $\hat{W}_{j^*}$ has null intersection with the remaining intervals, as in Figure 4.13. Obtaining the probability that W_{j^*} is the true

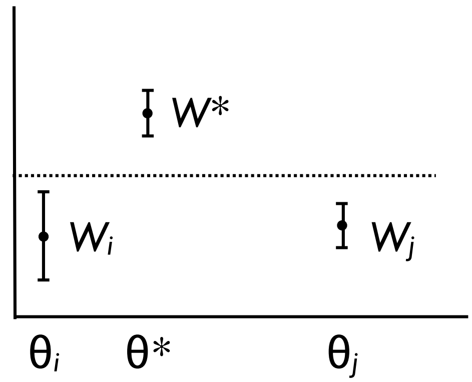


Figure 4.13: Non-overlapping intervals around the maximal and other points.

maximum is then straightforward, assuming the noise in each estimate is independent:

$$\Pr\{W_{j^*} \geq \max_{j \neq j^*} W_j\} = \prod_{j \neq j^*} \Pr\{W_{j^*} \geq W_j\} \geq \prod_{j \neq j^*} (1-\alpha_j)(1-\alpha_{j^*}) = \\ (1-\alpha_{j6*})^{J-1} \prod_{j \neq j^*} (1-\alpha_j).^{16} \tag{4.1}$$

This gives us a very crude bound on the probability of the estimated optimal design choice. There is a simple way to tighten this bound: we can use probability bounds for the appropriate half-intervals rather than whole intervals. Particularly, for any $j \neq j^*$, we need α_j such that $W_j \leq \hat{W}_j + q_j$ with probability at least $1 - \alpha_j$, and for j^* we need that $W_{j^*} \geq \hat{W}_{j^*} - q_{j^*}$ with probability at least $1 - \alpha_j$. The same result as in Equation 4.1 obtains, but the associated $(1 - \alpha)$s are considerably tighter.

Observe that in the above analysis no distributional assumptions are needed. There is no magic here, as these assumptions are simply hidden in the interval estimates. Indeed, as long as the estimates $\hat{W}_j$ have no ties (as they generically would not), it is possible to derive non-overlapping confidence intervals. To derive these intervals, we naturally would need the specifics of the distributions on the noise, but once we have them, the intervals already capture all the information sufficient for this (relatively crude) sensitivity analysis.

Since my domains will generally feature infinite design spaces, as did the TAC/SCM design problem, I would like to extend the bound in Equation 4.1 to infinite domains. In order to make any sensitivity analysis possible in such a case, additional assumptions

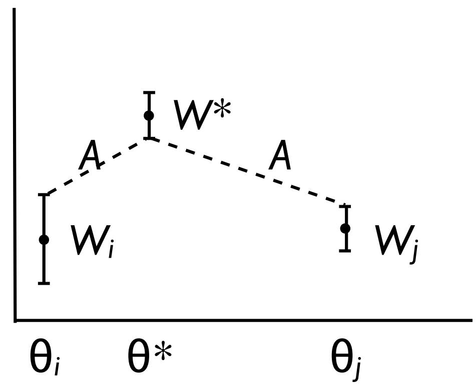


Figure 4.14: Non-overlapping intervals around the maximal and other points, such that any points in the intervals fall below the lower confidence bound on the maximal point by Lipschitz continuity.

need to be made about the structure of the objective function in θ. Here my assumption is Lipschitz continuity. Specifically, I assume that there is a constant A such that $|W(\theta_1) - W(\theta_2)| \leq A\|\theta_1 - \theta_2\|$ for any $\theta_1, \theta_2 \in \Theta$. Given this assumption, I now require that not only the confidence interval for the maximal and other points be non-overlapping, but that no intermediate points be above the maximal lower bound, as show in Figure 4.14. Formally, the requirement is that for every j,

$$A\|\theta_{j^*} - \theta_j\| + \hat{W}_j + q_j \leq \hat{W}_{j^*} - q*,$$

where q_j are upper and q_{j^*} the lower $1 - \alpha_j$ and $1 - \alpha_{j^*}$ confidence bounds respectively. The expression in Equation 4.1 can then be applied without change.

The sensitivity analysis in this section is understandable very abstract. Below, I provide an example of sensitivity analysis in the context of TAC/SCM.

4.10 Probabilistic Analysis in TAC/SCM

The simulation-based analysis of TAC/SCM above has produced evidence in support of the conclusion that no reasonable setting of storage cost was likely to sufficiently curb excessive day-0 procurement in TAC/SCM 2004. All of this evidence has been in the form of simple interpolation and extrapolation of estimates of the Nash equilibrium correspondence. These estimates are based on simulating game instances, and are subject to sampling noise contributed by the various stochastic elements of the game. In this section, I develop and apply methods for evaluating the sensitivity of the ϵ-bound calculations to such stochastic effects.

Suppose that all agents have finite (and small) pure strategy sets, A. Thus, it is feasible to sample the entire payoff matrix of the game. Additionally, suppose that noise is additive with zero-mean and finite variance, that is, $U_i(a) = u_i(a) + \tilde{\xi}_i(a)$, where $U_i(a)$ is the observed payoff to i when a was played, $u_i(a)$ is the actual corresponding payoff, and $\tilde{\xi}_i(a)$ is a mean-zero normal random variable. I designate the known variance of $\tilde{\xi}_i(a)$ by $\sigma_i^2(a)$. Thus, I assume that $\tilde{\xi}_i(a)$ is normal with distribution $N(0, \sigma_i^2(a))$.

I take $\bar{u}_i(a)$ to be the sample mean over all $U_i(a)$ in D, and follow Chang and Huang [2000] to assume that there is an improper prior over the actual payoffs $u_i(a)$ and sampling was independent for all i and a. I also rely on their result that $u_i(a)|\bar{u}_i(a) = \bar{u}_i(a) - Z_i(a)/[\sigma_i(a)/\sqrt{n_i(a)}]$ are independently distributed with posterior distributions $N(\bar{u}_i(a), \sigma_i^2(a)/n_i(a))$, where $n_i(a)$ is the number of samples taken of payoffs to i for pure profile a, and $Z_i(a) \sim N(0, 1)$.

I now derive a generic probabilistic bound that a profile $a \in A$ is an ϵ-Nash equilibrium. If $u_i(\cdot)|\bar{u}_i(\cdot)$ are independent for all $i \in I$ and $a \in A$, we have the following result (from this point on I omit conditioning on $\bar{u}_i(\cdot)$ for brevity):

Proposition 4.4

$$\begin{aligned} &\Pr\left(\max_{i \in I} \max_{b \in A_i} u_i(b, a_{-i}) - u_i(a) \le \epsilon\right) = \\ &= \prod_{i \in I} \int_{\mathbb{R}} \prod_{b \in A_i \setminus a_i} \Pr(u_i(b, a_{-i}) \le u + \epsilon) f_{u_i(a)}(u) du, \end{aligned} \tag{4.2}$$

where $f_{u_i(a)}(u)$ *is the pdf of* $N(\bar{u}_i(a), \sigma_i(a))$.

This result is restated in Proposition 7.22 and proved in the corresponding Appendix.

The posterior distribution of the optimum mean of n samples, derived by Chang and

Huang [2000], is

$$\Pr\left(u_i(a) \le c\right) = 1 - \Phi\left[\frac{\sqrt{n_i(a)}(\bar{u}_i(a) - c)}{\sigma_i(a)}\right], \tag{4.3}$$

where $a \in A$ and $\Phi(\cdot)$ is the $N(0,1)$ distribution function.

Combining the results (4.2) and (4.3), I obtain a probabilistic confidence bound that $\epsilon(a) \le \gamma$ for a given γ.

Now, I consider cases of incomplete data and use the results I have just obtained to construct an upper bound (restricted to profiles represented in data) on the distribution of $\sup\{\phi^*(\theta)\}$ and $\inf\{\phi^*(\theta)\}$ (assuming that both are attainable):

$$\Pr\{\sup\{\phi^*(\theta)\} \le x\} \le_D \Pr\{\exists a \in D : \phi(a) \le x \wedge a \in \mathcal{N}(\theta)\} \le$$
$$\sum_{a \in D:\phi(a) \le x} \Pr\{a \in \mathcal{N}(\theta)\} = \sum_{a \in D:\phi(a) \le x} \Pr\{\epsilon(a) = 0\},$$

where x is a real number and $\le_D$ indicates that the upper bound accounts only for strategies that appear in the data set D. Since the events $\{\exists a \in D : \phi(a) \le x \wedge a \in \mathcal{N}(\theta)\}$ and $\{\inf\{\phi^*(\theta)\} \le x\}$ are equivalent, this also defines an upper bound on the probability of $\{\inf\{\phi^*(\theta)\} \le x\}$. If we define $E = (-\infty, x]$, the above expression gives an upper bound on the value of the probability distribution of $\sup\{\phi^*(\theta)\}$ at x (restricted to D), and, in this particular case, an upper bound on the value of the distribution of $\inf\{\phi^*(\theta)\}$ at x[17] as well as the upper bound on the probability that $\cup_{\phi(a) \in \phi^*(\theta)}[\phi(a) \le x]$, with union restricted to $a \in D$. The values thus derived comprise the Tables 4.1 and 4.2.

$\phi^*(\theta)$	$\theta = 0$	$\theta = 50$	$\theta = 100$
<2.7	0.000098	0	0.146
<3	0.158	0.0511	0.146
<3.9	0.536	0.163	1
<4.5	1	1	1

Table 4.1: Upper bounds on the distribution of $\inf\{\phi^*(\theta)\}$ restricted to D for $\theta \in \{0, 50, 100\}$ when $\mathcal{N}(\theta)$ is a set of Nash equilibria.

Tables 4.1 and 4.2 suggest that the existence of *any* equilibrium with $\phi(a) < 2.7$ is

[17]This is true because the events $\cup_{\phi(a) \in \phi^*(\theta)}[\phi(a) \le x]$ and $\inf\{\phi^*(\theta)\} \le x$ are equivalent.

$\phi^*(\theta)$	$\theta = 150$	$\theta = 200$	$\theta = 320$
<2.7	0	0	0.00132
<3	0.0363	0.141	1
<3.9	1	1	1
<4.5	1	1	1

Table 4.2: Upper bounds on the distribution of $\inf\{\phi^*(\theta)\}$ restricted to D for $\theta \in \{150, 200, 320\}$ when $\mathcal{N}(\theta)$ is a set of Nash equilibria.

unlikely for any θ that I have data for, although this judgment, as I mentioned, is only with respect to the profiles I have actually sampled. I can then accept this as another piece of evidence that the designer could not find a suitable setting of θ to achieve his objectives—indeed, the designer seems unlikely to achieve his objective even if he could persuade participants to play a desirable equilibrium!

Table 4.1 also provides additional evidence that the agents in the TAC/SCM 2004 tournament were indeed rational in procuring large numbers of components at the beginning of the game. If we look at the third column of this table, which corresponds to $\theta = 100$, we can gather that no profile a in the data with $\phi(a) < 3$ is very likely to be played in equilibrium.

The bounds above provide some general evidence, but ultimately I am interested in a concrete probabilistic assessment of my conclusion with respect to the data I have sampled. Particularly, I would like to say something about what happens for the settings of θ for which I have no data. To derive an approximate probabilistic bound on the probability that no $\theta \in \Theta$ could have achieved the designer's objective, let $\cup_{j=1}^{J}\Theta_j$, be a partition of Θ, and assume that the function $\sup\{\phi^*(\theta)\}$ satisfies the *Lipschitz condition* with *Lipschitz constant* A_j on each subset Θ_j.[18] Since I have determined that raising the storage cost above 320 is undesirable due to secondary considerations, I restrict attention to $\Theta = [0, 320]$. I now define each subset j to be the interval between two points for which I have produced data. Thus,

$$\Theta = [0, 50] \bigcup (50, 100] \bigcup (100, 150] \bigcup (150, 200] \bigcup (200, 320],$$

[18]A function that satisfies the *Lipschitz condition* is called *Lipschitz continuous*.

with j running between 1 and 5, corresponding to subintervals above. I will further denote each Θ_j by $(a_j, b_j]$.[19] Then, the following Proposition gives us an *approximate* upper bound[20] on the probability that $\sup\{\phi^*(\theta)\} \leq \alpha$.

Proposition 4.5

$$\Pr\{\bigvee_{\theta \in \Theta} \sup\{\phi(\theta)\} \leq \alpha\} \leq_D$$
$$\sum_{j=1}^{5} \sum_{y,z \in D: y+z \leq c_j} \left(\sum_{a:\phi(a)=z} \Pr\{\epsilon(a) = 0\} \right) \left(\sum_{a:\phi(a)=y} \Pr\{\epsilon(a) = 0\} \right),$$

where $c_j = 2\alpha + A_j(b_j - a_j)$ *and* $\leq_D$ *indicates that the upper bound only accounts for strategies that appear in the data set* D.

Due to the fact that the bounds are approximate, I cannot use them as a conclusive probabilistic assessment. Instead, I take this as another piece of evidence to complement my findings.

Even if I can assume that a function that I approximate from data is Lipschitz continuous, I would rarely actually know the Lipschitz constant for any subset of Θ. Thus, I am faced with a task of estimating it from data. Here, I tried three methods of doing this. The first one simply takes the highest slope that the function attains within the available data and uses this constant value for every subinterval. This produces the most conservative bound, and in many situations it is unlikely to be informative.

An alternative method is to take an upper bound on slope obtained *within each subinterval* using the available data. This produces a much less conservative upper bound on probabilities. However, since the actual upper bound is generally greater for each subinterval, the resulting probabilistic bound may be deceiving.

A final method that I tried is a compromise between the two above. Instead of taking the conservative upper bound based on data over the entire function domain Θ, I take the average of upper bounds obtained at each Θ_j. The bound at an interval is then taken to be the maximum of the upper bound for this interval and the average upper bound for all

[19] The treatment for the interval [0,50] is identical.

[20] It is approximate in a sense that I only take into account strategies that are present in the data.

intervals.

The results of evaluating the expression for

$$\Pr\{\bigvee_{\theta\in\Theta} \sup\{\phi^*(\theta)\} \leq \alpha\}$$

when $\alpha = 2$ are presented in Table 4.3. In terms of my claims in this work, the expression

$\max_j A_j$	A_j	$\max\{A_j$,ave$(A_j)\}$
1	0.00772	0.00791

Table 4.3: Approximate upper bound on probability that some setting of $\theta \in [0, 320]$ will satisfy the designer objective with target $\alpha = 2$. Different methods of approximating the upper bound on slope in each subinterval j are used.

gives an upper bound on the probability that some setting of θ (i.e., storage cost) in the interval [0,320] will result in total day-0 procurement that is no greater in any equilibrium than the target specified by α and taken here to be 2. As I had suspected, the most conservative approach to estimating the upper bound on slope, presented in the first column of the table, provides us little information here. However, the other two estimation approaches, found in columns two and three of Table 4.3, suggest that I am indeed quite confident that no reasonable setting of $\theta \in [0, 320]$ would have done the job. Given the tremendous difficulty of the problem, this result is very strong.[21] Still, one must be very cautious in drawing too heroic a conclusion based on this evidence. Certainly, I have not "checked" all the profiles, but only a small proportion of them (infinitesimal, if I consider the entire continuous domain of θ and strategy sets). Nor can one expect ever to obtain enough evidence to make completely objective conclusions. Instead, the approach I advocate here is to collect as much evidence as is feasible given resource constraints, and make the most compelling judgment based on this evidence.

[21]Since I did not have all the possible deviations for any profile available in the data, the true upper bounds may be even lower.

4.11 Comparative Statics

As reluctant as a modeler may be to admit it, models are rarely perfect. Consequently, an important part of economic analysis has become *comparative statics*, that is, analysis of how the outcomes (e.g., optima, equilibria) are affected by small changes in the model. It is only natural to expect that such analysis be applied in the context of simulation-based mechanism design.[22]

Given that we have methods for approximating optimal mechanisms, comparative statics are actually conceptually not very difficult and can be done via the following procedure:

1. Select a Δ to represent a small change in a simulation model parameter V
2. Estimate the (nearly) optimal mechanism for $V = v^+ = v + \Delta$, and for $V = v^- = v - \Delta$, where v is the original parameter setting; say these have the objective values W^+ and W^- and the optimal parameter settings θ^+ and θ^- respectively
3. Obtain the estimate of the rate of change in the optimal objective value or optimal design choice using the *finite difference* method, that is, $D_W \approx \frac{W^+ - W^-}{v^+ - v^-}$ and $D_\theta \approx \frac{\theta^+ - \theta^-}{v^+ - v^-}$
4. Repeat the procedure to take n derivative estimates and take the average

The finite difference method yields a biased but consistent estimator and its convergence rate can be improved considerably by using common random variables [L'Écuyer, 1991]. It is just one, though perhaps the most commonly used, derivative estimator. Alternative procedures, for example, via simultaneous perturbations [Spall, 2003] in which a random small perturbation of the parameter vector is taken instead of a fixed Δ, may also be used in its place and provide some theoretical advantages. Since the area of derivative estimation is in itself very well explored in the simulations literature, I do not mean to claim much novelty here, but merely suggest using the ready-made methods for a specific

[22] It is just as natural to expect it to be applied in the context of simulation-based games. This discussion is deferred until I engage the topic in more detail in Part II.

purpose—in this case, simulation-based comparative statics. The question of which estimation methods are best suited for this purpose is, perhaps, interesting in its own right, although I do not see a priori why this setting has special structure that suggests particular advantage to any individual estimators beyond what has been previously explored in the literature.

4.12 Conclusion

In this chapter I introduced the problem of simulation-based mechanism design. The simulation-based mechanism design approach is valuable in settings in which some insight needs to be obtained about the economic mechanism design problem, but analytical techniques are intractable and the revelation principle need not hold. I model the general mechanism design problem as a one-shot two-stage game and use this model to obtain a general high-level framework for computational mechanism design, which is in the spirit of backwards induction. Furthermore, I present some ideas for solving problems which possess particular structure, as well as general design problems, and suggest how to do sensitivity analysis and comparative statics.

Throughout the chapter, I present a series of specific techniques for simulation-based mechanism design in the context of a TAC/SCM design problem. I am able to present considerable evidence about the structure of the Nash equilibrium correspondence based on simulation experiments, and use it to conclude that the particular approach undertaken by the TAC/SCM designers to alleviate the problem in practice had fundamental limitations.

CHAPTER 5

A General Framework for Computational Mechanism Design on Constrained Design Spaces

IN WHICH *I present a framework for mechanism design on parametrized design spaces. I focus on settings which induce one-shot Bayesian games and provide a series of examples of auction design problems which I solve computationally. In all the examples, I am able to verify the properties of computational solutions analytically, and, thus, can directly assess the efficacy of the approach.*[1]

5.1 Automated Mechanism Design for Bayesian Games

In the previous chapter I presented a general abstract framework for computational mechanism design on a constrained design space $\Theta \subset \mathbb{R}^n$. Here, I make it somewhat more concrete by focusing on settings in which the mechanism choices induce one-shot Bayesian games, that is, for every $\theta \in \Theta$

$$\Gamma_\theta = [I, \{A_i\}, \{T_i\}, F(\cdot), \{u_i(a, t, \theta)\}].$$

[1]The material in this chapter is taken from Vorobeychik *et al.* [2007a].

I require the designer to specify his objective in the form $W(s(t,\theta),t,\theta)$, where $s(t,\theta)$ is a *solution* or outcome of agent play. Significantly, the objective may be specified *algorithmically*; the restriction is merely that it outputs a real number representing the objective value for the combination of arguments just specified. Note that the solution $s(t,\theta)$ is a function of player types, since each player is presumed to observe his type prior to making a strategic choice. Below, I also use the short notation $s(\theta)$ to denote the equilibrium strategy profile, which in the Bayesian setting is a profile of functions of player types. Since the designer's objective will depend on player types, either indirectly due to its dependence on the player strategies, or directly through the type argument, we need to transform the type-dependent specification of the objective, $W(s(t,\theta),t,\theta)$, into $W(s(\theta),\theta)$; that is, rather than producing a function of types for a design choice θ, we need to output a real number. I refer to this transformation as *objective evaluation*. In Section 5.1.4 I present two principled approaches for evaluating the objective with respect to the distribution of player types.

In addition to a specification of the objective function, the designer may specify a collection of constraints on the outcomes (solutions) induced by the corresponding design choices. Let the constraints be specified as $\mathcal{C} = \{\mathcal{C}_i(s(t,\theta),t,\theta)\}$, although these may, again, be provided in an algorithmic form which returns *true* if the constraint is satisfied and *false* otherwise for a particular setting of the specified arguments.

As is common in the mechanism design literature, I evaluate mechanisms with respect to a *sample* Bayes-Nash equilibrium, $s(t,\theta)$. In other words, I adopt the weak implementation perspective.

Below, I instantiate the procedure I introduced in Section 4.7.2 using a concrete black-box optimization routine and elucidate its first three steps, thereby presenting a full parametrized mechanism design framework for Bayesian games.

5.1.1 Designer's Optimization Problem

I begin by treating the designer's problem as black-box optimization, where the black box produces a noisy evaluation of the designer's objective, $W(s(\theta),\theta)$, for the input design parameter, θ, given the game-theoretic predictions of play $s(\theta)$. Once I frame the problem as a black-box optimization problem, I can draw on a wealth of literature devoted to developing methods to approximate optimal solutions in this setting [Spall, 2003]. While I can in principle select any one of these, I have chosen simulated annealing, as it has proved quite effective for a great variety of simulation optimization problems in noisy settings with many local optima [Corana *et al.*, 1987; Fleischer, 1995; Siarry *et al.*, 1997]. By instantiating the high-level procedure in Section 4.7.2 with simulated annealing, we can obtain the following procedure, to which I refer below as the *automated mechanism design (AMD) framework*:

1. Begin with a randomly selected $\theta_0 \in \Theta$
2. In iteration k, select the next candidate mechanism, $\theta_{k+1}|\theta_k$ from a probability distribution $F_k(\theta_k)$
3. *Evaluate* θ_k and θ_{k+1}, obtaining W_k and W_{k+1} respectively. To evaluate a candidate mechanism θ, proceed as follows:
    (a) Compute or approximate a solution $s(t, \theta)$ of Γ_θ
    (b) Apply every constraint $\mathcal{C}_i(s(t, \theta), t, \theta) \in \mathcal{C}$ to the solution $s(t, \theta)$; return that θ is infeasible if any constraint fails (in my implementation, set $W(s(\theta), \theta) = -\infty$)[2]
    (c) If all the constraints are satisfied, evaluate the objective value $W(s(\theta), \theta)$ as described in Section 5.1.4
4. Set $\theta_{k+1} \leftarrow \theta_k$ w.p. $p_k(W_k, W_{k+1})$
5. Repeat steps 1-4
6. Return an approximately optimal design based on the resulting optimization path

In this procedure, $p_k(W_k, W_{k+1})$ is the *Metropolis acceptance probability* [Spall, 2003], under which the next iterate is "accepted" with probability 1 if it yields a higher objective

[2] A more sophisticated approach would perhaps use a penalty method, which would then require us to know the extent of constraint violation.

value; if θ_{k+1} yields a lower objective value, however, it is still accepted with small probability, but the probability of acceptance is exponentially decreasing in $|W_{k+1} - W_k|$. I opt for a relatively simple adaptive implementation of simulated annealing, with normally distributed random perturbations applied to the solution candidate θ_k in every iteration to obtain the candidate mechanism θ_{k+1}. That is, $F_k(\theta_k) = N(\theta_k, \sigma_k^2)$ for a specified variance sequence σ_k^2.[3]

As an application of black-box optimization, the mechanism design problem in my formulation is just one of many problems that can be addressed with one of a selection of methods. What makes it special is the subproblem of evaluating the objective function for a given mechanism choice, and the particular nature of mechanism design constraints which are evaluated based on Nash equilibrium outcomes and agent types.

5.1.2 Computing Nash Equilibria

As implied by the backwards induction process, I must obtain solutions (Bayes-Nash equilibria in the current setting) of the games induced by the design choice, θ, in order to evaluate the objective function. In general, this is simply not possible to do, since Bayes-Nash equilibria may not even exist in an arbitrary game, nor is there a general-purpose tool to find them. However, there are a number of tools that can find or approximate solutions in specific settings. For example, GAMBIT [McKelvey *et al.*, 2005] is a general-purpose toolbox of solvers that can find Nash equilibria in finite games, although it is often ineffective for even moderately sized games.

To the best of my knowledge, the only exact solver for a broad class of infinite games of incomplete information was introduced by Reeves and Wellman [2004] (henceforth, RW). Indeed, RW is a best-response finder, which has successfully been used iteratively to obtain sample Bayes-Nash equilibria for a restricted class of infinite two-player games of incomplete information. In Part II, I present a set of techniques for approximating Bayes-Nash equilibria in simulation-based infinite one-shot Bayesian games, thereby considerably extending the domain of applicability of my computational mechanism de-

[3] I use an exponentially decreasing sequence of variances in my implementation of the algorithm.

sign framework beyond what is presented in this chapter.

While RW is often effective in converging to a sample Bayes-Nash equilibrium, it does not do so always. Thus, we are presented with the first problem that makes automated mechanism design unique: how does one evaluate the objective if no solution can be obtained? There are a number of ways to approach this difficulty. For example, since RW is an iterative tool that will always (at least in principle) produce a best response to a given strategy profile, one can use the last best response in the non-converging finite series of iterations as the prediction of agent play. Alternatively, one may simply constrain the design to discard any choices for which the solver does not produce an answer. Here I employ the latter alternative, which is the more conservative of the two.

Since the goal of automated mechanism design is to approximate solutions to design problems with arbitrary objectives and constraints and to handle games with arbitrary type distributions, I treat the probability distribution over player types as a black box from which we can draw samples of profiles of joint player types. Thus, we must use numerical techniques to evaluate the objective with respect to player types, thereby introducing noise into the process.

5.1.3 Dealing with Constraints

Mechanism design can feature any of the following three classes of constraints: *ex ante* (constraints evaluated with respect to the joint distribution of types), *ex interim* (evaluated separately for each player and type with respect to the joint type distribution of other players), and *ex post* (evaluated for every joint type profile). When the type space is infinite we, of course, cannot numerically evaluate any expression for every type. I therefore replace these constraints with probabilistic constraints that must hold for a set of types which has a large probability measure. For example, an ex post individual rationality (IR) constraint would have to hold only for type profiles that can occur with probability greater than 0.95. Besides a computational justification, there is a practical justification for weakening the requirement that constraints be satisfied for every possible player type (or joint type profile), at least as far as individual rationality is concerned. When we introduce constraints that hold with high probability, we may be excluding a small measure set of types from participation (this is the case for the individual rationality constraints). But by excluding a small portion of types, the expected objective function will change very little, and, similarly, such a change will introduce little incentive for other types to deviate. Indeed, by excluding a subset of types with low valuations for an object, the designer may raise its expected revenue [Krishna, 2002].

Intuitively, it is unlikely to matter if a constraint fails on a set of types which occurs with probability zero. I conjecture, further, that in most practical design problems, violation of a constraint on a low-measure set of types will also be of little consequence, either because the resulting design is easy to fix, or because the other types will likely not have very beneficial deviations even if they account in their decisions for the effect of these unlikely types on the game dynamics. I support this conjecture via a series of applications of my framework: in none of these did my constraint relaxation lead the designer much astray.

Even when we weaken constraints based on agent type sets to their probabilistic equivalents, we still need a way to verify that such constraints hold by sampling from the type distribution. Since we can take only a finite number of samples, we will in fact verify a probabilistic constraint only at some level of confidence. The question we want to ask, then, is how many samples do we need in order to say with probability at least $1 - \alpha$ that the probability of seeing a type profile for which the constraint is violated is no more than p? That is the subject of the following theorem.

Theorem 5.1 *Let B denote a set on which a probabilistic constraint is violated, and suppose that we have a uniform prior over the interval $[0, 1]$ on the probability measure of B. Then, we need at least $\frac{\log \alpha}{\log(1-p)} - 1$ samples to verify with probability at least $1 - \alpha$ that the measure of B is at most p.*

In practice, however, this is not the end of the story for the ex interim constraints. The reason is that the ex interim constraint will take expectation with respect to the joint distribution of types of players other than the player i for which it is verified. Since we must evaluate this expectation numerically, we cannot escape the presence of noise in constraint evaluation. Furthermore, if we are trying to verify the constraint for many type realizations, it is quite likely that in at least one of these instances we will get unlucky and the numerical expectation will violate the constraint, even though the actual expectation does not. Although from a theoretical standpoint we should not be bothered by this issue, it has much practical significance: the search problem already faces a vast space of infeasibility, and such artificial constraint violations only add to its complexity and may well doom the approach from the start.

I heuristically circumvent this problem in two ways. First, I introduce a slight tolerance for a constraint, so that it will not fail due to small evaluation noise. Second, I split the set of types for which the constraint is verified into smaller groups, and throw away a small proportion of types in each group with the worst constraint evaluation result. For example, if we are trying to ascertain that ex interim individual rationality holds, we would throw away several types with the lowest estimated ex interim utility value.

I next describe three specific constraints employed in my applications.

Equilibrium Convergence Constraint Given that the game solutions are produced by a heuristic (iterative best-response) algorithm, they are not inherently guaranteed to represent equilibria of the candidate mechanism. We can instead enforce this property through an explicit constraint. The purpose of this constraint is to ensure that every mechanism is indeed evaluated with respect to a true equilibrium (or near-equilibrium) strategy profile (given my assumption that a Bayes-Nash equilibrium is a relevant predictor of agent play). For example, best response dynamics using RW need not converge at all.

I formally define this constraint as follows:

Definition 5.2 *Let $s(t)$ be the last strategy profile in a sequence of best response iterations, and let $s'(t)$ immediately precede $s(t)$ in this sequence. Then the equilibrium convergence constraint is satisfied if for every joint type profile of players, $|s(t) - s'(t)| < \delta$ for some a priori fixed tolerance level δ.*[4]

The problem that we cannot in practice evaluate this constraint for every joint type profile

[4]Note that if the payoff functions are Lipschitz continuous with a Lipschitz constant A, the condition above implies that $s(t)$ is a $A\delta$-Bayes-Nash equilibrium.

is resolved by making it probabilistic, as described above. Thus, I define a $(1-p)$-*strong equilibrium convergence constraint*:

Definition 5.3 *Let $s(t)$ be the last strategy profile produced in a sequence of solver iterations, and let $s'(t)$ immediately precede $s(t)$ in this sequence. Then the $(1-p)$-strong equilibrium convergence constraint is satisfied if for a set of type profiles t with probability measure no less than $1-p$, $|s(t)-s'(t)|<\delta$ for some a priori fixed tolerance level δ.*

Ex Interim Individual Rationality Ex-Interim-IR (EIIR) specifies that for every agent and type, the agent's expected utility conditional on its type is greater than its opportunity cost of participating in the mechanism. This idea is formalized in Definition 3.28. Again, in the automated mechanism design framework, I must modify the classical definition of EIIR to a probabilistic constraint as described above.

Definition 5.4 *The $(1-p)$-strong Ex-Interim-IR constraint is satisfied when for every agent $i \in I$, and for a set of types $t_i \in T_i$ with probability measure no less than $1-p$, $E_{t_{-i}} u_i(t, s(t)|t_i) \geq c_i(t_i) - \delta$, where $c_i(t_i)$ is the opportunity cost of agent i with type t_i of participating in the mechanism, and δ is some a priori fixed tolerance level.*

Commonly in the mechanism design literature the opportunity cost of participation, $c_i(t_i)$, is assumed to be zero but this assumption may not hold, for example, in an auction where not participating would be a give-away to competitors and entail negative utility.

Minimum Revenue Constraint The final constraint that I consider ensures that the designer will obtain some minimal amount of revenue (or bound its loss) in attaining a non-revenue-related objective.

Definition 5.5 *The minimum revenue constraint is satisfied if $E_t k(s(t), t) \geq C$, where $k(s(t), t)$ is the total payment made to the designer by agents with joint strategy $s(t)$ and joint type profile t, and C is the lower bound on revenue.*

5.1.4 Evaluating the Objective

As I mention above, if any constraint fails, the corresponding objective function value $W(s(\theta), \theta)$ is evaluated to $-\infty$. If all the constraints are satisfied, however, the objective must be evaluated with respect to the distribution of player types. Below, I present two approaches for doing this. The first is the traditional Bayesian approach, which I term *Bayesian mechanism design*, whereas the second is in the spirit of robust optimization, and I term it, correspondingly, *robust mechanism design*.

Bayesian Mechanism Design In much of this chapter I explore *Bayesian mechanism design*, in which the designer is presumed to have a belief about the distribution of agents' types. The Bayesian mechanism design problem evaluates the designer's objective value for a mechanism $\theta \in \Theta$ by taking the expectation of $W(s(t, \theta), t, \theta)$ with respect to the distribution of player types. That is,

$$W(s(\theta), \theta) = E_t[W(s(t, \theta), t, \theta)].$$

I assume that the designer has the same belief about agent types as the agents themselves, although this assumption is there purely for convenience.

Probably Approximately Robust Mechanism Design I address the problem of *robust mechanism design* by allusion to the analogous problem in the optimization literature, referred to as robust optimization [Ben-Tal and Nemirovski, 2002]. In robust optimization, uncertainty over parameters of an optimization program is accounted for by treating the uncertainty set essentially as an adversary which selects the worst outcome for the problem at hand. Thus, the solution to the robust program is one that gives the best outcome in the face of such an adversary.

The analogy here comes from treating the type space of agents as such an adversary. As a justification for such a pessimistic outlook, we can imagine that the designer is extremely averse to poor outcomes, perhaps envisioning a politician who is extremely worried about being reelected. While risk aversion can be treated formally in a Bayesian framework, such treatment requires the designer to be aware of his risk preferences. Robust treatment sidesteps this issue and may provide a useful approximation instead.

Formally, we can express the robust objective of the designer as

$$W(s(\theta), \theta) = \inf_{t \in T} W(s(t, \theta), t, \theta). \tag{5.1}$$

Note that this change is relatively minor and has no effect on the rest of the framework (replacing the expectation operator with the infimum). However, it entails a computationally infeasible problem of ensuring robustness for every joint type of a possibly infinite type space; anything short of that is no longer really robust. To address this problem, I relax the pure robustness criterion to probabilistic robustness.[5] My relaxation is that the designer is not worried about the worst subset of outcomes of the type space if that subset has very small measure. For example, if the set of types that has probability measure of 0.0001 are extremely unfavorable, their appearance is deemed sufficiently unlikely not to worry the designer. Furthermore, we can probabilistically ascertain that the worst outcome based on a finite number of samples from the type distribution is no better than a large measure of the type space. I call this paradigm *probably approximately robust* mechanism design.

To formalize this, suppose that in every exploration step using my framework one takes n samples from the type distribution, $T^n = \{T_1, \ldots, T_n\}$, and then selects the worst value of the objective over these n types:

$$\hat{W}(s(t, \theta), t, \theta) = \inf_{t \in T^n} W(s(t, \theta), t, \theta).$$

One would like to select a sufficiently high number of samples n, in order to attain high enough confidence, $1 - \alpha$, that the best objective value that he can obtain via L explorations using this framework is approximately robust. The following theorem gives such an n.

[5]To clarify, the critical issue is not so much the impossibility of computing the objective value exactly: this problem obtains even in the Bayesian mechanism design setting. Rather, the relaxation is necessary in order to enable us to speak in a meaningful way about objective estimation and to obtain probabilistic bounds, such I the one I present below.

Theorem 5.6 *Suppose we select the best design of L candidates, using n samples from the type distribution for each to estimate the value of $\inf_{t\in T\setminus T_A} W(s(t,\theta),t,\theta)$, where T_A is the set of types with value of $W(s(t,\theta),t,\theta)$ below $\hat{W}(s(t,\theta),t,\theta)$. If we want to attain the confidence of at least $1-\alpha$ that the measure of T_A is at most p, we need*

$$n \geq \frac{\log(1-(1-\alpha)^{\frac{1}{L}})}{\log(1-p)}$$

samples.

5.2 Extended Example: Shared-Good Auction (SGA)

5.2.1 Setup

Consider the problem of two people trying to decide between two options. Unless both players prefer the same option, no standard voting mechanism (with either straight votes or a ranking of the alternatives) can help with this problem. We can propose a simple auction: each player submits a bid and the player with the higher bid wins, paying some function of the bids to the loser in compensation. Reeves [2005] considered a special case of this auction and gave the example of two roommates using it to decide who should get the bigger bedroom and for how much more rent. Cramton [1987] considered this problem in the context of dissolving partnerships.

I define a space of mechanisms for this problem that are all budget balanced, individually rational, and (assuming monotone strategies) socially efficient. I then search the mechanism space for games that satisfy additional properties. The following is a payoff function defining a space of games parametrized by a payment function f.

$$u(t,a,t',a') = \begin{cases} t - f(a,a') & \text{if } a > a' \\ 0.5t & \text{if } a = a' \\ f(a',a) & \text{if } a < a', \end{cases} \tag{5.2}$$

where $u(\cdot)$ gives the utility for an agent who has a value t for winning and bids a against an agent who has value t' and bids a'. The ts are the agents' types and the as their actions. The semantics are that the winner (i.e., the player with the higher bid) pays $f(a, a')$ to the loser, where a in this case is the winning and a' the losing bid. In the tie-breaking case (which occurs with probability zero for many classes of strategies) the payoff is the average of the two other cases because the winner is chosen by a coin flip.

I now consider a restriction of the class of mechanisms defined above.

Definition 5.7 $\mathrm{SGA}(h, k)$ *is the mechanism defined by Equation 5.2 with* $f(a, a') = ha + ka'$, $h, k \in [0, 1]$.

For example, in $\mathrm{SGA}(1/2, 0)$ the winner pays half its own bid to the loser; in $\mathrm{SGA}(0, 1)$ the winner pays the loser's bid to the loser. More generally, h and k will be the relative proportions of winner's and loser's bids that will be transferred from the winner to the loser. I now give Bayes-Nash equilibria for such games when types are uniformly distributed.[6]

Theorem 5.8 *For* $h, k \geq 0$ *and types* $U[A, B]$ *with* $B \geq A + 1$ *the following is a symmetric Bayes-Nash equilibrium of* $\mathrm{SGA}(h, k)$*:*

$$s(t) = \frac{t}{3(h+k)} + \frac{hA + kB}{6(h+k)^2}.$$

We can now characterize the truthful (BNIC, per Definition 3.24) mechanisms in this space. According to Theorem 5.8, $\mathrm{SGA}(1/3, 0)$ is truthful for $U[0, B]$ types. I now show that this is the *only* truthful design in this design space.

Theorem 5.9 *With* $U[0, B]$ *types (*$B > 0$*),* $\mathrm{SGA}(h, k)$ *is BNIC if and only if* $h = 1/3$ *and* $k = 0$*.*

Below, I use this characterization to present concrete examples of the failure of the revelation principle for several sensible designer objectives. Since $\mathrm{SGA}(1/3, 0)$ is the only truthful mechanism in our design space, we can directly compare the objective value obtained from this mechanism and the best untruthful mechanism in the sections that follow. From here on I focus on the case of $U[0, 1]$ types.

[6] While not a significant contribution of this work, this is an original result to the best of my knowledge.

5.2.2 Automated Design Problems

Bayesian Mechanism Design Problems

Minimize Difference in Expected Utility First, I consider *fairness*, or negative differences between the expected utility of winner and loser, as the objective. Formally, the goal is to minimize

$$|E_{t,t'}[u(t, s(t), t', s(t'), k, h \mid a > a') - u(t, s(t), t', s(t'), k, h \mid a < a')]|. \tag{5.3}$$

I first use the equilibrium bid derived above to analytically characterize optimal mechanisms.

Theorem 5.10 *The objective value in (5.3) for* $\mathrm{SGA}(h, k)$ *is*

$$\frac{2h + k}{9(h + k)}.$$

Furthermore, $\mathrm{SGA}(0, k)$*, for any* $k > 0$*, minimizes the objective, and the optimum is 1/9.*

By comparison, the objective value for the truthful mechanism, $\mathrm{SGA}(1/3, 0)$, is 2/9, twice as high as the minimum produced by an untruthful mechanism. Thus, the revelation principle does not hold for this objective function in the specified design space. I can use Theorem 5.10 to find that the objective value for $\mathrm{SGA}(1/2, 0)$, the mechanism described by Reeves [2005], is 2/9.

Now, to test the automated mechanism design framework (as embodied by the procedure in Section 5.1), imagine we do not know about the above analytic derivations (including the derivation of the Bayes-Nash equilibrium). I, thus, run the automated mechanism design procedure in "black-box" mode. Table 5.1 presents results of AMD the search which is initialized to random values of h and k (taking the best outcome from 5 random restarts), and at the starting values of $h = 0.5$ and $k = 0$. Since the objective function turns out to be fairly simple, it is not surprising that I obtain the optimal mechanism for specific and random starting points (indeed, the optimal design was produced from every random starting point I generated).

Parameters	Initial Design	Final Design
h	0.5	0
k	0	1
objective	2/9	1/9
h	random	0
k	random	1
objective	N/A	1/9

Table 5.1: Design that approximately maximizes fairness (minimizes difference in expected utility between utility of winner and loser) when the optimization search starts at a fixed starting point, and the best mechanism from five random restarts.

Minimize Expected (Ex-Ante) Difference in Utility Here I modify the objective function slightly as compared to the previous section, and instead aim to minimize the expected ex ante difference in utility:

$$E|u(t, s(t), t', s(t'), k, h|a > a') - u(t, s(t), t', s(t'), k, h|a < a')|. \tag{5.4}$$

While the only difference from the previous section is the placement of the absolute value sign inside the expectation, this difference complicates the analytic derivation of the optimal design considerably. Therefore, I do not present the actual optimum design values.

Parameters	Initial Design	Final Design
h	0.5	0.49
k	0	1
objective	0.22	0.176
h	random	0.29
k	random	0.83
objective	N/A	0.176

Table 5.2: Design that approximately minimizes expected ex ante difference between utility of winner and loser when the optimization search starts at a random and a fixed starting points.

The results of application of my AMD framework are presented in Table 5.2. While the objective function in this example appears somewhat complex, it turns out (as I discovered through additional exploration) that there are many mechanisms that yield nearly optimal objective values.[7] Thus, both random restarts as well as a fixed starting point produced essentially the same near-optima. By comparison, the truthful design yields the objective value of about 0.22, which is considerably worse.

Maximize Expected Utility of the Winner Yet another objective in the shared-good-auction domain is to maximize the expected utility of the winner.[8] Formally, the designer is maximizing $E[u(t, s(t), t', s(t'), k, h \mid a > a')]$.

I first analytically derive the characterization of optimal mechanisms.

Theorem 5.11 *The problem is equivalent to finding* (h, k) *that maximize*

$$4/9 - k/[18(h + k)].$$

Thus, $k = 0$ *and* $h > 0$ *maximize the objective, and the optimum is* $4/9$*.*

Parameters	Initial Design	Final Design
h	0.5	0.21
k	0	0
objective	4/9	4/9
h	random	0.91
k	random	0.03
objective	N/A	0.443

Table 5.3: Design that approximately maximizes the winner's expected utility.

Here again my results in Table 5.3 are optimal or very nearly optimal, unsurprisingly for this relatively simple application.

Maximize Expected Utility of the Loser Finally, I try to maximize the expected utility of the loser.[9] Formally, the designer is maximizing

$$E[u(t, s(t), t', s(t'), k, h|a < a')].$$

[7] Particularly, I carried out a far more intensive exploration of the search space given the analytic expression for the Bayes-Nash equilibrium to ascertain that the values reported are close to actual optima. Indeed, I failed to improve on these.

[8] For example, the designer may be interested in minimizing the amount of money which changes hands, which is, by construction, an equivalent problem.

[9] For example, the designer may be interested in maximizing the amount of money which changes hands, which is, by construction, an equivalent problem.

I again first analytically derive the optimal mechanism.

Theorem 5.12 *The problem is equivalent to finding* h *and* k *that maximize*

$$2/9 + k/[18(h+k)].$$

Thus, $h = 0$ *and* $k > 0$ *maximize the objective, and the optimum is* $5/18$*.*

Parameters	Initial Design	Final Design
h	0.5	0
k	0	0.4
objective	2/9	5/18
h	random	0.13
k	random	1
objective	N/A	0.271

Table 5.4: Design that approximately maximizes the loser's expected utility.

Table 5.4 shows the results of running AMD in black-box mode in this setting. I can observe that my results are again either actually optimal when the search used a fixed starting point, or close to optimal when random starting points were used. While this design problem is relatively easy and the answer can be analytically derived, the objective function is non-linear, which, along with the presence of noise, adds sufficient complexity to blind optimization to suggest that my success here is at least somewhat interesting.

Robust Mechanism Design Problems

Minimize Nearly-Maximal Difference in Utility Here, I study the problem of probably approximately robust design for maximal difference in players' utility. The robust formulation of this problem is to minimize

$$\sup_{t,t' \in T} |u(t, s(t), t', s(t'), k, h|a > a') - u(u(t, s(t), t', s(t'), k, h|a < a')|.$$

Theorem 5.13 *The robust problem is equivalent to finding h, k that minimize*

$$\frac{h+2k}{3(h+k)}.$$

Thus, $k = 0$ is optimal for any $h > 0$, and the optimal value is 1/3.

Parameters	Initial Design	Final Design
h	random	0.01
k	random	0
objective	N/A	1/3

Table 5.5: Design that approximately robustly minimizes the difference in utility.

As one can see from the results in Table 5.5, the mechanism produced via the automated framework is optimally robust, as the optimum corresponds to one of the robust designs in Theorem 5.13.

Maximize Nearly-Minimal Utility of the Winner The second problem in robust design I consider is maximization of minimum utility of the winner given the type distribution on unit support set. This problem can be more formally expressed as maximizing

$$\inf_{t,t' \in T} u(t, s(t), t', s(t'), k, h | a > a').$$

Theorem 5.14 *The problem is equivalent to finding h, k that minimize*

$$\frac{k}{6(h+k)}.$$

Thus, $k = 0$ is optimal for any $h > 0$, with the optimal value of 0.

Parameters	Initial Design	Final Design
h	random	0.65
k	random	0
objective	N/A	0

Table 5.6: Design that approximately robustly maximizes the winner's utility.

Table 5.6 shows the results of optimizing this objective function using my automated mechanism design framework. As in the previous robust application of my framework, my design is optimally robust according to Theorem 5.14.

Maximize Nearly-Minimal Utility of the Loser The final robust design problem I will consider for the shared-good auction domain is that of robustly maximizing the utility of the loser. More formally, this is expressed as maximizing

$$\inf_{t,t' \in T} \; u(t, s(t), t', s(t'), k, h | a < a').$$

Theorem 5.15 *The problem is equivalent to finding h, k that maximize*

$$\frac{k}{6(h+k)}.$$

Thus, $h = 0$ is optimal for any $k > 0$, with the optimal value of 1/6.

Parameters	Initial Design	Final Design
h	random	0
k	random	0.21
objective	N/A	1/6

Table 5.7: Design that approximately robustly maximizes the loser's utility.

According to Table 5.7, I again observe that my automated process arrived at the optimal robust mechanism, as described in Theorem 5.15.

Of the examples I considered so far, most turned out to be analytic, and one I could only approach numerically. Nevertheless, even in the analytic cases, the objective function forms were not trivial, particularly from a blind optimization perspective. Furthermore, one must take into account that even the simple cases are somewhat complicated by the presence of noise, and thus one need not arrive at global optima even in the simplest of settings so long as the number of samples is not very large.

Having found success in the simple shared-good auction setting, I now turn my attention to a series of considerably more difficult problems.

5.3 Applications

I present results from several applications of my automated mechanism design framework to specific two-player problems. One of these problems, finding auctions that yield maximum revenue to the designer, has been studied in a seminal paper by Myerson [1981] in a much more general setting than the one I consider. Another, which seeks to find auctions that maximize social welfare, has also been studied more generally. Additionally, in several instances I was able to derive optima analytically. For all of these I have a known benchmark to strive for. Others have no known optimal design.

An important consideration in any optimization routine is the choice of a starting point, as it will generally have important implications for the quality of results. This could be especially relevant in practical applications of automated mechanism design, for example, if it is used as a tool to enhance an already working mechanism through parametrized search. Thus, I would already have a reasonable starting point and optimization could be far more effective as a result. I explore this possibility in several of my applications, using a previously studied design as a starting point. Additionally, I apply my framework to every application with completely randomly seeded optimization runs, taking the best result of five randomly seeded runs in order to alleviate the problem posed by local optima. Furthermore, I enhance the optimization procedure by using a *guided* restart, that is, by running the optimization procedure once using the current best mechanism as a new starting point.

In all of my applications, player types are independently distributed with uniform distribution on the unit interval. Finally, I used 50 samples from the type distribution to verify Ex-Interim-IR. This gives us 0.95 probability that 94% of types lose no more than the opportunity cost plus my specified tolerance which I add to ensure that the presence of noise does not overconstrain the problem. It turns out that every application that I consider produces a mechanism that is individually rational for all types *with respect to the tolerance level that was set.*

5.3.1 Myerson Auctions

The seminal paper by Myerson [1981] presented a theoretical derivation of revenue maximizing auctions in a relatively general setting. Here, my aim is to find a mechanism with a nearly-optimal value of some given objective function, of which revenue is an example.[10] However, I restrict myself to a considerably less general setting than did Myerson, constraining the design space to that described by the parameters $q, k_1, k_2, K_1, k_3, k_4$, and K_2 in (5.5).

$$u(t,a,t',a') = \begin{cases} U_1 & \text{if } a > a' \\ 0.5(U_1+U_2) & \text{if } a = a' \\ U_2 & \text{if } a < a', \end{cases} \tag{5.5}$$

where $U_1 = qt - k_1 a - k_2 a' - K_1$ and $U_2 = (1-q)t - k_3 a - k_4 a' - K_2$. I further constrain all the design parameters to be in the interval [0,1]. In standard terminology, this design space allows the designer to choose an allocation parameter, q, which determines the probability that the winner (i.e., agent with the winning bid) gets the good, and transfers, which I constrain to be linear in agents' bids.

While my automated mechanism design framework assures us that p-strong individual rationality will hold with the desired confidence, I can actually verify it by hand in this application. Furthermore, I can adjust the mechanism to account for lapses in individual rationality guarantees for subsets of agent types by giving to each agent the amount of the expected loss of the least fortunate type.[11] Similarly, if I do find a mechanism that is Ex-Interim-IR, I may still have an opportunity to increase expected revenue as long as the minimum expected gain of any type is strictly greater than zero.

Bayesian Mechanism Design Problems

Maximize Revenue In this section, I am interested in finding approximately revenue-maximizing designs in the specified constrained design space. Based on Myerson's feasi-

[10]Conitzer and Sandholm [2003] also tackled Myerson's problem, but assumed finite type and strategy spaces of agents, as well as a finite design space.

[11]Observe that such constant transfers will not affect agent incentives.

bility constraints, I derive in the following theorem that an optimal incentive compatible mechanism in the specified design space yields revenue of 1/3 to the designer,[12] as compared to 0.425 in the general two-player case.[13]

Lemma 5.16 *The mechanism in the design space described by the parameters in equation 5.5 is BNIC and Ex-Interim-IR if and only if* $k_3 = k_4 = K_1 = K_2 = 0$ *and* $q - k_1 - 0.5k_2 = 0.5$.

Theorem 5.17 *Optimal incentive compatible mechanism in my setting yields the revenue of 1/3, which can be achieved by selecting* $q = 1$, $k_1 \in [0, 0.5]$, *and* $k_2 \in [0, 1]$, *respecting the constraint that* $k_1 + 0.5k_2 = 0.5$.

Parameters	Initial Design	Final Design
q	random	0.96
k_1	random	0.95
k_2	random	0.84
K_1	random	0.78
k_3	random	0.73
k_4	random	0
K_2	random	0.53
objective	N/A	0.3

Table 5.8: Design that approximately maximizes the designer's revenue.

In addition to performing five restarts from random starting points, I repeated the simulated annealing procedure starting with the best design produced via the random restarts. This procedure yielded the design in Table 5.8. I now verify the Ex-Interim-IR and revenue properties of this design.

Proposition 5.18 *The design described in Table 5.8 is Ex-Interim-IR and yields the expected revenue of approximately 0.3. Furthermore, the designer could gain an additional 0.0058 in expected revenue without effect on incentives while maintaining the individual rationality constraint.*

[12]For example, Vickrey auction will yield this revenue.
[13]The optimal mechanism prescribed by Myerson is not implementable in my design space.

I have already shown that the best known design, which is also the optimal incentive compatible mechanism in this setting, yields a revenue of 1/3 to the designer. Thus, my AMD framework produced a design near to the best known. It is an open question what the actual global optimum is.

Maximize Welfare It is well known that the Vickrey auction is welfare-optimal. Thus, I know that the welfare optimum is attainable in the specified design space. Before proceeding with search, however, I must make one observation. While I am interested in welfare, it would be inadvisable in general to completely ignore the designer's revenue, since the designer is unlikely to be persuaded to run a mechanism at a disproportionate loss. To illustrate, take the same Vickrey auction, but afford each agent one billion dollars for participating. This mechanism is still welfare-optimal, but seems a senseless waste if optimality could be achieved without such spending (and, indeed, at some profit to the auctioneer). To remedy this problem, I use a minimum revenue constraint, ensuring that no mechanism that is too costly will be selected as optimal.

First, I present a general result that characterizes welfare-optimal mechanisms in my setting.

Theorem 5.19 *Welfare is maximized if either the equilibrium bid function is strictly increasing and $q = 1$ or the equilibrium bid function is strictly decreasing and $q = 0$. Furthermore, the maximum expected welfare in the specified design space is 2/3.*

Thus, for example, both first- and second-price sealed bid auctions are welfare-optimizing (as is well known).

In Table 5.9 I present the result of my search for optimal design with 5 random restarts, followed by another run of simulated annealing that uses the best outcome of 5 restarts as the starting point. I verified using the RW solver that the bid function $s(t) = 0.645t - 0.44$ is an equilibrium given this design. Since it is strictly increasing in t, I can conclude based on Theorem 5.19 that ***this design is welfare-optimal.*** I only need to verify then that both the minimum revenue and the individual rationality constraints hold.

Parameters	Initial Design	Final Design
q	random	1
k_1	random	0.88
k_2	random	0.23
K_1	random	0.28
k_3	random	0.06
k_4	random	0.32
K_2	random	0
objective	N/A	2/3

Table 5.9: Design that approximately maximizes welfare.

Proposition 5.20 *The design described in Table 5.9 is Ex-Interim-IR, welfare optimal, and yields the revenue of approximately 0.2. Furthermore, the designer could gain an additional 0.128 in revenue (for a total of about 0.33) without affecting agent incentives or compromising individual rationality and optimality.*

It is interesting that this auction, besides being welfare-optimal, also yields a slightly higher revenue to the designer than my mechanism in the previous section if I implement the modification proposed in Proposition 5.20. Thus, there appears to be some synergy between optimal welfare and optimal revenue in my design setting.

Robust Mechanism Design Problems

Maximize Nearly-Minimal Revenue The robust objective in this section is to maximize minimal revenue to the designer over the entire joint type space. That is, the robust objective function is

$$\begin{aligned}\inf_{t,t' \in T | s(t) > s(t')} & [k_1 s(t) + k_2 s(t') + k_3 s(t') + k_4 s(t)] + \\ & \inf_{t,t' \in T | s(t) < s(t')} [k_1 s(t') + k_2 s(t) + k_3 s(t) + k_4 s(t')] + \\ & \inf_{t,t' \in T | s(t) = s(t')} [(k_1 + k_2 + k_3 + k_4) s(t)] + K_1 + K_2.\end{aligned} \quad (5.6)$$

Assuming symmetry, here is a simple result about a set of mechanisms that yields 0 for the objective in Equation 5.6.

Theorem 5.21 *Any auction with $K_1 = K_2 = 0$ which induces equilibrium strategies in* *the form $s(t) = mt$ with $m > 0$ yields 0 as the value of the objective in Equation 5.6.*

Thus, both first-price and second-price sealed-bid auctions result in the value of 0 for the robust objective. Furthermore, by Lemma 5.16 it follows that the same is true for *any* BNIC and ex-interim individually rational mechanism in the specified design space.

Since it is far from clear what the actual optimum for this problem or for its probably approximately robust equivalent is, I ran my automated framework to obtain an approximately optimal design. In Table 5.10 I show the approximately optimal mechanism that

Parameters	Initial Design	Final Design
q	random	1
k_1	random	1
k_2	random	0.34
K_1	random	0.69
k_3	random	0
k_4	random	0
K_2	random	0
objective	N/A	0.0066

Table 5.10: Design that approximately robustly maximizes revenue.

results. I now verify its individual rationality and revenue properties.

Proposition 5.22 *The mechanism in Table 5.10 yields the value of 0.0066 for the robust objective. While it is not Ex-Interim-IR, it can be made so by paying each agent a fixed 0.000022, resulting in the adjusted robust objective value above 0.0065.*

Thus, I confirm that while not precisely individually rational, my mechanism is very nearly so, and with a small adjustment becomes individually rational with little cost to the designer. Furthermore, the designer is able to make a positive (albeit small) profit no matter what the joint type of the agents is.

5.3.2 Vicious Auctions

In this section I study a mechanism design problem motivated by the Vicious Vickrey auction [Brandt and Weiß, 2001; Brandt *et al.*, 2007; Morgan *et al.*, 2003; Reeves, 2005]. The essence of this auction is that while it is designed exactly like a regular Vickrey auction, the players get disutility from the utility of the other player, which is a function of parameter l, with the regular Vickrey auction the special case of $l = 0$.

I generalize the Vicious Vickrey auction design using the same parameters as in the previous section such that the Vicious Vickrey auction is a special case with $q = k_2 = 1$ and $k_1 = k_2 = k_3 = k_4 = K_1 = K_2 = 0$, and the utility function of agents presented in the previous section can be recovered when $l = 0$. I assume in this construction that payments, which will be the same (as functions of players' bids and design parameters) as in the Myerson auction setting, have a particular effect on players' utility parametrized by l. Hence, the utility function in (5.7).

$$u(t, a, t', a') = \begin{cases} U_1 & \text{if } a > a' \\ 0.5(U_1 + U_2) & \text{if } a = a' \\ U_2 & \text{if } a < a' \end{cases} \tag{5.7}$$

where $U_1 = q(1-l)t-(k_1(q(1-l)+(1-q))-(1-q)l)a-((1-q)l)t'-k_2(q(1-l)+(1-q))a'-K_1$ and $U_2 = (1-q)(1-l)t-(k_3((1-q)(1-l)+q)-ql)a-qlt'-k_4((1-q)(1-l)+q)a'-K_2$. In all the results below, I fix $l = 2/7$. Reeves [2005] reports an equilibrium for Vicious Vickrey with this value of l to be $s(t) = (7/9)t + 2/9$. Thus, we can see that we are no longer assured incentive compatibility even in the second-price auction case. In general, it is unclear whether there exist incentive compatible mechanisms in this design space, particularly because I constrain all the parameters to be in the interval $[0, 1]$.

Before proceeding, I would like to modify the definition of individual rationality in this setting to be as follows: every agent can earn non-negative expected value less expected payment (that is, expected surplus). To formalize this, $\mathrm{EU}(t) = v(t) - m(t) \geq 0$, where $v(t)$ is the expected value to agent with type t and $m(t)$ is the expected payment to the auctioneer by the agent with type t. This is in contrast with assuring each agent that every type will obtain non-negative expected utility. However, I believe that the alternative definition is more sensible in this setting, since it assures the agent that it will receive no less than the opportunity cost of participation in the auction, which we take to be zero.

Bayesian Mechanism Design Problems

Maximize Revenue The first objective is to (nearly) maximize revenue in this domain. The results of automated mechanism design in two distinct cases are presented in Table 5.11.

Parameters	Initial Design	Final Design
q	1	1
k_1	0	0
k_2	1	0.98
K_1	0	0.09
k_3	0	0.33
k_4	0	0
K_2	0	0
objective	0.48	0.49
q	random	1
k_1	random	1
k_2	random	0.33
K_1	random	0.22
k_3	random	0.22
k_4	random	0.12
K_2	random	0
objective	N/A	0.44

Table 5.11: Design that approximately maximizes revenue.

The top part of Table 5.11 presents the results of simulated annealing search that uses the previously studied Vicious Vickrey as a starting point. My purpose for doing so is two-fold. First, I would like to see if I can easily (i.e., via an automated process) do better than the previously studied mechanism. Second, I want to suggest automated mechanism design as a framework not only for finding good mechanisms from scratch, but also for improving mechanisms that are initially designed by hand. The latter could become especially useful in practice when applications are extremely complex and we can use theory and intuition to give us a good starting mechanism.

First, I determine the expected revenue and individual rationality properties of the Vicious Vickrey auction in the following Proposition.

Proposition 5.23 *The expected revenue from Vicious Vickrey auction with $l = 2/7$ is approximately 0.48. This auction is not Ex-Interim-IR, but can be adjusted by awarding* *each agent 0.021. The adjusted revenue would become 0.438.*

I now give the individual rationality and revenue properties of the auction that AMD obtains with Vicious Vickrey as the starting point.

Proposition 5.24 *The expected revenue from the auction {1,0,0.98,0.09,0.33,0,0} in Table 5.11 is approximately 0.49. This auction is Ex-Interim-IR, and will remain so if the designer charges a fixed entry fee of 0.0027, giving itself a total revenue of approximately 0.4932.*

Thus, I found a design which yields more revenue than the design previously studied in the literature (adjusted to be individually rational).

Now, I assume that I have never heard of Vicious Vickrey and need to find a good mechanism without any additional information. Consequently, I present results of search from a random starting point in the lower section of Table 5.11. Properties of the resulting auction are explored in Proposition 5.25.

Proposition 5.25 *The expected revenue from the auction {1,1,0.33,0.22,0.22,0.12,0} in Table 5.11 is approximately 0.44. This auction is Ex-Interim-IR, and can remain so if the designer charges all agents an additional fixed participation fee of 0.0199. This design change would increase the expected revenue to 0.4798.*

Thus, the design I obtained from a completely random starting point yields revenue that is not far below that of Vicious Vickrey (or the design that I found using Vicious Vickrey as a starting point), and is better than Vicious Vickrey if the latter is adjusted to be individually rational. Furthermore, this design can be improved considerably via a participation tax without sacrificing individual rationality.

Maximize Welfare In Table 5.12 I present an outcome of the automated mechanism design process with the goal of maximizing welfare. The procedure, as above, involves simulated annealing with five random restarts, and an additional run with the current optimum welfare as the starting point of the simulated annealing run. In the optimization, I utilized both the Ex-Interim-IR and minimum revenue constraints. In the following

Parameters	Initial Design	Final Design
q	random	0.37
k_1	random	0.8
k_2	random	1
K_1	random	0.49
k_3	random	0.29
k_4	random	0.67
K_2	random	0.48
objective	N/A	0.54

Table 5.12: Design that approximately maximizes welfare.

proposition I establish the welfare, revenue, and individual rationality properties of this mechanism.

Proposition 5.26 *The expected welfare of the mechanism in Table 5.12 is approximately 0.54 and expected revenue is approximately 0.225. It is Ex-Interim-IR for all types in [0.17,1] and can be made Ex-Interim-IR for every type at an additional loss of 0.13 in revenue.*

While individual rationality does not hold for almost 80% of types, this failure is easy to remedy at some additional loss in revenue (importantly, the adjusted expected revenue will be positive).

After a sequence of successful applications of AMD, I stand before an evident failure: the mechanism I found is quite a bit below the known optimum of $2/3$. Interestingly, recall that the optimal revenue mechanism in the vicious setting had a strictly increasing bid function and $q = 1$, and consequently was also welfare-optimal by Theorem 5.19.

Instead of plainly dismissing this application as a failure, I can perhaps derive some lessons as to why my results were so poor. I hypothesize that the most important reason is that I introduced minimum revenue of 0 as an additional hard constraint. From observing the optimization runs in general, I notice that the optimization problem both in the Myerson auctions and the vicious auctions design space seems to be rife with islands of local optima in the sea of infeasibility. Thus, the problem was difficult for black-box optimization already, and I only made it considerably more difficult by adding additional infeasible regions. In general, I would expect such optimization techniques to work best when the objective function varies smoothly and most of the space is feasible. Hard constraints make it more difficult by introducing (at least in my implementation) spikes in the objective value.[14]

I have seen some evidence to the correctness of my hypothesis already, since my revenue-optimal design also happens to maximize social utility. To test my hypothesis directly, I remove minimum revenue as a hard constraint in the next section, and instead try to maximize the weighted sum of welfare and revenue.

Maximize Weighted Sum of Revenue and Welfare In this section, I present results of AMD with the goal of maximizing the weighted sum of revenue and welfare. For simplicity (and having no reason for doing otherwise), I set weights to be equal. A

Parameters	Initial Design	Final Design
q	random	1
k_1	random	0.51
k_2	random	1
K_1	random	0.09
k_3	random	0.34
k_4	random	0.26
K_2	random	0
objective	N/A	0.6372

Table 5.13: Design that approximately maximizes the average of welfare and revenue.

design that my framework found from a random starting point is presented in Table 5.13. I verified using RW that $s(t) = 0.935t - 0.18$ is an (approximate) symmetric equilibrium bid function. Thus, by Theorem 5.19 this auction is welfare-optimal.

Proposition 5.27 *The expected revenue from the auction in Table 5.13 is 0.6078. However, it is not Ex-Interim-IR, and the least fortunate type loses nearly 0.044. However, by compensating the agents the designer can induce individual rationality without affecting incentives, at a revenue loss of 0.088. This would leave it with an adjusted expected revenue of 0.5198.*

[14]Recall that I implemented hard constraints as a very low value of the objective. Thus, adding hard constraints increases nonlinearity of the objective function, and the increase could be quite dramatic.

Robust Mechanism Design Problems

Maximize Nearly-Minimal Revenue I now apply my framework to the problem of robustly maximizing revenue of the designer. First, I present the result for the previously studied Vicious Vickrey auction.

Proposition 5.28 *By running the Vicious Vickrey auction, the designer can obtain at least 2/9 (approximately 0.22) in revenue for any joint type profile. By adjusting to make the auction individually rational, minimum revenue falls to 220/1089 (approximately 0.2).*

The results from running my automated design framework from a random starting point are shown in Table 5.14. I now verify the revenue and individual rationality properties of this mechanism.

Parameters	Initial Design	Final Design
q	random	0.86
k_1	random	1
k_2	random	0.71
K_1	random	0.14
k_3	random	0
k_4	random	0.09
K_2	random	0
objective	N/A	0.059

Table 5.14: Design that approximately robustly maximizes revenue.

Proposition 5.29 *The design in Table 5.14 yields revenue of at least 0.059 to the designer for any agent type profile, but is not ex-interim individually rational. It can be made such if the designer awards each agent 0.0135 for participation, yielding the adjusted revenue of 0.032.*

As we can see, the randomly generated design is considerably worse than the adjusted Vicious Vickrey. However, adjusted Vicious Vickrey requires negative settings of several of the design parameters. Since the parameters are initially constrained to be non-negative, it is unclear whether a better solution is indeed attainable in the specified constrained design space, even at a slight (< 0.02) sacrifice in individual rationality.

5.4 Conclusion

I presented a framework for automated design of general mechanisms (direct or indirect) using the Bayes-Nash equilibrium solver for infinite games developed by Reeves and Wellman [2004]. Results from applying this framework to several design domains demonstrate the value of our approach for practical mechanism design. The mechanisms that I found were typically either close to the best known mechanisms, or better.

My lone failure illuminated the difficulty of the automated mechanism design problem when too many hard constraints are present. After modifying the problem by eliminating the hard minimum revenue constraint and using multiple weighted objectives instead, we were able to find a mechanism with the best values of *both* objectives yet seen.

While in principle it is not at all surprising that we can find mechanisms by searching the design space—as long as we have an equilibrium finding tool—it was not at all clear that any such system will have practical merit. I presented evidence that mechanism design in a constrained space can indeed be effectively automated on somewhat realistic design problems that yield very large games of incomplete information. Undoubtedly, real design problems are vastly more complicated than any that I considered (or any that can be considered theoretically). In such cases, I believe that my approach could offer considerable benefit if used in conjunction with other techniques, either to provide a starting point for design, or to tune a mechanism produced via theoretical analysis and computational experiments.

Part II

One Nash, Two Nash, Red Nash, Blue Nash

CHAPTER 6

Related Work on Computational Game Theory

IN WHICH *I review some of the literature on computational game theory.*

6.1 Complexity of Computing Nash Equilibria

I begin the review of literature on computational Game Theory by citing some basic complexity results about computing Nash equilibria. The results are uniformly negative: there is considerable evidence that no efficient algorithm will be found to compute Nash equilibria. The complexity is worst case, and the algorithms which I describe in the following sections can be effective on many classes of games. Nevertheless, experience suggests that large unstructured games are indeed very difficult to solve.

The most direct result from the complexity theory is that the task of computing a single Nash equilibrium is *PPAD-complete* [Chen and Deng, 2006; Daskalakis *et al.*, 2006]. While not very much is yet known about *PPAD*, it is believed (with considerable evidence) that polynomial algorithms are unlikely to exist for problems in this class.

Gilboa and Zemel [1989] and Conitzer and Sandholm [2008] present a series of negative results about computing Nash equilibria with particular properties. For example, it is shown that the problem of determining the existence of a Nash equilibrium with social welfare of at least k is *NP-Hard*, and the problem of determining the number of Nash equilibria is *#P-Hard*. Additionally, Conitzer and Sandholm [2008] show that Nash equilibrium problems that involve optimization (e.g., optimizing social welfare) are even hard to approximate.

The results for Bayesian games are, naturally, even stronger. For example, the problem of determining whether there is a pure-strategy Bayes-Nash equilibrium is known to be *NP-Complete* [Conitzer and Sandholm, 2008].

While Conitzer and Sandholm [2008] demonstrate that computing Nash equilibria in symmetric games is no easier in the worst case than the general problem, there are many reasons to believe that symmetry yields advantages in practice. First, symmetric games can be represented much more compactly than general games. Additionally, symmetric equilibria always exist in finite games [Cheng *et al.*, 2004], and in computing such equilibria we need not worry about agent identity: we can search for equilibria in the space of single-agent strategies rather than all joint strategy profiles.

6.2 Solving Finite Two-Person Games

One of the most surprising results about complexity of Nash equilibrium computation is that two-player games are no easier to solve in the worst case than general m-player games. This is particularly strange considering the fact that special algorithms have been developed for two-player games and are known in practice to be much more effective than algorithms for general games.

There is, however, one class of two-player games which is known to be "easy" (that is, polynomial-time algorithms exist for solving games in this class): zero-sum games. The reason is that due to their special structure, zero-sum games can be solved by linear programming. Thus, I begin the discussion of algorithms for Nash equilibrium computation with the linear program formulation for zero-sum games.

6.2.1 Zero-Sum Games

Let $\{U_1, U_2\}$ be the payoff matrices of the two players (player 1 and player 2 respectively) and $\{A_1, A_2\}$ their respective pure strategy sets. Additionally, let x and y denote the mixed strategies of players 1 and 2 respectively. By the minimax theorem [Neumann and Morgenstern, 1980], the problem of solving a two-player zero-sum game can be formulated as the following program

$$\begin{aligned}
&\max_x \min_y x^T U_1 y \\
&s.t.: \\
&\sum_{i \in A_1} x_i = 1 \quad \text{and} \quad \sum_{j \in A_2} y_j = 1 \\
&x_i \geq 0, y_j \geq 0 \; \forall i \in A_1, j \in A_2.
\end{aligned}$$

Let $z = \min_y x^T U_1 y$. Note that since computing the value of z for a fixed x constitutes a decision problem for the second player, no randomization is necessary and player 2 need only focus on finding a "best response" (equivalently, a strategy which yields the greatest disutility to the competitor) among his pure strategies.

Thus, the subproblem of computing z for a fixed value of x can be reformulated as $z = \min_{j \in A_2} \sum_{i \in A_1} u_1^{ij} x_i$, where u_1^{ij} indicates the payoff to the first player when he plays his ith pure strategy and player 2 plays his jth, whereas x_i is the ith component of the first player's mixed strategy (that is, x_i is the probability that player 1 will play his ith pure strategy under x).

Since the minimum is taken over a finite set A_2 of strategies by player 2, the subproblem of computing z can relaxed to a set of constraints that $z \leq \sum_{i \in A_1} u_1^{ij} x_i \;\; \forall j \in A_2$, with the task of finding the greatest z satisfying these constraints. Consequently, we can rewrite the entire optimization program as the following linear program:

$$\max_x z$$
$$s.t.:$$
$$z \leq \sum_{i \in A_1} u_1^{ij} x_i \ \ \forall j \in A_2$$
$$\sum_{i \in A_1} x_i = 1; \quad x_i \geq 0 \, \forall i \in A_1$$

6.2.2 General Two-Person Games with Complete Information

The Lemke-Howson Algorithm

The most widely used method for solving general two-person games is the Lemke-Howson algorithm, which is, more generally, an algorithm for solving *linear complementarity problems (LCP)* [Lemke and Howson, 1964]. A *LCP* is characterized by vector of non-negative variables, $z \geq 0$, a set of linear constraints, $Cz = q$, and a partition of variables in z into two subvectors of equal length $\{z_a, z_b\}$ such that z_a and z_b are complementary in the sense that $z_a^T z_b = 0$. The implication of this complementarity condition is at most one of each respective component of z_a and z_b can be non-zero. For following discussion, I assume without loss of generality that all constraints are equality constraints.[1]

The Lemke-Howson algorithm for solving *LCP* is in many ways similar to the simplex method for solving linear programs [Nocedal and Wright, 2006]. In both, the algorithm follows a sequence of basic solutions, where the subsequent basis is generated from the previous basic solution by a procedure called *pivoting*. Concretely, suppose that the constraint matrix C has full rank (assuming some non-degeneracy conditions [von Stengel, 2002]) and let β be the basis which spans its column space. Then C_β (that is, the matrix composed only of basis columns) is a square matrix which is invertible and $C_\beta z_\beta = q$ has a unique basic solution, z_β^*, for the corresponding basic variables. All the variables z_j for $j \notin \beta$ are set to 0. Via linear algebra, we can obtain the following relationship between

[1]This is at no loss in generality, since inequality constraints can be handled by introducing *slack* variables to “fill” the inequality “gaps”.

basic and non-basic variables:

$$z_\beta = C_\beta^{-1} q - \sum_{j \notin \beta} C_\beta^{-1} C_j z_j.$$

The pivot step selects an entering variable and an exiting variable such that the new set of indeces generates another basic solution, which is also *feasible* in the sense that $z \geq 0$. In linear programming, an entering variable is selected to improve the value of the objective. In *LCP*, it is selected as a step towards finding a complementary solution. For both, the exiting variable is chosen to maintain feasibility. *LCP* algorithm proceeds by selecting an entering variable to be the complement of the last exiting variable, a process that is ultimately guaranteed to arrive at a complementary solution, albeit in the worst case after an exponential number of pivoting operations.

To formulate the problem of finding a Nash equilibrium in an arbitrary two-player game as a *LCP*, consider first the best response optimization problem for each player, which can be formulated as a linear program. Let $\{U_1, U_2\}$ be the $K \times L$ payoff matrices of players 1 and 2 choosing between K and L strategies respectively. Suppose that the mixed strategy y of player 2 is fixed. The best response of player 1 to y is the solution to the program

$$\max_{x \in \mathbb{R}^K} x^T (U_1 y) \;\; s.t. \;\; \sum_{k=1}^{K} x_k = 1, x_k \geq 0.$$

The dual of this LP is

$$\min_{u \in \mathbb{R}} u \;\; s.t. \;\; U_2 y \leq u 1_K,$$

where 1_K is a column vector of K ones. The constraint in the dual will bind for all strategies k which are player 1's best responses y and, naturally, all of these must yield the identical optimal payoff u. An analogous program can be formulated for player 2 for a fixed strategy x of player 1.

By strong duality of LP, a feasible optimal x and a dual optimum u must satisfy $x^T U_1 y = u$. Since $x^T 1_K^T = 1$, $x^T U_1 y = x^T 1_K u$. Rewriting, we obtain a complementarity condition $x^T (1_K u - U_1 y) = 0$. Since we know that $x^T \geq 0$ and $Ay \leq u 1_K$ by the primal and dual constraints, the complementarity condition implies that at most one of each element x_i of the vector x and each corresponding element $(1_K u - U_1 y)_i$ of the vector $1_K u - U_1 y$ can be positive. Repeating the same argument for player 2, we can obtain the following linear complementarity problem with variables x, y, u, and v:

$$
\begin{aligned}
& x^T(1_K u - U_1 y) = 0 \\
& y^T(1_L v - U_2^T x) = 0 \\
& x^T 1_K = 1 \\
& y^T 1_L = 1 \\
& 1_K u - U_1 y \geq 0 \\
& 1_L v - U_2^T x \geq 0 \\
& x \geq 0, y \geq 0.
\end{aligned}
\tag{6.1}
$$

By the arguments advanced above, a pair (x, y) is a solution of the matrix game if and only if it is a feasible solution of the LCP 6.1.[2] The corresponding variables u and v will then yield the equilibrium payoffs to both players.

We can simplify the problem slightly by noting that u and v are unnecessary to finding the equilibria: all we need is the condition that all the best response strategies for both of the players yield identical payoffs. The program can thus be reformulated as follows:

$$
\begin{aligned}
& x'^T r = 0 \\
& y'^T s = 0 \\
& A y' + r = 1_K \\
& B^T x' + s = 1_L \\
& x' \geq 0, y' \geq 0.
\end{aligned}
\tag{6.2}
$$

where r and s are slack variables used to convert the inequality constraints to equality constraints and $\{x', y'\}$ are real variables that can be mapped into the simplex of mixed strategies by normalization to yield the corresponding equilibrium mixed strategies. Note that either $(U_1 y')_k = 1$ or $r_k > 0$ and, similarly, either $(U_2^T x')_l = 1$ or $s_l > 0$.

[2]This is true if you consider the point $(x, y) = (0, 0)$ as an artificial Nash equilibrium.

The algorithm can be initialized with $(x', y') = (0, 0)$ (thus, $(r, s) = (1_K, 1_L)$), which is an artificial equilibrium. In the first step, an arbitrary k or l is removed from the basis. After an initial pivoting step, some index, say, k, will as a result fail the complementarity condition, since the basis will contain both $r_k > 0$ and $x'_k > 0$. By following a sequence of pivoting steps, however, a complementary solution and, thus, an equilibrium, will eventually be found. The procedure may be repeated by selecting different entering element in the first step, possibly computing different equilibria.

MIP Formulation

Sandholm *et al.* [2005] observe that the problem of computing a Nash equilibrium (or the entire set of Nash equilibria) can be characterized by a mixed integer program which seeks $s_i(a_i), v_i, v(a_i), b(a_i)$ such that

$$\begin{aligned}
&\sum_{a_i \in A_i} s_i(a_i) = 1 \quad \forall i \in I \\
&u_{a_i} = \sum_{a_{-i} \in A_{-i}} s_{-i}(a_{-i}) u_i(a_i, a_{-i}) \quad \forall i \in I, a_i \in A_i \\
&v_i \geq v(a_i) \quad \forall i \in I, a_i \in A_i \\
&v_i - v(a_i) \leq U_i b(a_i) \quad \forall i \in I, a_i \in A_i \\
&0 \leq s_i(a_i) \leq 1 - b(a_i) \quad \forall i \in I, a_i \in A_i \\
&v_i \geq 0, v(a_i) \geq 0, b(a_i) \in \{0, 1\}.
\end{aligned}$$

The interpretation of the variables is that s is the mixed strategy Nash equilibrium, v_i is the expected payoff of every strategy in the support of s, $v(a_i)$ is the (identical) expected payoff of the pure strategy a_i, and $b(a_i)$ is 1 if and only if the strategy a_i is played with probability 0 under s_i. The advantage of the MIP formulation is that any linear objective can now be added to this program to obtain Nash equilibria that maximize some desirable objective (e.g., social welfare).

6.3 Solving General Games

6.3.1 Simplicial Subdivision

A classical algorithm for solving general finite games is simplicial subdivision. This algorithm takes advantage of the fact that Nash equilibrium is a fixed point of the following function, defined on a joint simplex of mixed strategies:

$$y_{ij}(s) = \frac{s_{ij} + g_{ij}(s)}{1 + \sum_j g_{ij}(s)},$$

with

$$g_{ij}(s) = \max[u_i(a_{ij}, s_{-i}) - u_i(s), 0].$$

Here, I introduce for simplicity the notation that a_{ij} denotes a pure strategy j of player i; y_{ij} and g_{ij} have a corresponding interpretation. The function $y_{ij}(s)$ is a continuous function which maps a joint strategy simplotope (a product of simplexes of mixed strategies for all players) to itself. By Brouwer fixed point theorem, y has a fixed point, that is, there is a mixed strategy profile s such that $y(s) = s$. Furthermore, s is a fixed point of y iff s is a Nash equilibrium [Nash, 1951]. Consequently, any algorithm for finding or approximating a fixed point of a continuous map can be used to approximate a Nash equilibrium. The best known such algorithm was introduced by Scarf [1967]. A detailed description of this algorithm and its application in the context of computing Nash equilibria is provided by McKelvey and McLennan [1996].

6.3.2 The Liapunov Minimization Method

As an alternative to a fixed point problem, Nash equilibrium computation in finite games can be cast as global minimization [McKelvey, 1998]. Consider the function

$$h(s) = \sum_{i \in I} \sum_{a_i \in A_i} \max[u_i(a_i, s_{-i}) - u_i(s), 0]. \tag{6.3}$$

The following result, proven by McKelvey [1998], demonstrates that the problem of finding (or approximating) global minima to $h(s)$ is equivalent to finding (approximating) Nash equilibria.

Theorem 6.1 *The function $h(s)$ defined in Equation 6.3 is everywhere non-negative and differentiable. Furthermore, $h(s) = 0$ iff s is a Nash equilibrium.*

It is useful that $h(s)$ is everywhere differentiable, since gradient-based methods may be used to numerically minimize it. Unfortunately, it is not everywhere twice-differentiable, so a straightforward application of the Newton method, which assumes existence of a Hessian at every point, need not work well, even in local minimization. Additionally, since most well-understood numerical optimization techniques converge to a local solution, random restarts are necessary to guarantee global convergence.

As an alternative to the Newton method and other local non-linear optimization techniques, the Nedler-Mead algorithm [Nocedal and Wright, 2006] has proven effective for minimizing $h(s)$ in some settings (particularly, in symmetric games) [Reeves, 2005; Cheng *et al.*, 2004].

6.3.3 The Govindan-Wilson Algorithm

A relatively recent technique for computing sample Nash equilibria in general games was introduced by Govindan and Wilson [2003].[3] This technique relies heavily on the structure theorem in Kohlberg and Mertens [1986], who demonstrate that the graph of the Nash equilibrium correspondence is homeomorphic to the set of all games.

The homeomorphism of Kohlberg and Mertens can be constructed as follows. Let G denote a game and Γ a set of all games for a fixed set of pure strategy profiles. While it appears that Γ is a subset of $\mathbb{R}^K$ with K the size of the product space of pure strategy profiles, we will see shortly that it can actually be identified with $\mathbb{R}^L$ where $L = \sum_{i \in I} |A_i|$, that is, L is the total number of strategies of all players. For the presentation below, it will be convenient to collect all the pure strategies into a vector of size L, with each dimension j representing a strategy j of player i. Let $g \in R^L$ be a vector such that g_j is

[3]This technique was initially introduced for normal form games. Its use in extensive form games is presented in Govindan and Wilson [2002].

the expected payoff to player i for playing the corresponding strategy $j \in A_i$ when the remaining players play a uniform mixture over their pure strategies. It turns out that a game G can then be represented by a tuple $(\tilde{G}, g)$ such that

$$\sum_{t \in A_{-i}} \tilde{G}_{j,t} = 0$$

for every $j \in A_i$ [Govindan and Wilson, 2003]. The representations are equivalent because for any mixed strategy profile $s \in S$, the expected payoff of a pure strategy j of player i can be expressed by

$$v_j(s) = \sum_{t \in A_{-i}} G_{j,t} \prod_{n \neq i} s_n(t_n) = g_j + \sum_{t \in A_{-i}} \tilde{G}_{j,t} \prod_{n \neq i} s_n(t_n).$$

Let E denote the graph of the Nash equilibrium (N.E.) correspondence, that is $E = \{(G, s)|s \text{ is a N.E.}\} = \{(\tilde{G}, g, s)|s \text{ is a N.E.}\}$ and denote by H a homeomorphism between E and Γ. Specifically, let $H(\tilde{G}, g, s) = (\tilde{G}, z), z \in R^L$, defined by $z_j = s_j + v_j(s)$. In words, the homeomorphism H transforms s into z by adding to each element of the mixed strategy (that is, to the probability of playing a strategy j under the Nash equilibrium strategy s) the expected payoff from playing j when others play s. It is clear that this map is continuous. To prove that H is indeed a homeomorphism we must demonstrate that it has an inverse H^{-1} which is also continuous.

To obtain the inverse H^{-1}, we need to make use of the *retraction map* $r(z)$, mapping a vector $z \in R^L$ to a mixed strategy nearest to z in Euclidean distance [Gul *et al.*, 1993]. For a given $z \in R^L$, let $v_i(z)$ for a player $i \in I$ be a number such that $\sum_{j \in A_i}(z_j - v_i(z))^+ = 1$, where $(x)^+ = \max\{x, 0\}$. Then $r(z) = s$ such that $s_j = (z_j - v_i(z))^+$ for all $i \in I, j \in A_i$. Now, suppose that z is produced using the homeomorphism function H. Then the interpretation of the retraction map is that $v_i(z)$ is the payoff to all strategies of player i which are best responses to s. As such, $v_i(z) > v_j(s)$ for all j that are not best responses to s. Since a strategy j which is not a best response to s is played with probability 0 when s is a Nash equilibrium, $z_j = v_j(s)$ and, consequently, $(z_j - v_i(z))^+ = 0$ whenever j is not a best response to s. On the other hand, the payoff to every best response strategy is equal when s is a Nash equilibrium, and subtracting this payoff from every corresponding z_j will thus yield the probability of playing j in the Nash equilibrium s.

By the above argument, $r(z)$ provides one part of H^{-1} and is clearly continuous. Once we know the Nash equilibrium s of the game G, which we obtain by applying the retraction map to z, the corresponding g can be recovered by

$$g_j = z_j - s_j - \tilde{G}(r(z)),$$

where $\tilde{G}(s) = \sum_{t \in A_{-i}} \tilde{G}_{j,t} \prod_{n \neq i} s_n(t_n)$.

For the purposes of the algorithm, it will be sufficient to maintain a fixed $\tilde{G}$ and, consequently index the set of games by g. In other words, let $\Gamma(\tilde{G}) = \{(\tilde{G}, g) | g \in \mathbb{R}^L\}$. Note that $\Gamma(\tilde{G})$ can be identified with $\mathbb{R}^L$ and, thus, $r(z)$ can be viewed as a retraction map from $\Gamma(\tilde{G})$ to S. To make use of that, let $h : \Gamma(\tilde{G}) \to \Gamma(\tilde{G})$ be defined by

$$h(z) = z - r(z) - \tilde{G}(r(z)),$$

that is, h assigns to a vector z the vector of expected payoffs g which makes $r(z) = s$ a Nash equilibrium of $(\tilde{G}, g)$.

Now, suppose we would like to compute a Nash equilibrium of the game $G^* = (\tilde{G}^*, g^*)$ and suppose further that another game, $(\tilde{G}^*, \bar{g})$, has a unique Nash equilibrium $\bar{s}$ which is easy to compute. Define the function $\psi(z)$ by

$$\psi(z) = h(z) - g^*.$$

Since $h(z)$ corresponds to a game for which z can be retracted into a Nash equilibrium, $-\psi(z)$ can be interpreted as the displacement of this game from the game of interest, g^*. As such, $\psi(z^*) = 0$ implies that $r(z^*)$ is a Nash equilibrium of g^*.

Since zeros of the function $\psi(z)$ correspond to the Nash equilibria of g^*, the goal of the algorithm is to compute some zero of $\psi(z)$, z^*. Once z^* is found, we can apply $r(\cdot)$ to invert the homeomorphism and, thus, determine the Nash equilibrium of g^*. Various methods can be used to compute the zeros of a system of equations. The algorithm proposed by Govindan and Wilson [2003] uses a global Newton method developed by Hirsch and Smale [1979]. The idea is to begin with the game $g(0) = \bar{g}$ and its solution $z(0) = \bar{z}$ and trace the homotopy between $\bar{g}$ and g^* together with the corresponding solutions to obtain $g(1) = g^*$ and $z(1) = z^*$.

6.3.4 Logit Equilibrium Path Following Method

Another homotopy-based method, first mentioned in McKelvey and Palfrey [1995] and expanded on by Turocy [2005], makes use of the properties of Quantal Response Equilibria (QRE) [McKelvey and Palfrey, 1995]. A QRE of a game G is a Bayes-Nash equilibrium of another game in which each player's payoff function from G is independently perturbed. Specifically, let $v_{ij}(s)$ be the expected payoff to player i for playing a pure strategy j when others play a mixed strategy s. Rather than observing $v_{ij}(s)$, however, players observe

$$\hat{v}_{ij}(s) = v_{ij}(s) + \eta_{ij},$$

where η_{ij} are random variables drawn from some joint distribution of noise. McKelvey and Palfrey [1995] demonstrate the existence of a QRE in the induced game and provide a characterization of equilibrium strategies when η_{ij} come from an extreme value distribution. In that case,

$$s^*_{ij} = \frac{e^{\lambda v_{ij}(s)}}{\sum_{k \in A_i} e^{\lambda v_{ik}(s)}}.$$

These equilibria are referred to as *logit equilibria* and are parametrized by λ, which is a parameter of the noise distribution with the property that when $\lambda = 0$ noise dominates payoffs, while when $\lambda = \infty$, noise is negligible. The two extremes imply that the logit equilibrium for $\lambda = 0$ is a profile of uniformly fully mixed strategies (i.e., the centroid of the mixed strategy simplex for all players), whereas the logit equilibrium for $\lambda = \infty$ is a Nash equilibrium of the underlying game.

Define the function

$$H_{ij}(s,\lambda) = e^{\lambda v_{ij}(s)} - s \sum_{k \in A_i} e^{\lambda v_{ik}(s)}$$

and note that s^* which makes $H_{ij}(s^*, \lambda) = 0$ is a logit equilibrium of the game characterized by λ. Since H_{ij} defines a homotopy between $H_{ij}(s, 0)$ and $H_{ij}(s, \infty)$, we can use homotopy path following methods to compute the Nash equilibrium given by the zero of $H_{ij}(s, \infty)$.

6.3.5 Search in the Pure Strategy Support Space

An entirely different approach to finding a single Nash equilibrium was proposed by Porter *et al.* [2006]. The idea behind their algorithm is to perform a heuristic search in the space of possible Nash equilibrium supports. In the worst case this could take quite some time, since the space of supports grows exponentially in the number of players (for example, the joint strategy space for a game in which 5 players can each choose from 5 strategies is 3125, but the space of all possible supports is the powerset of these, amounting to 2^{3125}). Practically, however, the hope is that in most games of interest there are Nash equilibria with relatively small supports.

Since the authors are searching in the space of supports, they need a subroutine which determines whether there is a Nash equilibrium for a fixed support and, if so, computes one such equilibrium. The authors term this subroutine a *feasibility program*. Formally, suppose that $R_i \subset A_i$ is a fixed support. The feasibility program attempts to find $s \in S$ and a vector of corresponding equilibrium payoffs $v = \{v_1, \dots, v_m\}$ such that the following constraints hold:

$$
\begin{aligned}
&\forall i \in I, a_i \in R_i, \quad \sum_{a_{-i} \in R_{-i}} s_{-i}(a_{-i}) u_i(a_i, a_{-i}) = v_i \\
&\forall i \in I, a_i \notin R_i, \quad \sum_{a_{-i} \in R_{-i}} s_{-i}(a_{-i}) u_i(a_i, a_{-i}) \le v_i \\
&\forall i \in I, \quad \sum_{a_i \in R_i} s_i(a_i) = 1 \\
&\forall i \in I, a_i \in R_i, \quad s_i(a_i) \ge 0 \\
&\forall i \in I, a_i \notin R_i, \quad s_i(a_i) = 0,
\end{aligned}
\tag{6.4}
$$

where $s_{-i}(a_{-i}) = \prod_{j \neq i} s_j(a_j)$. Note that the feasibility program is linear when there are only two players and non-linear when the number of players is above two (its polynomial degree is one less the number of players, determined by the degree of $s_{-i}(a_{-i})$).[4]

The algorithm searches in the space of supports with a bias towards small balanced supports. It attempts to prune each support using iterated strict dominance. For each support which survives pruning, the feasibility program above is solved. If a Nash equilibrium exists for the given support, it is returned and the procedure terminates. Otherwise, the algorithm proceeds to heuristically select the next set of support strategies. This very simple algorithm is shown to be extremely effective in finding a single Nash equilibrium in a broad class of games.

6.4 "Learning" Algorithms

All the algorithms above guarantee that they will converge either to a local solution or to an actual Nash equilibrium, and even those with local convergence properties may be modified to guarantee completeness in probability. Below I describe several algorithms which have very limited convergence guarantees, but perform quite well on certain types

[4]We can observe that this program is quite similar to the MIP formulation in Section 6.2.2. Indeed, in two-player games the search method described in this section can be viewed as a heuristic for finding a feasible solution to the MIP. This insight can actually be useful to provide a lower bound for any specific optimization problem over Nash equilibria, such as the problem of finding a socially optimal Nash equilibrium.

of problems. All of these have been broadly classified as “learning” or “adaptive” algorithms, as they have historically been used to model ways in which agents may come to play a Nash equilibrium [Fudenberg and Levine, 1998].

6.4.1 Iterative Best Response

One of the best-known and most effective learning dynamics is “iterative best response”. The idea is that in each round the players play their best responses to the strategies that their opponents used in the previous round. It is clear that if this dynamics ever reaches a fixed point, such a fixed point would be a Nash equilibrium.

For ease of exposition, I describe the procedure for symmetric profiles (i.e., profiles in which all players play the same strategy):

1. Generate an initial symmetric profile r_0
2. Find best response, $\hat{r}$, to the current profile r_k
3. Set $r_{k+1} = \hat{r}$ and go back to step 2

There are two ways that the above procedure may be generalized to asymmetric settings. One method would have *all* players play their best response strategies to opponent play in the previous round. Another would have only one player (selected randomly or sequentially) best respond, but maintain the strategies of all others as they had been in the previous round.

The technique of iterative best response dynamics aimed at equilibrium computation is well-known, with Cournot duopoly being, perhaps, the most famous application. Under very restrictive assumptions (e.g., in supermodular games with unique Nash equilibria [Milgrom and Roberts, 1990] and in congestion games [Monderer and Shapley, 1996]) iterated best response is known to converge to a Nash equilibrium.

6.4.2 Fictitious Play

Another widely used algorithm for learning in games is *fictitious play*, in which all players best respond in each round to the empirical distribution over the strategies of their opponents.[5]

Formally, let $N_{-i}(a_{-i})$ denote the number of times a pure strategy profile $a_{-i} \in A_{-i}$ has been played by the players other than i before the current round. The presumed mixed strategy for the opponents is then characterized by the probabilities

$$s_{-i}(a_{-i}) = \frac{N_{-i}(a_{-i})}{\sum_{t \in A_{-i}} N_{-i}(t)}.$$

Player i then plays the best response to s_{-i}, that is, a strategy

$$a_i \in \arg\max_{a \in A_i} u_i(a, s_{-i}).$$

If there are multiple best response strategies, an arbitrary one is chosen.

Strict pure strategy Nash equilibria are absorbing states of fictitious play, and any pure strategy steady state must be a Nash equilibrium. Additionally, if empirical distributions converge, the corresponding mixed strategies constitute a mixed strategy Nash equilibrium. However, fictitious play is not generally convergent [Fudenberg and Levine, 1998].

6.4.3 Replicator Dynamics for Symmetric Games

Replicator dynamics was originally introduced as a model of evolution of strategies in a homogeneous population [Fudenberg and Levine, 1998]. The idea behind this model is that strategies which yield higher payoffs in pairwise encounters are more likely to be replicated by other players. Specifically, suppose that we have a homogeneous population of players and a (symmetric) strategy set A. The interactions between agents proceed in rounds, with each agent in the population adopting in every round a single strategy $a \in A$. We begin in the first round with some arbitrary distribution of strategies in A represented in the agent pool. The agents accrue payoffs by playing their respective strategies in a sequence of random pairwise encounters. Given the expected payoffs from such encounters in any given round g, the fraction of the agent pool representing a strategy

[5] Such behavior can be supported by a belief that opponents play a stationary mixed strategy.

a in the next round (that is, in round $g+1$) is proportional to a's relative expected payoff in round g.

Formally, let $p_g(a)$ denote the proportion of the population playing a strategy a in generation (round) g. Then,

$$p_g(a) \propto p_{g-1}(a)(Eu(a) - W),$$

where W is a payoff lower bound and $Eu(a)$ is the expected (symmetric) payoff of strategy a against the symmetric mixed strategy p_{g-1} corresponding to the population in round $g-1$.

It is not difficult to see that every symmetric Nash equilibrium of the game is a fixed point of replicator dynamics. The converse is only partially true, however: every interior fixed point (i.e., a fixed point with no extinct strategies) is a Nash equilibrium [Friedman, 1991]. Fortunately, any steady state that is a limit of a path that originates in the interior is a Nash equilibrium [Fudenberg and Levine, 1998]. On the other hand, and somewhat unfortunately, replicator dynamics need not converge to a steady state at all.

6.5 Solving Games of Incomplete Information

From the discussion above it should by now be intuitive that solving a game of incomplete information—even when it is one-shot—is a task substantially more difficult in general than solving complete information games. In principle, however, there is representational equivalence in that any game of incomplete information, just as a game of complete information, may be represented in normal form: simply allow strategies of players to be functions of types (and histories, if relevant), and let the payoff entries be the corresponding expected payoffs with respect to the type distributions. In practice, since the number of strategies for player i is then $O(|T_i|^{|A_i|})$, even relatively small Bayesian games can be quite intractable.

Needless to say, solving infinite games of incomplete information is an even more daunting task. Below, I describe some of the attempts in recent literature to approximating solutions to infinite Bayesian games. The common thread to these is to restrict strategy spaces and solve (or approximate Nash equilibria to) the games defined by the restricted strategy spaces. The hope is that the resulting approximations are reasonable in the context of the actual games of interest.

6.5.1 Restricted Strategy Spaces and Convergence

One promising approach to approximating solutions to Bayesian games, which is closely related to the techniques described in Chapter 9, was introduced by Armantier and Richard [2000] and Armantier *et al.* [2007]. The central idea behind this approach is to restrict strategy spaces of all players to be finite dimensional (for example, using polynomial approximations). The authors call the corresponding solution concept a *Constrained Strategic Equilibrium (CSE)* and demonstrate its convergence to a Bayes-Nash equilibrium in certain settings [Armantier *et al.*, 2007].

From the practical standpoint, these papers propose a series of numerical techniques for computing CSEs based on solving a system of non-linear equations representing the joint first-order optimality conditions for all players. Since formulating such equations requires smoothness of the expected utility functions, they resort to kernel function approximations to achieve smoothness where necessary.

6.5.2 Empirical Game Theory

Motivated in part by the work of Armantier *et al.* [2007] came a series of papers attempting to approximate solutions to extremely complex games such as simultaneous ascending auctions [Reeves *et al.*, 2005; Osepayshvili *et al.*, 2005], bidding strategies in a Trading Agent Competition scenario [Wellman *et al.*, 2006], and a four-player chess game [Kiekintveld *et al.*, 2006]. In each of these, the authors isolated a strategic element (usually a parameter) and attempted to approximate a Nash equilibrium in the resulting restricted strategy space, estimating payoffs of the induced normal-form game (or its restriction to a finite set of strategies) using Monte-Carlo. Since the current work is building on these empirical game theoretic methods, I discuss them in more detail in

6.6 Compact Representations of Games

Since finite games are notoriously difficult to solve, as attested to both by experience and worst-case computational complexity results above, computer scientists have for many years been interested in structured models of games which yield compact representations and are more computationally tractable. Below, I discuss three important graphical representations of games. The first and last of these are compact representations of one-shot (normal-form) games, whereas the second is a compact representation of games in extensive form.

6.6.1 Graphical Games

One of the earliest compact models of games was introduced by Kearns *et al.* [2001]. The idea behind this work is that a graphical model may be used to represent payoff dependencies between the players. Specifically, a graphical game is defined by a pair (G, u), where G is an undirected graph on vertices representing players in I and u is the set of payoff functions u_i. Each player i is represented by a vertex $i \in G$, and its *neighbors* $N_G(i) \subseteq I$ are vertices (players) $j \in G$ such that there exists an undirected edge $(i, j) \in G$. The utility function u_i of player i is then defined only on the set of joint strategies of players in $N_G(i)$ (which includes i by convention).

Kearns *et al.* [2001] describe two algorithms when a graphical game is represented by a tree: an exact exponential-time algorithm for computing all Nash equilibria of the game and a polynomial-time algorithm which approximates all equilibria. Ortiz and Kearns [2003] introduce algorithms for solving general graphical games. Singh *et al.* [2004] extend the graphical game model to games of incomplete information and demonstrate how to approximate Bayes-Nash equilibria based on these. Soni *et al.* [2007] connect general graphical games to constraint satisfaction problems (CSPs) and show how the CSP techniques can be used to compute approximate Nash equilibria.

6.6.2 Multi-Agent Influence Diagrams

Multi-agent influence diagrams (MAIDs) [Koller and Milch, 2003] follow in the footsteps of graphical models for probability distributions (Bayes Nets) and for single-agent decision settings (influence diagrams) [Russell and Norvig, 2003]. The key contribution of MAIDs is to model the relationship between the decision variables of all agents. Specifically, the strategic relevance relationship between decision nodes is represented by a relevance graph, in which a directed edge from one decision node to another indicates that this decision variables relies on the other. Koller and Milch [2003] introduce a sound and complete algorithm for determining strategic relevance between nodes and develop an algorithm for computing Nash equilibria which makes use of the strategic relevance graph.

6.6.3 Action-Graph Games

A compact representation which is somewhat related to both of the above is the action-graph game representation (AGG) [Bhat and Leyton-Brown, 2004]. Unlike graphical games (but similarly to MAIDs), action-graph games use nodes in a graph to represent strategies and edges to represent dependencies between particular strategic choices of the players. The incoming edges to a node a defines a's *neighborhood*, with all the neighbors of a being, thus, of some strategic relevance to it. Additionally, symmetry within the game can be represented by assigning to each node (i.e., to each strategy) a particular number of agents that can select it. The utilities of the players from selecting a particular strategy a then depends only on the numbers of agents selecting from the remaining strategies which are neighbors of a. Bhat and Leyton-Brown [2004] describe how to compute the expected utility and the Jacobian based on an action-graph representation of a game, which can then be used by standard methods to compute or approximate Nash equilibria of the game. Jiang and Leyton-Brown [2006] present a polynomial-time algorithm for computing expected payoffs in AGGs and extend the AGG representation to take advantage of additional structure in utility functions.

CHAPTER 7

Simulation-Based Game Modeling: Formal Notation and Finite-Game Methods

IN WHICH *I formalize the notions of simulation-based and empirical games and present methods, convergence analysis, and probabilistic confidence bounds for these game models when strategy sets of all players are finite.*[1]

In this chapter I begin the description of methods for analyzing simulation-based games. In the context of my overall work, which centers around the mechanism design problem, this would fall under the *mechanism analysis* rubric. Specifically, in Part I, I focus on the problem of mechanism design *search* and take for granted the ability to predict the outcomes of strategic interactions between players (participants) for any mechanism choice. In this and the following chapters, on the other hand, I assume a *fixed* choice of mechanism and attempt to answer the associated prediction question—that is, given a game model, I attempt to predict the outcomes of strategic interactions between players. The methods discussed in Part II can thus be viewed as *subroutines* to a mechanism design search algorithm.

Below, I begin by presenting formal notation for simulation-based and empirical games, both of which are instrumental concepts in the remaining chapters.

[1] Some of the material in this chapter is taken from Vorobeychik *et al.* [2006] and Wellman *et al.* [2005].

7.1 Simulation-Based and Empirical Games

Recall the basic model of a normal-form game, $\Gamma_N = [I, R, u(\cdot)]$, where I is the set of players, R the joint strategy set (pure, mixed, restricted, etc), and $u(\cdot)$ the function that maps joint strategy profiles $r \in R$ to a vector of payoff entries for all players.

A *simulation-based game* retains all the basic elements of the normal-form game, but makes the notion of utility functions $u(\cdot)$ somewhat more precise in a way pertinent to analysis. Specifically, a utility function in a simulation-based game is represented by an *oracle*, $\mathcal{O}$, which can be queried with any strategy profile $r \in R$ to produce a possibly noisy sample payoff vector U.

Formally, a simulation-based game is denoted by $\Gamma_S = [I, R, \mathcal{O}]$, where the oracle (simulator) $\mathcal{O}$ produces a sample vector of payoffs $U = (U_1, \ldots, U_m)$ to all players for a (usually pure) strategy profile r. I assume throughout that $E[U] = u(s)$. I denote an estimate of a payoff for profile r based on k samples from $\mathcal{O}$ by $\hat{u}_k(r) = \frac{1}{k}\sum_{j=1}^{k} U(r)^j$, where each $U(r)^j$ is generated by invoking the oracle, with profile r as input.

One key distinction between $\mathcal{O}$ and $u(\cdot)$ is, thus, that $u(\cdot)$ is presumed to provide easy access to exact payoff evaluations, whereas $\mathcal{O}$ evaluates payoffs with noise. Another distinction, no less vital but somewhat more subtle, is that by specifying the payoff function as an oracle, we in practice resign ourselves to the fact that payoffs are not available in any easily tractable form and the game must of necessity be analyzed using simulation experiments. Thus, for example, even though the payoff function may have a closed-form specification, the Nash equilibria of the game (or some useful qualitative properties thereof) cannot be obtained using analytic means.

A useful derivative of a simulation-based game is an *empirical game*, in which the payoff function is replaced by a data set, $D = \{(r_1, U_1), \ldots, (r_n, U_n)\}$, where r_j are profiles in R and U_j are possibly noisy samples from the oracle of the corresponding simulation-based game. I denote an empirical game by $\Gamma_{\mathcal{E}} = [I, R, D]$. Let $U_D(r)$ be the set of all payoffs in the data for a particular profile r, that is, $U_D(r) = \{U | (r, U) \in D\}$.

Implicit to the discussion of both the simulation-based games and empirical games is that they are defined with respect to an *underlying* game $\Gamma_{N \leftarrow S}$ characterized by the set of players I, a set of strategy profiles R, and the payoff function $u(\cdot)$ from which the oracle, in effect, is taking noisy samples. Given $\Gamma_{N \leftarrow S}$, the *true* regret of a profile $r \in R$ in both the simulation-based and the empirical game is evaluated with respect to $u(\cdot)$ of this underlying game. I do, however, introduce below *empirical regret*, which will be evaluated with respect to the information available in an empirical game data set.

7.2 A General Procedure for Solving Simulation-Based Games

Let us define a *game solver* to be an algorithm that takes an empirical game $\Gamma_{\mathcal{E}}$ as input and produces a set of probability distributions over strategy profiles as its output.

Suppose that we are given a simulation-based game and our mission (should we choose to accept it) is to estimate its solutions. The following outlines the general steps we can take towards this goal:

1. Generate an *empirical game* $\Gamma_{\mathcal{E}}$ by sampling the simulator (oracle) $\mathcal{O}$ to obtain a data set D of profile-payoff tuples (r, U).

2. Apply a game solver to $\Gamma_{\mathcal{E}}$ to obtain a set of solutions $F_1, \ldots, F_k$, where F_i is a probability distribution over profiles in R.

Several remarks are in order. First, I thus far have used the term *solutions* to games very vaguely, making it concrete only when I meant by it Nash equilibria. This is the first time I made the idea of a solution precise (as I use it in this work in any case): a solution is a set of distributions over profiles. Observe that this definition clearly captures Nash equilibria, trivially so if you consider that R can be comprised of *mixed strategy profiles*. Allowing for distributions over these, I anticipate the later discussion of *belief distributions of play* (see Chapter 11), that is, distributions which may incorporate Nash equilibria into beliefs, but may rely on other exogenous and endogenous information. Multiplicity of solutions is natural, since many solution concepts, including Nash equilibria, are not generally unique.

7.3 Simulation Control

Step 1 in the above routine abstractly suggests that the analyst produce an empirical game by sampling payoffs from the simulator. It does not, however, provide any guidance about how this should be done. The simplest idea, of course, is to sample profiles uniformly randomly from R (assuming that R is bounded). While asymptotically reasonable, such a procedure is not generally most effective in yielding the best answers given a limited number of samples. Consider, for example, a game in which there is a profile with extremely low payoffs relative to other profiles. Only a few samples will be necessary to have a high probability hypothesis that this profile is not a part of any Nash equilibrium. Consequently, one would not want to waste any more sampling actions on this profile, especially if each sampling action is quite expensive. To generalize, one should sample profiles selectively in such a way as to maximize the analyst's expected utility with respect to the ultimate outcomes. This utility would typically depend on the error in solution estimates: the higher the expected error, the lower the expected utility, and minimizing the expected error maximizes expected utility. In these cases, profiles should be selected for sampling in order to minimize the expected error in the solution estimates.

Russell and Wefald [1991] suggest a general principled approach to assess the relative value of computation actions, which in the setting of simulation control would translate to actions of taking samples from a simulator. I denote the current information state by S and the new information state that results from performing a computation action a by $S.a$.[2] The latter may be (and usually is) a random variable. I denote the analyst's *estimated* utility function under information state S by $\hat{U}^S$. Note that this utility is estimated since in general the actual utility is unknown. Finally, I let $x(S)$ denote an optimal computation choice with respect to $\hat{U}^S$. Under this setup, Russell and Wefald [1991] define

[2] I am overloading S in this context to be consistent with the notation of Russell and Wefald [1991]. Momentarily, I will switch notation to discuss the specific application of their ideas to simulation control in our setting.

the *expected value of computation* action a by

$$EVC(a|S) = E_{S.a|S}[\hat{U}^{S.a}(x(S.a)) - \hat{U}^{S.a}(x(S))].$$

In words, $EVC(a|S)$ evaluates how much the computation action a improves the estimated utility under the new information state. It is important to note that EVC is computed with respect to the utility estimate $\hat{U}^{S.a}$, thereby ignoring the effect of the computation action a on improving the estimate of the utility function. As a result, $EVC(a|S) \geq 0$ for any a. Note, however, that EVC does not in any way capture the quality of the utility estimate itself. Naturally, the better the utility estimate, the better choices we may expect to make in selecting an ***EVC*** maximizing action, although at an increased computational cost.

The adaptation of EVC to the problem of selecting the next sample from a game simulator was first proposed by Walsh *et al.* [2003] and dubbed EVI or *expected value of information*. Their first step is to define the information state to be an empirical game, $\Gamma_{\mathcal{E}}$, and utility estimate to be a measure of Nash equilibrium approximation error with respect to the empirical game, $\hat{f}_{\Gamma_{\mathcal{E}}}(\cdot)$. EVI is then defined in this setting as follows:

$$EVI(r|\Gamma_{\mathcal{E}}) = E_{\Gamma_{\mathcal{E}}.r|\Gamma_{\mathcal{E}}}[\hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}})) - \hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}}.r))],$$

where $\Gamma_{\mathcal{E}}.r$ designates an empirical game obtained by adding to $\Gamma_{\mathcal{E}}$ a payoff sample (or a constant number of samples) for the profile r.

A key assumption of EVC and its adaptation to the problem of taking payoff samples from a game simulator is the *single-step* assumption, that is, that the current sampling choice will be the last that the analyst intends to make. Alternatively, we may imagine that the analyst is choosing complete sampling sequences. Either assumption is unrealistic, the former because it is only actually true for the last sample taken, and the latter because it is an infeasible problem to consider all complete computation sequences, and seems a bit of an overkill in making an optimal choice for just one step. Additionally, the computational burden of EVI even in the case of one-step sampling may be non-trivial, since it could involve estimating expectations for a large number of profiles $r \in R$. I now observe, however, that when our error measure $\hat{f}(\cdot)$ is game-theoretic regret evaluated with respect to the empirical game payoffs, $\epsilon_{\Gamma_{\mathcal{E}}}(r)$, we need not be concerned about computing EVC for the entire profile space R, but for only a (generally) rather small subset of it.

As a first step, I define the error function to be with respect to *every* Nash equilibrium estimate in $\Gamma_{\mathcal{E}}$, that is, with respect to the set $\mathcal{N}(\Gamma_{\mathcal{E}})$ of equilibrium estimates. First, recall the definition of game-theoretic regret from Chapter 2. Extending the notion of regret to empirical games, let me define *empirical regret* of a profile r:

$$\epsilon_{\Gamma_{\mathcal{E}}}(t) = \max_{i \in I} \max_{b \in D_i(t)} [u_i(b, r_{-i}) - u_i(r)]. \tag{7.1}$$

In this expression, $D_i(t)$ is the set of profiles in the data set of the empirical game $\Gamma_{\mathcal{E}}$ with only player i deviating from t. Let the set of solutions to the empirical game $\mathcal{N}(\Gamma_{\mathcal{E}})$ to be the set of regret-minimizing profiles on the restricted strategy space R. Define the error of solutions to empirical game $\Gamma^1_{\mathcal{E}}$ with respect to another empirical game (perhaps, one with a better estimate of the payoff function), $\Gamma^2_{\mathcal{E}}$, to be

$$e(\Gamma^1_{\mathcal{E}}, \Gamma^2_{\mathcal{E}}) = \frac{1}{|\mathcal{N}(\Gamma^1_{\mathcal{E}})|} \sum_{t \in \mathcal{N}(\Gamma^1_{\mathcal{E}})} \epsilon_{\Gamma^2_{\mathcal{E}}}(t).$$

Using this notation, EVI becomes

$$EVI(r|\Gamma_{\mathcal{E}}) = E_{\Gamma_{\mathcal{E}}.r|\Gamma_{\mathcal{E}}}[e(\Gamma_{\mathcal{E}}, \Gamma_{\mathcal{E}}.r) - e(\Gamma_{\mathcal{E}}.r, \Gamma_{\mathcal{E}}.r)].$$

Now, assume that every $t \in \mathcal{N}(\Gamma_{\mathcal{E}})$ has zero game theoretic regret evaluated with respect to the game $\Gamma_{\mathcal{E}}$, that is, $\epsilon_{\Gamma_{\mathcal{E}}}(t) = 0$. If we impose no restrictions on our strategy set R, allowing it to be the set of all mixed strategy profiles, all solutions in $\mathcal{N}(\Gamma_{\mathcal{E}})$ are guaranteed to have zero regret and, thus, the assumption will necessarily hold. This need not be true, however, if R is restricted, for example, to the set of pure strategy profiles, as the empirical game may not possess a pure strategy Nash equilibrium. I do, however, expect my assumption usually to hold (at least approximately), particularly when many profiles (and, thus, many deviations) remain unsampled. Under this assumption,

$e(\Gamma_{\mathcal{E}}.r, \Gamma_{\mathcal{E}}.r) = 0$ for any $\Gamma_{\mathcal{E}}.r$, and EVI simplifies to

$$EVI(r|\Gamma_{\mathcal{E}}) = E_{\Gamma_{\mathcal{E}}.r|\Gamma_{\mathcal{E}}}[e(\Gamma_{\mathcal{E}}, \Gamma_{\mathcal{E}}.r)] = \frac{1}{|\mathcal{N}(\Gamma_{\mathcal{E}})|} \sum_{t \in \mathcal{N}(\Gamma_{\mathcal{E}})} E_{\Gamma_{\mathcal{E}}.r}[\epsilon_{\Gamma_{\mathcal{E}}.r}(t)].$$

Now, consider a profile r which is neither an equilibrium estimate (i.e., $r \notin \mathcal{N}(\Gamma_{\mathcal{E}})$) nor a neighbor to an equilibrium estimate (i.e., $r \notin D_i(t) \ \ \forall i \in I, \ \ \forall t \in \mathcal{N}(\Gamma_{\mathcal{E}})$). By sampling this profile, $E_{\Gamma_{\mathcal{E}}.r}[\epsilon_{\Gamma_{\mathcal{E}}.r}(t)]$ will remain unchanged for every $t \in \mathcal{N}(\Gamma_{\mathcal{E}})$, since r has no bearing on it. Consequently, $EVI = 0$ for every such $r \in R$. Since $EVI \geq 0 \forall r \in R$, we need only to consider profiles r that are either current solutions or their neighbors. Generically, the number of solutions to $\Gamma_{\mathcal{E}}$ is very small compared to the size of the game, and the number of neighbors is linear in the size of $\mathcal{N}(\Gamma_{\mathcal{E}})$, in the number of players I and in the number of player strategies R_i, whereas the size of R is, of course, exponential in the number of players.

While EVI is a somewhat principled approach to the problem of sampling, it is still heuristic for several reasons. First, as I already noted, it is only as good as the utility estimate $\hat{U}$ given the information state. If the utility estimate is suboptimal (or does not use all the information), EVI will naturally be suboptimal also, even given the assumptions above. In my description, computing EVI with respect to the game-theoretic regret on an empirical game $\Gamma_{\mathcal{E}}$ is suboptimal, since it does not use the distributional information. Note that what the analyst would ultimately care about in our setting is the regret as computed on the *underlying game*. Thus, we may actually be able to use the distributional information (if available) to estimate the best solutions (Nash equilibria) relative to the distribution of underlying games. Below I discuss methods to derive distributions of game-theoretic regret under various distributional assumptions about noise, and these would naturally have a bearing on how Nash equilibria of the underlying game may be estimated based on the information in an empirical game $\Gamma_{\mathcal{E}}$. Second, the single-step assumption is clearly false in our setting most of the time. Additionally, the computational cost of EVI, while alleviated considerably by restricting the set of profiles that need to be considered, is still non-trivial, since expectation needs to be estimated using Monte Carlo for a possibly large number of profiles.

The work by Walsh *et al.* [2003] indeed used an error metric that accounted for the distributional information in the game. Specifically, they let $f(r)$ denote the Nash equilibrium approximation error for a profile r as computed on the underlying game. Rather then using game-theoretic regret for this, they used a related notion of summed deviation benefits, that is:

$$f(r) = \sum_{i \in I} \sum_{a \in R_i} \max(0, u_i(a, r_{-i}) - u_i(r)).$$

Since only the empirical game is available in our context, the error must actually be defined as the expectation of $f(r)$ with respect to the empirical game payoffs: $f_{\Gamma_{\mathcal{E}}}(r) = E_{\Gamma_{\mathcal{E}}}[f(r)]$. This expectation is, of course, not possible to compute in general, and their estimated error $\hat{f}_{\Gamma_{\mathcal{E}}}(r)$ was defined as the sample average of the errors computed based on Monte Carlo samples from the distribution of underlying games. In order to approximate the solution $x(\Gamma_{\mathcal{E}})$, they computed the set of Nash equilibria of the empirical game and subsequently selected the equilibrium with the lowest estimated error. While this may yield a reasonable approximation of the best solution $x(\Gamma_{\mathcal{E}})$ with respect to their error metric, it clearly need not actually be the best. As a result, such a computation of EVI is fundamentally approximate and need no longer retain the property that $EVI(r) \geq 0$ for every profile r.

Note that EVI as just described is highly computationally intensive, requiring the computation of the entire set of Nash equilibria many times during the evaluation sequence for a single profile r. As a result, Walsh *et al.* [2003] propose an alternative with a much lower computational overhead, which they call $ECVI$ (*expected confirmation value of information*):

$$ECVI(r|\Gamma_{\mathcal{E}}) = E_{r|\Gamma_{\mathcal{E}}}[\hat{f}_{\Gamma_{\mathcal{E}}}(x(\Gamma_{\mathcal{E}})) - \hat{f}_{\Gamma_{\mathcal{E}} \cdot r}(x(\Gamma_{\mathcal{E}}))].$$

In words, this computes the amount by which sampling r reduces the expected error of a current solution $x(\Gamma_{\mathcal{E}})$. While the authors demonstrate the efficacy of this heuristic using simulations, there are problems with it. For illustration, suppose that $\hat{f}_{\Gamma_{\mathcal{E}}}(\cdot)$ is the game theoretic regret defined with respect to the empirical game $\Gamma_{\mathcal{E}}$. It is then clear that any Nash equilibrium is an optimum with respect to this error measure and, furthermore,

$\hat{f}_{\Gamma_{\mathcal{E}}}(x(\Gamma_{\mathcal{E}})) = 0$ and $\hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}}.r)) = 0$ as I already explained. Thus, $EVI(r)$ collapses to $E_{r|\Gamma_{\mathcal{E}}}[\hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}}))]$ and $ECVI(r)$ becomes $-E_{r|\Gamma_{\mathcal{E}}}[\hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}}))]$. As a result, the algorithm which uses $ECVI$ with the game theoretic regret as the error estimate will inevitably make precisely the worst decision with respect to EVI! While Walsh *et al.* [2003] use a somewhat different error measure, the problem just described seems quite fundamental, particularly if $\hat{f}_{\Gamma_{\mathcal{E}}}(x(\Gamma_{\mathcal{E}}))$ and $\hat{f}_{\Gamma_{\mathcal{E}}.r}(x(\Gamma_{\mathcal{E}}.r))$ are close to zero (at least relative to the other term), as they may well be when the number of samples is large. The intuition thus provided is indeed corroborated in the experiments of Jordan *et al.* [2008], which suggest that the performance $ECVI$ is quite poor in some settings.

Several alternative heuristic approaches for guiding sampling in finite games have been introduced in addition to the $ECVI$ heuristic: one by Sureka and Wurman [2005], and several by Jordan *et al.* [2008]. The approach of Jordan *et al.* [2008] is to select a profile that maximizes a Kullback-Leibler divergence, which is based on the distribution of minimum regret before and after sampling a chosen profile. The authors found that their *information-gain* heuristic typically outperformed $ECVI$ in their experiments.

The problem of sample selection is substantially easier if we consider a setting in which the simulator produces samples with no noise. In such a setting, the only question to be addressed is which profile that is not in our data set should be selected next. The earliest work known to me that addressed sample selection with noise-free simulation-based games is by Sureka and Wurman [2005], who introduce a *TABU best response* algorithm (or just *TABU*). In essence, their algorithm is an implementation of best response dynamics described in Chapter 6. However, it is well-known that the best response dynamics algorithm is only convergent in a very limited array of settings [Milgrom and Roberts, 1990; Monderer and Shapley, 1996]. In general, best response dynamics will end in a cycle and, consequently, is not complete even in finite games. To make it complete, Sureka and Wurman [2005] add every profile that has been sampled to a *TABU* list and do not allow any profile from the *TABU* list to be selected.

Another algorithm in the noise-free setting, *min-regret-first search (MRFS)*, is described and applied in Chapter 4. The *MRFS* algorithm proceeds from an arbitrary starting point to sample a profile which is a neighbor to some (arbitrary) approximate Nash equilibrium solution of the current empirical game. Jordan *et al.* [2008] analyzed both *MRFS* and *TABU* and found that *MRFS* tends to perform well according to the game-theoretic regret metric. The reason for this is, perhaps, that in the noise-free setting and given the regret error measure, *MRFS* is nearly equivalent to EVI—equivalent if we assume that solutions have zero regret with respect to the empirical game, as above. With this assumption, I already demonstrated that under EVI only solutions or neighbors need to be considered as sampling actions. When there is no sampling noise, there is no need to sample solution profiles again, since the values of the players' utilities at these profiles are already known exactly. Furthermore, the neighbors that have already been sampled need not be sampled again for the same reason. Therefore, the only profiles that need to be considered are the neighbors to solutions which have not been added to the data set of the empirical game, and that is precisely what *MRFS* would prescribe.

The approaches for guiding simulations discussed thus far assume that the game is finite. If the game is infinite, the problem is considerably more complex. For example, it is impossible in an infinite game to ever empirically confirm a profile to be a Nash equilibrium, even if payoffs contain no noise. Additionally, the probability (even posterior) that any particular profile is a Nash equilibrium will generically be zero. A technique like iterative best response, however, may still in principle be applicable, albeit it must inherently make use of approximate best responses in this setting. I discuss the problem of Nash equilibrium approximation in infinite games in Chapter 8, and present heuristic techniques for exploring the profile space of infinite games in order to approximate best response and Nash equilibria in Chapter 9.

7.4 Approximating Nash Equilibria in Empirical Games

Whatever method may be used to obtain a data set of profile-payoff tuples from the simulation-based game model, our ultimate goal is to estimate or approximate Nash equilibria of the underlying game. In this section, I first formally explore the meaning of Nash equilibrium approximation, and then present methods for estimating equilibria based on empirical games.

7.4.1 Nash Equilibrium Approximation Metrics

In general, no method is likely to be able to pinpoint actual Nash equilibria if the problem is complex enough and samples are noisy. Thus, the task is inherently that of approximation. There are two ways that approximate Nash equilibria may be defined: the first based on distance or norm, and the second based on game-theoretic regret, $\epsilon(r)$. Since Nash equilibria are not unique, there is a possible ambiguity in defining the distance between a profile r and a Nash equilibrium. I resolve this ambiguity by defining the distance from the profile r to a *set* of Nash equilibria using the *Hausdorff distance* from a point to a set.

Definition 7.1 *Let M be a metric space and $d(\cdot, \cdot)$ the associated distance metric. A* Hausdorff distance *from a point $r \in M$ to a set $K \subset M$, denoted $h(r, K)$ is defined to be*

$$h(r, K) = \inf_{t \in K} d(r, t).$$

In words, *Hausdorff distance* is the distance from r to the closest point in the set K (which exists if K is compact in M). The distance-based approximation may then be defined as follows:

Definition 7.2 *A profile r is a d-*approximate-Nash equilibrium *if $h(r, \mathcal{N}(\Gamma)) \leq d$, where $\mathcal{N}(\Gamma)$ is a set of Nash equilibria of the game Γ.*

The notion of distance-based approximation is useful to a mechanism designer who is only interested in actual Nash equilibrium solutions and would like to assess alternative designs. If we posit that the objective function is continuous in outcomes, that is $W(r, \theta)$ is continuous in r, then good distance-based approximations imply good approximations of the objective at some equilibrium. This approach, however, yields a philosophical difficulty due to the multiplicity of equilibria: how can the designer be sure that the players will choose to play the particular Nash equilibrium which is closest to the approximation r? If the Nash equilibrium can be computed exactly, such a philosophical problem can be at least in principle solved by suggesting that the designer presents a solution together with the design. Then, by the definition of a Nash equilibrium, no player will have an incentive to deviate, since all will presumably believe that this profile chosen by the designer will be played by others. In the approximation setting, the best that the designer can offer is his approximation. To address this philosophical difficulty, the designer may attempt to approximate the entire set of Nash equilibria. The relevant approximation metric is the Hausdorff distance between two sets, K and L.

Definition 7.3 *Let M be a metric space, $d(\cdot,\cdot)$ the associated distance metric, and $h(\cdot,\cdot)$ a Hausdorff distance from a point to a set. The (undirected) Hausdorff distance between two sets, $K, L \subset M$ is*

$$H(K,L) = \max\{\sup_{s\in K} h(s,L), \sup_{t\in L} h(t,K)\}.$$

Then, the distance-based set approximation of Nash equilibria is defined as follows:

Definition 7.4 *A set of profiles R is a d-*approximate-Nash equilibrium set *if*

$$H(R, \mathcal{N}(\Gamma)) \leq d,$$

where $\mathcal{N}(\Gamma)$ is a set of Nash equilibria of the game Γ.

That is, no profile $r \in R$ is very far from some Nash equilibrium and no Nash equilibrium is very far from some profile in R. This is a very strong notion of approximation and in practice almost never attainable, even asymptotically. Consequently, in most applications weaker notions must suffice.

I suggested above that the philosophical difficulty exists with evaluating designs with respect to a profile which is close to some Nash equilibrium in terms of a distance (say, Euclidean distance). This difficulty may be alleviated if we suppose that such an approximation yields every player very little to gain by deviating to a best response, since presumably the action of computing a best response may then not be worth its cost. This idea gives rise to the second approximation notion based on regret, one that is most significant in this work and which has already been used in the preceding chapters.

Definition 7.5 *A profile r is an ϵ-*Nash equilibrium *if $\epsilon(r) \leq \epsilon$.*

In words, r is an ϵ-Nash equilibrium if its game-theoretic regret as previously defined is bounded by ϵ. This approximation measure focuses directly on the strategic stability of a profile rather than its actual proximity to a Nash equilibrium. As such, it captures, I believe, the vital flavor of what approximation means in the context of games: a profile that is nearly an equilibrium yields little incentive to deviate to any player. One standard philosophical justification I have already mentioned: if there is little incentive to deviate and some cost to finding an improving deviation, agents will not be expected to deviate at all. Practically, I believe that this idea is of great usefulness, since there is typically cost to finding improving deviations which is not directly modeled by the game. As a result, I use the regret function $\epsilon(r)$ as the equilibrium approximation metric throughout this thesis, with a few exceptions which I state explicitly.

7.4.2 Estimating Nash Equilibria in (Small) Finite Games

The most straightforward—although not necessarily optimal—method for estimating Nash equilibria in empirical games in which every pure strategy profile has at least one sample is the following:

1. Use the empirical game $\Gamma_{\mathcal{E}}$ to estimate the payoff matrix of the underlying game.

2. Numerically compute Nash equilibria of the estimated game (e.g., using the GAMBIT toolbox [McKelvey *et al.*, 2005]).

Note that I have not as yet specified how the empirical game can be used to estimate the payoff matrix of the underlying game.The most direct way is to construct the estimated payoff matrix by using sample average payoffs for every profile. That is, for every $r \in R$, let

$$\bar{u}(r) = \frac{1}{|U_D(r)|} \sum_{u \in U_D(r)} u.$$

Later in this chapter I develop some basic confidence bounds and convergence results for this method of estimating Nash equilibria.

A level of sophistication can be added if we use variance reduction techniques rather than sample means to estimate payoffs of the underlying game [Reeves, 2005; Wellman *et al.*, 2005]. For example, control variates, conditioning, and quasi-random sampling can achieve a considerable increase in sampling efficiency [Ross, 2001; L'Écuyer, 1994].

Observe that the method for estimating Nash equilibria I just described uses only a part of the available information. Specifically, it does not use any information about the sampling noise. Intuitively, if such information can be used, we can perhaps reduce estimation variance. I follow this intuition to define a *maximum likelihood equilibrium estimator (MLEE)* in Section 7.8 below, which uses probabilistic information gleaned from the prior noise distribution and empirical game data in producing a Nash equilibrium estimate.

7.4.3 Estimating Nash Equilibria in Large and Infinite Games

In estimating Nash equilibria based on empirical game data in small finite games, I took advantage of the fact that we can collect payoff samples for every strategy profile in the game, thereby constructing an estimate of the entire payoff matrix. The resulting empirical game can be viewed as a normal-form game *estimate*, and the Nash equilibria of the underlying game can be estimated by Nash equilibria of the estimated game.

When strategy sets of players are very large, the approach above, at least in its direct form, is infeasible, since we cannot obtain samples for every strategy profile of the game. Perhaps the easiest way to extend the above approaches to analysis of infinite games is by restricting the original strategy profile sets R to some small finite subsets $R_k \subset R$ (with the restriction that R_k is a cross-product of some player strategy sets) such that every strategy profile in R_k has at least one payoff sample. The downside to this approach is that typically the granularity of R_k must be quite small in order for us to estimate the entire payoff matrix based on simulations.

As an alternative, we can generalize the idea of approximation to games in which we do not have estimates of payoffs for some of the profiles. The key step to doing this is in generalizing the notion of game-theoretic regret to empirical games. I have done this above by defining *empirical regret* in Equation 7.1.

Estimating a Single Nash Equilibrium

Just as the regret function $\epsilon(r)$ can be used to characterize Nash equilibria and approximate Nash equilibria of the game, we can use the empirical regret as a proxy when only limited and noisy empirical data is available. The first step to using empirical regret in equilibrium approximation or estimation is to generalize the idea of approximate Nash equilibria to empirical games, particularly those in which only a small subset of all strategy profiles is contained in the data. This is done by defining *candidate* equilibria of empirical games.

Definition 7.6 *r is a* candidate δ-equilibrium *of the empirical game $\Gamma_{\mathcal{E}}$ if $\epsilon_{\Gamma_{\mathcal{E}}}(r) \leq \delta$.*

Thus, a candidate δ-equilibrium in an empirical game is analogous to a δ-Nash equilibrium in a normal-form game, and the two coincide when an empirical game simply represents a game in normal form.

Just as Nash equilibria minimize game-theoretic regret, we can use minimizers of empirical regret to estimate Nash equilibria based on empirical games.

Definition 7.7 *Let $\epsilon^*_{\Gamma_{\mathcal{E}}} = \min\{\epsilon_{\Gamma_{\mathcal{E}}}(r) : r \in D\}$. We refer to a candidate $\epsilon^*_{\Gamma_{\mathcal{E}}}$-equilibrium $r \in D$ as a* candidate Nash equilibrium estimate*.*

If the game is finite, R is a set of mixed strategies, and the empirical game contains the actual payoffs of the underlying game for every $r \in R$ (indeed, it is sufficient to have every payoff vector for the pure strategy profile space), $\epsilon^*_{\Gamma_{\mathcal{E}}} = 0$ and the corresponding set of minimizing profiles is the set of Nash equilibria of the underlying game. More generally, for any R, the set of minimizers will be profiles which are the best Nash equilibrium approximations in the joint strategy space.

An alternative approach to estimating Nash equilibria based on empirical games that is particularly valuable when the underlying game is infinite is to use machine learning to estimate the payoff function over the profile space, generalizing from profiles that have been sampled. If there are many players, the task of learning the payoff function over the space of individual player choices may be impractical. To alleviate this problem, we can use summarization functions to reduce the dimension of the payoff function domain.

Additionally, if the game is symmetric, we need worry about learning only a single payoff function, rather than one for each player. I explore the learning approach in detail in Chapter 8.

Estimating Sets of Nash Equilibria

The simplest extension of the above techniques to estimate the entire set of Nash equilibrium outcomes of the underlying game is to use the set of candidate Nash equilibria of the empirical game. Alternatively, we can use the set of all candidate δ-equilibria for some fixed δ towards this end. In practice, however, it is often more useful to estimate a *range* of equilibrium outcomes, particularly if we can use some summarization function to map profiles to a low-dimensional statistic. To obtain such estimates, we can first generate a set of candidate δ-equilibria and then use a *convex hull* (that is, the smallest convex set containing these) as our estimated "range". Consequently, in Definition 4.3, I introduced the following estimator for a set of (approximate) Nash equilibria.

Definition 7.8 *For a set K, define $Co(K)$ to be the convex hull of K. Let B_δ be the set of candidates at level δ. I define $\hat{\phi}^*(\theta) = Co(\{\phi(r) : r \in B_\delta\})$ for a fixed δ to be an estimator of $\phi^*(\theta)$, where $\phi(r)$ is some aggregation function which maps strategy profiles to a (possibly) lower-dimensional summary.*

In words, the estimate of a set of equilibrium outcomes is the convex hull of all (aggregated) strategy profiles with ϵ-bound below some fixed δ. I present some theoretical support for this method of estimating the set of Nash equilibria below.

7.5 Applications to TAC/SCM Analysis

In Chapter 4 I described the mechanism design problem in the context of TAC/SCM 2004. I now revisit the day-0 procurement story in its beginnings in TAC/SCM 2003, focusing on the strategic issues rather than on the design problem. For more details, refer to Section 4.1 and Appendix B.

As I mentioned in Section 4.1, in response to the aggressive day-0 procurement by

TAC agents, team Deep Maize introduced a *preemptive* strategy. The preemptive strategy consisted of two elements: (1) a large day-0 order for every component and (2) a short-term deadline. As it is impossible for suppliers to complete the order by the proposed deadline, the model specifies that the supplier will make two offers as a result. The first offer commits to deliver the entire quantity of the requested supplies, but on a new (considerably later) date. The second offers as much of the quantity of the component as can be produced by the specified deadline, given other outstanding offers. As a general rule, Deep Maize would reject the former and accept the latter offer. The key, however, is that the initial request is so large as to effectively preempt any component requests that the supplier would consider after Deep Maize: these can be at best completed towards the end of the game and, consequently, are not useful to the manufacturers. A more complete description of the preemptive strategy that is of interest here is presented in Appendix B.

The preemptive strategy worked quite well: the initial component orders were considerably more subdued on average and profits increased. Deep Maize team, while encouraged by the positive social effect of the preemptive strategy, did not, of course, introduce it exclusively for common good. There were strong selfish reasons too. One of these reasons was that the design of Deep Maize was particularly adept at dynamic optimization and did not focus very much on optimal day-0 procurement. As long as initial procurement strategy dominated the game, Deep Maize was unlikely to stand out. Once aggressive day-0 procurement was neutralized, however, the designers of Deep Maize believed that their agent would be much more effective.

7.5.1 Analysis of Day-0 Procurement in TAC/SCM 2003

In this section, I present game-theoretic analysis of procurement in TAC/SCM 2003 which suggests that jockeying for components at the start of the game was an outcome of rational decision making by the agents, rather than a transient result of irrational behavior.

To begin, I define the relevant elements of the simulation-based game: players, strategies, and the simulation-based payoff function. Two games are considered, one with five and another with six players, as described below. While the actual strategies employed by agents in a typical tournament game are extremely complex, for the purposes of this analysis it will suffice to focus on two distinct variants of Deep Maize: *aggressive* (A) and *baseline* (B), characterized by the particular day-0 procurement behavior.

In strategy A (aggressive), the agent requests large quantities of components from every supplier on day 0. The specific day-0 RFQs (requests-for-quote) issued correspond to aggressive day-0 policies observed for actual TAC/SCM 2003 participants. Strategy A randomly selects among these at the beginning of each game instance. In strategy B (baseline), the agent treats day 0 just like any other day, issuing requests according to its standard policy of serving anticipated demand and maintaining a buffer inventory [Kiekintveld *et al.*, 2004]. Both of these strategies follow the standard Deep Maize procurement policy after day 0.

The final element of the simulation-based game in this setting is the actual TAC/SCM simulation model, complete with the participating six agents. As required, each simulation (that is, a TAC/SCM game) produces a vector of payoffs for all players given their respective strategic choices.

Two versions of this game are considered in the analysis below. The first is an *unpreempted* six-player game, where agents are restricted to playing A or B. The second is a five-player game, with the sixth place taken up by a fixed agent playing the preemptive strategy. I refer to this as the *single-preemptor* game.[3]

Since the two strategies incorporate specified policies for conditioning on private information, the game is represented in normal form. By symmetry there are only seven distinct profiles for the unpreempted game, corresponding to the number j of agents playing A, $0 \leq j \leq 6$. There are six distinct profiles for the single-preemptor game. Payoffs for each profile represent the expected profits for playing A or B, respectively, given the other agents, with expectation taken over all stochastic elements of the game environment.

To estimate our game's expected payoff matrix and thereby generate the data set for

[3]The full six-player game where all agents are allowed to choose the preemptive strategy is considered in Wellman *et al.* [2005].

the corresponding empirical game, approximately 30 game instances were sampled for each strategy profile—about 400 in total. For each sample, the average profits were collected for the As and Bs, as well as the demand level, $\bar{Q}$. The control variates variance reduction method (using $\bar{Q}$ as the variate) was used for each strategy to estimate its payoff in that profile, thereby estimating the payoff matrix of the underlying game.

Based on the empirical game and the estimated payoff matrix which corresponds to the *unpreempted* setting, we can verify that increasing the prevalence of aggressiveness degrades aggregate profits. In contrast, the analysis of the *single-preemptor* game shows that inserting a single preemptive agent neutralizes the effect of aggressiveness, diminishing the incentive to implement an aggressive strategy, and also ameliorating its negative effects. Moreover, the presence of a preemptor tends to improve performance *for all agents* in profiles containing a preponderance of agents playing A.

To study the equilibrium behavior of each of the two versions of the game, asymmetric pure-strategy equilibria, as well as symmetric mixed-strategy equilibria, are computed for each of the games.[4] Comparison of the features of equilibrium behavior in the respective games confirms the tournament observations about the effects of strategic preemption.

Let iA denote the profile with no preemption and i agents playing A (the rest playing B). The unique pure-strategy Nash equilibrium in this game based on the estimated payoff matrix is 4A. That this is an equilibrium can be seen by comparing adjacent columns in the bar chart of Figure 7.1. Arrows indicate for each column, whether an agent in that profile would prefer to stay with that strategy (arrow head), or switch (arrow tail). Solid black arrows denote statistically significant comparisons. Profile 4A is the only one with only in-pointing arrows.

Let PiA denote the profile with a preemptive agent and i As. In the game with preemption, we find three pure-strategy Nash equilibria: P4A, P2A, and P0A. The payoff comparisons are illustrated by Figure 7.2. Significantly, observe that the last equilibrium involves all players playing the baseline strategy, although overall the distinction between aggressive and baseline strategies seems much less important when a preemptive agent is

[4]It can be shown that for any N-player two-strategy symmetric game, there must exist at least one equilibrium in pure strategies, and there also must exist at least one symmetric equilibrium (pure or mixed) [Cheng *et al.*, 2004].

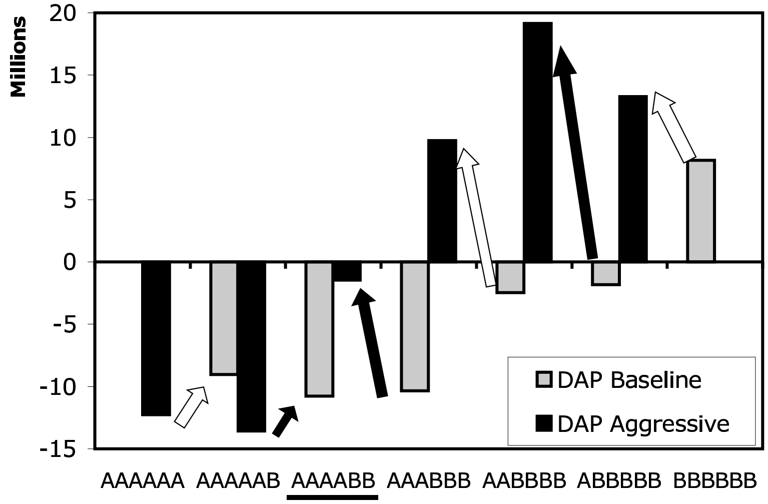


Figure 7.1: Estimated payoffs for unpreempted strategy profiles.

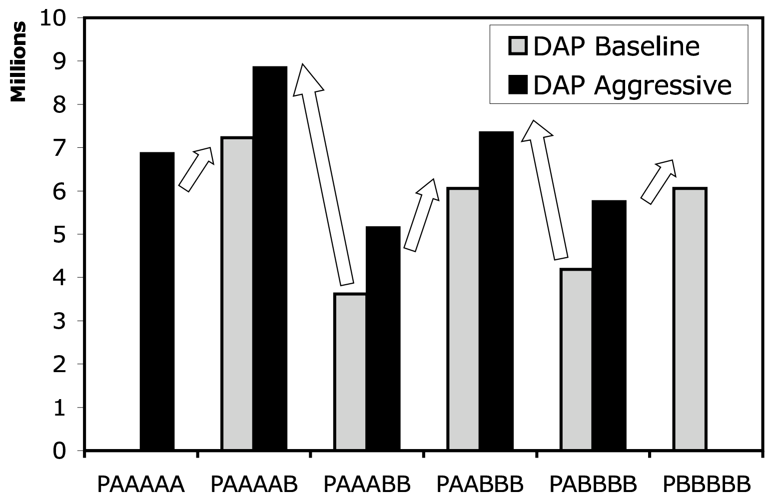


Figure 7.2: Estimated payoffs, with preemption.

present.

7.5.2 Analysis of Day-0 Procurement in TAC/SCM 2004

As I already pointed out in Section 4.1, in spite of the efforts by the game masters, aggressive day-0 procurement behavior dominated the 2004 TAC/SCM tournament. After the tournament, the strategic incentives of agents given the game rules were again evaluated in a stylized way using simulations.

Let a_i be a particular (pure strategy) choice of day-0 procurement for agent i, with

a a profile of such choices. The procurement choices in the analysis are represented by multipliers of the day-0 procurement strategy used by the Deep Maize agent in the actual tournament. Recall from Section 4.1 that $\phi(a) = \sum_{i=1}^{6} a_i$ is the aggregation function representing the sum of day-0 procurement of the six agents participating in a particular supply-chain game.

Figure 7.3 presents the estimate of a set of Nash equilibria when storage cost is set to 100% of storage inventory, charged daily—the actual setting used in the TAC/SCM 2004 tournament. The lowest prediction of day-0 procurement is at $\phi(a) = 3.3$, which corre-

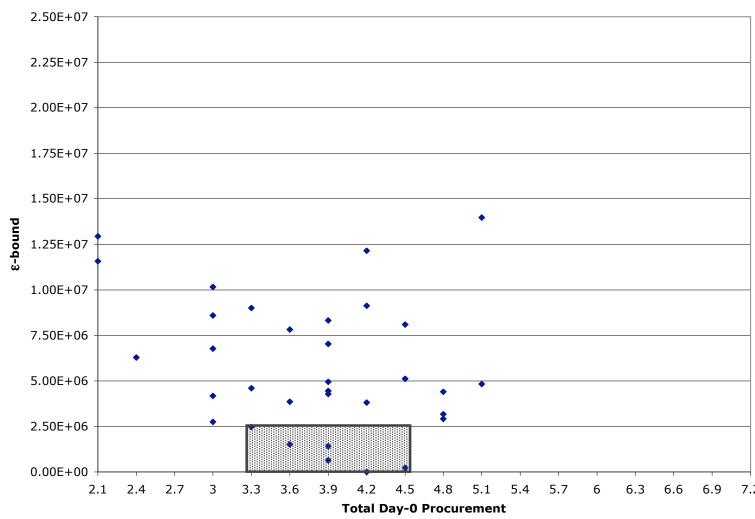


Figure 7.3: Values of $\epsilon_{\Gamma_{\mathcal{E}}}$ for profiles explored using search when storage cost is 50%. Strategy profiles explored are presented in terms of the corresponding values of $\phi(a)$.

sponds to about 39,000 components, which amounts to roughly 1/3 of the total supplier capacity for the entire game. The highest prediction is about 58,400—nearly 1/2 of total capacity. The actual procurement in the 2004 competition was even higher.[5]

7.6 Consistency Results About Nash Equilibria in Empirical Games

In much of what was described above, the set of Nash equilibria of the empirical game with estimated payoff matrices is used as the estimator of Nash equilibria of the

[5]For further details on this analysis, see Section 4.5.3.

underlying game. This seems to be a sensible approach, and I now explore convergence of sets of Nash equilibria computed on small finite empirical games to sets of equilibria in the underlying games. The results in this section will often need an explicit distinction between pure and mixed strategy profiles. As the reader may recall, pure strategy profiles are denoted by $a \in A$, while mixed strategy profiles are denoted by $s \in S$. I use notation $u_{n,i}(a)$ to refer to a payoff function estimate of player i based on a pointwise average over n i.i.d. samples from the distribution of payoffs. I also assume that estimates $u_{n,i}(a)$ are independent for all $a \in A$ and $i \in I$. I use the notation Γ_n to refer to the game $[I, R, \{u_{i,n}(\cdot)\}]$ estimated from the corresponding empirical game, whereas Γ will denote the underlying game, $[I, R, \{u_i(\cdot)\}]$, with $R = A$ or $R = S$, depending on context below. Similarly, I define $\epsilon_n(r)$ to be the regret function computed with respect to the game Γ_n.

In this section, I show that $\epsilon_n(s) \to \epsilon(s)$ a.s. uniformly on the mixed strategy space S for any finite game, and, furthermore, that all mixed strategy Nash equilibria in empirical games eventually become arbitrarily close to *some* Nash equilibrium strategies in the underlying game. I use these results in Chapter 12 to show that under certain conditions, the optimal choice of the design parameter based on simulation-based games converges almost surely to the actual optimum.

Theorem 7.9 *Suppose that* $|I| < \infty, |A| < \infty$*. Then* $\epsilon_n(s) \to \epsilon(s)$ *a.s. uniformly on* S*.*

Recall that $\mathcal{N}$ is a set of all Nash equilibria of Γ. If we define $\mathcal{N}_{n,\gamma} = \{s \in S : \epsilon_n(s) \leq \gamma\}$, we have the following corollary to Theorem 7.9:

Corollary 7.10 *For every* $\gamma > 0$*, there is* M *such that* $\forall n \geq M$*,* $\mathcal{N} \subset \mathcal{N}_{n,\gamma}$ *a.s.*

Proof. Since $\epsilon(s) = 0$ for every $s \in \mathcal{N}$, we can find M large enough such that $\Pr\{\sup_{n \geq M} \sup_{s \in \mathcal{N}} \epsilon_n(s) < \gamma\} = 1$. $\square$

By the corollary, for any game with a finite set of pure strategies and for any $\epsilon > 0$, all Nash equilibria lie in the set of empirical ϵ-Nash equilibria if enough samples have been taken. As I now show, this provides some justification for our use of a set of profiles with a non-zero ϵ-bound as an estimate of the set of Nash equilibria, which I have done in Section 4.5.1.

Let us assume that $\phi^*(\theta) : \Theta \to \mathbb{R}$. First, suppose we conclude that for a particular setting of θ, $\sup\{\hat{\phi}^*(\theta)\} \le \alpha$. Then, since for any fixed $\epsilon > 0$, $\mathcal{N}(\theta) \subset \mathcal{N}_{n,\epsilon}(\theta)$ when n is large enough,

$$\sup\{\phi^*(\theta)\} = \sup_{s \in \mathcal{N}(\theta)} \phi(s) \le \sup_{s \in \mathcal{N}_{n,\epsilon}(\theta)} \phi(s) = \sup\{\hat{\phi}^*(\theta)\} \le \alpha,$$

for any such n. Thus, since we defined the welfare function of the designer to be $\mathbf{I}\{\sup\{\phi^*(\theta)\} \le \alpha\}$ in our domain of interest, the empirical choice of θ satisfies the designer's objective, thereby maximizing his welfare function.

Alternatively, suppose we conclude that $\inf\{\hat{\phi}^*(\theta)\} > \alpha$ for every θ in the domain. Then,

$$\alpha < \inf\{\hat{\phi}^*(\theta)\} = \inf_{s \in \mathcal{N}_{n,\epsilon}(\theta)} \phi(s) \le \inf_{s \in \mathcal{N}(\theta)} \phi(s) \le \sup_{s \in \mathcal{N}(\theta)} \phi(s) = \sup\{\phi^*(\theta)\},$$

for every θ, and we can conclude that no setting of θ will satisfy the designer's objective.

Next, I show that when the number of samples is large enough, every Nash equilibrium of Γ_n is close to *some* Nash equilibrium of the underlying game. This result will lead me to consider convergence of optimizers based on empirical data to actual optimal mechanism parameter settings in Chapter 12.

I first note that the function $\epsilon(s)$ is continuous in a finite game.

Lemma 7.11 *Let S be a mixed strategy set defined on a finite game. Then $\epsilon : S \to \mathbb{R}$ is continuous.*

For the exposition that follows, we need a bit of additional notation. First, let (Z, d) be a metric space, and $X, Y \subset Z$ and define *directed Hausdorff distance* from X to Y to be

$$h(X, Y) = \sup_{x \in X} \inf_{y \in Y} d(x, y).$$

Observe that $U \subset X \Rightarrow h(U, Y) \le h(X, Y)$. Further, define $B_S(x, \delta)$ to be an open ball in $S \subset Z$ with center $x \in S$ and radius δ. Now, let $\mathcal{N}_n$ denote all Nash equilibria of the game Γ_n and let

$$\mathcal{N}_\delta = \bigcup_{x \in \mathcal{N}} B_S(x, \delta),$$

that is, the union of open balls of radius δ with centers at Nash equilibria of Γ. Note that $h(\mathcal{N}_\delta, \mathcal{N}) = \delta$.

We can then obtain the following general result, the proof of which is in the appendix.

Theorem 7.12 *Suppose $|I| < \infty$ and $|A| < \infty$. Then $h(\mathcal{N}_n, \mathcal{N})$ converges to 0 a.s.*

7.7 Probabilistic Bounds on Approximation Quality

It is intuitive that reporting results of empirical game analysis would require some kind of statistical significance claims, just as reporting sample means generally requires the associated t-test results or confidence intervals. In the literature on empirical game analysis, no theoretical confidence bounds have been available to make such analysis possible for approximate equilibria in empirical games. Instead, much of the past work had resorted to worst case sensitivity analysis procedures [Walsh *et al.*, 2002] and indirect evidence [Wellman *et al.*, 2005]. While these can be used to support certain claims, they are not entirely satisfactory, and it certainly seems possible to engage in statistical analysis of empirical equilibria given appropriate distributional assumptions. [Reeves, 2005] discusses sensitivity analysis procedure in the same spirit as my derivations which follow. His analysis involved sampling payoffs from a normal distribution centered at the sample mean and deriving an empirical probability distribution that particular strategies are played in an actual equilibrium.

In this section, I introduce a statistical framework for sensitivity analysis of solutions based on empirical games. Specifically, I derive probabilistic confidence bounds on the quality of empirical equilibria given various assumptions about the nature of noise and the structure of the underlying game. I then experimentally assess the quality of my bounds when some of these assumptions do not hold.

7.7.1 Distribution-Free Bounds

I begin by deriving distribution-free bounds about the likelihood that the specified profiles are Nash equilibria. While most general, these bounds are likely the least useful, as they will in practice rarely be very tight.

The first lemma provides a (deterministic) bound on $\epsilon(r)$, where r is some strategy profile in the set of joint strategies, given that we have a bound on the quality of the payoff function approximation for every point in the domain.

Lemma 7.13 *Let $u_i(r)$ be the underlying set of payoff functions for all players and $\hat{u}_i(r)$ be an approximation of $u_i(r)$ for each $i \in I$. Suppose that $|u_i(r) - \hat{u}_i(r)| \leq \delta$ for strategy profiles r and $\forall(a_i, r_{-i}) : a_i \in A_i$. Then $|\epsilon(r) - \hat{\epsilon}(r)| \leq 2\delta$.*

Note the notation $\hat{u}_i(r)$ for the approximate payoff function and $\hat{\epsilon}(r)$ for the game-theoretic regret defined with respect to $\hat{u}$.

Next, I derive a distribution-free bound, first for pure strategy and thereafter for mixed strategy approximate equilibria. For pure strategy bounds, I assume that the following condition holds *pointwise* on the joint strategy space R (pure or mixed):

$$\Pr\{|u(r) - \hat{u}(r)| \geq \gamma\} \leq \delta. \tag{7.2}$$

The desired probabilistic bound is then described by the following lemma.

Theorem 7.14 *Suppose (7.2) holds* pointwise *on R (pure or mixed). Then*

$$\Pr\{|\epsilon(r) - \hat{\epsilon}(r)| \geq 2\gamma\} \leq m(K+1)\delta = 1 - \alpha,$$

where $K = \max_{i \in I} |A_i|$.

Applying Chebyshev's inequality and Chernoff bound, we get the following corollaries for $a \in A$:

Corollary 7.15 *For any random variable with finite variance,*

$$\Pr\{|\epsilon(a) - \hat{\epsilon}(a)| \geq \gamma\} \leq \frac{4(K+1)m\sigma^2}{n\gamma^2}.$$

Corollary 7.16 *For a random variable that is bounded between L and U,*

$$\Pr\{|\epsilon(a) - \hat{\epsilon}(a)| \geq \gamma\} \leq m(K+1) \exp\left\{-\frac{\gamma^2 n}{2(U-L)^2}\right\}.$$

To find a distribution-free bound for a mixed strategy profile, we need a somewhat stronger assumption:

Lemma 7.17 *Suppose (7.2) holds* uniformly *on A. Then*

$$\Pr\{|u(s) - \hat{u}(s)| \geq \gamma\} \leq \delta \; \forall s \in S.$$

Theorem 7.18 *Suppose (7.2) holds* uniformly *on A and let $s \in S$. Then*

$$\Pr\{|\epsilon(s) - \hat{\epsilon}(s)| \geq 2\gamma\} \leq m(K+1)\delta.$$

Proof. By Lemmas 7.14 and 7.17. □

Corollary 7.19 *For any random variable with finite variance, $s \in S$,*

$$\Pr\{|\epsilon(s) - \hat{\epsilon}(s)| \geq \gamma\} \leq \frac{4|A|(K+1)m\sigma^2}{n\gamma^2}.$$

Corollary 7.20 *For a random variable that is bounded between L and U, $s \in S$,*

$$\Pr\{|\epsilon(s) - \hat{\epsilon}(s)| \geq \gamma\} \leq m|A|(K+1) \exp\left\{-\frac{\gamma^2 n}{2(U-L)^2}\right\}.$$

For general applicability of the above result, I observe in the following simple proposition that we can use them to bound a probability that a particular profile is a δ-Nash equilibrium for a given δ.

Proposition 7.21 *Let δ be given. Then,* $\Pr\{\epsilon(r) \leq \delta\} \geq Pr\{|\epsilon(r) - \hat{\epsilon}(r)| \leq \delta - \hat{\epsilon}(r)\}$.

Proof. $\Pr\{\epsilon(r) \leq \delta\} = \Pr\{\epsilon(r) - \hat{\epsilon}(r) \leq \delta - \hat{\epsilon}(r)\} \geq \Pr\{|\epsilon(r) - \hat{\epsilon}(r)| \leq \delta - \hat{\epsilon}(r)\}$. □

7.7.2 Confidence Bounds for Finite Games with Normal Noise

Suppose that it is feasible to sample the entire payoff matrix of the game. To derive a generic probabilistic bound for a pure strategy profile $a \in A$, suppose that we have an improper prior on $u_i(a)$ for all $a \in A$, and the sampling noise is Gaussian with known variance σ^2. Notationally, I will use $\bar{u}_i(a)$ to indicate a sample mean estimate of $u_i(a)$ based on $n_i(a)$ samples.

The results below build on the derivation of the distribution of the maximum of k variables based on samples of these with zero-mean additive Gaussian noise by [Chang and Huang, 2000]. In this setting, [Chang and Huang, 2000] demonstrate that the posterior distributions of $u_i(\cdot)|\bar{u}_i(\cdot)$ are Gaussian random variables with mean $\bar{u}_i(\cdot)$ and variance σ^2. Furthermore, if the payoffs are sampled independently, the actual payoffs $u_i(\cdot)$ are also independently distributed, conditional on $\bar{u}_i(\cdot)$. Given these assumptions, the following general bound, which I already presented in Proposition 4.4, can derived (in the sequel I omit conditioning on $\bar{u}_i(\cdot)$ for brevity):

Proposition 7.22

$$\Pr\left(\epsilon(a) \leq \epsilon\right) = \prod_{i \in I} \int_{\mathbb{R}} \prod_{b \in A_i \setminus a_i} \Pr(u_i(b, a_{-i}) \leq u + \epsilon) f_{u_i(a)}(u) du,$$

where $f_{u_i(a)}(u)$ is the pdf of $N(\bar{u}_i(a), \sigma_i(a))$.

The posterior distribution of the actual mean under the assumption of Gaussian noise was derived in [Chang and Huang, 2000] and also presented in Section 4.10:

$$\Pr\left(u_i(a) \leq c\right) = 1 - \Phi\left[\frac{\sqrt{n_i(a)}(\bar{u}_i(a) - c)}{\sigma_i(a)}\right], \tag{7.3}$$

where $a \in A$ and $\Phi(\cdot)$ is $N(0, 1)$ distribution function. Applying Lemma 7.22, we get

$$\Pr\left(\epsilon(a) \leq \epsilon\right) = \prod_{i \in I} \int_{\mathbb{R}} \prod_{b \in A_i \setminus a_i} \Pr\left(u_i(b, a_{-i}) \leq u + \epsilon\right) f_{u_i(a)}(u) du = 1 - \alpha. \tag{7.4}$$

Having derived bounds on pure strategy profiles, it is not difficult to extend the results to bounds on mixed strategy profiles, as we do in the following theorem:

Proposition 7.23 *Let $s \in S$ be a mixed strategy profile. Then,*

$$\Pr\left(\epsilon(s) \le \epsilon\right) = \prod_{i \in I} \int_{\mathbb{R}} \prod_{b \in A_i} \left[\Pr(W_i(b) \le u + \epsilon)\right] f_{W_i^*}(u) du,$$

where

$$\Pr(W_i(b) \le u + \epsilon) = 1 - \Phi\left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c) s_{-i}(c) - u - \epsilon}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c) s_{-i}^2(c)}{n_i(b,c)}}}\right]$$

and

$$W_i^* \sim N\left(\sum_{a \in A} \bar{u}_i(a) s(a), \sum_{a \in A} \frac{\sigma^2(a) s^2(a)}{n_i(a)}\right).$$

7.7.3 Confidence Bounds for Infinite Games That Use Finite Game Approximations

Suppose that we are trying to estimate a Nash equilibrium for a game $\Gamma = [I, R, u(\cdot)]$ with R infinite. Let $R_k \subset R$ be finite and define $\Gamma_k = [I, R_k, u(\cdot)]$ to be a finite approximation of Γ. To draw any conclusions about Γ based on its finite approximation we must make some assumptions about the structure of the actual payoff functions on the infinite domain. I assume that the payoff functions, $u_i(\cdot)$ of all players satisfy the Lipschitz condition with Lipschitz coefficient B.

Define d to be the maximum distance from a point in $R_{k,i}$ to its closest neighbor in R_i:

$$d = \max_{i \in I} \sup_{r \in R_i} \inf_{r' \in R_{k,i}} \{|r - r'|\} < \infty.$$

Then if r is an ϵ-Nash of Γ_k with probability $1 - \alpha$, then it is an $(\epsilon + Bd)$-Nash of Γ with probability at least $1 - \alpha$. Consequently, we have the following bound:

Proposition 7.24

$$\Pr\left(\max_{i \in I} \sup_{t \in R_i} u_i(t, r_{-i}) - u_i(r) \le \epsilon\right) \ge$$
$$\ge \prod_{i \in I} \int_{\mathbb{R}} \prod_{t \in R_{k,i}} \Pr(u_i(t, r_{-i}) \le u + \epsilon - Bd) f_{u_i(r)}(u) du.$$

7.7.4 Experimental Evaluation of Bounds

In this section I experimentally explore several questions about the quality of some of our theoretical bounds for a fixed confidence level $1-\alpha$. Define $\hat{\alpha} = \#\{r : \bar{\epsilon}(r) < \epsilon(r)\}/\#\{r\}$.[6] In words, $\hat{\alpha}$ is the proportion of profiles with probabilistic bound below actual. Our first metric of quality of the bounds is *accuracy*, which we define to be $\max\{\hat{\alpha}-\alpha, 0\}$. This will indicate whether our empirical confidence level is below the theoretical. The second metric is *tightness*, defined as $|\bar{\epsilon}(r) - \epsilon(r)|$. In practice, tight bounds would limit the number of samples we need for good equilibrium estimates.

My questions primarily attack scenarios in which the assumptions on which our bounds are based do not hold. In finite games explored below, I compare the quality of bounds I derived under normal and uniform additive noise. For this evaluation, I used a finite game generation engine, GAMUT [Nudelman *et al.*, 2004], to compare the results across several classes of games. I bounded payoffs for games between 0 and 1000,000 and fixed variance of additive noise at 17,000,000. The confidence level was fixed at 95%. For each random game generated I randomly selected a pure profile in one set of experiments and a symmetric mixed profile in another. I then proceeded to compute $\epsilon(r)$ and $\bar{\epsilon}(r)$ for the randomly selected profile r.

As a first sanity check, I experimentally verified that the quality of bounds for finite games does not significantly vary with $\epsilon(r)$. Consequently, I report results in this section solely in terms of *tightness*. From Figure 7.4 (a) we can see that the tightness of bounds on pure strategy profiles does not appear to be affected by the size of the game. For symmetric mixed profiles, however, the bounds become tighter as the size of the game increases. This result is likely due to the fact that we uniformly randomly generated weights on each strategy in the strategy set. From Figure 7.4 (b) we can see that Prisoner's Dilemma and Chicken seem to have a slightly higher bounds than random zero-sum and 2x2 games. In all cases mixed strategy profiles tend to have tighter bounds than pure strategy profiles. Finally, uniform noise does not seem to have a significant impact on tightness.

[6] Recall that $\bar{\epsilon}(r)$ is the smallest ϵ for which the probabilistic bound for profile r holds with at least $(1-\alpha)100\%$ confidence.

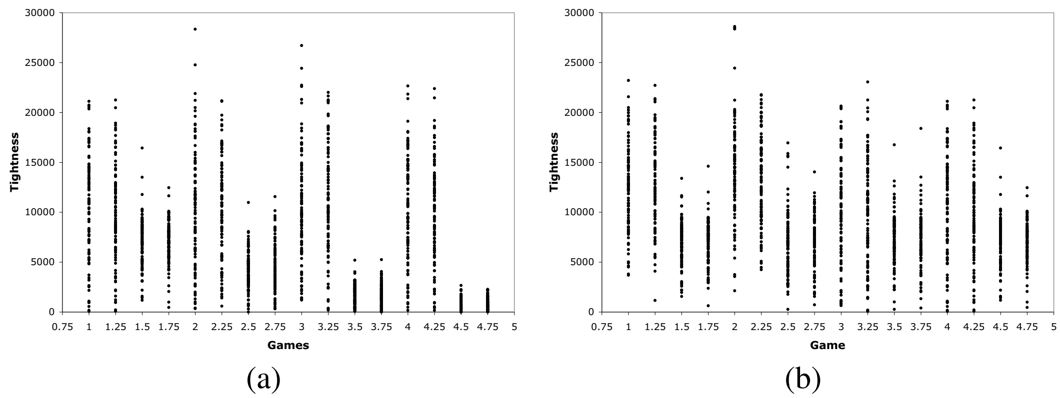


Figure 7.4: Comparison of tightness of bounds for 95% confidence level across (a) random games (1.x=2x2; 2.x=3x3; 3.x=4x4; 4.x=5x5) and (b) several classes of 2x2 games (1.x=Prisoners Dilemma, 2.x=Chicken, 3.x=random 2-strategy zero sum game, 4.x=random 2x2 game). For both (a) and (b), y.0=pure, normal noise; y.25=pure, uniform noise; y.5=mixed, normal noise; y.75=mixed, uniform noise.

Accuracy (when noise was uniform) was not an issue in all but two games: random 4x4 (2% difference) and random 5x5 (15% difference). However, the average tightness for profiles with bounds that fell below actual $\epsilon(r)$ was quite low, roughly 0.006 for the 4x4 game and 0.0075 for the 5x5 game, whereas the average tightness over all profiles was 0.03 and 0.02 respectively. Thus, it does not appear that uniform (instead of normal) noise would significantly impact results reported for these classes of games.

7.8 Maximum Likelihood Equilibrium Estimation

To this point, the techniques for estimating Nash equilibria based on empirical games have involved using either Nash equilibria of the empirical game or profiles with low empirical regret. Neither of these techniques, however, takes into account available distributional information about the game. Intuitively, such information could be of great value, as we can use it to establish precise probabilistic bounds on the likelihood that each profile is a Nash (or an ϵ-Nash equilibrium) of the underlying game, as shown in Section 7.7. I now attempt to utilize such distributional information in defining another equilibrium estimator, one which uses a profile most likely to be a Nash equilibrium of the underlying game. While certainly not the only method for using distributional information in the empirical game to estimating Nash equilibria, this method has some appealing properties, which I demonstrate below.[7]

Under the assumption of an improper prior on actual payoffs and normal noise with known finite variance, I derived above exact probabilistic bounds on ϵ-Nash equilibria of pure and mixed strategy profiles. By setting the $\epsilon(r)$ in these expressions to 0, these bounds describe the probability that a profile r is a Nash equilibrium. I now define the maximum likelihood Nash equilibrium estimator (MLEE) $\hat{r}_{MLEE}$ to be

$$\hat{r}_{MLEE} = \arg\max_{r \in R} \Pr\{\epsilon(r) = 0\}. \tag{7.5}$$

For this expression to be well-defined, we need to ensure that the maximum actually exists. If R is finite, that is entirely obvious. Let us take a more general case where R is the set of mixed strategy profiles S. The following lemma takes us most of the way.

Lemma 7.25 $\Pr\{\epsilon(r) = 0\}$ *is continuous.*

Based on this result, we can now readily confirm that the expression for the MLE estimator is well-defined.

Theorem 7.26 *The maximum in Equation 7.5 exists when* $R = S$.

Proof. By Lemma 7.25, $\Pr\{\epsilon(r) = 0\}$ is continuous. Since S is an n-dimensional simplex, it is compact in $\mathbb{R}^n$. By Weierstrass theorem, the maximum exists on S. □

Corollary 7.27 *The maximum in Equation 7.5 naturally also exists when* R *is a closed subset of* S*, with the case of finite* R *being a (trivial) special case.*

Proof. This follows because a closed subset of a compact set is compact. □

Having verified that the MLEE is well-defined, I demonstrate that the estimate which it produces for a particular underlying game is *essentially unique*.

Definition 7.28 *An estimator is* essentially unique *if it is unique with probability 1.*

[7] An alternative, closely related method, was suggested by Jordan *et al.* [2008]. Their method uses the profile which is most likely to have the smallest $\epsilon(r)$ as the estimate of the best approximate Nash equilibrium.

Suppose we have a game in which every profile has been sampled n times and let $P(r) = \Pr\{\epsilon(r) = 0\}$. The following theorem states that with probability 1 there will not be two profiles $r, r' \in S$ with the same value of $P(r)$.

Theorem 7.29 *$P(r)$ is essentially unique in the mixed strategy space S.*

Corollary 7.30 *The estimate $\hat{r}_{MLEE}$ is essentially unique.*

Finally, I verify that the MLEE is consistent.

Definition 7.31 *A Nash equilibrium estimator $\hat{r}_n$ is* consistent *if $\epsilon(\hat{r}_n) \to 0$ in probability as the minimum number of samples n of each strategy profile in the game goes to infinity.*

Note that the notion of estimator consistency that I use is somewhat weak in that I only require it to be consistent in the ϵ quasi-metric. Nevertheless, this should be a satisfactory guarantee for practical purposes.

I first show that the probability that a profile r is a Nash equilibrium computed on a game in which every profile has been sampled at least n times (and denoted by $\Pr_n(\epsilon(r) = 0)$) is consistent for every profile r. In other words, $\Pr_n(\epsilon(r) = 0)$ is (strongly) consistent pointwise on S.

Lemma 7.32 *Suppose that the payoffs of the underlying payoff function $u(\cdot)$ has no ties. Then $\Pr_n(\epsilon(r) = 0) \to \Pr(\epsilon(r) = 0)$ for every $r \in R$, where $\Pr(\epsilon(r) = 0) = 1$ when r is a Nash equilibrium of the underlying game and 0 otherwise.*

This lemma implies the following theorem.

Theorem 7.33 *If the game is generic, the MLEE 7.5 is consistent in the above sense.*

It is intuitive that the Nash equilibria of sampled games are related to the profiles that are most likely to be Nash equilibria of the underlying game (as number of samples goes to infinity). Furthermore, equilibria of sampled games can also serve as Nash equilibrium estimators, as discussed above. It is well known, however, that there could be, and often is, a multitude of equilibria. Thus, the estimation problem that relies on Nash equilibria of sampled games to estimate a sample Nash will need to select one of these, and there

	a	b
a	(1,1)	(2,1-ϵ)
b	(1 - ϵ,2)	(2 - ϵ,∞)

Figure 7.5: Example matrix game.

is, as yet, no principled way to do this. The essential uniqueness result above, then, is useful as it eliminates the selection problem. One may perhaps indeed suggest using the MLEE as a principled way of selecting an equilibrium from a sampled game. However, as I now show, MLEE need not correspond to *any* equilibrium.

Theorem 7.34 *The MLEE need not correspond to any Nash equilibrium of the sampled game.*

Proof. Consider the matrix game in Figure 7.5. We use the ∞ as an indicator of a payoff that is sufficiently high to make the below approximations close. Note that $\Pr(\epsilon(a,a) = 0) \approx 0.5 * 0.5 = 0.25$ and $\Pr(\epsilon(b,b) = 0) \approx 0.5 * 1 = 0.5$. It is clear that (b,b) has higher likelihood of being a Nash equilibrium, but (a,a) is a strictly dominant (and therefore unique) Nash equilibrium of the sampled game. □

Thus, MLE estimators, although closely related to equilibria of empirical games, are fundamentally different estimators. Which estimation method is better in practice in terms of quality of equilibrium approximation is an empirical question.

7.9 Comparative Statics

In the game theory literature, it is common to expect some analysis of the effect of small changes to model parameters on the predictions. Insofar as the computational analysis techniques based on simulation-based games would be applied to gain insights about the practical applications, similar *comparative statics* analysis would be desirable.

I have already discussed comparative statics in the context of mechanism design in Section 4.11, and the same ideas would in principle apply here. The deep practical problem of doing comparative statics computationally in the context of games—more acute than in mechanism design—is non-uniqueness of solutions.

Recall that the procedure of comparative statics would approximate the derivative of the solutions with respect to some model parameters (e.g., mechanism design choices). To do this, we would have to approximate solutions at several points in some neighborhood. If we could estimate all solutions or at least create ranges of outcomes as discussed above, we would simply make evaluations based on these ranges. Insofar as we would obtain *sample* approximate Nash equilibria, however, alternative design choices, even very close, may yield very different approximate equilibrium solutions and, hence, very different outcomes in games. These differences may, however, be almost entirely due to the fact that we simply happened to compute a different approximate equilibrium. It is not as yet clear to me exactly how severe the extent of this problem is, nor how to deal with it in a principled way. The problem is, however, considerably alleviated if we focus on computing belief distributions of play, rather than Nash equilibria, which I discuss in Chapter 11: there, a distribution of play imposes uniqueness and may also have desirable continuity properties for effective comparative statics analysis.

7.10 Conclusion

In this chapter I defined the notions of simulation-based and empirical games and demonstrated how Nash equilibria can be estimated from empirical games. I used the application to the analysis of strategic TAC/SCM day-0 procurement to demonstrate some of the basic techniques, both in the case of small game stylized models and a model in which player strategy spaces are infinite. The application presented considerable evidence supporting the claim that tournament behavior was a result of strategic interactions between players, rather than irrational outcome of heuristic approximations in which players seem to typically engage.

To enhance the methodology of empirical game analysis, I derived in this chapter probabilistic bounds on quality of Nash equilibrium estimates based on empirical games, both of the distribution-free variety, and based on the assumption of Gaussian sampling noise. Additionally, I presented convergence results about Nash equilibria computed based on empirical game data.

My final contribution in this chapter was to introduce a new estimator of Nash equilibria, which I termed *maximum likelihood equilibrium estimator*. The idea behind this estimator is to use distributional assumptions of Gaussian noise and the resulting probabilistic bounds derived earlier in assessing the likelihood that any particular profile is a Nash equilibrium of the underlying game. The estimator then simply selects the profile most likely to be an actual Nash equilibrium from some restricted set of strategy profiles which can be considered and may include the entire space of mixed strategies. I demonstrate that this estimator is well-defined in the general setting of finite games, and is, additionally, consistent and essentially unique.

CHAPTER 8

Learning Payoff Functions in Infinite Games

IN WHICH *I introduce and analyze statistical learning techniques for approximating payoff and regret functions in infinite games. My purpose for doing so is to approximate game-theoretic solutions, as well as to learn the functions themselves.*[1]

In the previous chapter I focused primarily on Nash equilibrium approximation methods based on empirical regret. For the purposes of this chapter, I call this technique the *sample best* method, having in mind Definition 7.7, since it selects the profile with the lowest (best) empirical regret evaluated with respect to the empirical data (payoff samples) directly. In this chapter I present and evaluate alternative Nash equilibrium estimation techniques for infinite games. The idea behind these techniques is to use regression learning to estimate either payoff functions or actual regret functions based on empirical game data. I investigate various regression model forms—low-degree polynomials, local regression, and support vector machines (SVMs)—applied to infinite games with strategy sets defined by real intervals. I explore two example games, both with incomplete information and real-valued actions. First is the standard first-price sealed-bid auction, with two players and symmetric value distributions. The solution to this game is well-known [Krishna, 2002], and its availability in analytical form proves useful for benchmarking my learning approaches. My second example is a five-player market-based scheduling game

[1]The material in this chapter is taken from Vorobeychik *et al.* [2007b].

[Reeves *et al.*, 2005], where time slots are allocated by simultaneous ascending auctions [Cramton, 2006; Milgrom, 2000]. This game has no known solution, though previous work has identified equilibria on discretized subsets of the strategy space [Reeves *et al.*, 2005].

8.1 Payoff Function Approximation

8.1.1 Problem Definition

We are given a set of data points (a, U), each describing an instance where agents played a pure strategy profile a and realized value $U = (U_1, \ldots, U_m)$. For deterministic games of complete information, U is simply $u(a)$. With incomplete information or stochastic outcomes, U is a random variable, more specifically an independent draw from a distribution function of a, with expected value $u(a)$.

The *payoff function approximation task* is to select a function $\hat{u}$ from a candidate set $\mathcal{H}$ minimizing some measure of deviation from the true payoff function u. Because the true function u is unknown, of course, we must base our selection on evidence provided by the given data points.

The goal in approximating payoff functions is typically not predicting payoffs themselves, but rather in evaluating strategic behavior. Therefore, for assessing my results, I measure approximation quality not directly in terms of a distance between $\hat{u}$ and u, but rather in terms of the *strategies dictated by* $\hat{u}$ evaluated with respect to u. For this I appeal to the notion game-theoretic regret. Specifically, let $\hat{r}$ be a (pure or mixed) approximate Nash equilibrium of the game $[I, R, \{\hat{u}(\cdot)\}]$ and let the regret function $\epsilon(r)$ be computed with respect to the underlying game $[I, R, \{u(\cdot)\}]$. Then the measure of approximation quality of $\hat{u}$ is just $\epsilon(\hat{r})$. Since in general u will either be unknown or not amenable to this analysis, I also developed a method for estimating $\epsilon(\hat{r})$ from data, described in some detail below.

For the remainder of this chapter, I focus on a special case of the general problem, where action sets are real-valued intervals, $A_i = [0, 1]$. Moreover, I restrict attention to symmetric games and introduce several forms of aggregation of other agents' actions as another knob for controlling model complexity.[2] The assumption of symmetry allows me to adopt the convention for the remainder of the paper that payoff $u(a_i, a_{-i})$ is to the agent playing a_i.

8.1.2 Polynomial Regression

One class of models I consider are the *nth-degree separable polynomials*:

$$\begin{aligned} u(a_i, \phi(a_{-i})) = {} & b_n a_i^n + \cdots + b_1 a_i \\ & + c_n \cdot (\phi(a_{-i}))^n + \cdots + c_1 \cdot \phi(a_{-i}) + e, \end{aligned} \tag{8.1}$$

where $\phi(a_{-i})$ represents some aggregation of the strategies played by agents other than i.[3] For two-player games, ϕ is simply the identity function. We refer to polynomials of the form (8.1) as separable, since they lack terms combining a_i and a_{-i}. We also consider models with such terms, for example, the *non-separable quadratic*:

$$\begin{aligned} u(a_i, \phi(a_{-i})) = {} & b_2 a_i^2 + b_1 a_i + c_2 \cdot (\phi(a_{-i}))^2 \\ & + c_1 \cdot \phi(a_{-i}) + d \cdot a_i \phi(a_{-i}) + e. \end{aligned} \tag{8.2}$$

Note that (8.2) and (8.1) coincide in the case $n = 2$ and $d = \mathbf{0}$. In the experiments described below, we employ a simpler version of non-separable quadratic that takes $c_1 = c_2 = \mathbf{0}$.

One advantage of the quadratic form is that we can analytically solve for Nash equilibrium. Given a general non-separable quadratic (8.2), the necessary first-order condition for an interior solution is

$$a_i = -\frac{b_1 + d \cdot \phi(a_{-i})}{2b_2}.$$

[2] Although none of these restrictions are inherent in the approach, one must of course recognize the tradeoffs in complexity of the hypothesis space and generalization performance. Thus, I strive to build in symmetry to the hypothesis space whenever applicable.

[3] Note that $\phi(a_{-i})$ may be a vector, for example, when $\phi(a_{-i}) = a_{-i}$. In this case, $(\phi(a_{-i}))^n$ would be a vector in which every element of $\phi(a_{-i})$ is raised to the nth power.

This reduces to

$$a_i = -\frac{b_1}{2b_2}$$

in the separable case. For the non-separable case with *additive aggregation*, $\phi_{sum}(a_{-i}) = \sum_{j \neq i} a_j$, we can derive an explicit first-order condition for *symmetric* equilibrium:

$$a_i = -\frac{b_1}{2b_2 + (m-1)d}.$$

It has been observed that any game in which $u_i(a)$ are continuous separable payoff functions on compact strategy sets has a pure-strategy Nash equilibrium [Balder, 1996]. This result is quite intuitive, since the absence of interactions entails the existence of dominant strategies and continuous functions are guaranteed to have maxima on compact domains by the Weierstrass theorem [Luenberger, 1969]. However, a pure-strategy equilibrium need not exist in non-separable games, even if payoff functions are quadratic. In the experiments that follow, whenever the polynomial approximation yields no pure-strategy Nash equilibrium, I randomly select a symmetric pure strategy profile from the joint strategy set. If, on the other hand, I find multiple pure equilibria, a sample equilibrium is selected arbitrarily.

8.1.3 Local Regression

In addition to polynomial models, I explore learning using two local regression methods: locally weighted average and locally weighted quadratic regression [Atkeson *et al.*, 1997]. Unlike model-based methods such as polynomial regression, local methods do not attempt to infer model coefficients from data. Instead, these methods weigh the training data points by distance from the query point and estimate the answer—in our case, the payoff at the strategy profile point—using some function of the weighted data set. I used a Gaussian weight function:

$$w = e^{-k^2},$$

where k is the distance of the training data point from the query point and w is the weight that is assigned to that training point.

In the case of locally weighted average, I simply take the weighted average of the payoffs of the training data points as the payoffs for a given strategy profile. Locally weighted quadratic regression, on the other hand, fits a quadratic function to the weighted data set for each query point.

8.1.4 Support Vector Machine Regression

The third category of learning methods I investigate is Support Vector Machines (SVMs). For details regarding this learning method, I refer an interested reader to Vapnik [1995]. In my experiments, I used the *SVM light* package [Joachims, 1999], which is an open-source implementation of SVM classification and regression algorithms, and chose Gaussian radial basis function of the form $e^{-2||x-x'||^2}$ as the kernel.

8.1.5 Finding Mixed-Strategy Equilibria

In the case of polynomial regression models, we are able to find either analytic or simple and robust numeric methods for computing pure Nash equilibria. With local regression and SVM learning we are not so fortunate, as we do not have access to a closed-form description of the learned function.

When a particular learned model is not amenable to a closed-form solution, I use it to approximate a Nash equilibrium of the underlying game as follows. First, I restrict the learned function to a finite strategy subset. Since this restriction produces a finite game, we can thereafter apply any generic finite game solver to find an approximate Nash equilibrium of the *learned* game. For this task, I employ replicator dynamics [Fudenberg and Levine, 1998], which searches for a symmetric mixed equilibrium using an iterative evolutionary algorithm. I treat the result after a fixed number of iterations as an approximate Nash equilibrium of the learned game.[4]

To summarize, I use the following procedure:

1. Randomly produce a training set of data points D in the empirical game in the

[4]Replicator dynamics does not necessarily converge, but when it does reach a locally asymptotically stable fixed point the result is a Nash equilibrium [Friedman, 1991]. For cases where replicator dynamics fails to converge, I still treat the final result as an approximate Nash equilibrium of the game.

form (a, U), where a is a pure strategy profile and U is a sample of corresponding payoffs to the players.

2. Run the learning algorithm on D to obtain a predictor of expected payoffs given a pure strategy profile a (for example, we would thereby obtain the coefficients of polynomial regression or the weights of the local learning methods and SVM)

3. Produce a grid of pure strategy profiles as a discrete approximation of the actual continuous strategy sets.

4. Use the predictor of choice (e.g., SVM) trained as above to obtain the approximate payoff matrix in the discrete strategy subset, thereby obtaining an approximation of the *learned* game.

5. Apply replicator dynamics to the discretized approximation of the learned game to approximate a Nash equilibrium.

8.1.6 Strategy Aggregation

As noted above, I consider payoff functions on two-dimensional strategy profiles in the form $u(a_i, a_{-i}) = f(a_i, \phi(a_{-i}))$. As long as $\phi(a_{-i})$ is invariant under permutation of the strategies in a_{-i}, the payoff function is symmetric. Since the actual payoff functions for our example games are also known to be symmetric, I constrain that $\phi(a_{-i})$ preserve the symmetry of the underlying game.

In my experiments, I compared three variants of $\phi(a_{-i})$. First and most compact is the simple sum, $\phi_{sum}(a_{-i})$. Second is the ordered pair (ϕ_{sum}, ϕ_{ss}), where $\phi_{ss}(a_{-i}) = \sum_{j \neq i}(a_j)^2$. The third variant, $\phi_{identity}(a_{-i}) = a_{-i}$, simply takes the strategies in their direct, unaggregated form. To enforce the symmetry requirement in this last case, I sort the strategies in a_{-i}.

8.2 First-Price Sealed-Bid Auction

In the standard first-price sealed-bid (FPSB) auction game [Krishna, 2002], agents have private valuations for the good for sale, and simultaneously choose a bid price representing their offer to purchase the good. The bidder naming the highest price gets the good and pays the offered price. Other agents receive and pay nothing. In the classic setup first analyzed by Vickrey [1961], agents have identical valuation distributions, uniform on $[0, 1]$, and these distributions are common knowledge. The unique (Bayesian) Nash equilibrium of this game is for agent i to bid $\frac{m-1}{m} v_i$, where v_i is i's valuation for the good.

Note that strategies in this game (and generally for games of incomplete information), $b_i : [0, 1] \to [0, 1]$, are functions of the agent's private information. I consider a restricted case, called "ray bidding" by Selten and Buchta [1994], where bid functions are constrained to the form

$$b_i(v_i) = a_i v_i, \; a_i \in [0, 1].$$

This constraint transforms the action space to a real interval, corresponding to choice of strategy parameter a_i. Additionally, transformation simplifies representation of the game in normal-form. One can easily see that the restricted strategy space includes the known equilibrium of the full game, with $a_i = \frac{m-1}{m}$ for all i, which is also an equilibrium of the restricted game in which agents are constrained to strategies of the given form. While the fact that our restricted strategy space happens to include the actual equilibrium seems extremely fortunate, strategy spaces of this form are intuitively very natural: if we restrict bidders to submit bids in the space of valuations, any valuation that they may wish to bid is also some fraction of their actual value, and they will generally not wish to bid more than the object is worth to them. However, a criticism still applies that we generally do not a priori know if our restriction of the strategy space contains any equilibria of the actual game, and it is certainly not difficult to construct cases in which it is not. Thus, we will at best find an equilibrium of the restricted game, assuming generally that it is a reasonable equilibrium approximation of the actual game. While the question of finding appropriate restrictions of games is an important one, it is beyond the scope of this work.

I further focus on the special case $m = 2$, with corresponding equilibrium at $a_1 = a_2 = 1/2$. For the two-player FPSB, we can also derive a closed-form description of the actual expected payoff function [Reeves, 2005]:[5]

$$u(a_1, a_2) = \begin{cases} 0.25 & \text{if } a_1 = a_2 = 0, \\ \frac{(a_1-1)\left[(a_2)^2-3(a_1)^2\right]}{6(a_1)^2} & \text{if } a_1 \geq a_2, \\ \frac{a_1(1-a_1)}{3a_2} & \text{otherwise.} \end{cases} \tag{8.3}$$

The maximum benefit to deviation (ϵ) as a function of joint pure strategy profile $a \in [0, 1]^2$ is shown in Figure 8.1.

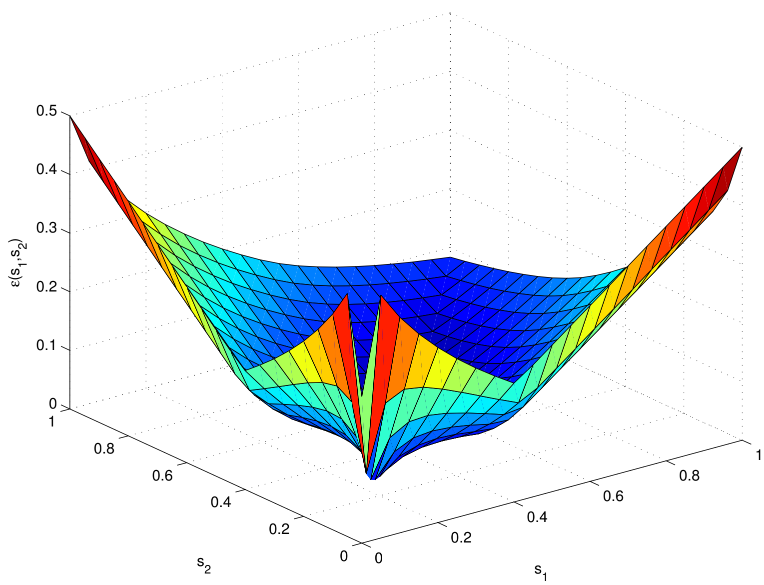


Figure 8.1: $\epsilon(a)$ surface in two-player First-Price Sealed-Bid auction.

The availability of known solutions for this example facilitates analysis of the learning approach. My first set of results is summarized in Figure 8.2. For each of the learning methods (classes of functional forms), I measure average ϵ for varying training set sizes. For instance, to evaluate the performance of separable quadratic approximation with training size N, I independently draw N strategies, $\{a^1, \ldots, a^N\}$, uniformly on $[0, 1]$. The corresponding training set comprises N^2 points: $((a^i, a^j), u(a^i, a^j))$, for

[5] Recall that $u(a_1, a_2) = u_1(a_1, a_2)$ and the payoff to the second player is symmetric.

$i, j \in \{1, \dots, N\}$, with $u(a)$ as given by (8.3). I find the best separable quadratic fit $\hat{u}$ to these points, and find a Nash equilibrium corresponding to $\hat{u}$. I then calculate the least ϵ for which this strategy profile is an ϵ-Nash equilibrium with respect to the *actual* payoff function u. I repeat this process 200 times, averaging the results over strategy draws, to obtain each value plotted in Figure 8.2.

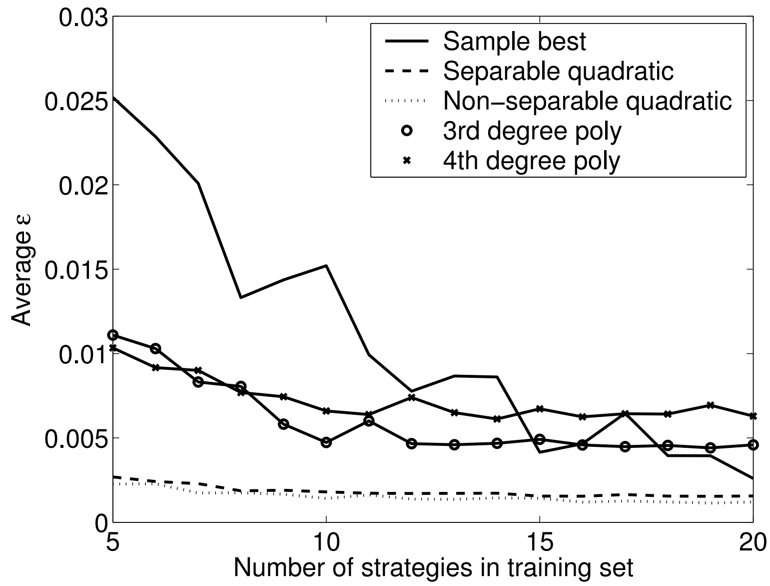


Figure 8.2: Performance comparison of discrete (sample best) and polynomial function approximation models trained on noiseless data sets of varying size in two-player First-Price Sealed-Bid auction.

As we can see, both second-degree polynomial forms I tried performed quite well on this game. For $N < 20$, quadratic regression outperformed in my experiments the model labeled “sample best”, in which the payoff function is approximated by the discrete training set directly. The derived equilibrium in this model is simply a Nash equilibrium over the discrete strategies in the training set. At first, the success of the quadratic model may be surprising, since the actual payoff function (8.3) is only piecewise differentiable and has a point of discontinuity. However, as we can see from Figure 8.3, it appears quite smooth and well approximated by a quadratic polynomial. The higher-degree polynomials apparently overfit the data, as indicated by their inferior learning performance displayed in this game.

The results of this game provide an optimistic view of how well regression might

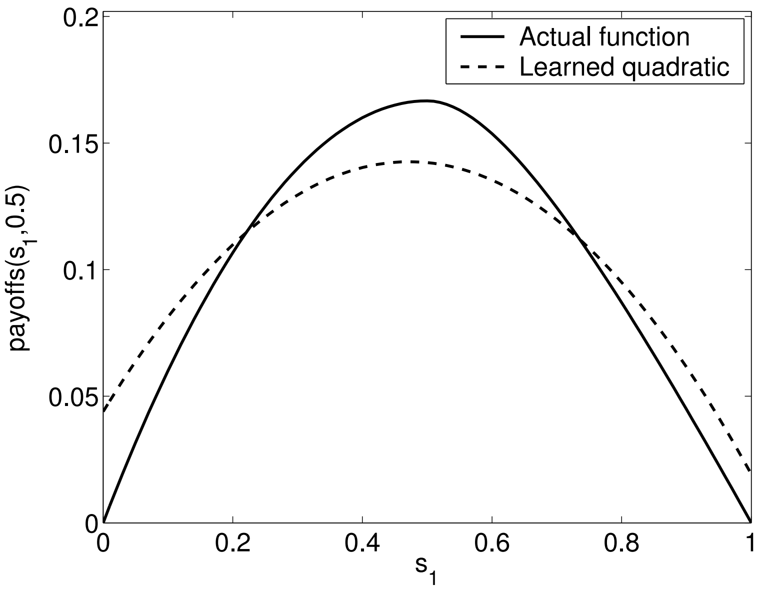


Figure 8.3: Learned and actual payoff function in two-player First-Price Sealed-Bid auction when the other agent plays 0.5. The learned function is the separable quadratic, for a particular sample with $N = 5$.

be expected to perform compared to discretization. This game is quite easy for learning since the underlying payoff function is well captured by our lower-degree model.

The results thus far were based on a noiseless data set. In another set of experiments I artificially added noise to the samples from (8.3) according to a zero-mean Normal distribution. In order to test the effect of noise on the performance of function approximation, I used a control variable, k, such that variance of noise is $1/k^2$. For each value of k and for each function approximation method, I randomly selected 5 strategies between 0 and 1. I then took noisy samples from (8.3) as described to produce a data set, $((a^i, a^j), U(a^i, a^j))$, for $i, j \in \{1, \ldots, 5\}$ (where $U = u(a^i, a^j) + \eta$ and $\eta \sim N(0, 1/k^2)$). After the model was fit to the data set, I found the corresponding equilibrium, and evaluated its ϵ with respect to the actual payoff function. The results, averaged over 200 random draws of 5 strategies, are shown in Figure 8.4. It is interesting to observe that when variance is very high, all methods performed extremely poorly, with non-separable quadratic and sample best (discrete approximation) methods approximately tied for best when variance was 1. However, as variance falls, we can see considerable improvement of the quadratic methods compared to discrete approximation, whereas the third-degree

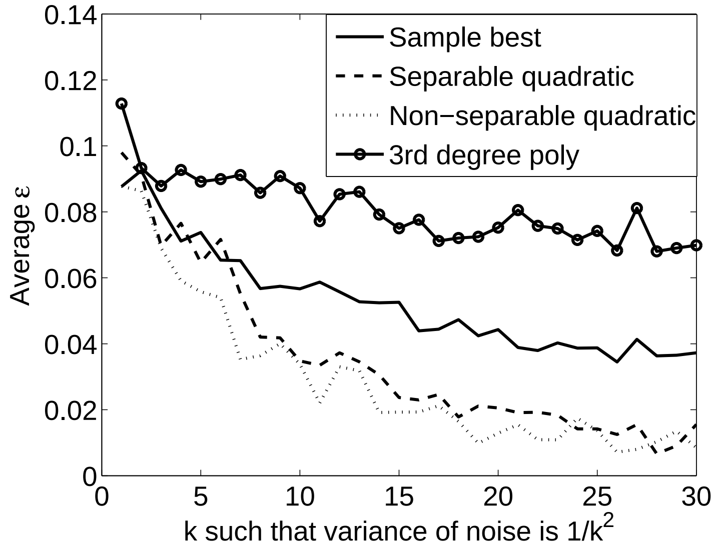


Figure 8.4: Performance comparison of discrete (sample best) and polynomial function approximation models in two-player First-Price Sealed-Bid auction as the variance of noise in the data decreases.

polynomial fit remains on average quite poor. Additionally, including the strategy interaction term produced better results in my experiments, as the non-separable quadratic outperformed other methods in most cases.

8.3 Market-Based Scheduling Game

The second game I investigate presents a significantly more difficult learning challenge. It is a five-player symmetric game, with no analytic characterization, and no (theoretically) known solution. The game hinges on incomplete information, and training data is available only from a simulator that samples from the underlying distribution.

The game is based on a market-based scheduling scenario [Reeves *et al.*, 2005], where agents bid in simultaneous ascending auctions for time-indexed resources necessary to perform their given jobs. Agents have private information about their job lengths, and values for completing their jobs by various deadlines. Note that the full space of strategies is quite complex—dependent on multi-dimensional private information about preferences as well as price histories for all the time slots. As in the FPSB example, I transform this policy space to a real interval by constraining strategies to a parametrized form.

In particular, I start from a simple myopic policy—*straightforward bidding* [Milgrom, 2000], in which the agents choose the bundle of goods to bid on based on their current perceived prices. Following Reeves *et al.* [2005], I generalize this strategy by introducing a scalar parameter, $a_i \in [0, 1]$, called "sunk awareness", which modifies the perceived-price calculation for slots the agent is currently winning. Specifically, the agent treats the effective price of such a slot as $a_i\beta$, where β is the current winning price. Playing strategy $a_i = 1$ is therefore the same as straightforward bidding, and $a_i = 0$ treats the goods as sunk costs. Intuitively, the appropriate value should be somewhat in between, since the agent is committed and the cost is sunk if the current price prevails, but may be let off the hook if another agent outbids it for this slot. Lower values of a_i increase the agent's tendency to stick with slots that it is currently winning.

Though sunk awareness represents a crude approximation to a complex tradeoff, including the parameter provides an agent with room for improvement over straightforward bidding. Most significantly for our current purposes, the optimal setting of a_i is generally dependent on other agents' behavior. This is precisely the relationship I aim to induce through learning. Toward this end, I collected data for all strategy profiles over the discrete set of values $a_i \in \{0, 0.05, \ldots, 1\}$. Accounting for symmetry, this represents 53,130 distinct strategy profiles. For evaluation purposes, I treat the sample averages for each discrete profile as the true expected payoffs on this grid.

I again acknowledge that no guarantees can be made about the actual game (bidding in simultaneous ascending auctions) based solely on analysis of the restricted setting (straightforward bidding with variable sunk awareness). However, in this and many other settings, analysis of the entire strategic space is hopeless, and one generally must rely on restricted analysis to say anything about the actual game. Thus, I proceed to analyze the restricted game, presuming that it sheds light on the strategic interactions in the actual game.

The previous empirical study of this game by Reeves *et al.* [2005] estimated the payoff function over a discrete grid of profiles assembled from the set $\{0.8, 0.85, 0.9, 0.95, 1\}$ of strategies, computing an approximate Nash equilibrium using replicator dynamics. I therefore used the training set based on the data for these strategies (300,000 samples per profile), regressed to the quadratic forms, and calculated empirical ϵ values with respect to the entire data set of 53,130 profiles by computing the maximum benefit from deviation within the data, which I already defined in Section 7.3 (Equation 7.1); I repeat the definition here to keep the chapter reasonably self-contained:

$$\epsilon_{\Gamma_{\mathcal{E}}} = \max_{i \in I} \max_{a_i \in D_i} \left[u_i(a_i, \hat{a}_{-i}) - u_i(\hat{a}) \right],$$

where D_i is the strategy set of player i represented within the data set.[6] Since the game is symmetric, the maximization over players can be dropped, and all the agent strategy sets are identical.

If we compare the Nash equilibria of the learned functions presented in Table 8.1 to the inner product between the restricted strategy set and the corresponding mixed strategy produced using the discrete approximation (a kind of "expected" equilibrium strategy), the results are quite close. However, the learning methods produced much better approximations of Nash equilibria in terms of $\epsilon_{\Gamma_{\mathcal{E}}}$.[7]

Method	Equilibrium a_i	ϵ
Separable quadratic	0.876	0.0027
Non-separable quadratic	0.876	0.0027
Discrete approximation	(0,0.94,0.06,0,0)	0.0238

Table 8.1: Values of ϵ for the symmetric pure-strategy equilibria of games defined by different payoff function approximation methods. The quadratic models were trained on profiles confined to strategies in {0.8,0.85,0.9,0.95,1}. The mixed-strategy equilibrium of the discrete model is presented as the probabilities with which the corresponding strategies are to be played.

In a more comprehensive trial, I collected 2.2 million additional samples for each of 53,130 profiles, and ran the learning algorithms on 100 training sets, each uniformly randomly selected from the discrete grid $\{0, 0.05, \ldots, 1\}$. Each training set included all profiles generated from between five and ten of the twenty-one agent strategies on the

[6]Observe that unlike the first-price sealed-bid auction game, the problem of computing the actual best response in the market-based scheduling game is a computationally intractable POMDP due to both incomplete and imperfect information (player types are private information, and the only observable feature of other agents' bids is the value of the highest for each good). Thus, for the purposes of this study, I treat the data set of 53,130 profiles as the true game.

[7]Since 0.876 is not a grid point, I determined $\epsilon_{\Gamma_{\mathcal{E}}}$ post hoc, by running further profile simulations with all agents playing 0.876, and where one agent deviates to any of the strategies in $\{0, 0.05, \ldots, 1\}$.

grid. Since in this case an approximate equilibrium produced by a polynomial regression model does not typically appear in the complete data set, I developed a method for estimating ϵ for pure symmetric approximate equilibria in symmetric games based on a mixture of neighbor strategies that do appear in the test set. Let us designate a pure symmetric equilibrium strategy of the approximated game by $\hat{a}$. I first determine the closest neighbors to $\hat{a}$ in the symmetric pure strategy set D represented within the data. Let these neighbors be denoted by a' and a''. I define a mixed strategy α over support $\{a', a''\}$ as the probability of playing a', computed based on the relative distance of $\hat{a}$ from its neighbors:

$$\alpha = 1 - \frac{|\hat{a} - a'|}{|a' - a''|}.$$

Note that symmetry allows a more compact representation of a payoff function if agents other than i have a choice of only two strategies. Thus, I define $U(a_i, j)$ as the payoff to a (symmetric) player for playing strategy $a_i \in D$ when j other agents play strategy a'. If $m - 1$ agents each independently choose whether to play a' with probability α, then the probability that exactly j will choose a' is given by

$$\Pr(\alpha, j) = \binom{m-1}{j} \alpha^j (1 - \alpha)^{m-1-j}.$$

We can thus approximate ϵ of the mixed strategy α by

$$\max_{a_i \in D} \sum_{j=0}^{m-1} \Pr(\alpha, j) \left[U(a_i, j) - \alpha U(a', j) - (1 - \alpha) U(a'', j) \right].$$

Using this method of estimating ϵ on the complete data set, I compared results from polynomial regression to the method which simply selects from the training set the pure-strategy profile with the smallest value of ϵ. I differentiate between the case where we only consider symmetric pure profiles (labeled "sample best (symmetric)") and all pure profiles (labeled "sample best (all)").[8]

From Figure 8.5 we see that regression to a separable quadratic produced on average a

[8]Observe in Figures 8.5 and 8.6 that the method which restricted the search for a best pure strategy profile to symmetric profiles on average did better in terms of ϵ then when this restriction was not imposed.

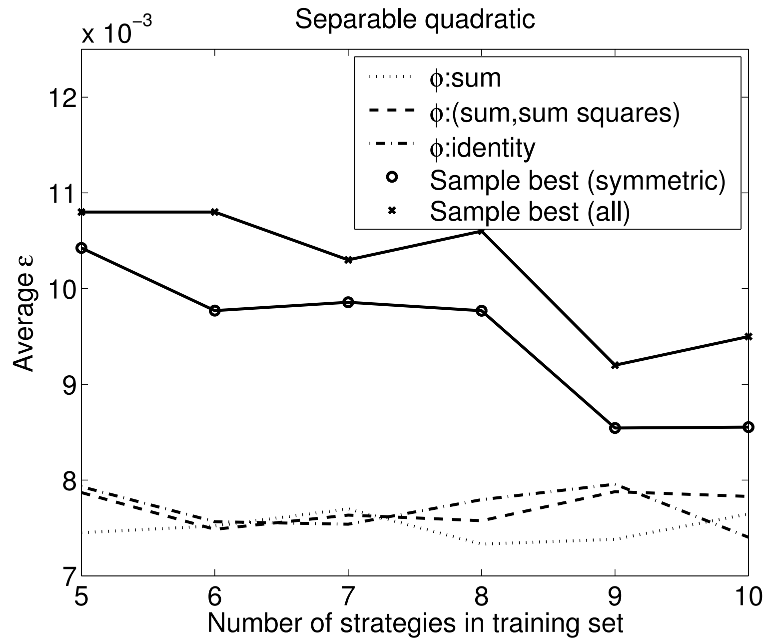


Figure 8.5: Performance comparison of discrete and separable quadratic function approximation models with several forms of strategy aggregation in the market-based scheduling game.

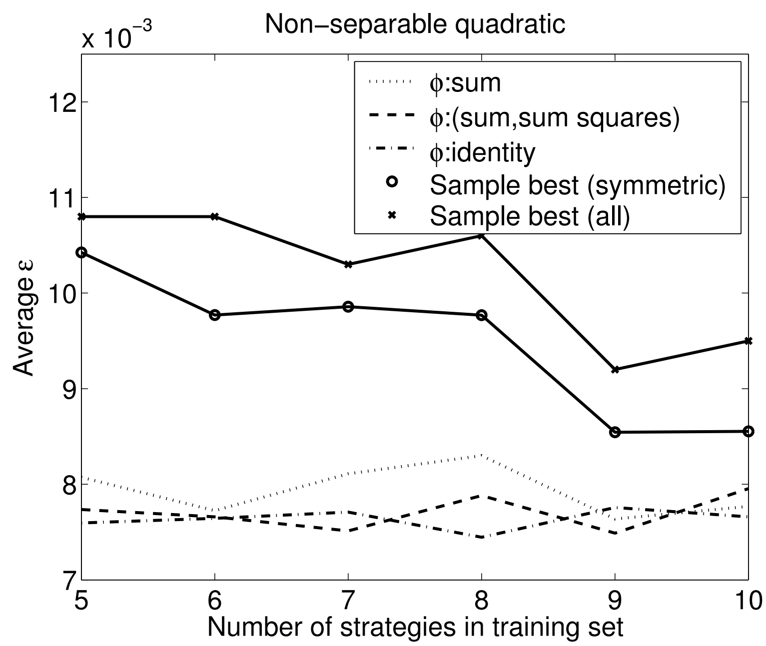


Figure 8.6: Performance comparison of discrete and non-separable quadratic function approximation models with several forms of strategy aggregation in the market-based scheduling game.

considerably better approximate equilibria when the size of the training set was relatively small. Figure 8.6 shows that the non-separable quadratic performed similarly. The results appear relatively insensitive to the degree of aggregation applied to the representation of other agents' strategies.

The polynomial regression methods I employed yield pure-strategy Nash equilibria. I further evaluated four methods that generally produce mixed-strategy equilibria: two local regression learning methods, SVM with a Gaussian radial basis kernel, and direct estimation using the training data. As discussed above, I compute mixed-strategy equilibria by applying replicator dynamics to discrete approximations (using a fixed ten-strategy grid) of the learned payoff functions. In the case of direct estimation from training data, the data itself was used as input to the replicator dynamics algorithm. Since I ensure that the support of any mixed-strategy equilibrium produced by these methods is in the complete data set, I can compute ϵ of the equilibria directly.

As we can see in Figure 8.7, locally weighted average method performed better in my experiments than the other three for most data sets that included between five and ten strategies. Additionally, locally weighted regression performed better than discrete approximation in our experiments on four of the six data set sizes I considered, and SVM consistently beat discrete approximation for all six data set sizes.[9]

It is somewhat surprising to see how irregular our results appear for the local regression methods. I cannot explain this irregularity, although of course there is no reason for us to expect otherwise: even though increasing the size of the training data set may improve the quality of fit, improvement in quality of equilibrium approximation does not necessarily follow.

In order to investigate the effect of noise on the function approximation methods as I had done in the FPSB setting, I ran another set of experiments in which I used the original data set of 300,000 samples per profiles as the training data set, and the data set of 2.5 million samples per profile as the evaluation data set. The remainder of the setup was as

[9] Note that I do not compare these results to those for the polynomial regression methods. Given noise in the data set, mixed-strategy profiles with larger supports may exhibit lower ϵ simply due to the smoothing effect of the mixtures. Thus, when any profile with the lowest ϵ is desired, mixed-strategy profiles would likely be preferred. However, when pure strategy profiles are preferred, polynomial methods are more desirable as they produce pure strategy approximate equilibria.

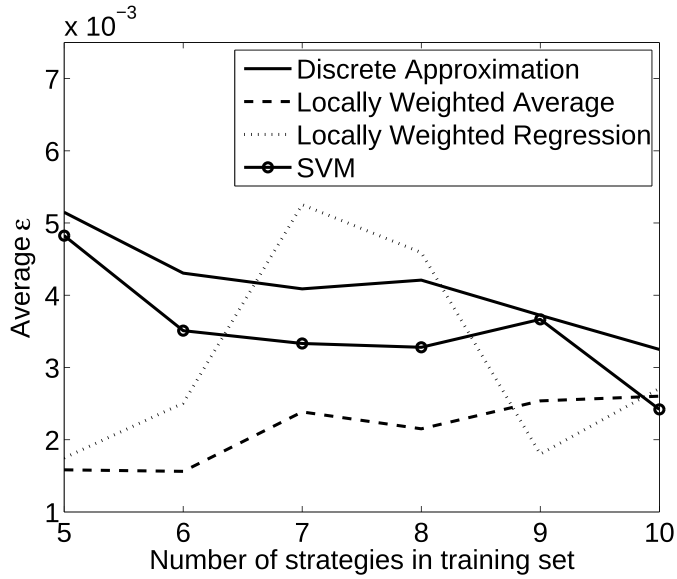


Figure 8.7: Performance comparison of discrete approximation (with replicator dynamics used to find an equilibrium) and local and SVM regression methods with strategy aggregation of the form $\phi(a_{-i}) = (\phi_{sum}, \phi_{ss})$ in the market-based scheduling game.

above.

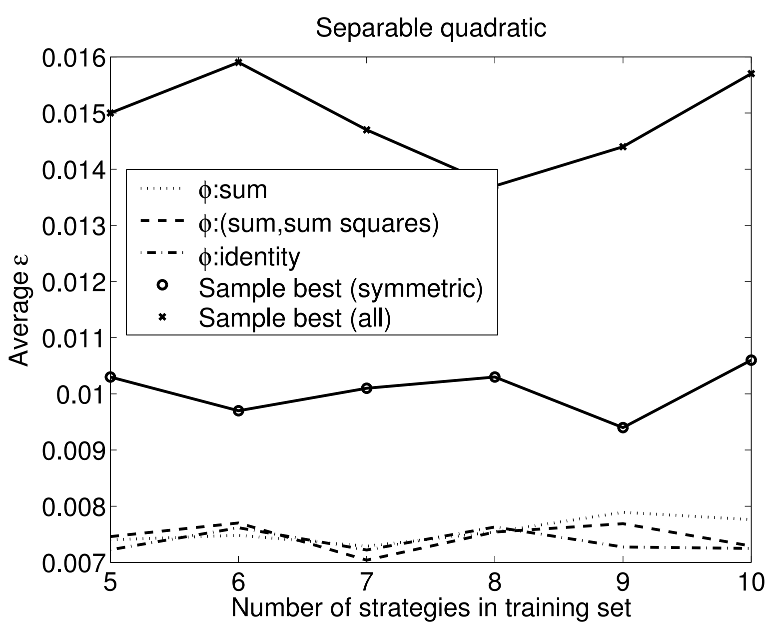


Figure 8.8: Performance comparison of discrete and separable quadratic function approximation models with several forms of strategy aggregation when noise is present in training data in the market-based scheduling game.

Figures 8.8 and 8.9 show the performance of separable and non-separable quadratic

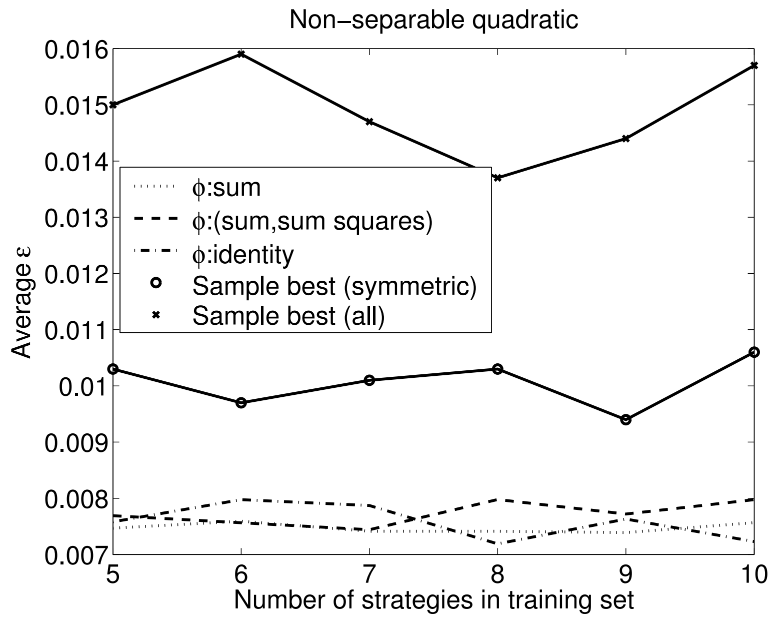


Figure 8.9: Performance comparison of discrete and non-separable quadratic function approximation models with several forms of strategy aggregation when noise is present in training data in the market-based scheduling.

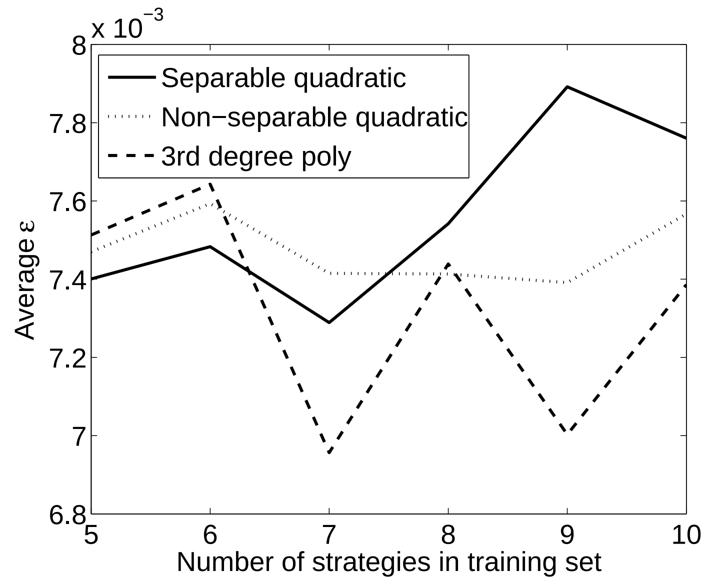


Figure 8.10: Performance comparison of low-degree polynomial models trained on a noisy data set in the market-based scheduling game. All use strategy aggregation of the form ϕ_{sum}.

models with different forms of strategy aggregation, comparing them to simply selecting the best symmetric and asymmetric pure strategy profiles. The quadratic methods still significantly outperformed the methods based on discrete payoff function approximation

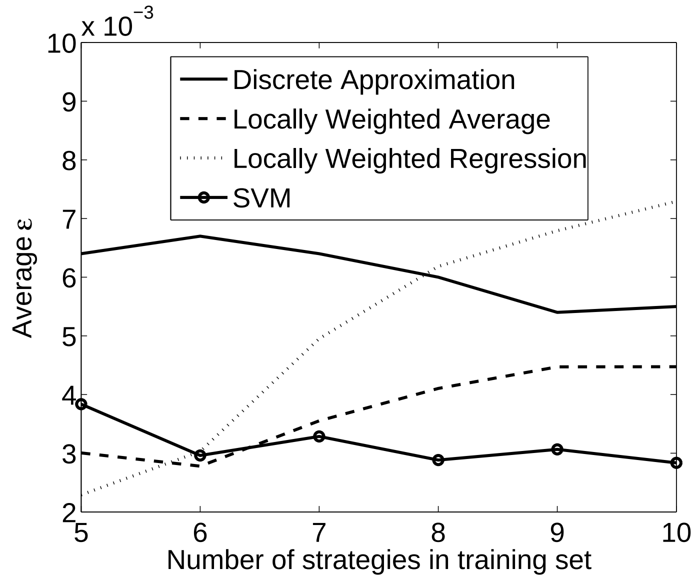


Figure 8.11: Performance comparison of discrete approximation (with replicator dynamics used to find an equilibrium) and local and SVM regression methods with strategy aggregation of the form $\phi(a_{-i}) = (\phi_{sum}, \phi_{ss})$ trained on noisy data in the market-based scheduling game.

in our experiments, and we can again see little difference between the three forms of strategy aggregation I tried.

Figure 8.10 suggests that the third order separable polynomial is relatively robust to noise, having outperformed the quadratic methods in the experiments when training data was more abundant.

The comparison of methods that yield mixed-strategy equilibrium approximations is shown in Figure 8.11. As in the noiseless setting, learning methods tended to outperform discrete approximation. SVM produced the best approximations for almost all training data set sizes I had considered, while local learning methods were generally better than discrete approximation.

Finally, I compared the approximation quality of the SVM model with and without strategy aggregation in Figure 8.12. I used the noisy training data set for these comparisons. Unlike several comparisons between aggregation functions that we had seen already, in this case we do see a distinct advantage to using the particular form of strategy aggregation I chose here over using none at all.

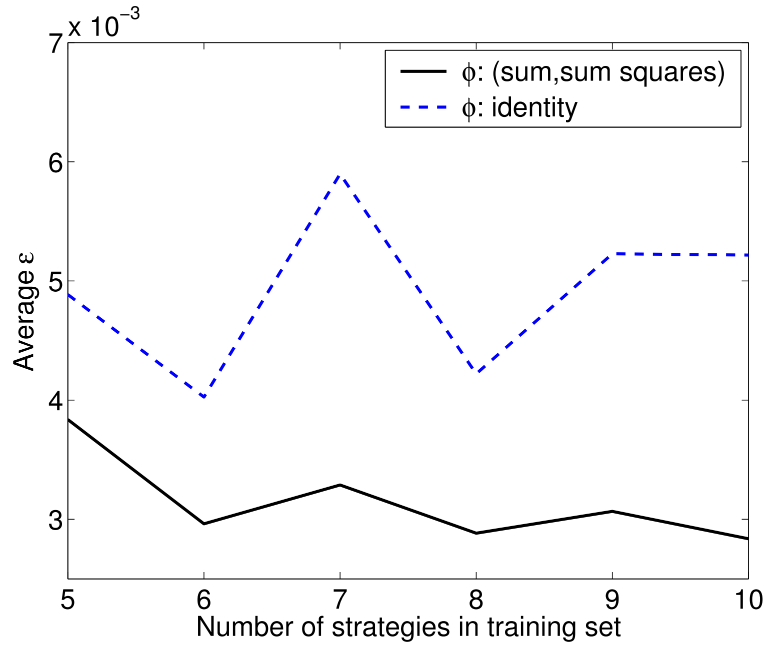


Figure 8.12: Performance comparison of SVM model without strategy aggregation (identity) and with aggregation of the form $\phi(a_{-i}) = (\phi_{sum}, \phi_{ss})$ in the market-based scheduling game.

8.4 Using ϵ in Model Selection

By using the ϵ metric in this study, I implicitly suggested that it may also be a useful guide for selecting a learning model to guide the final choice of an approximate equilibrium. A standard method for model selection that has been studied extensively is mean squared error computed on a test data set (see, for example, Hastie *et al.* [2001]). An alternative I would like to suggest here is to select the model with lowest ϵ (as defined above) estimated from the union of training and test data.

In the experiments here, I compared three model selection methods in the FPSB setting: mean-squared error on test data and two methods based on ϵ of the model fit on the union of training and test data. The mean-squared error method simply selects a model that has the lowest error on the test data set. The first ϵ-based method (I refer to it as "Best ϵ" in the experiments) finds the exact value of ϵ with respect to the restricted set of deviations represented in training and test data. Since the equilibrium of the learned model may not itself be contained in the training or the test data, we may need to take additional samples from the noisy payoff function.[10]

An alternative method estimates ϵ based only on the available training and test data (requiring no additional payoff samples) by taking a mixture of neighborhood points in the data set as a mixed-strategy candidate equilibrium profile in the place of the actual equilibrium of the learned game, as I discussed in some detail in Section 8.3. I refer to this method as "Best Approximate ϵ".

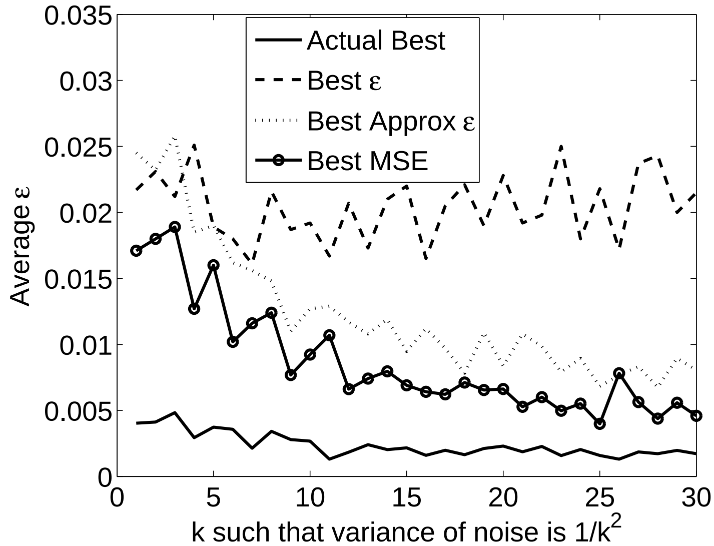


Figure 8.13: Comparison of model selection criteria in FPSB as variance of the noise in the data decreases.

Figure 8.13 compares the three methods for model selection, varying the parameter, k, of the noise distribution, $N(0, 1/k^2)$, and keeping the size of the training set fixed at 25 profiles, representing all profiles for a random draw of 5 strategies. The model choices are restricted to separable and non-separable quadratics, as well as second- and third-degree separable polynomials. As a baseline for comparisons, I chose the model with the lowest actual ϵ in addition to applying the above selection criteria. This is referred to as "Actual Best" in the figure.

As another comparison between the same model selection methods, I fixed the distribution of noise to be $N(0, 1)$ and varied the number of strategies in the training data set between 5 and 10. The results of this comparison are shown in Figure 8.14.

[10] I add noise artificially to the known FPSB payoff function.

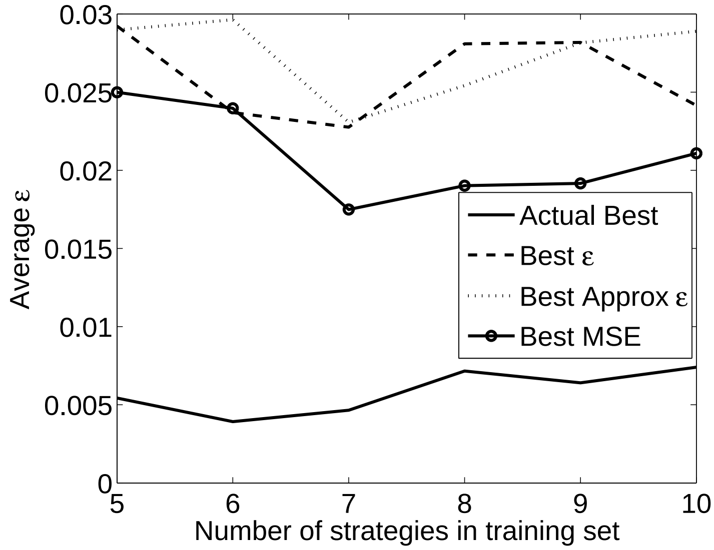


Figure 8.14: Comparison of model selection criteria in FPSB for different training data set sizes, fixing variance of noise at 1.

Both plots show that mean-squared error selection criterion consistently outperformed the others. Additionally, we can see from Figure 8.13 that "Best Approximate ϵ" method tended to outperform the other ϵ-based method in the experiments. This provides some justification for my use of this as the estimate of actual ϵ in the experiments involving the Market-Based Scheduling game.

8.5 Using ϵ as the Approximation Target

Until now, the target of my function approximation endeavors had been the players' payoff functions. In this section, I would like to explore using $\epsilon(a)$ as defined in Section 8.1.1 as an alternative target, restricting the domain of this function to pure strategy profiles.

Since $\epsilon(a) \geq 0$ for all $a \in A$, and $\epsilon(a) = 0$ in equilibrium, any pure-strategy equilibrium minimizes $\epsilon(a)$. Now, suppose that we have an approximation, $\hat{\epsilon}(a)$ of $\epsilon(a)$. Then we can take a minimizer of $\hat{\epsilon}(a)$ to be an approximate Nash equilibrium.

There are several settings in which we may use $\epsilon(a)$ as a target. The most intuitive involves a data set in the form $(a, \epsilon(a))$ in place of the data set of payoff experience I had assumed thus far. This is a standard noiseless function approximation setting, and in my experiments I refer to it as "Target: $\epsilon(a)$".

In general, obtaining a data set in the form $(a, \epsilon(a))$ seems quite extraordinary. Typically, data representing payoff experience as I had supposed previously will be considerably easier to come by. In such a setting, we can still use an approximate $\epsilon(a)$ for each profile represented in the data set, as long as we restrict the strategy sets A_i to deviations available in data.[11] Making this empirical $\epsilon(a)$ as a target of function approximation is the second setting, which provides a more direct comparison with payoff function approximation techniques I had already explored.

My first set of experiments, presented in Figure 8.15, compares the two function approximation settings that employ $\epsilon(a)$ as the target to the setting of payoff function approximation already explored, using either a noiseless data set of FPSB payoff data points, $(a, u(a))$, or of data points $(a, \epsilon(a))$, where appropriate. For all methods presented, I used the separable quadratic model described in Section 8.1.2. The remainder of the setup involved randomly selecting N strategies, for N ranging between 5 and 20, sampling all profiles over these strategies for either the data set of payoff experience or samples from $\epsilon(s)$, and learning the model for the appropriate target function. I repeated this process 200 times in each setting.

As we can see from Figure 8.15, $\epsilon(a)$ approximation target generally performed somewhat better than payoff function approximation. It is noteworthy that even using the $\epsilon(a)$ approximated based on the payoff data as the target tended to produce better equilibrium approximations than those based on learned payoff functions.

Most surprising, perhaps, is the result that the approximate $\epsilon(a)$ provided a better target in my experiments than the actual $\epsilon(a)$. I propose several conjectures to explain this phenomenon. One possibility is that this is a purely idiosyncratic result due to my choice of model and my approximation setting, which manifests itself in a systematic bias of the model that is learned based on the actual examples of $\epsilon(a)$. The fact that this

[11] We can, in effect, drop the assumption here that all profiles in the restricted strategy space of players are available in data. Instead, I simply restrict my computation of ϵ to deviations available to the players within data. Note that this approach gives a lower bound on empirical ϵ. For example, when no deviations are available for some profile a, we can take $\epsilon(a) = 0$. Nevertheless, I use data sets composed of all profiles in the restricted strategy spaces of players in my experiments.

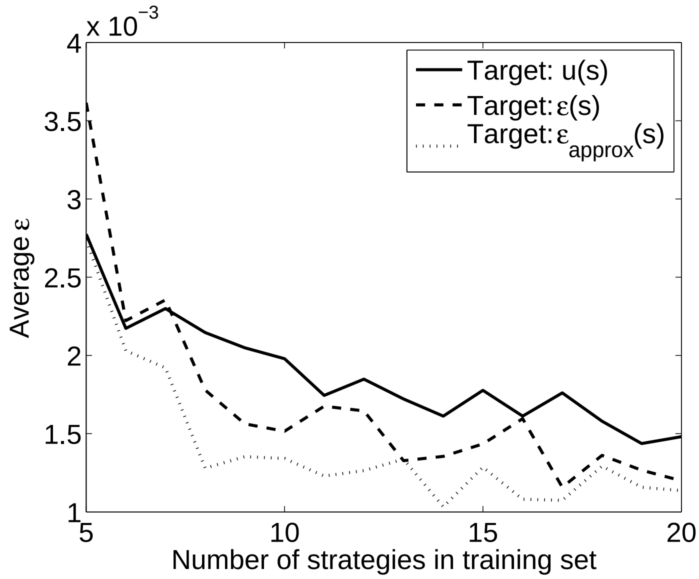


Figure 8.15: Comparison of learning different targets on a noiseless FPSB data set of varying size.

method still outperformed payoff function approximation may be evidence against this conjecture.

Another possibility is that the worst-case nature of $\epsilon(a)$ may tend to introduce irregularities into a data set that make the learning task more difficult in terms of the goal of subsequent equilibrium approximation, and restricting the set of possible deviations has a smoothing effect that actually facilitates learning. From limited exploration, I found some evidence supporting this conjecture. However, I can at this point say very little in general about this phenomenon.

In another set of experiments, I added noise, $\eta \sim N(0,1)$, to the payoff data, affecting the learning of payoff functions and approximate $\epsilon(s)$. The remaining setup of these experiments was as above, and I present the performance of learning $\epsilon(s)$ based on noiseless samples from the actual function as a part of the plots for calibration. The results are presented in Figures 8.16, 8.17, and 8.18 for the settings of $k = 1, 4$, and 10 respectively. According to these plots approximating Nash equilibria based on payoff function approximation yielded results that showed generally better performance when variance was relatively high than using the same data set to derive an $\epsilon(a)$ as the learning target. On the other hand, when variance was small (Figure 8.18) or zero (Figure 8.15),

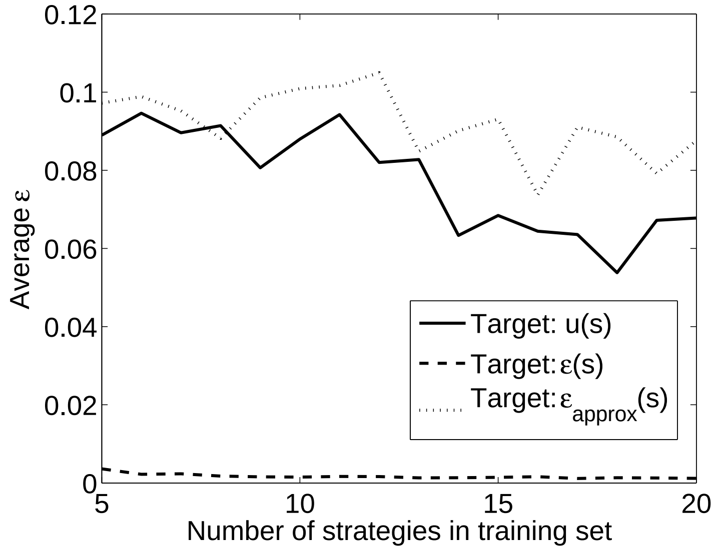


Figure 8.16: Comparison of learning different targets in FPSB when data has additive noise of the form $N(0, 1)$.

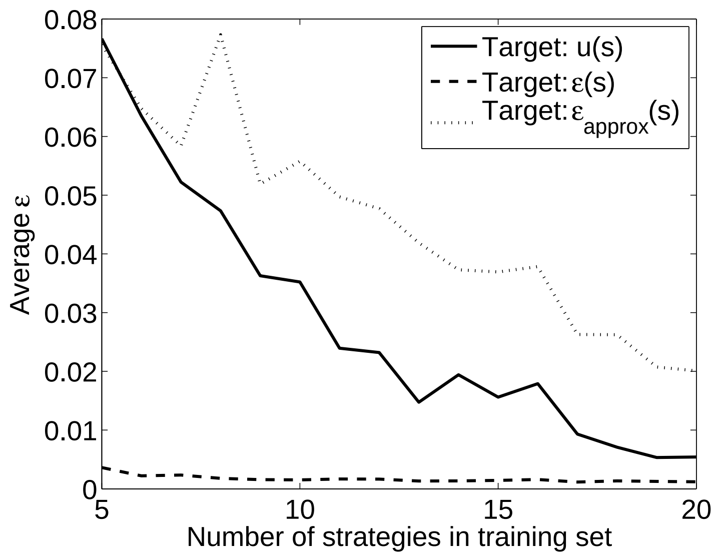


Figure 8.17: Comparison of learning different targets in FPSB when data has additive noise of the form $N(0, 0.0625)$.

approximate $\epsilon(a)$ became nearly as good or even better on average.

Overall, the results in this section provide a first limited exploration into the relative performance of learning the regret function, $\epsilon(a)$, suggesting that at least in certain settings it could be reasonably effective.

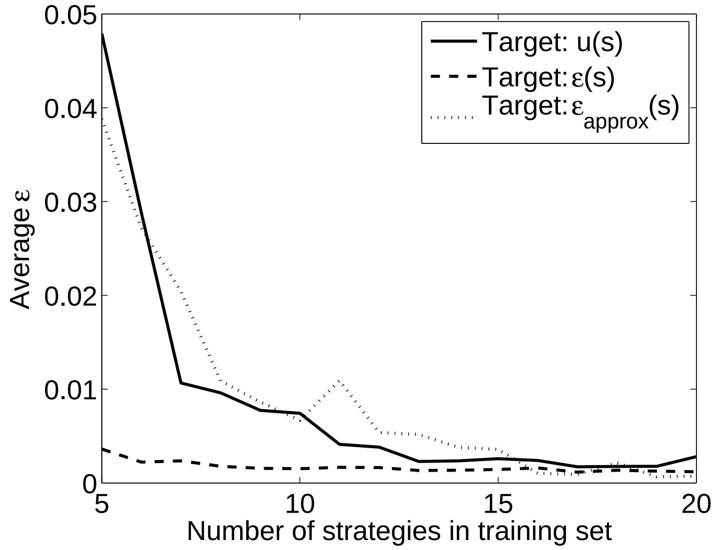


Figure 8.18: Comparison of learning different targets in FPSB when data has additive noise of the form $N(0, 0.01)$.

8.6 Future Work

8.6.1 Active Learning

In the next chapter I explore stochastic search methods for generating an empirical game which can then be used as data for payoff function and subsequent Nash equilibrium approximation. I note that my assumption is that the tasks of search and equilibrium approximation are not tightly coupled. This assumption may well not be very good, and, perhaps, particularly poor in the case of learning. Indeed, function approximation is often enhanced when we can select data points to sample in order to minimize the variance of the approximate function (thereby also reducing the mean squared error, since the bias of the model space remains constant). A number of techniques had been developed that leverage the specific properties of the model class under consideration in order to selectively sample the function domain [Cohn *et al.*, 1996]. These fall under the rubric of *active learning*.

The primary goal of active learning is to improve the expected quality of function fit in terms of mean squared error. Our goal, however, is somewhat different. We do not necessary care whether the quality of function fit is improved, as long as we can extract a better approximation of a Nash equilibrium from the approximate payoff function. While the two goals are intimately related (a poor function fit will generally not represent the strategic interactions very well), the techniques for achieving each may well be different.

On the other hand, I have considered an alternative setting of approximating the $\epsilon(a)$ function of the game, rather than the payoff functions of players, from data. In this setting, the goals of active learning and equilibrium approximation are considerably closer than in the setting of payoff function approximation. In the future, I intend to explore such connections between the two active learning goals and expect to engage in an experimental study of the relative utility of different active learning methods with respect to our end goals.

8.6.2 Learning Bayes-Nash Equilibria

Let us suppose that we have an extended data set of experience, (a, t, U), where t is the vector of player types that play the strategy profile a and accrue the resulting payoffs, U. Given such a data set, we can extend the input to our function approximation models to include the players' type vector, t as the additional set of inputs. A particularly appealing model class is one studied by Reeves and Wellman [2004]. For any model in this class Reeves and Wellman derived a method for computing a best response to a piecewise-linear strategy function (of players' types) exactly. This best-response finder can also be used in an iterative best-response dynamic to find Bayes-Nash equilibria.

If we know the actual distribution of players' types, we can use it directly in an equilibrium computation tool, whereas an unknown type distribution can be estimated from the data.

8.7 Conclusion

The results in both the FPSB and market-based scheduling games suggest that when data is sparse, regression methods can provide better approximations of the underlying game than direct estimation of a discrete model from the given data set. I observed this in cases where data points corresponded to actual or supposed actual payoff values, as well as cases where the given data comprises noisy samples of such payoffs. In the FPSB setting, I was able to artificially control for the amount of noise present in the data set, observing the dynamics of the payoff function approximation methods as the noise in the system decreased. A particularly interesting result in this case was that low-degree polynomial models exhibited an increasing advantage over discrete approximation as the variance of noise decreased.

In the market-based scheduling setting, I obtained two sets of training data with varying number of samples per profile. As a result, I was able to test my methods both in a noiseless setting, by using the data with the same number of samples per profile for both training and testing, and in a noisy setting, by using data with fewer samples per profile as the training set. I found that local regression and SVM learning methods generally outperformed discrete approximation in my experiments, both on noiseless and noisy training data. While locally weighted average shined when training data contained no noise, SVM model yielded better performance when noise was present.

I also introduced strategy aggregation as a way to control regression model complexity. The comparison between the various degrees of strategy aggregation provided mixed conclusions. In several settings, with and without noise, aggregating the strategies of other players appeared to provide no advantage in function approximation. In one setting, however, I did observe improved quality of equilibrium approximation when a particular form of strategy aggregation was used, as compared to no aggregation at all.

Having used the quality of the approximate equilibrium, ϵ, as a metric for the success of functional fit, I felt it natural to consider it explicitly as a model selection criterion. I evaluated it in comparison to model selection based on mean squared error of the payoffs on a test data set. My results in the FPSB setting, however, were not favorable, as model selection based on the more direct measure of function accuracy provided better average performance.

Finally, I considered the possibility of learning $\epsilon(a)$ directly, rather than deriving it from a learned payoff-function model. In a series of experiments using FPSB as the game of choice, I observed mixed results. Whereas the noisy environment favored payoff function approximation, $\epsilon(a)$ performed relatively well when the data contained little or no noise.

In summary, I performed a series of experiments comparing several methods of payoff function regression, an alternative approach of learning $\epsilon(a)$, and exploring the possibility of using ϵ as a model selection criterion. Regression or other generalization methods offer the potential to extend game-theoretic analysis to strategy spaces (even infinite sets) beyond directly available experience. By selecting target functions that support tractable equilibrium calculations, I render such analysis analytically convenient. By adopting functional forms that capture known structure of the payoff function (e.g., symmetry or strategy aggregation), I facilitate learnability. By using $\epsilon(a)$ as a target of function approximation, we may in some cases be able to exploit structure that may not be as apparent in the more direct payoff function. This chapter provides some evidence of the efficacy of the different methods that I considered and the tradeoffs between them that emerge in different environments.

CHAPTER 9

Stochastic Search Methods for Approximating Equilibria in Infinite Games

IN WHICH *I describe a convergent algorithm based on a hierarchical application of simulated annealing for approximating Nash equilibria (if they exist) in simulation-based games with finite-dimensional strategy sets. Additionally, I present alternative algorithms for best response and Nash equilibrium approximation, with a particular focus on one-shot infinite games of incomplete information.*[1]

In this chapter, I introduce several general-purpose techniques for search in the profile space of infinite games, with a particular focus on one-shot infinite games of incomplete information. The ultimate goal of these techniques is to produce an empirical game which yields (via empirical regret minimization) good approximations of Nash equilibria. All of my techniques rely on a best response approximation subroutine, as I schematically show in Figure 9.1. The methods that I describe all take as input a black-box specification of the players' payoff functions (and type distributions, if we are dealing with a game of incomplete information) and output an empirical game which can then be used to approximate a Nash equilibrium. In a fairly general setting, I am able to demonstrate theoretical convergence (in the regret metric) of one of the methods to an actual Nash equilibrium. Additionally, I experimentally demonstrate efficacy of all methods I study.

[1] The material in this chapter is taken from Vorobeychik and Wellman [2008].

The experimental evidence focuses on relatively simple auction settings, with most examples involving only one-dimensional types, and, consequently, allowing the use of low-dimensional strategy spaces. It is as yet unclear how my approaches will fare on considerably more complex games, as such games are not easily amenable to an experimental evaluation.

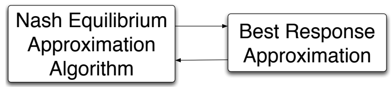


Figure 9.1: Approximation of Nash equilibria using a best response approximation subroutine.

For all the results that follow, I assume that strategy sets R_i of each player i are restricted to be finite-dimensional, that is, $R_i \subset \mathbb{R}^n$ for each i.

9.1 Best Response Approximation

Best-response approximation is a subroutine in all the methods for equilibrium approximation I discuss below. Thus, I first describe this problem in some detail and present a globally convergent method for tackling it.

9.1.1 Continuous Stochastic Search for Black-Box Optimization

At the core of my algorithms for best response approximation lies a stochastic search subroutine which can find an approximate maximizer of a black-box objective function on continuous domains. The topic of black-box continuous optimization has been well-explored in the literature [Spall, 2003]. In this work, I utilize two algorithms: stochastic approximation and simulated annealing. The overall approach, of course, admits any satisfactory black-box optimization tool that can be effective in continuous settings. A part of my goal is to assess the relative performance of a local and a global search routine.

Stochastic Approximation

Stochastic approximation [Spall, 2003] is one of the early algorithms for continuous stochastic search. The idea of stochastic approximation is to implement gradient descent algorithms in the context of a noisy response function. While the application of gradient descent techniques to noiseless numeric optimization has very old roots, its application to domains with noise was first studied by Robbins and Monro [1951] and Kiefer and Wolfowitz [1952]. The work by Kiefer and Wolfowitz is particularly interesting, as it demonstrates convergence in the case where the gradient is approximated using a finite-difference method (i.e., the difference of function value at neighboring points serves as the approximate gradient at a query point). As with all gradient descent algorithms, convergence is guaranteed only to a local optimum. However, together with random restarts and other enhancements, stochastic approximation can perform reasonably well even in global optimization settings.

Simulated Annealing

Simulated annealing is a well-known black-box optimization routine [Spall, 2003] with provable global convergence [Ghate and Smith, 2008]. Simulated annealing takes as input an oracle, f, that evaluates candidate solutions, a set X of feasible solutions, a candidate kernel $K(X_k, \cdot)$ which generates the next candidate solution given the current one, X_k, and the temperature schedule t_k that governs the Metropolis acceptance probability $p_k(f(x), f(y))$ at iteration k, defined as

$$p_k(f(x), f(y)) = \begin{cases} \exp[-\frac{f(y)-f(x)}{t_k}] & \text{if } f(y) < f(x), \\ 1 & \text{if } f(y) \geq f(x), \end{cases}$$

where $f(\cdot)$ is the (possibly noisy) evaluation of the function to be maximized, x is the current candidate, and y is the next candidate. It then follows a 3-step algorithm, iterating steps 2 and 3:

1. Start with $X_0 \in X$.

2. Generate Y_{k+1} using candidate kernel $K(X_k, \cdot)$.

3. Set $X_{k+1} = Y_{k+1}$ with Metropolis acceptance probability $p_k(f(X_k), f(Y_{k+1}))$, and $X_{k+1} = X_k$ otherwise.

In my applications of the simulated annealing routine, I define the solution space to be $X = [0, 1]$. My candidate kernel is a modified normal distribution with support contracted to [0,1]. I set the initial variance to be 0.5, and decrease it at every iteration.[2]

9.1.2 Globally Convergent Best Response Approximation

I present the application of simulated annealing search to the problem of best response in games in Algorithm 1, where t_k be a schedule of temperatures, and n_k be a schedule of the number of samples used to evaluate the candidate solutions at iteration k.

Algorithm 1 BR($\mathcal{O}, R_i, r_{-i}, K(\cdot,\cdot), t_k, n_k$)

1: Start with $a_0 \in R_i$
2: For $k > 0$, generate $b_{k+1} \in R_i$ using $K(a_k, \cdot)$
3: Generate $U_1 = \hat{u}_{n_k,i}(a_k, r_{-i})$ and $U_2 = \hat{u}_{n_k,i}(b_{k+1}, r_{-i})$ from $\mathcal{O}$
4: Set $a_{k+1} \leftarrow b_{k+1}$ w.p. $p_k(U_1, U_2)$ and $a_{k+1} \leftarrow a_k$ o.w.

For the analysis below, I need to formalize the notion of a *candidate Markov kernel*, $K(\cdot)$, which defines the distribution over the next candidate given the current:

Definition 9.1 *A function $K : A \times \mathcal{B} \to [0, 1]$ is a* candidate Markov kernel *if $A \subset R^n$ and $\mathcal{B}$ is a Borel σ-field over A. The first argument of $K(\cdot, \cdot)$ is the current candidate, and the second is a subset of candidates, for which K gives the probability measure.*

In order for simulated annealing to have any chance to converge, the kernel must satisfy several properties, in which case I refer to it as an *admissible kernel*.

Definition 9.2 *A kernel $K : A \times \mathcal{B} \to [0, 1]$ is* admissible *if*

- *K is absolutely continuous in second argument*

[2]For the convergence result below, I assume that this variance is constant, but a decreasing schedule has been shown better in practice [Spall, 2003].

- *Its density is uniformly bounded away from 0, that is* $K(x, B) = \int_B g(x, y)dy$ *with* $\inf_{x,y \in A} g(x, y) > 0$*, and*

- *For every open* $B \subset A$*,* $K(x, G)$ *is continuous in* x*.*

To gain some intuition about the admissibility criterion, let's look at an example of an admissible kernel.

Example 9.3 *Suppose* $A = [a, b]^n$ *with* $-\infty < a < b < \infty$*. Define the kernel to be*

$$K(x, B) = \frac{\int_B g(x, y)^n \prod_{j=1}^n dy_j}{\int_{[a,b]^n} g(x, y)^n \prod_{j=1}^n dy_j}. \tag{9.1}$$

where

$$g(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-y)^2}{2\sigma^2}\right\}$$

with σ^2 *denoting the variance. Clearly, since the interval is bounded, the Normal density is everywhere positive for any mean* x*. Since the density is continuous in* x*, so is* K*, and absolute continuity is trivial since Lebesgue measure zero sets integrate to 0.*

The following conditions map directly to the sufficient conditions for global convergence of simulated annealing observed by Ghate and Smith [2008]:

1. *EXISTENCE* holds if R_i is closed and bounded and the payoff function $u_i(r_i, r_{-i})$ is continuous on R_i. This condition is so named because it implies that the best response exists by the Weierstrass theorem.

2. *ACCESSIBILITY* holds if for every maximal $a^* \in R_i$ and for any $\epsilon > 0$, the set $\{a \in R_i : \|a - a^*\| < \epsilon\}$ has positive Lebesgue measure.

3. *DECREASING TEMPERATURES (DT)* holds if the sequence t_k of temperatures converges to 0.

4. *CONVERGENCE OF RELATIVE ERRORS (CRE)* holds if the sequences

$$\frac{|\tilde{u}_{n_k,i}(a_k, r_{-i}) - u_i(a_k, r_{-i})|}{t_k}$$

and

$$\frac{|\tilde{u}_{n_k,i}(b_{k+1}, r_{-i}) - u_i(b_{k+1}, r_{-i})|}{t_k},$$

where b_{k+1} is the next candidate generated by the kernel, converge to 0 in probability.

The first two conditions ensures that the global optimum actually exists and can be reached by random search with positive probability. The third and fourth conditions ensure that the iterates stabilize around optima, but do so slowly enough so that the noise does not lead the search to stabilize in suboptimal neighborhoods.

Theorem 9.4 (Ghate and Smith [2008]) *If the problem satisfies* EXISTENCE *and* ACCESSIBILITY*, and the algorithm parameters satisfy* DT*, and* CRE*, Algorithm 1 utilizing an admissible candidate kernel converges in probability to* $u_i^*(r_{-i}) = \max_{a \in R_i} u_i(a, r_{-i})$.

Proof. All the conditions map directly to the sufficient conditions for convergence of simulated annealing in a stochastic setting shown by Ghate and Smith [2008]. □

Let $\hat{u}_{i,k}(r_{-i})$ to be the answer produced when Algorithm 1 is run for k iterations. This will be the estimate of $u_i^*(r_{-i})$, which, by Theorem 9.4, is consistent.

9.2 Nash Equilibrium Approximation

My goal in this chapter is to take a simulation-based game as an input and return a profile constituting an approximate Nash equilibrium in the underlying game—that is, the game constructed from expected payoffs in the simulation-based game. More formally, I take as input a specification of a simulation-based game, $[I, \{R_i\}, \mathcal{O}]$ and the maximum number of allotted queries from the oracle, Q, and output $\hat{r} \in R$, such that $\epsilon(\hat{r})$ is small (as evaluated with respect to the underlying game). Below, I present two general approaches to Nash equilibrium approximation: a well-known iterative best response approach, as well as my own algorithm based on simulated annealing. Both of these can use *any* best response approximation algorithm as a subroutine.

Since I take Q as input, I am faced with a general problem of dividing up the stock of queries between taking multiple samples to improve the estimate of $u(r)$ and exploring the profile space to obtain sufficient information about the game to effectively approximate a Nash equilibrium. I do not address this "division" problem in a principled way in this chapter, but rather heuristically allocate samples. In the convergence analysis, I simply ensure that samples are taken at a sufficiently increasing rate, whereas in my experiments, I merely keep this rate fixed.

9.2.1 Equilibrium Approximation via Iterated Best Response

While the problem of best-response approximation is interesting in its own right, it may also be used iteratively to approximate a Nash equilibrium. The procedure of *iterative best response*, which I describe in Section 6.4.1 in the context where best response can be computed exactly, can be applied directly by iterating *approximate* best responses. When the procedure terminates after a finite number of steps K, I return the final iterate r_K as an approximate Bayes-Nash equilibrium. The presumption is that after enough steps the best response dynamics roughly converges, although in Chapter 6 I had already observed that there are few general convergence guarantees for iterative best response even when the best response is computed exactly at each step.

9.2.2 A Globally Convergent Algorithm for Equilibrium Approximation

In this section I develop a globally convergent algorithm for computing approximate Nash equilibria in infinite games. The approach I take, visualized in Figure 9.2, is to minimize the approximate regret, $\hat{\epsilon}(r)$, where approximations are produced by running Algorithm 1. For the task of minimizing regret I again use an adaptation of simulated

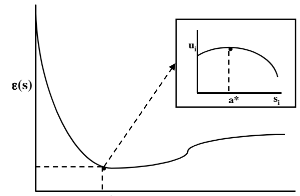


Figure 9.2: A diagrammatic view of my algorithm based on approximate regret minimization.

annealing, but now need to establish the convergence conditions for this meta-problem.

First, let me define a candidate kernel for this problem as a combination of admissible kernels for each agent i:

$$K(x, B) = \int_B \prod_{i \in I} g_i(x, y_i) \prod_{i \in I} dy_i, \tag{9.2}$$

where $K^i(x, C) = \int_C g_i(x, y)dy$, where $g_i(\cdot)$ denotes the Kernel density used by the simulated annealing routine for player i. I now confirm that the resulting kernel is admissible, a result that will allow us to focus only on defining an admissible kernel on each player's strategy set.

Lemma 9.5 *The candidate kernel defined in Equation 9.2 is admissible.*

Proof. Since each $g_i(x, y_i)$ is positive everywhere, so is the product. Furthermore, it is clear that if B is of measure-zero, then so is $K(x, B)$. Finally, $K(x, B)$ is continuous on x since each $g_i(x, y_i)$ is continuous and therefore so is the product. $\square$

Given the candidate kernels for each player and the constructed candidate kernel for the regret minimization problem, I present a meta-algorithm—Algorithm 2—for approximating Nash equilibria in infinite games.

Algorithm 2 EQEstimate($\mathcal{O}, S, K(\cdot,\cdot), K^i(\cdot,\cdot), t_l, n_l, t_k^i, n_k^i$)

```
Start with r_0 ∈ R_i
Generate q_{l+1} ∈ R_i using K(r_l, ·)
Generate û_{n_l,i}(r_l) and û_{n_l,i}(q_{l+1}) from O
Let û_{i,l}(r_{-i,l}) ← BR(O, r_{-i,l}, R_i, K^i(·,·), t_k^i, n_k^i)
Let û_{i,l}(q_{-i,l+1}) ← BR(O, q_{-i,l}, R_i, K^i(·,·), t_k^i, n_k^i)
Set r_{l+1} ← q_{l+1} w.p. p_l(r_l, q_{l+1}) and r_{l+1} ← s_l o.w.
```

I now present the sufficient conditions for convergence of Algorithm 2. First, I verify what properties on the underlying game are needed for continuity of $\epsilon(r)$.

Lemma 9.6 *If $u_i(r)$ are uniformly continuous on R for every i, then $\epsilon(r)$ is continuous on R.*

Based on Lemma 9.6, we need to modify the *EXISTENCE* criterion slightly as follows:

*EXISTENCE** holds if R_i is closed and bounded and the payoff function $u_i(r_i, r_{-i})$ is uniformly continuous on R_i for every player i. This condition is so named because it implies that the minimum exists by the Weierstrass theorem.

Since I am concerned about every player now and, furthermore, need to avoid "undetectable" minima in $\epsilon(r)$, I also modify the *ACCESSIBILITY* condition:

*ACCESSIBILITY** holds if for any $\delta > 0$, for every profile r, for every player i, and for every maximal $a^* \in R_i$ the set $\{a \in R_i : \|a - a^*\| < \delta\}$ has positive Lebesgue measure; furthermore for every minimal $r^* \in R$ the set $\{r \in R : \|r - r^*\| < \delta\}$ has positive Lebesgue measure.

We also need to augment the conditions on algorithm parameters to include both the conditions on the parameters for the problem of minimizing $\epsilon(r)$, as well as the conditions on parameters for finding each player's best response. For clarity, I will let l denote the iteration number of the meta-problem of minimizing $\epsilon(r)$ and k denote the iteration number of the best response subroutine.

- *DECREASING TEMPERATURES* (DT*)* holds if for every agent i the sequence t_k^i of temperatures converges to 0, and the sequence t_l of temperatures converges to 0

- *CONVERGENCE OF RELATIVE PAYOFF ERRORS (CRPE)* holds if for every agent i the sequences of ratios

$$\frac{|\hat{u}_{n_k^i,i}(a_k, r_{-i}) - u_i(a_k, r_{-i})|}{t_k^i}$$

and

$$\frac{|\hat{u}_{n_k^i,i}(b_{k+1}, r_{-i}) - u_i(b_{k+1}, r_{-i})|}{t_k^i},$$

where b_{k+1} is the next candidate generated by the kernel, converge to 0 in probability.

Now, define

$$\hat{\epsilon}_l(r) = \max_{i \in I}[\hat{u}_{i,n_l}(r_{-i}) - \hat{u}_{i,n_l}(r)].$$

Lemma 9.7 *If* EXISTENCE*, ACCESSIBILITY*, DT*, *and* CRPE *hold,* $\hat{\epsilon}_l(r)$ *converges to* $\epsilon(r)$ *in probability for every* $r \in R$.

I need one more condition on the algorithm parameters before proving convergence:

CONVERGENCE OF RELATIVE EPSILON ERRORS (CREE) holds if the sequences of ratios

$$\frac{|\hat{\epsilon}_{l,i}(r_k, r_{-i,k}) - \epsilon_i(r_k, r_{-i,k})|}{t_k}$$

and

$$\frac{|\hat{\epsilon}_{n_k,i}(r_{k+1}, r_{-i,k}) - \epsilon_i(r_{k+1}, r_{-i,k})|}{t_k},$$

where r_{k+1} is the next candidate generated by the kernel, converge to 0 in probability.

Theorem 9.8 *Under the conditions* EXISTENCE*, ACCESSIBILITY*, DT*, CRPE, *and* CREE, *Algorithm 2 converges to* $\bar{\epsilon} = \min_{r \in R} \epsilon(r)$.

Proof. While Ghate and Smith [2008] prove convergence for functions which are expectations of the noisy realizations, their proof goes through unchanged under the above sufficient conditions, as long as we ascertain that $\hat{\epsilon}_l \to \epsilon(r)$ for every $r \in R$. This I showed in Lemma 9.7. □

Corollary 9.9 *If there exists a Nash equilibrium on* R, *Algorithm 2 converges to a Nash equilibrium when the conditions* EXISTENCE*, ACCESSIBILITY*, DT*, CRPE, *and* CREE *obtain.*

9.3 Examples

9.3.1 Mixed Strategy Equilibria in Finite Games

Consider finite normal form games with exact payoff realizations for every pure strategy profile $a \in A$. Since the payoff functions are uniformly continuous on the mixed strategy space (see Claim E.2 in the appendix), and since we can find $\epsilon(s)$ exactly for any mixed strategy s if the strategy sets are small enough, convergence only requires a proper choice of the kernel and the choice of a temperature schedule which converges to 0. Algorithm 2 will then converge to a Nash equilibrium for an arbitrary finite game in normal form. A similar idea for this domain that had at its root an optimization routine was AMOEBA, which uses the Nedler-Mead non-linear optimization technique [Reeves, 2005], and the Liapunov minimization method described in Section 6.3.2. However, Nedler-Mead has few known convergence guarantees [Spall, 2003], although it does perform well in many practical settings, and the Liapunov method has typically been implemented using algorithms which only guarantee local convergence. Of course, many other very effective convergent algorithms exist for finding or approximating equilibria in finite games. Thus, application of my methods to finite games is interesting mostly academically, but may well be less effective than the best known methods.

9.3.2 Symmetric Games

Most solution tools available do not account for symmetry in games. My algorithm can be easily modified to be considerably more effective in games that are symmetric if we restrict our focus to symmetric equilibria. In this case, we can restrict the domain of $\epsilon(r)$ to be the strategy set of one player and only look at one player's unilateral deviations. Running time for this modified algorithm in symmetric games is then independent of the number of players, which is an exponential win for Nash computation.

9.4 Infinite Games of Incomplete Information

Perhaps the most important application of the methods I have discussed is to infinite games of incomplete information. In what follows, I define one-shot games of incomplete information and adapt my methods to this domain. Additionally, I introduce another best response approximation method specifically designed for strategies that are functions of private information.

9.4.1 Best Response Approximation

Given a best response approximation subroutine, Algorithm 2 can be applied to the infinite games of incomplete information, although convergence can be guaranteed only when best response is approximating using Algorithm 1. Below, I describe two methods for approximating best response functions: the first is a direct adaptation of the techniques I described above; the second is based on regression. I note that both methods rely on an assumption that we can define a relatively low-dimensional hypothesis class for each player which contains good approximations of the actual best response. Later, I experimentally verify that this is indeed possible for a number of interesting and non-trivial games. More generally, an analyst may need to hand-craft low-dimensional restricted strategy sets in order to effectively apply my techniques.

9.4.2 Strategy Set Restrictions

Since I restrict the strategy sets to be subsets of $\mathbb{R}^n$, I cannot represent games of incomplete information perfectly in this restricted normal form. What I can do, however, is restrict the sets of strategies allowed for each player to a finite-dimensional function space on reals, and thereby parametrize each strategy using a vector $\theta_i \in \Theta_i \subset \mathbb{R}^n$. Let me denote the joint parameter space thus created by $\Theta = \Theta_1 \times \cdots \times \Theta_m$, and denote the restricted space of the incomplete information game by $\mathcal{H}_i$ for each player i. Then, $h_{\theta_i,i}(t) \in \mathcal{H}_i$ is a particular type-conditional strategy of player i. I aggregate over all players to obtain $h_\theta(t) = (h_{\theta_1,1}, \ldots, h_{\theta_m,m})$. I then describe a restricted game of incomplete information by $[I, \{\mathcal{H}_i\}, \{T_i\}, F(\cdot), \{u_i(\cdot)\}]$, where T_i is the set of player i's types and $F(\cdot)$ is the joint distribution over player types. This game can be mapped into the normal form, as I described in Chapter 2, by letting $A_i = \Theta_i$ and defining $\tilde{u}_i(\theta) = E_F u_i(h_\theta(t))$ for any $\theta \in \Theta$. Thus, the transformed game is $[I, \{\Theta_i\}, \{\tilde{u}_i(\theta)\}]$. Now, Algorithm 2 is directly applicable and will guarantee convergence to a strategy profile with the smallest expected regret.

Direct Method My first method for approximating best response functions in infinite games is simply an application of Algorithm 1. Here, the oracle $\mathcal{O}$ performs two steps: first, generate a type $t \in T$ from the black-box type distribution; and next, generate a payoff from the simulation-based payoff function for the strategy profile *evaluated* at t. As I have noted above, convergence to global best response function can be guaranteed in the finite-dimensional hypothesis class $\mathcal{H}_i$; indeed convergence obtains even for an arbitrary black-box specification of the strategies of other players.

Regression to Pointwise Best Response My second method takes an indirect route to approximating the best response, approximating best response *actions* for each of a subset of player types, and thereafter fitting a regression to these. The outline of this algorithm is as follows:

1. Draw L types, $\{t_1, \ldots, t_L\}$, from the black-box type distribution
2. Use simulated annealing to approximate a pointwise best response for each t_j, $\hat{s}_j$
3. Fit a regression $\hat{s}(t)$ to the data set of points $\{t_j, \hat{s}_j\}$

The regression $\hat{s}(t)$ is the resulting approximation of the best response function.

9.5 Experimental Evaluation of Best Response Quality

9.5.1 Experimental Setup

In this section, I explore the effectiveness of the two methods I introduced as best-response routines for infinite one-shot games of incomplete information. The best response methods were both allowed 5000 queries to the payoff function oracle for each iteration, and a total of 150 iterations. For both, the total running time was consistently under 4 seconds. I compared my methods using both stochastic approximation (indicated by "_stochApprox_" in the plots) and the simulated annealing stochastic search subroutines. Besides comparing the methods to each other, I include as a reference the results of randomly selecting the slope parameter of the linear best response. This method sets the intercept to zero, since many of the best response strategies, and particularly equilibria, in the games I consider have a zero-intercept. I want to emphasize that my goal is *not* merely to beat the random method, but to use it as *calibration* for the approximation quality of the other two.

I test my methods on three infinite auction games. The first is the famed Vickrey, or second-price sealed-bid, auction [Krishna, 2002], with payoff function specified in Equation 9.3, where m denotes the number of players and v the number of ties for highest bid.

$$u_i(a_i, a_{-i}, t_i, t_{-i}) = \begin{cases} \frac{1}{v}(t_i - \max_{j \neq i} a_j) & \text{if } a_i = \max_{j \neq i} a_j, \\ t_i - \max_{j \neq i} a_j & \text{if } a_i > \max_{j \neq i} a_j, \\ 0 & \text{otherwise.} \end{cases} \tag{9.3}$$

The second is first-price sealed-bid auction [Krishna, 2002], with payoff function specified in Equation 9.4.

$$u_i(a_i, a_{-i}, t_i, t_{-i}) = \begin{cases} \frac{1}{v}(t_i - a_i) & \text{if } a_i = \max_{j \neq i} a_j, \\ t_i - a_i & \text{if } a_i > \max_{j \neq i} a_j, \\ 0 & \text{otherwise.} \end{cases} \tag{9.4}$$

The final game to which I apply my techniques is a *shared-good auction*, with payoff function specified in Equation 9.5.

$$u_i(a_i, a_{-i}, t_i, t_{-i}) = \begin{cases} \frac{1}{v}(t_i - \frac{m-1}{m} a_i + \frac{1}{m} \max_{j \neq i} a_j) & \\ \quad\quad\quad\quad \text{if } a_i = \max_{j \neq i} a_j, & \\ t_i - \frac{m-1}{m} a_i & \text{if } a_i > \max_{j \neq i} a_j, \\ \frac{1}{m} \max_{j \neq i} a_j & \text{otherwise.} \end{cases} \tag{9.5}$$

I experimented with two- and five-player games with uniform type distributions, noting that the best-response finder proposed by Reeves and Wellman [2004] cannot be directly applied to games with more than two players. In my experiments with these auctions, I focus on the hypothesis class $\mathcal{H}$ of linear functions, with $\theta = (\alpha, \beta)$ such that $h_\theta(t_i) = \alpha t_i + \beta$, which includes an actual best response in many auction domains. In all best-response experiments, I sought a best response to a linear strategy of the form $s(t) = kt$, with k generated uniformly randomly in $[0, 1]$.

The results were evaluated based on regret, $\epsilon(s)$, computed on the underlying game. I took the average value of $\epsilon(s)$ over 100–1000 trials and based the statistical tests on these samples. Statistically significant difference at the 99% confidence level between my two methods when both use simulated annealing was indicated by a "*". In the plots, I also include 99% confidence intervals for further statistical comparisons.

9.5.2 Two-Player One-Item Auctions

My first experiment is largely a sanity check, as there is an *exact* best response finder for all three auction games I consider here [2004].[3] The results are shown in Figure 9.3. I group the initial results into two categories. The first category is comprised of settings in which there is a linear best response function. This is satisfied by Vickrey for any value of k, first-price sealed-bid auction (fpsb) with $k \geq 1/2$, and shared-good auction (sga) with $k \geq 2/3$.[4] In all of these settings, my best response approximations are orders of magnitude better than random. Indeed, *in every auction I study, the difference between all*

[3] There is not a significant difference in running times between the exact best response finder and my approximators.

[4] Recall that k is the randomly generated slope of the line to which I am approximating a best response function.

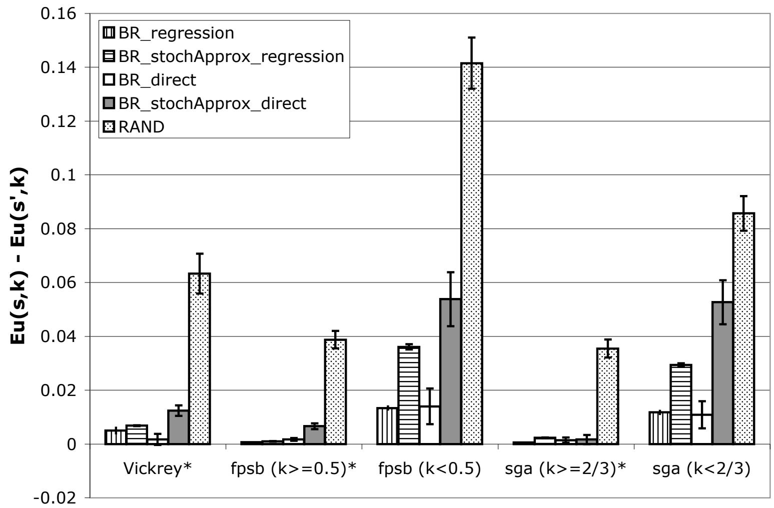


Figure 9.3: Comparison of best response methods in 2-player games with reserve price = 0.

of my methods (using stochastic approximation or simulated annealing) and random is quite statistically significant (p-value $< 10^{-10}$*).* Therefore, I omit the results for random from the subsequent figures. Additionally, in all but Vickrey, the regression-based method is better than direct, most likely because this method is particularly sample-efficient went the actual best response is linear and there are not many alternative best response options. This may seem surprising given that my evaluation metric favors the direct method. However, in such settings, fitting regression to pointwise best responses allows, perhaps, a more efficient use samples, particularly when the number of iterations is not too large. This observation, however, does not hold true for the Vickrey auction. I conjecture that the reason that the direct method outperformed the regression-based method here is that there is an infinite set of best response functions, and all but one are non-linear. Thus, the results of pointwise best responses need not in tandem produce the desired line.

The settings in the second category yield non-linear best response functions. All the remaining comparisons in Figure 9.3 fall into this category. As expected, the performance of linear best response approximation is somewhat worse here, although in all cases far better than random. It is worth noting that in all of these the direct method performs statistically no worse, and in several cases much better than the regression-based method. The result is intuitive, since the direct method seeks the most profitable linear best response, whereas regression presumes that linearity is a good fit for the *actual* best response, and may not do well when this assumption does not hold.

While it is good to see the effectiveness of my methods in settings for which exact numerical solvers exist, my goal is to apply them to problems for which no general-purpose numerical solver exists. My first such examples are two-player Vickrey and first-price sealed-bid auctions with reserve prices, denoted by vickrey_rp and fpsb_rp respectively (Figure 9.4).[5] In both of these, the direct method far outperforms the regression-based

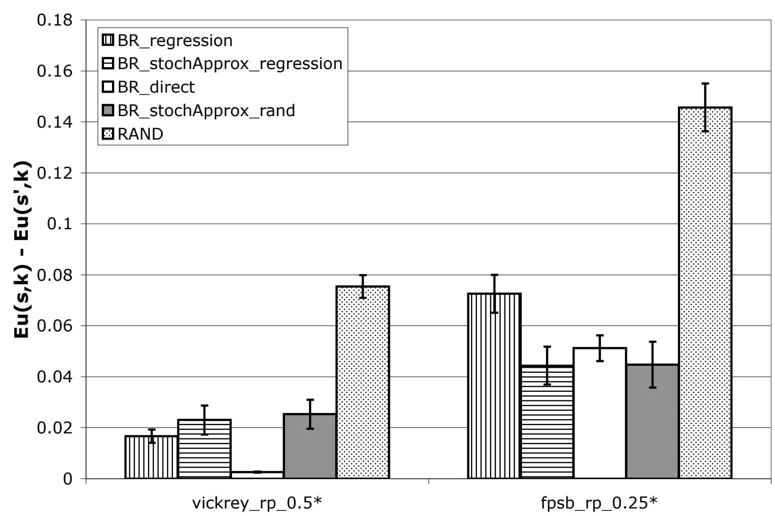


Figure 9.4: Comparison of best response methods in 2-player games with reserve price > 0. The reserve price for Vickrey is 0.5, while it is 0.25 for the first-price sealed-bid auction.

method. It appears, then, that the regression-based best response approximation method is at a considerable disadvantage by my evaluation metric when the actual best response is non-linear or when there are many best response functions, most of which are non-linear.

In nearly all cases simulated annealing yielded statistically significant improvement over stochastic approximation. Indeed, at times it was better by more than a factor of magnitude. The reason, I believe, is that stochastic approximation expends much of its computing budget estimating gradients, while simulated annealing can guide and make use of all the function evaluations it generates along the search path. Additionally, stochastic approximation is a local search method (even with random restarts, which

[5]Of course, both of these are analytically tractable. My study, however, is solely concerned with *numerical* methods for solving games.

considerably enhance its performance), and the problems at hand appear more suited to global search.

9.5.3 Five-Player One-Item Auctions

Since there is no general-purpose best response finder for five-player infinite games of incomplete information, the only viable comparison of my results is to each other. As we can see from Figure 9.5, in the five-player setting the direct method tends to

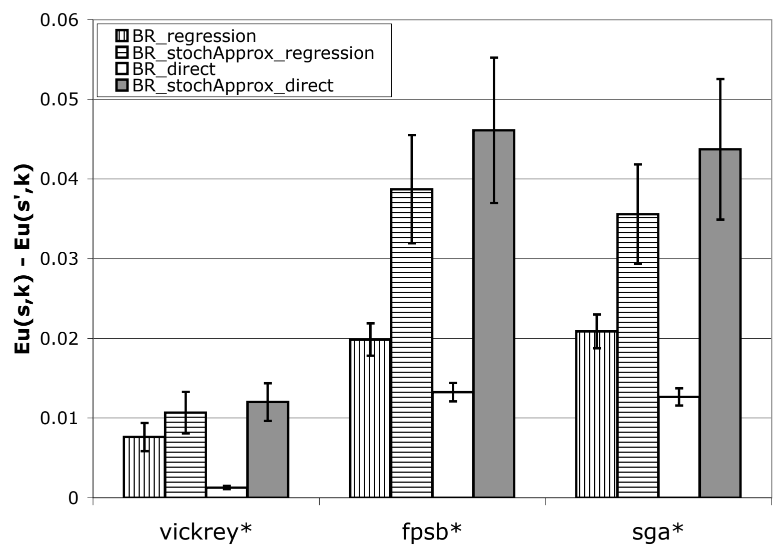


Figure 9.5: Comparison of best response methods in 5-player games with uniform type distribution.

produce substantially better approximate best response than the regression-based method. Additionally, in two of the three auctions in this setting, simulated annealing showed substantial advantage over stochastic approximation. Since these results echo those in smaller games, the reasons are likely to be the same.

9.5.4 Sampling and Iteration Efficiency

In this section, I compare the "regression" and "direct" methods in terms of efficiency in their use of both samples from the payoff function and iterations of the optimization algorithm. The results below are roughly representative of the entire set of results involving several auction games with varying numbers of players.

First, I consider sampling efficiency. As Figure 9.6 suggests, the "direct" method

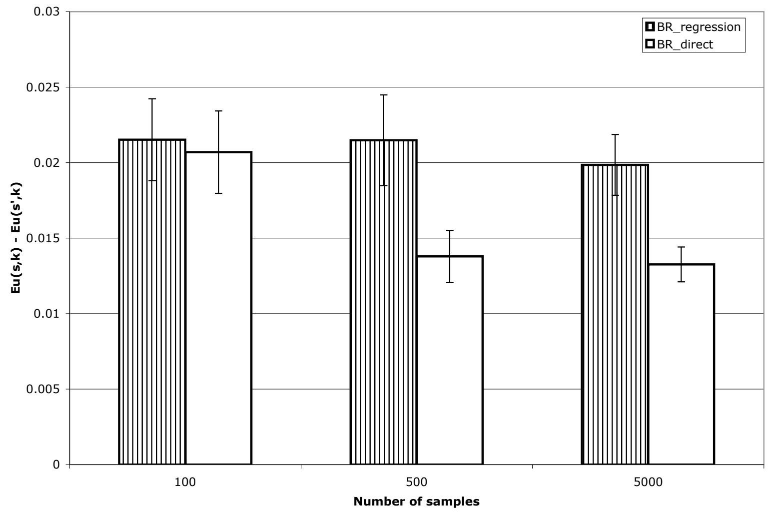


Figure 9.6: Comparison of sampling efficiency of best response methods in two-player first-price sealed-bid auctions.

seems no worse and at times substantially better than the "regression"-based method for various sample sizes I consider: when very few samples are taken, both methods seem to perform almost equally poorly, but as the number of samples per iteration increases, "direct" method quickly surpasses "regression".

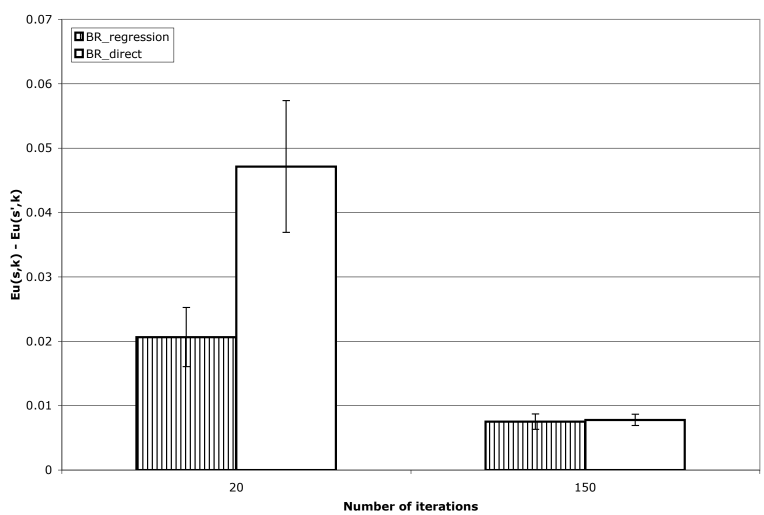


Figure 9.7: Comparison of iteration efficiency of best response methods in two-player first-price sealed-bid auctions.

Iteration efficiency results are presented in Figure 9.7. Interestingly, these results appear somewhat different from those for sampling efficiency: the "direct" method seems particularly affected by a dearth of iterations, while "regression" is quite robust. Note, however, that even in the case of sampling efficiency, regression is quite robust across different sample sizes; its flaw is that it fails to take sufficient advantage of additional sampling. I conjecture that the robustness of regression is partly because linear approximation of actual best response is somewhat reasonable in this setting, and, if so, regression smoothes out the noise much better when iterations are few.

9.6 Experimental Evaluation of Equilibrium Quality

9.6.1 Experimental Setup

I now turn to an application of best response techniques to Bayes-Nash equilibrium approximation in infinite one-shot games of incomplete information. Since the games I consider are all symmetric, I focus on approximating *symmetric* equilibria, that is, equilibria in which all agents adopt the same strategy. If we can compute a best response to a particular strategy, we can use iterated best response dynamics to find equilibria, assuming, of course, that the dynamics converge. Here, I will avoid the issue of convergence by taking the last result of five best response iterations as the final approximation of the Bayes-Nash equilibrium. In all cases, I seed the iterative best response algorithm with truthful bidding, i.e., $s(t) = t$. All other elements of experimental setup are identical to the previous section. *As before, in every application the difference between both of my methods and random is quite statistically significant and the actual experimental difference is always several orders of magnitude*.

9.6.2 Two-Player One-Item Auctions

I first consider three two-player games for which we can numerically find the exact best response and two for which we cannot. I present the results in Figure 9.8. In all three games that can be solved exactly (vickrey, fpsb, and sga), the regression-based method outperforms the direct method. In the case of Vickrey auction with reserve price of 0.5, this result is reversed, and the performance of the two methods on first-price sealed-bid

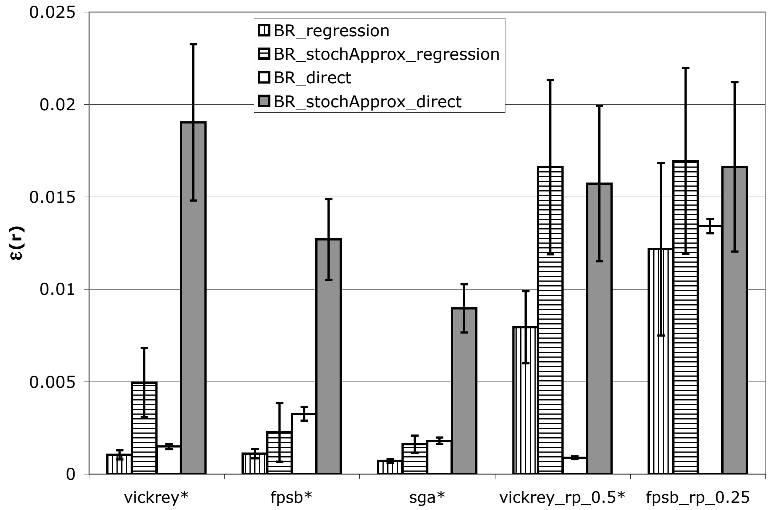


Figure 9.8: Comparison of equilibrium approximation quality in two-player games.

auction with reserve price of 0.25 is not statistically different. Since in the first three cases the best response to truthful bidding is the linear equilibrium bid function, we may again be observing a more efficient use of samples on the part of the regression-based method. When the reserve price is introduced, there is a large set of types indifferent between an infinite number of best response strategies (since they will not get the good anyway), and we have previously seen such settings to be advantageous to the direct method. As I have observed previously, simulated annealing tends to be considerably better than stochastic approximation.

9.6.3 Five-Player One-Item Auctions

Now I consider five-player games, for which no general-purpose numerical tool exists to compute a Bayes-Nash equilibrium or even a best response. The results are presented in Figure 9.9. While the two methods are statistically indistinguishable from one another, the direct method was considerably better than the regression-based method in Vickrey auction experiments, and the opposite was true in the experiments involving the first-price sealed-bid auction. Interestingly, we can observe here considerable advantage from using simulated annealing, as compared to stochastic approximation, in all instances for both the regression-based and the direct methods.

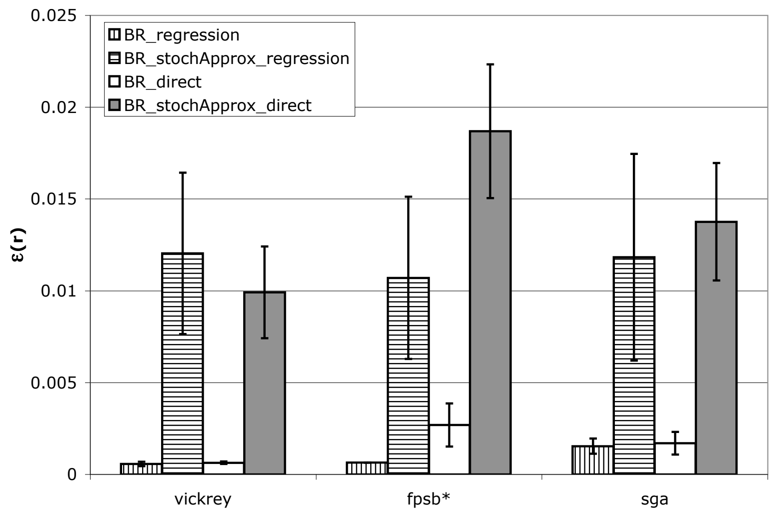


Figure 9.9: Comparison of equilibrium approximation quality in five-player games with uniform type distribution.

9.6.4 Two-Player Two-Item First-Price Combinatorial Auction

In all the experiments above I faced auctions with one-dimensional player types. Here, I apply my methods to a first-price combinatorial auction—a considerably more complex domain—although I restrict the auction to two players and two items. I allocate the items between the two bidders according to the prescription of *winner determination problem* [2006], which takes a particularly simple form in this case. I further restrict the problem to bidders with complementary valuations. Specifically, each bidder draws a value v^i for each item i from a uniform distribution on [0,1] and draws the value for the bundle of both items v^b from the uniform distribution on $[v^1 + v^2, 2]$. I let each player's value vector be denoted by $v = \{v^1, v^2, v^b\}$.

Since the game is symmetric, I seek a symmetric approximate equilibrium. Since the joint strategy space is an intractable function space, I restrict the hypothesis class to the functions of the form:

$$b^1(v) = k_1 v^1; b^2(v) = k_2 v^2; b^b(v) = b^1 + b^2 + k_3(v^b - b^1 - b^2).$$

Unlike the experiments above, verifying the approximation quality with respect to actual best responses is extremely difficult in this case. Thus, I instead measure the quality of the approximations against the best possible in the restricted strategy space. Finally, I use here the direct method with simulated annealing as the best response approximation tool. As we can see from Figure 9.10, the best response dynamics appears to converge

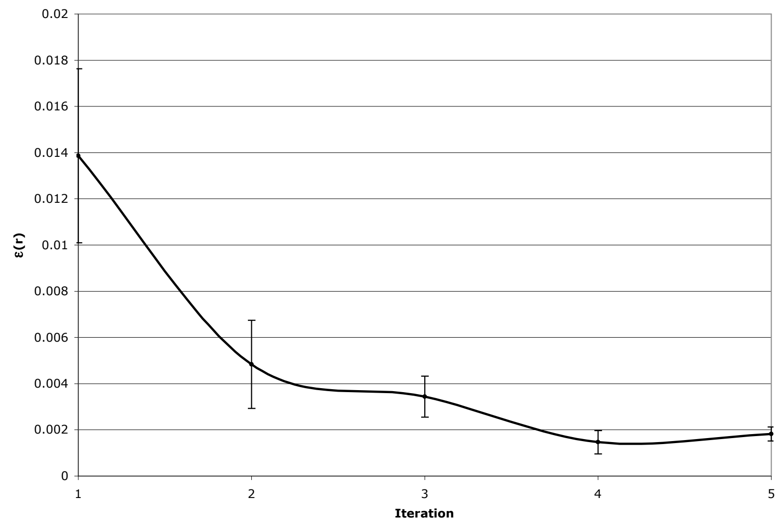


Figure 9.10: Iterations of approximate best response dynamics in the combinatorial auction.

quite quickly on the restricted strategy space. Thus, at least according to this limited criterion, the direct method is quite effective in approximating equilibria even in this somewhat complicated case for which no general analytic solution is known to date.

9.7 Experimental Comparison of Equilibrium Approximation Methods

In this section I compare the approximation quality of best response dynamics and Algorithm 2—using simulated annealing in one case, and stochastic approximation in another.[6] In this setup, both best response dynamics and Algorithm 2 use the *direct* approximate best response method as a subroutine, and I only look at the five-player first-price sealed-bid auction (although I do not expect the results to be very different for the other auction domains above). As we can see from Figure 9.11, while not guaranteed to converge in general, best response dynamics seems more effective than Algorithm 2; thus,

[6] Naturally, when stochastic approximation is used, we lose the global convergence properties.

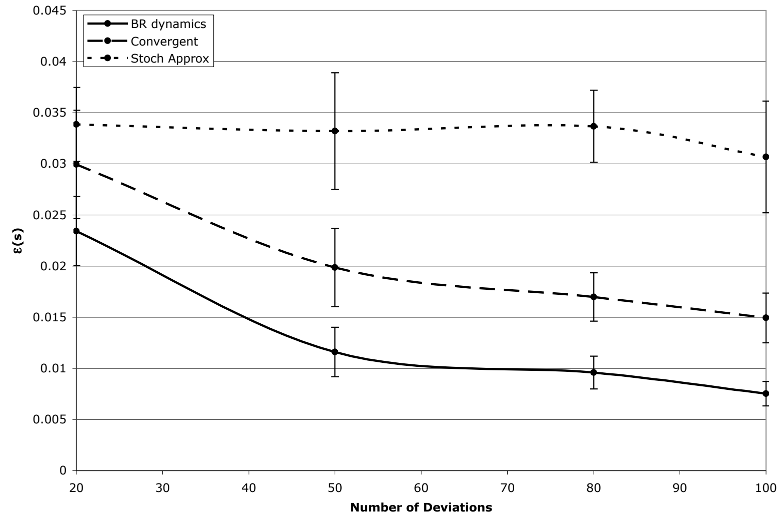


Figure 9.11: Comparison of equilibrium approximation quality of best response dynamics and the convergent method.

while convergence is guaranteed, it appears somewhat slow. Additionally, we can observe that simulated annealing is significantly better than stochastic approximation even in the capacity of stochastic regret minimization.

9.8 Conclusion

I study Nash equilibrium approximation techniques for games that are specified using simulations. My algorithmic contributions are a set of methods, including convergent algorithms, for best response and Nash equilibrium approximation. On the experimental side, I demonstrate that all methods that I introduce can effectively be used to approximate best responses and Nash equilibria. However, there is considerable evidence in favor of using simulated annealing rather than a gradient descent-based algorithm as a black-box optimization workhorse.

Of the two methods for approximating best response in games of incomplete information which I described, I found that the method that directly optimized the parameters of the best response function outperformed a regression-based method on the more difficult problems, and was generally not very much inferior on others. Thus, faced with a new problem, the direct method seems preferable. There is a caveat, however: the regression-based method appears more robust when the number of iterations is not very large. My final result shows that, in spite of weak convergence guarantees, best response dynamics outperforms the globally convergent algorithm in the first-price sealed-bid auction setting.

While my results are generally very optimistic, the experimental work was restricted to relatively simple games. To be applicable to more difficult problems, particularly those with high-dimensional strategy sets, the techniques will likely require the analyst to hand-craft restricted strategy sets given some knowledge of the problem structure.

CHAPTER 10

Analysis of Dynamic Bidding Strategies in Sponsored Search Auctions

IN WHICH *I analyze symmetric pure strategy equilibria in dynamic sponsored search auction games. I show that a convergent strategy also exhibits high stability to deviations. On the other hand, a strategy which yields high payoffs to all players is not sustainable in equilibrium play. Finally, I demonstrate how collusion between bidders can be asymptotically achieved even under incomplete information.*[1]

In previous chapters I presented a number of techniques for analyzing games which are not analytically tractable and applied them to the strategic analysis of a supply-chain simulation. In this chapter, I present simulation-based analysis of another application, one which has generated considerable interest in recent years: sponsored search auctions.

10.1 Motivation

Sponsored search—the placement of advertisements along with search results—is currently a multi-billion dollar industry, with Google and Yahoo! the key players [Lahaie, 2006]. In the academic literature, much progress has been made in modeling sponsored search auctions as one-shot games of complete information, in which the players' values

[1] The material in this chapter is taken from Vorobeychik and Reeves [2008].

per click and click-through-rates are common knowledge. A typical justification for such an approach is the abundance of information in the system, since the advertisers have ample opportunity to explore, submitting and resubmitting bids at will. As the complexity of modeling the full dynamic game between advertisers that is actually taking place is quite intractable, static models provide a good first approximation. However, it ultimately pays to understand how relevant the dynamics really are to strategic choices of players.

One question which has been addressed in the dynamic setting is whether it is reasonable to expect simple dynamic strategies to converge to Nash equilibria. Cary *et al.* [2007] explored several *greedy bidding strategies*, that is, strategies under which players submit bids with the goal of obtaining the most profitable slot given that other players' bids are fixed. One of these strategies, *balanced bidding*, provably converges to a minimum revenue symmetric Nash equilibrium of the static game of complete information. This solution concept happens to be analytically tractable and has therefore received special attention in the literature [Varian, 2007; Lahaie and Pennock, 2007; Edelman *et al.*, 2007].

Convergence of dynamic bidding strategies is only one of many relevant questions that arise if we try to account for the dynamic nature of the sponsored search game. Another is whether we can identify Nash equilibrium strategies in the dynamic game. This problem in general is quite hard as there are infinitely many actions and ways to account for the changing information structure. One approach, taken by Feng and Zhang [2007], is to model the dynamic process using a Markovian framework. My approach focuses on the set of greedy bidding strategies from Cary *et al.* [2007]. In motivating greedy bidding strategies, Cary et al. have argued that advertisers are unlikely to engage in highly fine-grained strategic reasoning and will rather prefer to follow relatively straightforward strategies. This motivation, however, only restricts attention to a set of plausible candidates. To identify which are likely to be selected by advertisers, we need to assess their relative stability to profitable deviations. For example, while we would perhaps like advertisers to follow a convergent strategy like balanced bidding, it is unclear whether players will find it more profitable to follow a non-convergent strategy.

My goal is to provide some initial information about stability of a small set of greedy bidding strategies under incomplete information. Specifically, I use simulations to estimate the gain any advertiser can accrue by deviating from pure strategy symmetric equilibria in greedy bidding strategies. The results are promising: the convergent balanced bidding strategy is typically the most stable of the strategies I study.

To complement the analysis above, I examine the incentives when joint valuations are common knowledge, but the game is repeated indefinitely. Folk theorems [Mas-Colell *et al.*, 1995] suggest that players may be able to increase individual profits (and decrease search engine revenue) by colluding. I demonstrate several such collusion strategies and show them to be effective over a range of sponsored search auction environments. My analysis complements other approaches to studying collusion in auctions, both in the dynamic sponsored search context [Feng and Zhang, 2007] and in a general one-shot context [Krishna, 2002]. Finally, I extend the collusion result to the case of incomplete information.

10.2 Modeling Sponsored Search Auctions

A traditional model of sponsored search auctions specifies a ranking rule, which ranks advertisers based on their bid and some information about their relevance to the user query, click-through-rates for each player and slot, and players' valuations or distributions of valuations per click. Let a player i's click-through-rate in slot s be denoted by c_s^i and his value per click by v_i. Like many models in the literature (e.g., [Lahaie, 2006; Lahaie and Pennock, 2007]) I assume that the click-through-rate can be factored into $e_i c_s$ for every player i and every slot s. If a player i pays p_i^s in slot s, then his utility is $u_i = e_i c_s (v_i - p_i^s)$. The parameter e_i is often referred to as *relevance* of the advertiser i, and c_s is the slot-specific click-through-rate. I assume that the search engine has K slots with slot-specific click-through-rates $c_1 > c_2 > \ldots > c_K$.

Lahaie and Pennock [2007] discuss a family of ranking strategies which rank bidders in order of the product of their bids b_i and some weight function w_i. They study in some depth a particular weight function $w(e_i) = e_i^q$, where q is a real number. In the simulation analysis below, I consider two settings of q: 0 and 1. The former corresponds to rank-by-bid, b_i, whereas the latter is typically called rank-by-revenue, $e_i b_i$.

When players are ranked by their bids, two alternative pricing schemes have been studied: first-price (set price equal to player's bid) and generalized second-price (set price equal to next highest bid). For more than one slot, neither is incentive compatible. However, stability issues have induced the major search engines to use generalized second-price auctions. These have been generalized further to ranking by weighted bid schemes by using the price rule $p_i^s = \frac{w_{s+1} b_{s+1}}{w_i}$. The interpretation is that the bidder i pays the amount of the lowest bid sufficient to win slot s.

10.3 Dynamic Bidding Strategies

In much of this work I restrict the strategy space of players to four dynamic strategies.[2] While this is a dramatic restriction, it allows us to gain some insight into the stability properties of the dynamic game and to identify particularly interesting candidates for further analysis in the future. Additionally, it has been argued as unlikely that players will engage in full-fledged strategic reasoning and will rather follow relatively straightforward dynamic strategies [Cary *et al.*, 2007] such as the ones we consider. We now define a simple class of dynamic bidding strategies.

Definition 10.1 (Greedy Bidding Strategies [Cary *et al.*, 2007]) *A* greedy bidding strategy *for a player i is to choose a bid for the next round of a repeated keyword auction that obtains a slot which maximizes its utility u_i assuming the bids of all other players remain fixed.*

If the player bids so as to win slot s which it is selecting according to a greedy bidding strategy, any bid in the interval (p_i^s, p_i^{s-1}) will win that slot at the same price. The particular rule which chooses a bid in this interval defines a member of the class of greedy bidding strategies. I analyze strategic behavior of agents who can select from four greedy bidding strategies specified below. For all of these, let s^* designate the slot which myopically maximizes player i's utility as long as other players' bids are fixed.

[2]By dynamic strategies, I mean strategies which are functions of history. A dynamic game would then be a game which admits dynamic strategies and allows non-null and non-terminal histories.

Definition 10.2 (Balanced Bidding [Cary *et al.*, 2007]) *The* Balanced Bidding *strategy* BB *chooses the bid* b *which solves* $c_{s^*}(v_i - p_i^{s^*}) = c_{s^*-1}(v_i - b)$. *If* s^* *is the top slot, choose* $b = (v_i + p_i^1)/2$.

The Balanced Bidding strategy bids what the next higher slot would have to be priced to make the player indifferent about switching to it.

Definition 10.3 (Competitor Busting [Cary *et al.*, 2007]) *The* Competitor Busting *strategy* CB *selects the bid* $b = \min\{v_i, p_i^{s^*-1} - \epsilon\}$.

Thus the CB strategy tries to cause the player that receives the slot immediately above s^* to pay as much as possible.

Definition 10.4 (Altruistic Bidding [Cary *et al.*, 2007]) *The* Altruistic Bidding *strategy* AB *chooses the bid* $b = \min\{v_i, p_i^{s^*} + \epsilon\}$.

This strategy ensures the highest payoff (lowest price) of the player receiving the slot immediately above s^*.

Definition 10.5 (Random Bidding) *The* Random *strategy* RAND *selects the bid* b *uniformly randomly in the interval* (p_i^s, p_i^{s-1}).

10.4 Empirical Bayesian Game Analysis

In this section I construct and analyze a Bayesian game played between advertisers (alternatively, bidders or players) who may choose one of four greedy bidding strategies described above. It may be helpful for this analysis to envision that each dynamic strategy is implemented by a proxy agent. The players may select exactly one proxy agent and inform it of their preferences (in this context, value per click). The proxies thereafter participate on the behalf of each player, submitting bids based on the predefined strategies and player preferences.

Being a one-shot game of incomplete information, the bidders can condition their strategic (proxy) choices only on their own valuations. I do not allow conditioning based on relevances, as these are assumed to be a priori unknown both to the search engine and to the bidders.

In order to construct the game, we need to define player payoffs for every joint realization of values and relevances, as well as the corresponding choice of dynamic strategies. As is common for dynamic interactions, I define the payoff in the meta-game as the discounted sum of stage payoffs. In each stage, exactly one bidder, selected uniformly randomly, is allowed to modify its bid according to its choice of dynamic bidding strategy.[3] The corresponding stage payoff is an expected payoff given the ranking and payments of players as a function of joint bids, as defined in Section 10.2. I model the entire dynamic process—once relevances, values, and strategies are determined—using a simulator, which outputs a sample payoff at the end of a run of 100 stages. The discount factor is set to 0.95.[4] With this discount factor, the total contribution from stage 101 to infinity is 0.006, and I thus presume that the history thereafter is negligible.

Expected payoff to a particular player for a fixed value per click, relevance, and strategy is estimated using a sample average of payoffs based on 1000 draws from the distribution of valuations and relevances of other players. The metric for quality with which a particular strategy profile s approximates a Bayes-Nash equilibrium is the estimate of $\epsilon(s)$, which is the sample average gain from playing a best response to s over 100 draws from the player's value and relevance distributions. For each of these 100 draws, the gain from playing a best response to s is computed as the difference between the highest expected payoff for any strategy in the restricted set and the expected payoff from s_i, estimated as described above.

Since the meta-game is constructed numerically for every choice of values, relevances, and strategies of all players, an in-depth analysis of all strategies in the game is hopeless. Instead, I focus much of my attention on four pure symmetric strategy profiles, in which each player chooses the same dynamic strategy for any valuation. While this seems an enormous restriction, it turns out to be sufficient for our purposes, as these

[3]This condition ensures the convergence of *balanced bidding* dynamics [Cary *et al.*, 2007].

[4]While this is a very conservative discount for each bidding stage, our offline experiments suggest that our results are not particularly sensitive to it (for example, results which use average payoff per round as long-term utility seem to be qualitatively similar).

happen to contain near-equilibria.

10.4.1 Equal Relevances

In this section I focus on the setting in which all players' relevances are equal and assume that values per click are distributed normally with mean 500 and standard deviation 200.[5] Three sponsored search auction games are considered: in one, 5 advertisers compete for 2 slots; in the others, 20 and 50 advertisers respectively compete for 8 slots.

Figure 10.1 presents average $\epsilon(s)$ and payoffs for all four pure symmetric profiles in strategies which are constant functions of player values per click. The first observation we can make is that BB has a very low $\epsilon(s)$ in every case, suggesting that it has considerable strategic stability in the restricted strategy space. This result can also be claimed with high statistical confidence, as 99% confidence intervals are so small that they are not visible in the figure. In contrast, AB manifests very high $\epsilon(s)$ in the plot and we can be reasonably certain that it is not sustainable as an equilibrium. The picture that emerges is most appealing to the search engine: AB, which yields the greatest payoffs to players (and least to the auctioneer), is unlikely to be played, whereas BB yields the lowest player payoffs in the restricted strategy space.

10.4.2 Independently Distributed Values and Relevances

I now consider the setting in which relevances of players are not identical, but are rather identically distributed—and independently from values per click—according to a uniform distribution on the interval [0,1]. Since now the particulars of the bid ranking scheme come into play, I present results for the two schemes that have received the most attention: rank-by-bid ($q = 0$) and rank-by-revenue ($q = 1$).

Figures 10.2a and b present the results on stability of each symmetric pure strategy profile to deviations for $q = 0$ and $q = 1$ respectively. We can see that there are really no qualitative differences between the two settings, and indeed between the setting of

[5] For this and subsequent settings we repeated the experiments with an arguably more realistic lognormal distribution and found the results to be qualitatively unchanged.

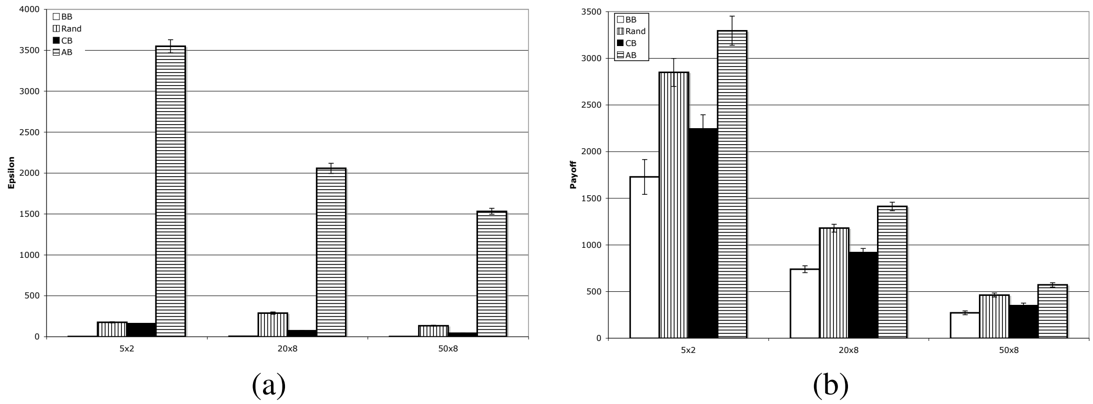


Figure 10.1: (a) Experimental $\epsilon(s)$ and (b) symmetric payoff for every pure symmetric profile in constant strategies with associated 99% confidence bounds.

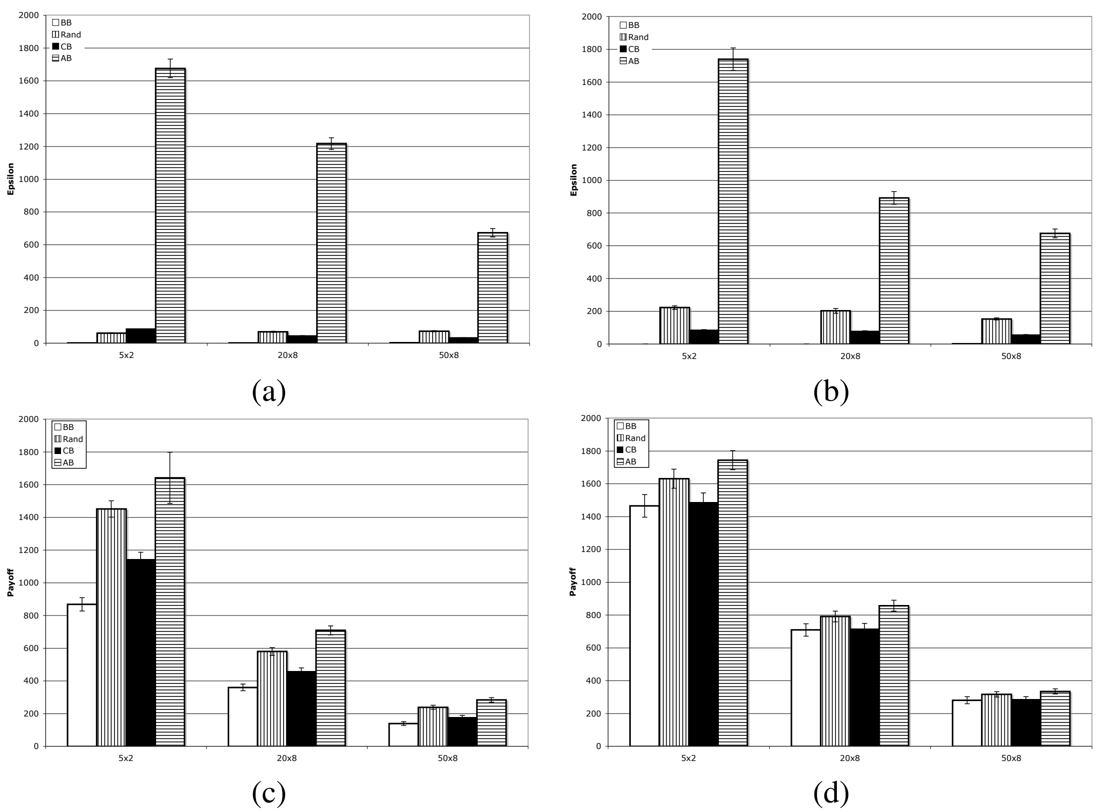


Figure 10.2: Experimental $\epsilon(s)$ (a) when $q = 0$ (b) when $q = 1$ for every pure symmetric profile; experimental payoff (c) when $q = 0$ (d) when $q = 1$ for every pure symmetric profile.

independently distributed values and relevances and the previous one in which relevances were set to a constant for all players. A possible slight difference is that RAND and CB strategies appear to have better stability properties when $q = 0$. However, this could be misleading since the payoffs to players are also generally lower when $q = 0$. The most notable quality we previously observed, however, remains unchanged: BB is an equilibrium (or nearly so) in all games for both advertiser ranking schemes, and AB is highly unstable, whereas BB yields a considerably lower payoff to advertisers than AB in all settings.

10.4.3 Correlated Values and Relevances

In the final set of experiments I draw values and relevances from a joint distribution with a correlation coefficient of 0.5. As may by now be expected, BB remains a near-equilibrium both when we set $q = 0$ and $q = 1$ (Figures 10.3a and b). However, when $q = 0$, RAND and CB are now also near-equilibria when the number of players and slots is relatively large—and, indeed, more so as the number of players grows from 20 to 50. As a designer, this fact may be somewhat disconcerting, as BB remains the strategy with the lowest payoffs to players (and, consequently, will likely yield the highest search engine payoffs) when $q = 0$; by comparison, payoffs to players when RAND is played are considerably higher than BB (Figure 10.3c). In all the cases, however, altruistic bidding remains highly unstable, to the bidders' great chagrin, as it is uniformly more advantageous in terms of payoffs (Figures 10.3c and d).

10.4.4 Truthfulness in Balanced Bidding

Thus far, I presented considerable evidence for the stability of the balanced bidding strategy in the face of deviations to the other four greedy bidding strategies under consideration. One question that now arises, however, is whether there is any incentive for a bidders to misrepresent their valuations. In the context of balanced bidding, this would mean that a bidder submits bids as if it had a valuation other than its own. Myopically this should not be the case since each bidder submits bids that greedily maximize his utility.

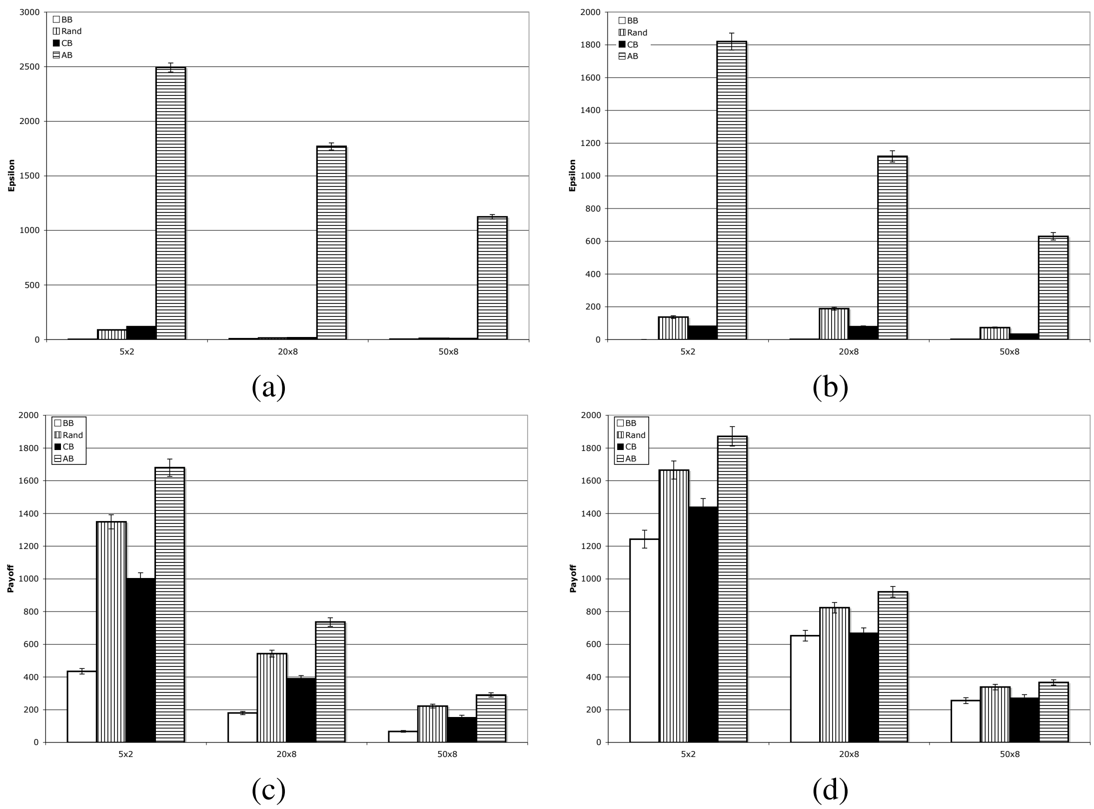


Figure 10.3: Experimental $\epsilon(s)$ (a) when $q = 0$ (b) when $q = 1$ for every pure symmetric profile; experimental payoff (c) when $q = 0$ (d) when $q = 1$ for every pure symmetric profile.

In the long term, however, perhaps such deception would pay off. That untruthfulness in this sense is unprofitable at a fixed point seems a foregone conclusion, since balanced bidding is guaranteed to converge in our settings to a symmetric equilibrium. However, there may well be an asymmetric equilibrium with the resulting allocation and bids. Below, I show that this is impossible if the fixed point is a minimum symmetric equilibrium. Particularly, I now show that at a fixed point of balanced bidding the utility of an untruthful bidder is no higher than if it were truthful (and converged to the truthful fixed point). For convenience, I restrict the remainder of the analysis in this section to a setting in which all relevances are constant, although the analysis would remain qualitatively unchanged if I had not.

Lemma 10.6 *Consider a situation in which all but one player is bidding according to balanced bidding dynamics, and one, the deviant, is considering whether to bid truthfully. The utility of the deviant in a truthful fixed point is no lower than in a fixed point reached when it is playing balanced bidding as if it had another value per click. Furthermore, bidding in order to get a higher slot yields a strictly lower utility in a fixed point for generic values per click.*

The proof of this and other results is in the appendix. Considering Lemma 10.6 and the fact that balanced bidding is greedy and, consequently, a player cannot obtain immediate gain by deviating suggests that the only gain from an untruthful variant of balanced bidding is through transient payoffs—that is, as a side-effect of responses by agents who follow balanced bidding honestly. Such effects are difficult to study analytically; thus, I use simulations to gauge regret from truthful balanced bidding. In Figure 10.4 I present results for constant relevances, as well as settings when relevances and values are independent and correlated. The sponsored search settings considered are as above. In the experiments, I allowed deviations to be factors $k = \{0, 0.5, 1.5, 2\}$ of the player's valuation. That is, a player i would submit kv_i rather than v_i to the balanced bidding proxy as his value per click. Thus, for example, a player with $k = 0.5$ and value per click of 200 would play balanced bidding as if its value were 100. As the figure suggests, when there are few players, gains from bidding as if you are someone else appear relatively high:

when values and relevances are correlated the results suggest that regret can be as high as 90 (roughly 22% of total payoff). However, regret drops off considerably as the number of players increases, down to about 10% of payoff with 50 players. Regret is lower when relevances are drawn independently of values or when they are constant.

Overall, we can observe that balanced bidding seems to be somewhat less stable when we consider the possibility that bidders may play untruthfully, that is, play as if they their value per click was different from what it actually is. Note, however, that the discount factor of 0.95 that I use is actually extremely conservative: while reasonable as an annual rate, it is unlikely to be so low per bidding round. As such, the importance of the result in Lemma 10.6 is likely to be considerably greater than our empirical results suggest. In any case, whether the observed incentives to deviate are strong enough or not would depend on other factors, such as the cost of determining a deviation that carries substantial gain.

As a final piece of evidence for the efficacy of truthful balanced bidding, I compare its regret to that from playing several variants of untruthful bidding. Figure 10.5 displays the regret for several symmetric profiles with varying degrees of untruthfulness exhibited by the players in a game with 5 players and 2 slots. As we can see, the regret from truthful balanced bidding ($k = 1$) is far overshadowed by that from untruthful profiles.

10.5 Repeated Game

10.5.1 Common Knowledge of Values

It is common in the sponsored search auction literature to assume that the player valuations and click-through-rates are common knowledge, suggesting that the resulting equilibria are rest points of natural bidder adjustment dynamics. The justification offered alludes to the repeated nature of the agent interactions. Yet, the equilibrium concept used is a static one. If a game is infinitely repeated, the space of Nash equilibrium strategies expands considerably [Mas-Colell *et al.*, 1995]. Thus, if we take the dynamic story seriously, it pays to seek subgame perfect equilibria in the repeated game, particularly if they may offer considerably better payoffs to players than the corresponding static Nash

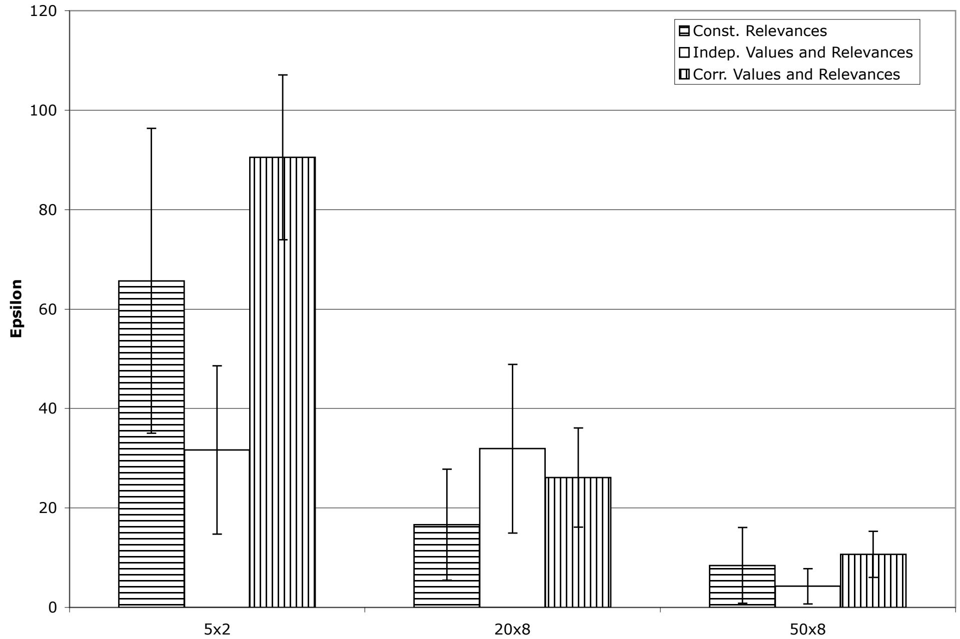


Figure 10.4: Experimental $\hat{\epsilon}(s)$ and the associated 95% confidence intervals for balanced bidding based on allowed deviations to untruthful play.

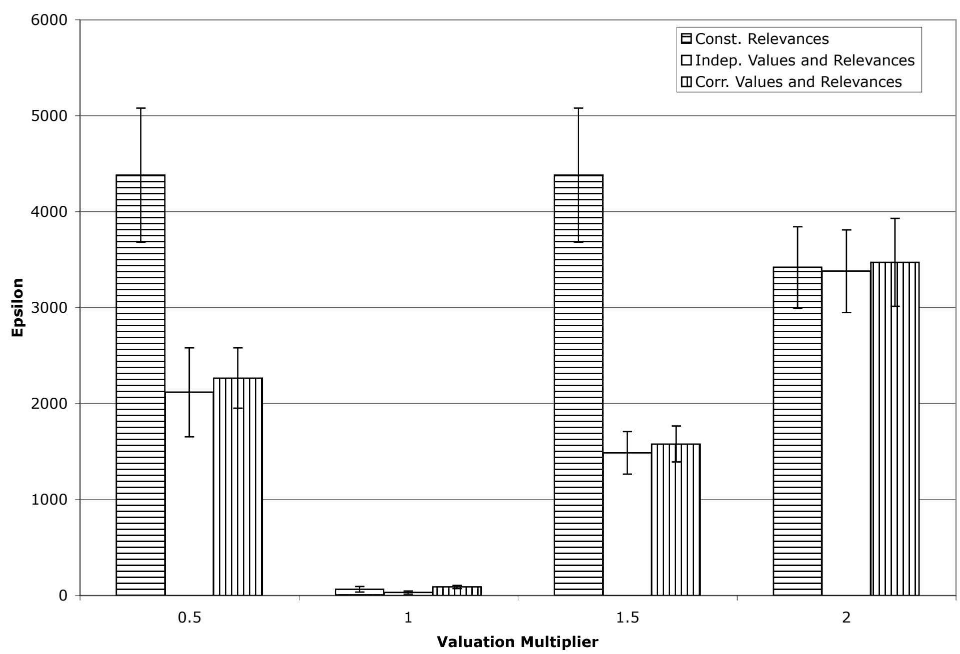


Figure 10.5: Experimental $\hat{\epsilon}(s)$ and the associated 95% confidence intervals for balanced bidding profiles with $k = \{0.5, 1, 1.5, 2\}$, where k is the multiplier of player's value; note that $k = 1$ is truthful balanced bidding. Deviations are allowed to $k = \{0, 0.5, 1, 1.5, 2\}$. The setting is a game with 5 players and 2 slots.

equilibria.

As typical analysis of repeated interactions goes, the subgame perfect equilibrium consists of two parts: the main path, and the deviation-punishment path. The main path has players jointly follow an agreed-upon profitable strategy profile, whereas the deviation path punishes any deviant. The trick, of course, is that for the equilibrium to be subgame perfect, the punishment subgame must itself be in equilibrium, yet must be sufficiently bad to discourage deviation.

A natural candidate for punishment is the worst (in terms of player payoffs) Nash equilibrium in the static game. Clearly, such a path would be in equilibrium, and is likely to offer considerable discouragement to deviants. A desirable main path would have players pay as little as possible, but needs to nevertheless discourage bidders who do not receive slots from outbidding those who do. Furthermore, all "slotless" bidders should remain slotless in the deviation subgame, since it is then clear that no incentives to deviate exist among such bidders, and we need only consider bidders who occupy some slot.

For the remainder of this section, I assume that the valuations are generic and bidders are indexed by the number of the slot they obtain in a symmetric Nash equilibrium.[6] Define the dynamic strategy profile *COLLUSION* as follows:

- *Main path*: $\forall s > K, b_s = v_s$. For all others, $b_s = \frac{w_{K+1}}{w_s} v_{K+1} + (K - s + 1)\epsilon$, where ϵ is some very small (negligible) number. Note that this yields the same ordering of bidders who receive slots as any symmetric Nash equilibrium of the game.

- *Deviation path*: play the maximum revenue symmetric Nash equilibrium strategies in every stage game. This yields the maximum revenue to the auctioneer and the lowest utilities to the players of any Nash equilibrium in the stage game [Varian, 2007].

Whether the delineated strategy constitutes a subgame perfect Nash equilibrium depends on the magnitude of the discount factor, γ_i, of every player i. The relevant question

[6] Via a simple extension of the results by Varian [2007] we can show that in a symmetric Nash equilibrium, bidders are ranked by $w_s b_s$.

is then how large do γ_i need to be to enable enforcement of *COLLUSION*. For example, $\gamma_i = 0$ will deter nothing, since there are no consequences for deviation (the game is effectively a one-stage game). Below, I give the general result to this effect.

Theorem 10.7 *The* COLLUSION *strategy profile is a subgame perfect Nash equilibrium if, for all players* i*,*

$$\gamma_i \geq \max_{s \leq K, t \leq s} \frac{(c_t - c_s)(v_s - \frac{w_{K+1}v_{K+1}}{w_s}) - (c_t \frac{w_t}{w_s}(K-t+1) - c_s(K-s))\epsilon}{c_t(v_s - \frac{w_{K+1}v_{K+1}}{w_s}) - c_s v_s - c_t \frac{w_t}{w_s}(K-t+1)\epsilon + V_{sum}}, \quad (10.1)$$

where $V_{sum} = \sum_{t=s+1}^{K} w_{t-1}v_{t-1}(c_{t-1} - c_t) + w_K v_K c_K$.

The lower bound on the discount factor in Equation 10.1 depends on the particular valuation vector, the relative merits of slots, and the total number of slots, and it is not immediately clear whether there actually are reasonable discount factors for which deviations can be discouraged. To get a sense of how sustainable such an equilibrium could be, I study the effect of these parameters on the lower bound of the discount factor. To do this, I let the relevances of all players be constant, fix the number of players at 20 and take 100 draws of their valuations from the normal distribution with mean 500 and standard deviation 200. I vary the number of slots between 2 and 15, recording the average, minimum, and maximum values of the lower bound. Furthermore, I normalize c_1 to 1 and let $\frac{c_s}{c_{s+1}} = \delta$ for all $s \leq K-1$. The results are displayed in Figure 10.6 for different values of δ.

First, focus on Figure 10.6c which shows the results for $\delta = 1.428$, an empirically observed click-through-rate ratio [Lahaie and Pennock, 2007]. As the figure suggests, when the number of slots is between 0 and 5, it seems likely that *COLLUSION* can obtain as a subgame perfect equilibrium, as the requirements on the discount factor are not too strong. When the number of slots grows, however, the incentives to deviate increase, and when the number of slots is above 11, such a collusive equilibrium no longer seems likely.

Figures 10.6a, b, and d display similar plots for other settings of δ. These suggest that as δ rises, incentives to deviate rise, since when there is a greater dropoff in slot quality

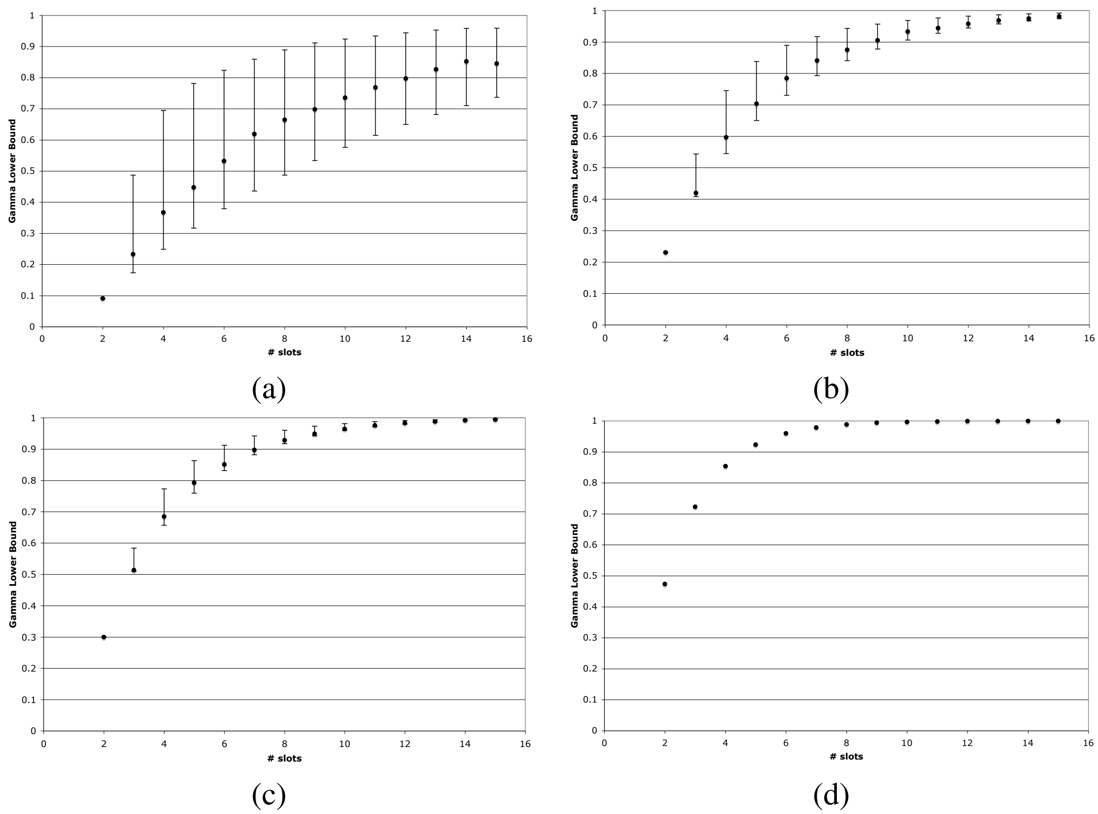


Figure 10.6: Lower bounds on the discount factor as the number of available slots varies when (a) $\delta = 1.1$, (b) $\delta = 1.3$, (c) $\delta = 1.428$, and (d) $\delta = 1.9$.

for lower slots, players have more to gain by moving to a higher slot even for a one-shot payoff.

Above, the punishment path involved bidders playing a maximum symmetric equilibrium. Such a punishment path is quite strong, since, as I mentioned, it yields the lowest utilities to the players of any Nash equilibrium in the stage game. An alternative and considerably weaker punishment would be to play a *minimum* symmetric equilibrium. Define the dynamic strategy profile *COLLUSION2* as follows:

- *Main path*: $\forall s > K$, $b_s = v_s$. For all others, $b_s = v_{K+1} + (K - s + 1)\epsilon$, where ϵ is some very small (negligible) number. Again, this yields the same ordering of bidders who receive slots as any symmetric Nash equilibrium of the game.

- *Deviation path*: play the *minimum* revenue symmetric Nash equilibrium strategies in every stage game.

One may wonder why I would ever stipulate a weaker punishment. The answer is that a maximum revenue symmetric equilibrium deviation path actually requires the common knowledge of values, which in the generalized price auction will be different from bids for all players who receive slots. As we will see below, there is considerable value in being able to relax this assumption. Particularly, it is convenient that balanced bidding converges to a minimum symmetric Nash equilibrium as its unique fixed point. Thus, if we assume that all bidders follow this strategy (and do so truthfully), we can simply punish by stipulating that bidders revert to their fixed point bids, making *COLLUSION2* a much more plausible strategy.

We can easily extend Theorem 10.7 to *COLLUSION2*:

Theorem 10.8 *The* COLLUSION2 *strategy profile is a subgame perfect Nash equilibrium if, for all players i,*

$$\gamma_i \geq \max_{s \leq K, t \leq s} \frac{(c_t - c_s)(v_s - \frac{w_{K+1}v_{K+1}}{w_s}) - (c_t \frac{w_t}{w_s}(K - t + 1) - c_s(K - s))\epsilon}{c_t(v_s - \frac{w_{K+1}v_{K+1}}{w_s}) - c_s v_s - c_t \frac{w_t}{w_s}(K - t + 1)\epsilon + V_{sum}}, \qquad (10.2)$$

where $V_{sum} = \sum_{t=s+1}^{K} w_t v_t (c_{t-1} - c_t) + w_{K+1} v_{K+1} c_K$.

Naturally, we would like to know how much weaker *COLLUSION2* is than *COLLUSION*. The answer can be observed in Figure 10.7: it does appear substantially weaker. While collusion still seems likely when the number of slots is below about 8, even with few slots there are instances when collusion is not feasible, as the higher range of bounds on γ seem very near 1.

A final question I would like to raise here is whether the choice of ranking function affects feasibility of collusion and, if so, how much. I study this in the cases when values and relevances are distributed independently and when they are correlated. Otherwise, the setup is as above. The results are presented in Figure 10.8. As the plots suggest, increasing q from 0 to 1 (that is, increasingly emphasizing relevances in the ranking) does reduce incentives to collude somewhat. However, the reduction is not all that significant—the bound on γ increases by at most 5%. It does appear, however, that with higher settings of q there are more instances in which collusion is not feasible at all, with higher ranges of γ near 1 for most slot configurations.

10.5.2 Incomplete Information

I now consider the question of whether there exists a collusive strategy for agents who do not initially know each other's valuations. Since players will not have incentive to reveal these to each other honestly, they will need to be induced. The idea is to let players play a symmetric Nash equilibrium for as many stages as necessary so that, given the discount factor, it will not pay for players to misrepresent themselves initially in order to exploit their fellow colluders later. After these initial stages playing a symmetric equilibrium, the players will be able to infer each other's valuations and play *COLLUSION2* from then on.

Lemma 10.9 *For generic valuations, a deviation from a minimum symmetric equilibrium bid to a bid that obtains a higher slot yields a strictly lower than equilibrium utility.*

Proof. For generic values, in a minimum symmetric equilibrium $w_{j+1}b_{j+1} < w_j b_j$. Thus $c_i(v_i - p_i) \geq c_j(v_i - \frac{w_{j+1}b_{j+1}}{w_i}) > c_j(v_i - \frac{w_j b_j}{w_i})$. □

Theorem 10.10 *Let b_{sym} be a vector of symmetric equilibrium bids and suppose that the* COLLUSION2 *subgame is in equilibrium for a particular vector of γ_i. Then there exists an $L > 0$ such that the following is an SPNE:*

1. *Play b_{sym} for L stages.*

2. *Play* COLLUSION2 *from stage $L + 1$ to ∞.*

Now, consider the following dynamic strategy:

1. Play balanced bidding until a fixed point is reached.

2. Play fixed point for L stages.

3. Play *COLLUSION2* from stage $L + 1$ to ∞.

I just showed that if *COLLUSION2* itself is in equilibrium, we can find L large enough such that the subgame comprised of steps 2 and 3 above is also. I have also experimentally demonstrated that balanced bidding may itself be nearly a Bayes-Nash equilibrium in an array of settings when deviations to several other greedy bidding strategies are allowed. When incentives to lie are small, we expect the entire dynamic strategy in steps 1-3 to have relatively low regret, as long as the *COLLUSION* subgame does.

10.6 Conclusion

I analyzed a set of greedy bidding strategies in a dynamic sponsored search auction setting. Many of the results are more favorable for search engines than advertisers: a high-payoff strategy profile is not sustainable in equilibrium, whereas a low-payoff profile is reasonably stable. On the other hand, when complete information about valuations and click-through-rates is available, there are possibilities for collusion that yield high payoffs to players, sustainable over a range of settings. In the case of incomplete information, a three-stage profile may be employed in which for some number of rounds depending on the discount factors bidders play a minimum symmetric equilibrium and then switch to a collusive strategy once the valuations become known. This strategy profile is expected to be reasonably stable to deviations, so long as the discount factors are small enough to support the ultimate collusion.

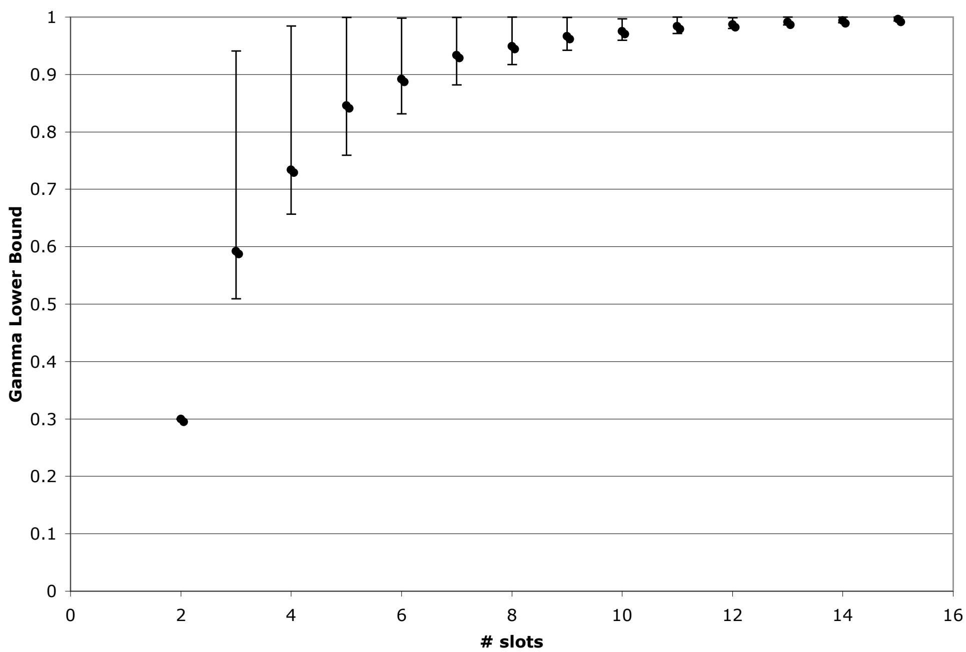
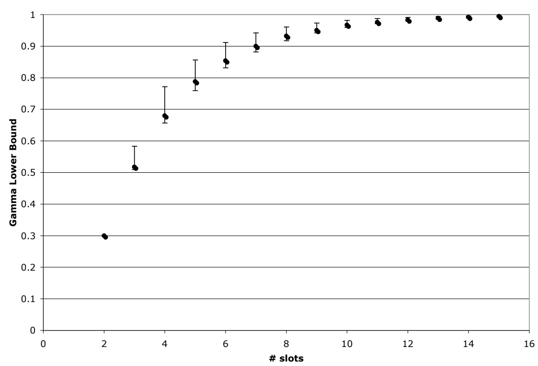
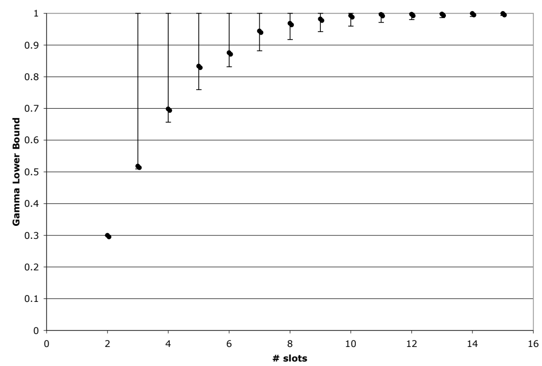


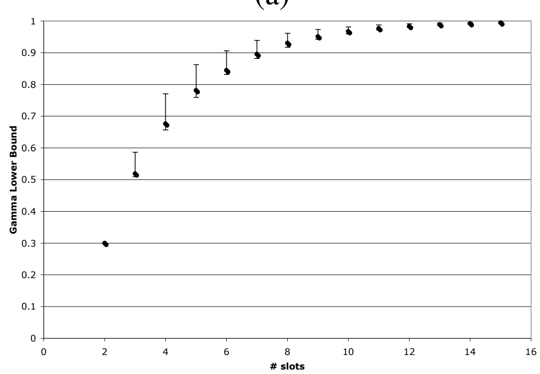
Figure 10.7: Lower bounds on the discount factor as the number of available slots varies when $\delta = 1.428$.



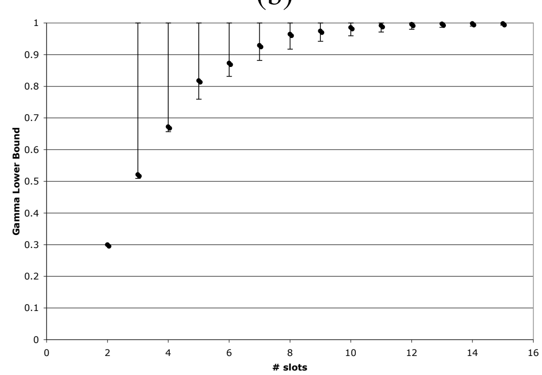
(a)



(b)



(c)



(d)

Figure 10.8: Lower bounds on the discount factor as the number of available slots varies when (a) and (b) values and relevances are independent; and (c) and (d) when values and relevances are correlated. The figures on the left (a) and (c) correspond to $q = 0$, while those on the right (b) and (d) correspond to $q = 1$.

Part III

Convergence of Simulation-Based Mechanism Design and Conceptual Issues

CHAPTER 11

Beyond Nash Equilibria: Belief Distributions of Play

IN WHICH *I discuss the notion of belief distributions of play, which are the designer's predictions (beliefs) of the outcomes of strategic interactions for a given mechanism choice. I compare belief distributions of play to the traditional equilibrium solution concepts and suggest several ways for constructing these heuristically in practice. I also provide probabilistic confidence bounds for the approximation quality of the introduced solution concepts.*[1]

Generally, mechanism design theory has focused on very rigid solution concepts, such as Bayes-Nash and dominant strategy equilibria, presuming (usually implicitly) that these are ideal predictors of play, or, rather, that a profile that is not a solution will certainly

[1] Some of the material in this chapter is taken from Vorobeychik and Wellman [2006].

not be played. Thus far, the methods I had proposed have all appealed to these solution concepts or approximations thereof.

The traditional approach runs up against two significant practical hurdles. First, in the world of boundedly rational agents and considerable complexity that will generally not be accounted for in models, relying on specific solutions (e.g., Nash equilibria) as the sole predictions of play seems somewhat impractical. Additionally, the equilibrium solution concepts have come under considerable scrutiny from experimental economists, and have in a number of cases shown to be poor predictors of actual play [Selten, 1991; Erev and Roth, 1998]. Second, even if we can believe that agents will indeed play a prescribed solution, we would still like to know which one. Even if we follow the typical approach within game theory and seek refinements [Kreps, 1990], we still generally fail to achieve point predictions. Typical treatment within the mechanism design literature has either made the optimistic assumption that agents will play the most favorable equilibrium (weak implementation), or has approached the problem pessimistically, designing for the worst possible equilibrium outcome (strict implementation) [Mas-Colell *et al.*, 1995; Osborne and Rubinstein, 1994]. The former relies on the designer's ability to persuade the agents to play the desirable strategies. The latter allows agents to choose a solution arbitrarily, but as a result may sacrifice good design choices. To sum up, few will argue that a mechanism designer is primarily concerned with predicting agent play given a particular mechanism choice, but perfectly accurate point predictions seem elusive both in theory and in practice, while multiplicity of predictions introduces ambiguity into the design process that is not easily resolved.

In this chapter, I suggest that unique prediction of play is undesirable. Instead, I believe that the designer should formulate a *belief distribution of agent play* that leverages a flexible solution concept. By using distributions of agent play, we move away from the rigidity of relying on a particular solution concept as *the* prediction, allowing rather that agents may play a non-solution (non-equilibrium) profile with some positive probability, albeit perhaps very small. We can, if needed, still fall back upon the traditional weak or strong implementation by setting the probability that agents play the desirable equilibrium to one in the former case, and setting the probability of the least desirable equilibrium (conditional on the design choice) to one in the latter. We can also restrict distributions to put positive probabilities of play exclusively on solutions. This would put complete faith in a solution concept of choice, but allow the designer to retain flexibility in dealing with multiplicity of solutions.

However, in order to fully take advantage of the flexibility offered by my approach, one would need solution concepts that provide “hooks” upon which probabilities of play could hinge. I suggest a number of such concepts, many of which are relaxations of the traditional ones such as Nash and dominant-strategy equilibria.

An important element of mechanism design is the actual modeling of strategic interactions between agents, as well as the effect that mechanism choice has upon it. Any modeling effort will generally have some uncertainty in regards to the agent preferences or preference distributions, insofar as the designer may have made mistaken assumptions or failed to account for important secondary elements that influence agent play. Such uncertainty may be expressed as a distribution of agent payoffs around an estimate expressed by the model. If the designer has developed a belief distribution of agent play based on his model, he would necessarily need to subsequently adjust it to account for this uncertainty. One approach would have the designer merely specify the distribution of play as a function of the game model—akin to the meta-strategies described by Kiekintveld and Wellman [2008]. He would then simply integrate this function to obtain a posterior distribution of agent play with respect to the noise in the model. The difficulty with this approach is that it may require the designer to find a solution correspondence, which is a formidable task in general. Furthermore, in real settings, the designer may have considerable trouble formalizing his beliefs even so simply as above. To address these problems, albeit very imperfectly from the theoretical perspective, I introduce *heuristic beliefs*, that is, beliefs that are derived heuristically from probabilistic indicators of particular game theoretic solutions. This approach allows the designer to rely on systematic analysis, while avoiding explicit modeling of much of what he understands “intuitively”. Furthermore, it allows considerable flexibility in selecting the solution concept most applicable to the setting at hand.

The idea of using multiple sources of evidence to derive a distribution of play in a strategic setting has recently been studied by Duong *et al.* [2008], while the entire notion of forming predictions in strategic contexts using distributions of play rather than formulating solution concepts is extensively discussed by Wolpert [2008].

11.1 Preliminaries

Recall that the goal of the designer is to optimize the value of some welfare function, $W(r, \theta)$, dependent on the mechanism parameter and resulting play, r. Since I allow for many possible outcomes of agent play, I can evaluate the objective function for a given game abstractly as follows. I define $W_T(\hat{R}, \theta) = T_{\hat{R}} W(r, \theta)$, where T is some functional acting on $W(r, \theta)$. Several examples of T commonly found in the literature are $\inf_{\hat{R}}$ (representing strict implementation) and $\sup_{\hat{R}}$ (representing weak implementation). I have already argued that both of these are somewhat extreme. Instead, I will concentrate on an alternative: I let $T_{F,\hat{R}}$ to be the expectation with respect to some probability distribution F over $\hat{R}$. Then, $W_{T_{F,\hat{R}}}(\hat{R}, \theta) = E_{F,\hat{R}} W(r, \theta)$. Given a description of the solution correspondence $\mathcal{N}(\theta)$ and $W_T(\mathcal{N}(\theta), \theta)$, the designer faces a standard optimization problem.

11.2 Solution Concepts

In evaluating the welfare function with respect to the distribution of agent play, I relied until now on a specification of a solution concept which in effect provides the support of the distribution of play. In this section, I explore a number of ideas about how solution concepts can be defined. I begin with a standard Nash equilibrium concept and its immediate relaxation to approximate Nash equilibria, and go on to relax several other solution concepts in a similar fashion with the hope that one can thereby incorporate all strategy profiles that the designer may find to be plausible agent play. I then describe a complementary solution concept that may serve as an additional hook to deriving distributions of play by allowing the designer to indirectly model risk aversion of agents. Indeed, all of the solution concepts I propose have relaxation parameters which can serve the designer in defining probabilities of play for various profiles.

11.2.1 Nash and Approximate Nash Equilibria

Perhaps the most common solution concept for games is *Nash equilibrium*, which I define formally in Chapter 2. A closely related concept is an *approximate* or ϵ*-Nash equilibrium*, where ϵ is the maximum benefit to any agent for deviating from the prescribed strategy (also defined formally in Chapter 2).

One reason for relaxing the Nash equilibrium solution concept is so that we can express the likelihood that a particular profile will be played in terms of its proximity to being a Nash equilibrium (that is, if agents have considerable incentives to deviate from a particular profile, we deem such a profile an unlikely outcome of play). An example of this would be to posit that a profile r is played with probability proportional to $\exp^{-\epsilon(r)}$, where $\epsilon(r)$ is the regret of profile r (that is, r is an $\epsilon(r)$-Nash equilibrium). Since the choice of Nash equilibrium for such relaxation is somewhat arbitrary, I introduce similar relaxations to other solution concepts in the sections that follow.

11.2.2 Alternatives to (ϵ)-Nash Equilibria

While Nash equilibrium has long been the primary solution concept for games, many have expressed dissatisfaction with it in the context of real mechanism design problems and agent strategic considerations. For example, Erev and Roth [1998] provide experimental evidence that a reinforcement learning algorithm tends to be a better predictor of actual play in games with a unique equilibrium. In a similar vein, Selten [1991] presents a series of arguments against the Bayesian rationality as a reasonable predictor and, even, as an effective normative system. One appealing alternative that has been proposed is *rules of thumb* [Rosenthal, 1993b,a], that is, strategies that are essentially simple rules or series of rules conditioned on context that have proven effective over time. A similar notion of heuristic strategies has been studied by Walsh *et al.* [2002], and there has been work to find a Nash equilibrium in a relatively small set of heuristic strategies [Walsh *et* *al.*, 2002; Reeves *et al.*, 2005].

In this section, I describe several concepts that allow one to consider strategic choices of agents that have appealing properties that may qualify them as reasonable rules of thumb.[2] Thus, I expand the set of solutions that a designer may consider as plausible agent play, and thereby allow the mechanism designer to make the most appropriate choice for a particular problem.

Nearly Dominant Profiles

An intuitive property of a rule of thumb is that it *usually* works *reasonably* well. The way we can translate this idea into a solution concept is by introducing ϵ-dominant strategies.

Definition 11.1 *A profile, r is ϵ-dominant if* $\forall i \in I, \forall t \in R_{-i}$,

$$u_i(r_i, t) + \epsilon \geq u_i(r', t),\ \forall r' \in R_i.$$

While dominant strategies are rare, strategies that are nearly dominant may be more common, and may indeed provide a solid basis for certain rules of thumb. Of course, there will always be an ϵ-dominant strategy if we set ϵ to be high enough (as long as the payoff functions are bounded). However, once ϵ is large, such strategies are no longer nearly dominant in any meaningful way. Still, this solution concept may be a useful and reasonably compelling way to model agent play without appealing to the hyperrational Nash equilibrium. From the players' viewpoint, nearly dominant pure strategies are easy to find with respect to a given game, although the algorithmic question of finding the entire set of (pure and mixed) nearly dominant profiles may be a bit more involved. From a designer's viewpoint, they are reasonable things to expect agents to play, since no player has much to gain by deviating no matter what the other players do.

[2]Note that often rules of thumb have been used to restrict the strategic landscape [Walsh *et al.*, 2002; Reeves *et al.*, 2005]. Here I assume that any such restrictions have been made, if appropriate, and concentrate rather on narrowing down the set of profiles that may plausibly be played in order to subsequently derive distributions of agent play.

As in the case of ϵ-Nash equilibria, an important advantage of this relaxation of the dominant strategy profiles is that we can derive distributions of play based on "how dominant" a particular profile is. That is, we can assess a relatively low likelihood of play to profiles in which at least one agent may be significantly better off by playing another strategy for some deviation of other agents. Alternatively, we may fix ϵ and assess zero probability of play to profiles which are not ϵ-dominant (assuming, of course, that there is at least one strategy profile that is).

Treating ϵ-dominance as a solution concept, we can also derive probabilistic confidence bounds (under the standard normality assumption) on any profile in an empirical game:

Theorem 11.2 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance. Then,*

$$\Pr\left(r \text{ is } \epsilon\text{-dominant}\right) = \prod_{i \in I} \prod_{t \in R_{-i}} \int_{\mathbb{R}} \prod_{r' \in R_i \setminus r_i} \Pr\{u_i(r', t) \leq u + \epsilon\} f_{u_i(r_i,t)}(u) du.$$

Nearly Undominated Profiles

In his seminal work, Pearce [1984] describes the notion of rationalizable strategies. While the set of all rationalizable strategies is not always identical to the set of strictly undominated strategies, the two concepts are closely related, and are indeed appealing on similar grounds. The argument of Pearce was that the Nash equilibrium concept was too strong to describe actual behavior. By weakening it to a set of *plausible* strategy profiles that may be observed, actual behavior may be explained, although no longer modeled precisely.

While the idea that players are unlikely to play a profile that is strictly dominated (or not rationalizable) is very intuitive, there is experimental evidence to suggest that dominated strategies (for example, cooperative play in Prisoner's Dilemma) may indeed be played in practice [Cooper and Ross, 1996]. As a consequence, I introduce here an even weaker concept of nearly undominated or ϵ-undominated strategies, which include strategies that, while dominated, are very close to being optimal for some strategy profile that other agents may play.

Definition 11.3 *A profile, r, is ϵ-undominated if $\forall i \in I, \exists t \in R_{-i}$, such that*

$$u_i(r_i, t) + \epsilon \geq u_i(r', t),\ \forall r' \in R_i.$$

This concept is not entirely new: Cheng and Wellman [2007] actually explored iterative elimination of ϵ-dominated strategies as a computational tool to facilitate Nash equilibrium approximation in large finite games.

Since this solution concept is very weak, it allows the designer to retain most strategy profiles in a game as plausible rules of thumb, eliminating only those that are clearly extremely harmful to at least one agent. The assumption that a very poorly performing strategy would never be played is quite plausible in real situations. For example, a strategy that consistently loses the agent one million dollars, as compared to any other strategy that incurs no loss, carries considerable appeal of being assessed probability zero of play.

Given a set of nearly undominated strategies, we can also imagine that a likelihood of nearly dominant or nearly Nash profiles would be greater based on their nearness to the corresponding solution concept. Thus, we can actually combine all three of the relaxed solution concepts we have so far discussed to obtain a distribution of agent play.

If we treat ϵ-undominatedness as a solution concept in its own right, we can, as before, derive probabilistic confidence bounds describing the likelihood that a profile in an empirical game is ϵ-undominated for a fixed ϵ. I present an expression for such a bound under the assumption of normal noise in Theorem 11.4.

Theorem 11.4 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance.*

Then,

$$\Pr\left(r \text{ is } \epsilon\text{-undominated}\right) =$$
$$= \prod_{i \in I} \left[1 - \prod_{t \in R_{-i}} \int_{\mathbb{R}} \left(1 - \prod_{r' \in R_i \setminus r_i} \Pr\{u_i(r', t) \leq u + \epsilon\} \right) f_{u_i(r_i, t)}(u) du \right].$$

Safety of Pure and Mixed Profiles

Risk aversion has been a topic of much economic research, and it is wise for the designer to expect that the players will be risk averse to some degree. However, the precise models of risk aversion may not be easy to come by. Instead, I introduce a concept that allows the designer to avoid precise modeling, while still capturing some of the intuition behind risk aversion. To that end, I define the notion of δ-safety.

Definition 11.5 *Let R_{-i} be the joint space of deviations of agents other than i. A profile r is δ_i-safe for agent i if*

$$\delta_i(r) \geq \max_{t \in R_{-i}} (u_i(r) - u_i(r_i, t)).$$

A profile r is then δ-safe if it is δ_i-safe for all agents, that is, if

$$\delta(r) \geq \max_{i \in I} \max_{t \in R_{-i}} (u_i(r) - u_i(r_i, t)).$$

Alternatively, r is δ-safe if, for every player $i \in I$,

$$u_i(r) \leq u_i(r_i, t) + \delta, \; \forall t \in R_{-i}.$$

Again, for completeness, I present the probabilistic confidence bound on the δ-safety of profiles in an empirical game.

Theorem 11.6 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance.*

Then,

$$\Pr\left(\max_{i\in I}\max_{t\in R_{-i}}(u_i(r)-u_i(r_i,t))\le\delta\right)=$$
$$=\prod_{i\in I}\int_{\mathbb{R}}\prod_{t\in R_{-i}}\left[1-\Pr(u_i(r_i,t)\le u-\delta)\right]f_{u_i(r)}(u)du.$$

The theorem and proof are analogous to cases already discussed, with r a pure or a mixed profile.

I do not see the notion of δ-safety as having much independent value. Instead, I view it as a useful way to distinguish particular types of rule-of-thumb strategies that players may consider. For example, we can imagine that in a set of approximate Nash equilibria, there may be profiles that would be extremely sensitive to deviations by players, and, therefore, have a higher bound on δ-safety. The notion of δ-safety thus provides a mechanism designer with an additional assessment of likelihood of play by indirectly accounting for risk aversion of agents without having to quantify it.

As for the other solution concepts, we may need a way to answer questions about probability of a particular profile being a δ-safe ϵ-equilibrium for a given δ and ϵ, as well as similar questions about other solution notions I have developed. I present the corresponding probabilistic confidence bounds below. First, I present a bound on probability that a profile r is a δ-safe ϵ-Nash equilibrium.

Theorem 11.7 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance. Then,*

$$\Pr(r \text{ is } \delta\text{-safe, } \epsilon\text{-Nash})=$$
$$\Pr([\max_{i\in I}\max_{t\in R_{-i}}(u_i(r)-u_i(r_i,t))\le\delta]\&[\max_{i\in I}\max_{r'\in R_i}(u_i(r',r_{-i})-u_i(r))\le\epsilon])=$$
$$\prod_{i\in I}\int_{\mathbb{R}}\left[\left(\prod_{t\in R_1}\Pr\{u_i(r_i,t)\ge u-\delta\}\right)\left(\prod_{r'\in R_i\setminus r_i}\Pr\{u_i(r',r_{-i})\le u+\epsilon\}\right)\right]dF_{u_i(r)}(u).$$

We can combine δ-safety and ϵ-dominance or ϵ-undominatedness analogously, although in these cases the derivations are somewhat more laborious. The former bound is presented in Theorem 11.8 and the latter in Theorem 11.9.

Theorem 11.8 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance. Then,*

$$\Pr\{r \text{ is } \delta\text{-safe, } \epsilon\text{-dominant}\} =$$
$$\Pr\{\max_{i\in I}\max_{t\in R_{-i}}[u_i(r)-u_i(r_i,t)] \le \delta \;\&\; \max_{i\in I}\max_{t\in R_{-i}}\max_{r'\in R_i}[u_i(r',t)-u_i(r_i,t)] \le \epsilon\} \ge$$
$$\prod_{i\in I}\int_{\mathbb{R}}\left[\left(\prod_{t\in R_{-i}}\Pr\{u_i(r_i,t)\ge u-\delta\}\right)\left(\prod_{r'\in R_i\backslash r_i}\Pr\{u_i(r',t)\le u+\epsilon-\delta\}\right)\right]dF_{u_i(r)}.$$

Theorem 11.9 *Suppose that the payoffs for all profiles are independently sampled. Further, suppose that the sampling noise for every profile is Gaussian with known variance. Then,*

$$\Pr\{r \text{ is } \delta\text{-safe, } \epsilon\text{-undominated}\} =$$
$$\Pr\{\max_{i\in I}\max_{t\in R_{-i}\backslash r_{-i}}[u_i(r)-u_i(r_i,t)] \le \delta \;\&\; \max_{i\in I}\min_{t\in R_{-i}}\max_{r'\in R_i\backslash r_i}[u_i(r',t)-u_i(r_i,t)] \le \epsilon\} =$$
$$\prod_{i\in I}\int_{\mathbb{R}}\left[\prod_{t\in\mathbb{R}_{-i}\backslash r_{-i}}\Pr\{u_i(r_i,t)\ge u-\delta\}\right]\times$$
$$\times\left[1-\left(1-\prod_{r'\in R_i\backslash r_i}\Pr\{u_i(r',r_{-i})\le\epsilon+u\}\right)\times\right.$$
$$\left.\times\left(\prod_{t\in\mathbb{R}_{-i}\backslash r_{-i}}\frac{\int_{u-\delta}^{\infty}\{1-\prod_{r'\in R_i\backslash r_i}\Pr(u_i(r',t)\le\epsilon+v)\}f_{u_i(r_i,t)}(v)dv}{\int_{u-\delta}^{\infty}f_{u_i(r_i,t)}(v)dv}\right)\right]dF_{u_i(r)}(u).$$

11.3 Constructing Distributions of Agent Play

My goal in this section is to give several ideas about constructing belief distributions of agent play that rely on game-theoretic solution concepts (thus taking the players' incentives seriously), but do not necessarily "commit" to them.

The first approach is for the designer to decide exactly which solution concept is the best model for the strategic behavior of the agent pool. The designer may choose a very weak concept and make as few assumptions as possible about agent rationality and common knowledge, and we provided some guidance about such choices in the preceding section. Once the solution concept is chosen, the designer will need to assess the relative likelihood of solutions, for example, modeling each solution as equally likely to be played. Alternatively, the designer may wish combine a solution concept with some notion of δ-safety as I previously defined, and thereafter devise a distribution that puts higher probability on solutions with low δ. Note that inherent in this approach is the assumption that any profile that is not a solution will be played with probability zero.

Yet another approach is to develop a distribution of play based on the relaxation parameter within a solution concept. Previously, I defined several ϵ- and δ-concepts (ϵ-Nash, ϵ-dominant, ϵ-undominated, δ-safe). Each pure strategy profile in the game will most certainly be any such ϵ-concept for some value of ϵ (the same is true of δ-safety). One could then assess the probability of play for a particular profile to be inversely proportional to its value of ϵ or δ for the selected solution concept. One could also develop similar probabilistic models of agent play based on combinations of these solution concepts. The advantage of this method is that the assessment of probability of play will be positive for every pure strategy profile in the game. Thus, in a way, the designer will be hedging.

Thus far, I presumed that the designer is certain about his model of the players' utilities. This, of course, is suspect, and indeed some effort has been made within the game theory community to assess the quality of particular solution concepts based on how well they survive such modeling noise [Fudenberg *et al.*, 1988]. Here, I will take another approach. I assume that payoffs specified by the designer are samples from a Normal distribution with some variance. This could reasonably be the case, for example, when the payoffs in the game are sample averages based on simulation data, as I discussed in preceding chapters. Variance here would need to be specified by the designer, again calling forward his modeling prowess, but hopefully this would not be a very difficult exercise. If the designer is only slightly uncertain about his model, he would specify a small variance, whereas if his uncertainty is great, so would be the variance.

The distributions of play that I described above are all *conditional* on a particular game. Thus, in order to find the distribution of play in the face of uncertainty about the actual game, the designer would need to take the expectation of this conditional distribution with respect to the distribution of games. I now state this in a somewhat more formal language. Suppose we fix the game, Γ, that the agents will play, and choose a solution concept, $\mathcal{C}$. I designate the distribution of agent play conditional on Γ and the solution concept by

$$P_{\Gamma,\mathcal{C}}(r) = \Pr\{r \text{ is played}|\Gamma, \mathcal{C}\},$$

where r is a pure strategy profile. This may be specified by the designer as I suggested above. Now, in order to derive the probability of agents playing a pure strategy profile r given only a particular solution concept, one would simply take the expectation with respect to the distribution of games:

$$P_{\mathcal{C}}(r) = \Pr\{r \text{ is played}|\mathcal{C}\} = E_{\Gamma}[P_{\Gamma,\mathcal{C}}].$$

This can be done for every pure strategy profile to obtain a distribution of agent play.

There are several important shortcomings in the approach I just suggested. The first is simply that it requires numerical integration, as we do not have a closed-form expression of this expectation for any solution concept that I have discussed. This happens to be a relatively significant problem, since numerical techniques here would require computing the entire set of solutions for each of some finite set of games, which may quickly become impractical when the game is relatively large. Another limitation is that the designer is still required to specify a model of his beliefs given a game, even though he may be uncertain about his model of the players' payoffs. Often, the designer may find himself incapable of doing even that very sensibly, but would instead like to have a more heuristic, though systematic, approach to modeling agent play. To this end, I propose a *heuristic* distribution of play, which does not require any modeling on designer's part, except his choice of a solution concept.

Observe that the distribution over agent payoff functions induces a probability that each pure strategy profile r is a solution $\mathcal{C}$. I derived expressions for these probabilities in Chapter 7 in the case of approximate Nash equilibria, and analogous expressions for several other solution concepts are presented above. We can use these to obtain a very simple heuristic distribution over play by normalizing as follows:

$$\Pr\{r \text{ is played}|\mathcal{C}\} = \frac{\Pr\{r \text{ is } \mathcal{C}\}}{\sum_{r' \in R} \Pr\{r' \text{ is } \mathcal{C}\}}.$$

Since we have an expression for $\Pr\{r' \text{ is } \mathcal{C}\}$, computation is greatly simplified.[3]

11.4 Conclusion

In this chapter, I focused on an important practical shortcoming of mechanism design theory: lack of effective methods for point predictions of actual play. Indeed, I believe that making very precise predictions of agent play is undesirable, as complexity of real strategic settings will generally make models imperfect enough that even the most appropriate solution concept will not necessarily make a good predictor of play. Instead, I argued for the need to evaluate mechanism choices with respect to belief distributions of play based on flexible solution concepts. Thus, game-theoretic notions may eagerly enter the distributions of play, but need not define them entirely.

It seems most useful in devising distributions of agent play to have solution concepts with relaxation parameters that allow relative assessment of the likelihood of play of different strategy profiles. I suggested the degree of approximation of a solution concept as an example of such a parameter. Additionally, I suggested a complementary solution concept which may be used to incorporate the designer's beliefs about the risk aversion of agents in the distribution of play. Finally, I presented examples of how solution concepts that I suggested may be used in deriving distributions of agent play.

My approaches can be easily extended to incorporate uncertainty about the designer's model of the strategic scenario, although incurring considerable computational effort. To alleviate this difficulty, as well as to consider yet another alternative sensible way to develop distributions of play, I introduced heuristic distributions, which are based on a probabilistic assessment—with respect to the distribution of payoffs in the game model—of

[3] While we may still need to do numerical integration, we are now free from the burden of computing solutions for every game on the integration path, which will almost always yield substantial numerical savings.

the closeness of every profile to a solution. This closeness, it must be noted, is not evaluated based on Euclidean distance, but is rather determined by the value of the relaxation parameter of the chosen solution concept (say, ϵ in an approximate Nash equilibrium).

I believe that there is still a considerable gap between theoretical mechanism design and its practical applications. In this chapter, I suggested that this gap may be narrowed if the designer has effective ways to determine the distribution of agent play based on game-theoretic notions. While I proposed a number of methods to this end, much work needs to be done to verify whether these are truly effective in practical settings, or whether others need to be developed in their place.

CHAPTER 12

Convergence of Optimal Mechanisms on Finite Simulation-Based Games Using Belief Distributions of Play

IN WHICH *I present strong consistency results about approximately optimal mechanisms based on empirical games and distribution-based solution concepts.*

In this chapter I present sufficient conditions for convergence of nearly-optimal mechanisms in the simulation-based game setting. Since convergence of mechanism design choices requires that the corresponding solution concept is also convergent, I proceed to demonstrate convergence of several solution concepts based on specific forms of the belief distributions of play.

As we will see, requirements for convergence are quite strong. In the case of the Nash equilibrium solution concept, the results requires that there is a unique Nash equilibrium in every game induced by a feasible design choice. With belief distributions of play, the requirement is that the support of the distribution be finite and fixed for all empirical games.

Before proceeding with the technical part of this chapter, let me review some relevant notation. Recall that R denotes a set of strategy profiles (pure or mixed). I use S to refer to the set of mixed strategy profiles and A to refer to the set of pure strategy profiles.

Recall that $W(r, \theta)$ denotes the designer's objective function, Θ the design space, $\theta \in \Theta$ a particular design choice, and $r \in R$ a strategy profile (analogously, $a \in A$ and $s \in S$ denote pure and mixed strategy profiles respectively). Additionally, recall from the previous chapter that $W_T(\hat{R}, \theta) = T_{\hat{R}} W(r, \theta)$, where T is a functional acting on W. Here, I deal with the case where T is the expectation with respect to a belief distribution of play. Henceforth, I will also use R in place of $\hat{R}$ where I mean it to be some restricted subset of the joint strategy space.

12.1 Convergence of Mechanism Design Based on Nash Equilibria in Empirical Games

I first show that in the special case when Θ and A are finite and each Γ_θ has a unique Nash equilibrium, the estimates $\hat{\theta}$ of an optimal mechanism almost surely converge to an actual optimizer.

Let $\mathcal{N}_n(\theta)$ denote the set of Nash equilibria of an empirical game induced by θ in which the payoffs for every strategy profile have been sampled at least n times. Define $\hat{\theta}$ to be

$$\hat{\theta} = \arg\max_{\theta \in \Theta} W(\mathcal{N}_n(\theta), \theta),$$

and let θ^* be defined by

$$\theta^* = \arg\max_{\theta \in \Theta} W(\mathcal{N}(\theta), \theta).$$

Theorem 12.1 *Suppose $|\mathcal{N}(\theta)| = 1$ for all $\theta \in \Theta$ and suppose that Θ and A are finite. Let $W(s, \theta)$ be continuous at the unique $s^*(\theta) \in \mathcal{N}(\theta)$ for each $\theta \in \Theta$. Then $\hat{\theta}$ is a consistent estimator of θ^* if $W(\mathcal{N}(\theta), \theta)$ is defined as a supremum, infimum, or expectation (with respect to any probability distribution) over the set of Nash equilibria. In fact, $\hat{\theta} \to \theta^* a.s.$ in each of these cases.*

The shortcoming of the above result is that, within my framework, the designer has no way of knowing or ensuring that Γ_θ do, indeed, have unique equilibria. However, it does lend some theoretical justification for pursuing design based on Nash equilibrium estimates formed on empirical games, and, perhaps, will serve as a guide for more general results in the future.

12.2 Convergence of Mechanism Design Based on Heuristic Distributions of Play

Suppose that R is finite. Let $P(r, \theta)$ be the distribution of play in the actual game[1] and let $P_n(r, \theta)$ be the distribution of play in the empirical game which contains at least n payoff samples for every profile. I assume that both P and P_n have their support in R.

Define the actual expected objective value of the designer with respect to R as follows:

$$W(R, \theta) = \sum_{r \in R} W(r, \theta) P(r, \theta). \tag{12.1}$$

Similarly, for a game in which every profile is sampled at least n times,

$$W_n(R, \theta) = \sum_{r \in R} W(r, \theta) P_n(r, \theta). \tag{12.2}$$

I first demonstrate some basic convergence properties of the objective function defined with respect to a finite support R. Specifically (and intuitively), as long as the distribution of play P_n converges to P almost surely and pointwise on R and Θ, then the objective function estimate W_n converges to the actual objective W almost surely.[2]

Lemma 12.2 *Suppose $P_n(r, \theta) \to P(r, \theta)$ a.s. for every $r \in R$ and for every $\theta \in \Theta$. Then $W_n(R, \theta) \to W(R, \theta)$ a.s. for every $\theta \in \Theta$. If furthermore Θ is compact, $W(r, \theta)$ is continuous in θ, and $P_n(r, \theta) \to P(r, \theta)$ a.s.* uniformly *on Θ, then $W_n(R, \theta) \to W(R, \theta)$ a.s. uniformly on Θ.*

Next, I present the intuitive result that as long as P and $W(r, \theta)$ are continuous, so is $W(R, \theta)$. This result in combination with the Weierstrass theorem will ensure the

[1] For example, distributions constructed from game-theoretic solution concepts as described in the previous chapter; some specific examples distributions of play are presented below

[2] Analogous results can be obtained for convergence in probability.

existence of the optimal mechanism on compact Θ, which is stated in the theorem that follows.

Lemma 12.3 *Suppose $P(r, \theta)$ and $W(r, \theta)$ are continuous in θ. Then $W(R, \theta)$ is continuous in θ.*

Proof. Follows since product of continuous functions is continuous, and a finite sum of continuous functions is continuous. □

Theorem 12.4 *Suppose $P(r, \theta)$ and $W(r, \theta)$ are continuous in θ and Θ is compact. Then $\exists\, \theta^* \in \Theta$ such that $W(R, \theta^*) = \max_{\theta \in \Theta} W(R, \theta)$.*

Proof. By Lemma 12.3, $W(R, \theta)$ is continuous. Since Θ is compact, the result follows from the Weierstrass theorem. □

Above I showed that the objective values converge. What I am really interested in, however, is the convergence of the estimates of optimal objective values obtained from simulation data. This is be the subject of the main result below. Towards this end, I state one more useful lemma, which takes as a precondition convergence of W_n to W and demonstrates how it implies convergence in terms of optimal objective values.

Lemma 12.5 *Suppose $W_n(R, \theta) \to W(R, \theta)$ a.s. uniformly on Θ, $W(R, \theta)$ is continuous on Θ, and $W_n(R, \theta)$ is continuous on Θ for every n. Let $\hat{\theta} = \arg\max_{\theta \in \Theta} W_n(R, \theta)$, $\theta^* = \arg\max_{\theta \in \Theta} W(R, \theta)$. Then $W(R, \hat{\theta}) \to W(R, \theta^*)$ a.s.*

Combining the partial results, I can now state my main convergence result.

Theorem 12.6 (Main MD Convergence Result) *Suppose $W(r, \theta)$ and $P(r, \theta)$ are continuous on Θ, $P_n(r, \theta)$ is continuous on Θ for all n. Suppose further that Θ is compact, and $P_n(r, \theta) \to P(r, \theta)$ a.s. for every $r \in R$ and uniformly on Θ. Then, $W(R, \hat{\theta}) \to W(R, \theta^*)$ a.s., where $\hat{\theta} = \arg\max_{\theta \in \Theta} W_n(R, \theta)$, $\theta^* = \arg\max_{\theta \in \Theta} W(R, \theta)$.*

Proof. By Lemma 12.3, $W(R, \theta)$ is continuous, and $W_n(R, \theta)$ is continuous for every n. By Lemma 12.2, $W_n(R, \theta) \to W(R, \theta)$ a.s. uniformly on Θ. Combining these results with Lemma 12.5 gives us the desired result. □

When Θ is finite, pointwise and uniform convergence are equivalent and continuity of $P_n(r, \theta)$ on Θ is unnecessary,[3] implying the following corollary.

Corollary 12.7 *Suppose Θ is finite and $P_n(r, \theta) \to P(r, \theta)$ a.s. for every $r \in R$ and for every $\theta \in \Theta$. Then, $W(R, \hat{\theta}) \to W(R, \theta^*)$ a.s.*

Proof. Since Θ is finite, it is compact, and since it is discrete, every function of θ is continuous in θ, since every subset of Θ is both open and closed. □

I have now distilled convergence of mechanism design based on simulation data into the problem of satisfying the basic preconditions in the main result of this section. Let me review these again for clarity:

- $W(r, \theta)$ and $P(r, \theta)$ are continuous on Θ
- $P_n(r, \theta)$ is continuous on Θ for all n
- Θ is compact
- $P_n(r, \theta) \to P(r, \theta)$ a.s. for every $r \in R$ and uniformly on Θ

One key item on this list that would generally be non-trivial to show is the last, that is, convergence of $P_n(r, \theta) \to P(r, \theta)$. In the sections which follow I demonstrate convergence for several techniques of constructing distributions of play in the sense of Chapter 11.

12.3 Convergence of Heuristic Distributions of Play

In this section, I study some distributions of agent play which satisfy the last sufficient condition above, useful in establishing convergence of approximately optimal mechanisms to actual optima.

[3] More precisely, continuity in a topological sense is guaranteed if Θ is finite.

12.3.1 Proportion-Based Distributions of Play

Let $m(r) \geq 0$ be some function which maps strategy profiles to non-negative real values. Suppose R is finite and assume that there is $r \in R$ such that $m(r) > 0$. I define a heuristic distribution of play based on the relative magnitudes of $m(r)$ for profiles $r \in R$ as follows:

$$P^p(r) = \frac{m(r)}{\sum_{t \in R} m(t)}. \tag{12.3}$$

That is, the probability that any profile $r \in R$ is played is proportional to its weight $m(r)$. The following general result demonstrates convergence of any distribution of play constructed in such a manner, as long as the corresponding weight function $m_n(r)$ converges to $m(r)$ with respect to the underlying game.

Theorem 12.8 *Suppose R is finite and $\exists r \in R \;\; s.t. \;\; m(r) > 0$. Suppose, forthermore, that $m_n(r) \to m(r)$ a.s. for every $r \in R$. Then $P_n^p(r, \theta) \to P^p(r, \theta) \;\; a.s. \; \forall \, \theta$ uniformly on R.*

Below, I present several convergence results for specific examples of weight functions $m(r)$, representing the relative likelihood of profiles under the belief distribution of play.

12.3.2 Proportions Based on $\epsilon(r)$

Let $m(r) = \exp\{-\epsilon(r)\}$. Intuitively, exponentially lower likelihood is assigned to profiles with higher game-theoretic regret. First, I restate a convergence theorem from Chapter 7.

Theorem 7.9 *Suppose that we have a finite game with joint mixed strategy set S. Then $\epsilon_n(s) \to \epsilon(s)$ a.s. uniformly on S.*

Corollary 12.9 *Let $R \subset S$. Then $m_n(r) \to m(r)$ a.s. uniformly on R.*

Proof. The result follows by the continuity of the exponential function. □

12.3.3 Proportions Based on Probability Distribution over $\epsilon(r)$

Suppose again that the game is finite with R a finite subset of the mixed strategy profile space (e.g., the set of all pure strategy profiles) Let $m(r) = \Pr\{\epsilon(r) = 0|\Gamma\}$ and $m_n(r) = \Pr\{\epsilon(r) = 0|\Gamma_n\}$, where Γ is the underlying game and Γ_n is the game in which payoffs are estimated based on n samples. Note that $m(r) = 1$ if r is a Nash equilibrium and $m(r) = 0$ otherwise.

Theorem 12.10 *Suppose the payoffs of Γ have no ties. Then $m_n(r) \to m(r)$ a.s. for every $r \in R$.*

12.3.4 Proportions Based on Approximate Equilibria

Definition 12.11 *r is a* strict *δ-Nash equilibrium if $\epsilon(r) < \delta$.*

Let $m(r) = I(r)$, where $I(r) = 1$ if r is an $\bar{\epsilon}$-Nash equilibrium, and $I(r) = 0$ otherwise. Assume also that $\bar{\epsilon}$ partitions R into $\{R_1, R_2\}$, R_1 and R_2 non-empty, where all $r \in R_1$ are strict $\bar{\epsilon}$-Nash equilibria and all $r \in R_2$ are not $\bar{\epsilon}$-Nash equilibria. First, let us ascertain that such an $\bar{\epsilon}$ always exists if R is finite and not all profiles have the same $\epsilon(r)$.

Lemma 12.12 *If R is finite and there are $r, t \in R$ such that $\epsilon(r) \neq \epsilon(t)$, there is $\bar{\epsilon}$ that partitions R into $\{R_1, R_2\}$, both non-empty, where all $r \in R_1$ are strict $\bar{\epsilon}$-Nash equilibria and all $r \in R_2$ are not $\bar{\epsilon}$-Nash equilibria.*

Proof. Let us order all profiles by $\epsilon(r)$. Suppose without loss of generality that $\epsilon(r) < \epsilon(t)$. Then, pick any $\bar{\epsilon} \in (\epsilon(r), \epsilon(t))$. □

Now I can prove convergence of $I_n(r)$ to $I(r)$ for every $r \in R$ (and, thus, uniformly on R).

Theorem 12.13 *$I_n(r) \to I(r)$ a.s. for every $r \in R$.*

12.4 Conclusion

This chapter addresses convergence of mechanism design choices made based on simulation-based games. My main result yields a set of sufficient conditions in terms of problem characteristics as well as the characteristics of the belief distributions of play used to evaluate the designer's objective. A key condition that must be satisfied is that the distributions of play are themselves convergent. I thus complement the main result with a series of positive examples of convergence of heuristic distributions of play.

CHAPTER 13

Conclusion and Future Work

13.1 Summary of Contributions

As the fields of mechanism design and game theory mature, both in the theoretical literature and in practical applications, the boundary of analytic tractability is increasingly being tested. Many applications have emerged for which theory provides relatively little guidance because no analytic characterization of the solutions can be obtained.

In my thesis, I propose a series of computational techniques for mechanism design and game-theoretic analysis applicable when analytical approaches become intractable. My work is divided into two main parts, corresponding to the two key questions that arise from a fairly natural one-shot two-stage mechanism design model.[1] In the first part, I tackle the mechanism design *search* problem, assuming that a subroutine for predicting game-theoretic outcomes for given mechanism choices is available. The second part is devoted to *game-theoretic analysis*, which corresponds to the problem of *predicting* the outcomes of strategic interactions between players for a *fixed* mechanism.

By combining the techniques which I describe in the first two parts of my thesis, I obtain a complete framework for mechanism design when the payoff functions of players, as well as the objective and constraints of the designer, are all specified algorithmically. As such, I provide the first (to my knowledge) complete methodology for performing

[1]The third part addresses some broad conceptual questions and convergence of simulation-based mechanism design. It does not introduce new computational techniques.

mechanism design in highly complex settings.

The ultimate goal of my methodology is not merely in producing some candidate mechanism given a particular problem, although that is certainly a necessary capability. Rather, the computational mechanism design techniques can be used to provide insight into problems that have until now been addressed using intuition and back-of-the-envelope calculations. For example, I was able to systematically demonstrate above that the design flaw in the supply-chain simulation could not be satisfactorily addressed by simple parameter tweaking. While I do not completely close the loop in computational analysis of simulation-based models, I provide some of the necessary means and make initial strides by the way of applications.

In the sections that follow, I summarize some of the specific contributions that I make towards the overall simulation-based mechanism design goal.

13.1.1 The Simulation-Based Mechanism Design Framework

At the root of my simulation-based mechanism design framework lies a simulated annealing stochastic search technique, which uses a game analysis tool as a subroutine. *The framework allows algorithmic specification of the designer objective and constraints, as well as the player payoff functions (and type distributions, when relevant).*

Instantiating my framework on one-shot Bayesian games, I present several approaches to evaluate the designer's objective with respect to the type distribution for specific design choices, and introduce a principled method for relaxing constraints in order to evaluate them based on a finite number of draws from the infinite space of joint player types. I demonstrate the effectiveness of the entire framework in the context of several non-trivial mechanism design problems. Finally, I provide some convergence guarantees in certain simulation-based mechanism design settings.

13.1.2 Solving Games Specified by Simulations

Much of my work aims to develop numerical tools to approximate solutions to games which are specified using simulations. *The purpose is to form* predictions *of play for a* *fixed mechanism design choice, and then use the methods as a subroutine in the mechanism design framework above.* In complete generality the problem is, of course, quite difficult. Thus, many of my tools attempt to exploit structure inherent in games (but unknown a priori). Additionally, since many interesting games possess strategy spaces that render numerical analysis intractable, appropriate choice of strategy space restriction is crucial to successful analysis. Some theoretical foundation for game-theoretic solution approximation via strategic restriction have been developed by Armantier *et al.* [2007]. My work builds on these to a limited degree, mostly borrowing inspiration and tools from the operation research and machine learning communities.

Methods and Sensitivity Analysis for Finite Games I begin my foray into solving simulation-based games by considering game models which have relatively small sets of players and strategies, but yield noisy payoff samples when queried with specific strategy profiles. In this setting, there is a straightforward approach to estimating Nash equilibria: first, estimate the payoff matrices of all players by taking samples from the simulator, and second, apply any finite-game solver to the estimated game. I extend this method to large games by appropriately generalizing the Nash equilibrium estimator based on an incompletely specified payoff matrix.

In the analysis of simulations it is standard to expect a report of probabilistic error bounds on the results. These would come, for example, in the form of confidence intervals around the reported means. It is natural, therefore, to expect that equilibrium results based on noisy simulations would be reported together with some confidence bounds. With this motivation in mind and armed with some convenient (and standard) normality assumptions, I derive a closed-form expression for the probability distribution of game-theoretic regret for pure and mixed strategy profiles in finite games based on simulation samples. Extensions to infinite games are straightforward when a Lipschitz continuity assumption is added.

Stochastic Search for Nash Equilibria An important question in the context of games specified using simulations is how to best collect samples to generate a data set of payoff experience which is most amenable to equilibrium approximation techniques.

In this thesis, I explore a set of techniques based on stochastic black-box optimization, which are geared specifically towards the Nash equilibrium approximation problem. I utilize the black-box optimization techniques from operations research by observing that Nash equilibrium approximation can be viewed as a two-tiered optimization problem: at the first (main loop) tier, we need to minimize the game-theoretic regret function, whereas at the second (subroutine) tier, we evaluate regret by approximating best responses for all players. I prove that an algorithm which uses simulated annealing in both the main loop and the subroutine is globally convergent in a black-box setting, even if the function is highly non-linear, payoff samples are stochastic, and the strategy profile space is infinite. As a result, the method is quite broadly applicable, and I demonstrate its effectiveness in the context of infinite one-shot Bayesian games.

While the two-tiered equilibrium approximation method is convergent and performs quite well in practice, I show empirically that it is outperformed by approximate best-response dynamics. The latter is not generally convergent, but will often reach very good Nash equilibrium approximations very quickly.

Nash Equilibrium Approximation in Infinite Games Using Machine Learning Since I allow games to be specified using simulations, I cannot make any assumptions on the structure of the game directly. Nevertheless, I can hypothesize a structure; that is, I can posit a hypothesis class of functions which contains good approximations of the underlying payoff function, and then perform regression learning based on the data set obtained from simulation experiments—for example, the data trail of the stochastic search techniques I just described.

The techniques in supervised learning typically focus on optimal generalization ability of the regression. My purpose for using them, however, is a different sort of generalization: I would like to obtain good approximations of Nash equilibria based on the learned payoff function. Thus, I empirically evaluate the performance of learning algorithms with respect to game-theoretic regret. Additionally, I contrast the techniques that use the payoff function as the learning target with those using as target (empirical) game-theoretic regret.

In the end, I demonstrate that machine learning techniques provide substantial improvement as compared to selecting the best approximate Nash equilibrium (that is, a profile with lowest empirical regret) based on the empirical game data alone, and provide some empirical guidance in choosing the appropriate learning target *for the purpose of Nash equilibrium approximation based on a data set obtained from simulation-based games*.

13.1.3 Applications

While much of my work has focused on the development of analysis tools, much of it was performed in the context of particular applications. One such application was the strategic and mechanism design analysis of the Supply-Chain scenario in the Trading Agent Competition, where I demonstrated that a well-recognized problem with the supply-chain simulation design could not be fixed by parameter tweaking alone. Another was in the analysis of dynamic strategies in sponsored search auctions. In this domain I explore the stability and payoff properties of several greedy bidding strategies, demonstrating that the one which is convergent is also the most stable. In this work, I also study collusive bidding strategies, both in the context of complete and incomplete information in the repeated game setting. One of the main results is to demonstrate, using both theoretical and simulation-based evidence, that players can construct strategies which result in collusion after a finite number of rounds.

13.2 Future Work

In my thesis, I have built the groundwork for addressing complex mechanism design and game-theoretic analysis problems. Below, I describe some of the applications which may now be tackled using the techniques presented above.

13.2.1 Strategic Analysis of Electricity Markets

Electricity markets have been very actively studied in recent years, both by academics and policy-makers. What makes this domain particularly interesting from my perspective is the fact that, while many electricity markets are run as uniform-price auctions, relatively little is understood about these in general. Indeed, the broader problem of auctioning divisible goods has proven very difficult to grapple analytically in many settings. One consequence of this has been the increasing importance of empirical research. Another, I hope, will be the great value of computational tools in providing additional insights about the problem, both from the strategic bidding and the mechanism design perspectives.

I expect to begin by analyzing uniform-price multi-unit auctions to obtain better insight about the shape of bidding strategies and the magnitude of economic inefficiency, and the relationship of these to various parameters which characterize the auction environment. I believe that the results will then be generalizable to settings with infinitely divisible goods (such as electricity), and will proceed thereafter to study the effect of additional complications, such as interactions between spot and futures markets.

13.2.2 Healthcare Economics: Analysis Using Simulations

My agenda for studying the Economics of healthcare and, particularly, the incentives in the healthcare system, is fairly ambitious: I hope to make use of and enhance healthcare system simulations, focusing particularly on the strategic interactions between the important players (healthcare providers, insurers, etc). A venue that may facilitate considerably the development of an appropriate simulation environment is the International Trading Agent Competition (TAC). TAC is an annual competition which features a series of simulation scenarios developed specifically to engage competition of autonomous agents designed by various independent research groups. Perhaps a healthcare scenario which features competition between large healthcare providers (e.g., hospitals) or health insurers would be feasible as another TAC game and would be of interest to the community. The ultimate goal of this simulation environment is to provide a testbed for healthcare policy analysis.

13.2.3 Other Applications

Besides the applications to electricity markets and healthcare policy analysis, numerous other problems can provide fertile ground for applications of my thesis research. My work already features applications to a supply-chain simulation, albeit somewhat focused on the particulars of the simulation for very restricted agent strategy spaces. The actual strategies of agents trading as a part of supply chains—even in the simplified simulation setting—are extremely complex, and it may be fruitful to distill and analyze a richer game than had thus far been considered. Additionally, I am interested in the problem of efficient resource sharing in large distributed systems, such as GRID computing.

APPENDICES

APPENDIX A

Glossary of Acronyms

- *AMD:* automated mechanism design
- *BMD:* Bayesian mechanism design
- *BNIC:* Bayes-Nash incentive compatible
- *DSIC:* dominant strategy incentive compatible (strategyproof)
- *DSMD:* dominant strategy mechanism design
- *EIIR:* ex interim individually rational
- *EPE:* ex post efficient
- *EPIR:* ex post individually rational
- *LCP:* linear complementarity problem
- *MIP:* mixed integer program
- *PRM:* partial revelation mechanism
- *SCF:* social choice function
- *SWF:* social welfare function(al)
- *VCG:* Vickrey-Clarke-Groves mechanism

APPENDIX B

Trading Agent Competition and the Supply-Chain Game

B.1 Trading Agent Competition (TAC)

Many a year ago, in an Artificial Intelligence Laboratory far, far away, an idea was born to create a simulation environment in which autonomous trading agents can compete. The brain-child of an adventurous Computer Science professor and his faithful students, the idea matured quickly and in year 2000 burgeoned into the first annual Trading Agent Competition.

The first TAC featured a travel shopping scenario, in which trading agents compete to satisfy customer orders given the limited hotel and entertainment resources, and unlimited (but, in later versions, progressively more expensive) flights. After several successful years, and having attracted considerable international attention, the competition expanded to include a new game, in which trading agents compete as manufacturers in a supply chain. Since the supply-chain game is of particular relevance to this work, I proceed with a detailed description below.

B.2 The TAC Supply-Chain Game

In the TAC/SCM scenario,[1] six agents representing PC (personal computer) assemblers operate in a common market environment, over a simulated production horizon. The environment constitutes a *supply chain*, in that agents trade simultaneously in markets for supplies (PC components) and the market for finished PCs. Agents may assemble for sale 16 different models of PCs, defined by the compatible combinations of the four component types: CPU, motherboard, memory, and hard disk.

Figure B.1 diagrams the basic configuration of the supply chain. The six agents (arrayed vertically in the middle of the figure and to be supplied by the competing teams) procure components from the eight suppliers on the left, and sell PCs to the entity representing customers, on the right. Trades at both levels are negotiated through a *request-for-quote* (RFQ) mechanism, which proceeds in three steps:

1. Buyer issues RFQs to one or more sellers.

2. Sellers respond to RFQs with *offers*.

3. Buyers accept or reject offers. An accepted offer becomes an *order*.

The suppliers and customer implement fixed negotiation policies, defined in the game specification, and discussed in detail below where applicable.

The game runs for 220 simulated days. On each day, the agent may receive offers and component delivery notices from suppliers and RFQs and offer acceptance notifications from customers. It then must make several decisions:

1. What RFQs to issue to component suppliers.

2. Given offers from suppliers (based on the previous day's RFQs), which to accept.

3. Given component inventory and factory capacity, what PCs to manufacture.

4. Given inventory of finished PCs, which customer orders to ship.

[1]For more details and latest information, see `http://www.sics.se/tac`. Below, we follow with a specifications of the game for the 2003 and 2004 competitions.

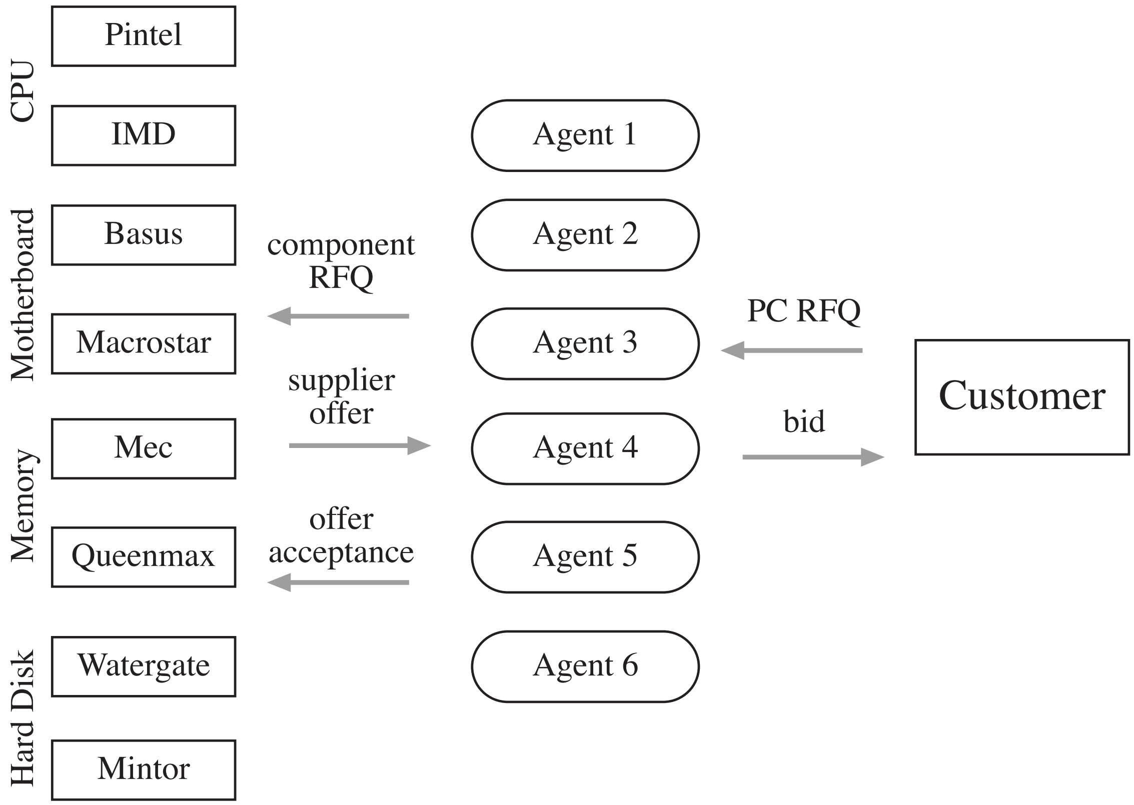


Figure B.1: TAC/SCM supply chain.

5. Given RFQs from customers, to which to respond and with what offers.

In the simulation, the agent has 15 seconds to compute and communicate its daily decisions to the game server. At the end of the game, agents are evaluated by total profit, with any outstanding component or PC inventory valued at zero.

A key stochastic feature of the game environment is level of demand for PCs. The underlying demand level is defined by an integer parameter Q (called RFQ_{avg} in the specification document [Arunachalam *et al.*, 2003, Section 6]). Each day, the customer issues a set of $\hat{Q}$ RFQs, where $\hat{Q}$ is drawn from a Poisson distribution with mean value defined by the parameter Q for that day. Since the order quantity, PC model, and reserve price are set independently for each customer RFQ, the number of RFQs serves as a sufficient statistic for the overall demand, which in turn is a major determinant of the potential profits available to the agents.

The demand parameter Q evolves according to a given stochastic process. In the TAC/SCM 2003 specification, the process was defined as follows. In each game instance, an initial value, Q_0, is drawn uniformly from [80,320]. If Q_d is the value of Q on day d, then its value on the next day is given by [Arunachalam *et al.*, 2003, Section 6]:

$$Q_{d+1} = \min(320, \max(80, \tau_d Q_d)), \tag{B.1}$$

where τ is a trend parameter that also evolves stochastically. The initial trend is neutral, $\tau_0 = 1$, with subsequent trends updated by a perturbation $\epsilon \sim U[-0.01, 0.01]$:

$$\tau_{d+1} = \max(0.95, \min(1/0.95, \tau_d + \epsilon)). \tag{B.2}$$

In a given game, the demand may stay at predominantly high or low levels, or oscillate back and forth. As it turns out, the resulting distribution of average RFQs per day (Figure B.2) has spikes at the low and high levels; indeed, we can see that games are most likely to have average demand in the low ranges. As this was considered anomalous

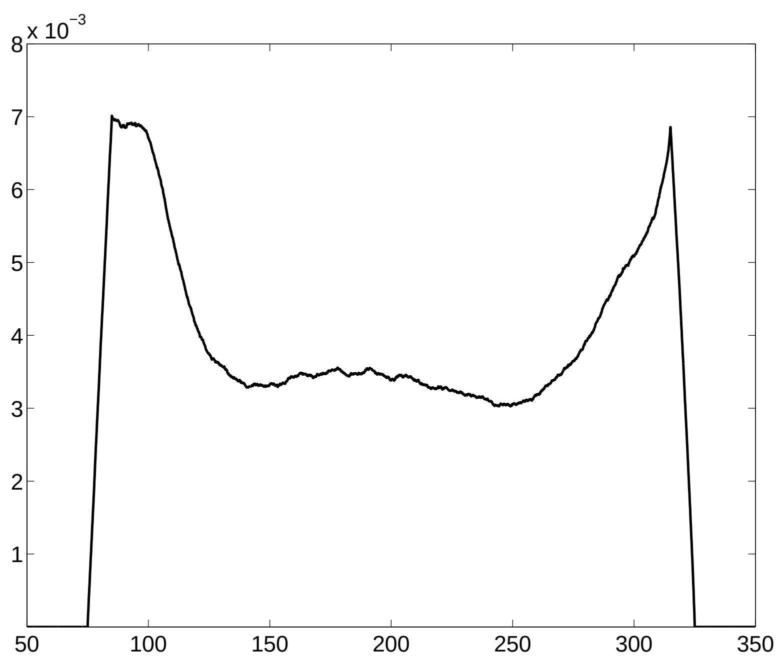


Figure B.2: Probability density for average RFQs per day ($\bar{Q}$) in TAC/SCM 2003.

and undesirable by the game masters, the demand process was modified for TAC/SCM 2004 to make the average demand unimodal, with medium-range values occuring with the highest probability.

In the TAC/SCM 2003/04 market, suppliers set prices for components based on an analysis of their available capacity. Conceptually, there exist separate prices for each type of component, from each supplier. Moreover, these prices vary over time: both the time that the deal is struck, and time that the component is promised for delivery.

The TAC/SCM 2003 component catalog [Arunachalam *et al.*, 2003, Figure 3] associates every component c with a *base price*, b_c. The correspondence between price and quantity for component supplies is defined by the suppliers' pricing formula [Arunachalam *et al.*, 2003, Section 5.5]. The price offered by a supplier at day d for an order to be delivered on day $d + i$ is

$$p_c(d+i) = b_c - 0.5 b_c \frac{\kappa_c(d+i)}{500i}, \tag{B.3}$$

where for any j, $\kappa_c(j)$ denotes the cumulative capacity for c the supplier projects to have available from the current day through day j. The denominator, $500i$, represents the *nominal capacity* controlled by the supplier over i days, not accounting for any capacity committed to existing orders.

Supplier prices according to Eq. (B.3) are date-specific, depending on the particular pattern of capacity commitments in place at the time the supplier evaluates the given RFQ. A key observation is that component prices are never lower than at the start of the game ($d = 0$), when $\kappa_c(i) = 500i$ and therefore $p_c(i) = 0.5b_c$, for all c and i.[2] As the supplier approaches fully committed capacity ($\kappa_c(d+i) \to 0$), $p_c(d+i)$ approaches b_c. While some changes were made to the supplier model for TAC/SCM 2004, this key design aspect was retained.

In general, one would expect that procuring components at half their base price would be profitable, up to the limits of production capacity. Customer reserve prices range between 0.75 and 1.25 the base price of PCs, defined as the sum of base prices of components. Therefore, unless there is a significant oversupply, prices for PCs should easily exceed the component cost, based on day-0 prices.

[2]As discussed below, this creates a powerful incentive for early procurement, with significant consequences for game balance. In retrospect, the supplier pricing rule was generally considered a design flaw in the game, and has been substantially revised for the 2003 TAC/SCM tournament.

An agent's procurement strategy must also take into account the specific TAC/SCM RFQ process. Each day, agents may submit up to 10 RFQs, ordered by priority, to each supplier. The suppliers then repeatedly execute the following, until all RFQs are exhausted: (1) randomly choose an agent,[3] (2) take the highest-priority RFQ remaining on its list, (3) generate a corresponding offer, if possible. In responding to an RFQ, if the supplier has sufficient available capacity to meet the requested quantity and due date, it offers to do so according to its pricing function. If it does not, the supplier instead offers a partial quantity at the requested date and/or the full quantity at a later date, to the best of its ability given its existing commitments. In all cases, the supplier quotes prices based on Eq. (B.3), and reserves sufficient capacity to meet the quantity and date offered.

Although it necessarily simplifies the PC market and manufacturing process to a great extent, the TAC/SCM game does introduce several realistic elements not typically incorporated in trading games. Specifically, it embeds trading in a concrete production context, and incorporates stochastic (and partially observable) exogenous effects. Like actors on real supply chains, TAC/SCM agents make decisions under uncertainty over time, dealing with both suppliers and customers, in a competitive market environment. Negotiation concerns several facets of a deal (price, quantity, delivery time, penalty), and takes place simultaneously at multiple tiers.

B.2.1 TAC/SCM 2003: Preemptive Strategy

The Deep Maize preemptive strategy operates by submitting a large RFQ to each supplier for each component produced. The preemptive RFQ requests 85000 units—representing 170 days' worth of supplier capacity—to be delivered by day 30. See Figure B.3. It is of course impossible for the supplier to actually fulfill this request. Instead, the supplier will offer us both a partial delivery on day 30 of the components they can offer by that date (if any), and an earliest-complete offer fulfilling the entire quantity (unless the supplier has already committed 50 days of capacity). With these offers, the

[3] At the start of the game, suppliers select among agents with equal probability. Thereafter, suppliers employ a reputation mechanism whereby the probability of agent choice depends on its record of accepting previous offers. We discuss the operation and effectiveness of this mechanism below.

supplier reserves necessary capacity. This has the effect of preempting subsequent RFQs, since we can be sure that the supplier will have committed capacity at least through day 172. (The extra two days account for negotiation and shipment time.) We will accept the partial-delivery offer, if any (and thereby reject the earliest-complete), giving us at most 14000 component units to be delivered on day 30, a large but feasible number of components to use up by the end of the game.

Figure B.3: Deep Maize's preemptive RFQ.

In the situation that our preemptive RFQ gets considered after the supplier has committed 50 days of production to other agents, we will not receive an offer, and our preemption is unsuccessful. For this reason, we also submit backup RFQs of 35000 to be delivered on day 50, and 15000 to be delivered on day 70.

The effect of a preemptive RFQ clearly depends on the random order by which its target supplier selects agents to consider. On each selection, Deep Maize will be picked with probability 1/6, which means that with probability $1 - \frac{5}{6}^k$ it is selected within the first k RFQs. For example, the preemptive RFQ appears among the first four 51.8% of the time. Given these orderings are also generated independently for each supplier-component combination, with high probability Deep Maize is expected to successfully preempt a significant number of the other agents' day-0 RFQs.

The TAC/SCM designers anticipated the possibility of preemptive RFQ generation, (there was much discussion about it in the original design correspondence), and took steps to inhibit it. The designers instated a *reputation mechanism*, in which refusing offers from suppliers reduces the priority of an agent's RFQs being considered in the future. This is accomplished by adjusting agent i's selection probability π_i as follows [Arunachalam *et* *al.*, 2003, Section 5.1]:

$$\begin{aligned} \text{weight}_i &= \max\left(0.5, \frac{\text{QuantityPurchased}_i}{\text{QuantityRequested}_i}\right), \\ \pi_i &= \frac{\text{weight}_i}{\sum_x \text{weight}_x}. \end{aligned}$$

Even with this deterrent, we felt our preemptive strategy would be worthwhile. Since most agents were focusing strongly on day 0, priority for RFQ selection on subsequent days might not turn out to be crucial. In any event, we did not seem to have viable alternatives.

We deployed the preemptive strategy in the semifinal rounds of TAC/SCM 2003. Although none had anticipated it explicitly, it turned out that most agents playing in the finals were individually flexible enough to recover from day-0 preemption. By preempting, it seemed that Deep Maize had leveled the playing field, but RedAgent's apparent adaptivity in procurement and sales [Keller *et al.*, 2004] earned it the top spot in the competition rankings.

APPENDIX C

Proofs for Chapter 4

C.1 Proof of Proposition 4.5

First, let us suppose that some function, $f(x)$ defined on $[a_i, b_i]$, satisfy Lipschitz condition on $(a_i, b_i]$ with Lipschitz constant A_i. Then the following claim holds:

Claim C.1 $\inf_{x \in (a_i, b_i]} f(x) \geq 0.5(f(a_i) + f(b_i) - A_i(b_i - a_i)$.

To prove this claim, note that the intersection of lines at $f(a_i)$ and $f(b_i)$ with slopes $-A_i$ and A_i respectively will determine the lower bound on the minimum of $f(x)$ on $[a_i, b_i]$ (which is a lower bound on infimum of $f(x)$ on $(a_i, b_j]$). The line at $f(a_i)$ is determined by $f(a_i) = -A_i a_i + c_L$ and the line at $f(b_i)$ is determined by $f(b_i) = A_i b_i + c_R$. Thus, the intercepts are $c_L = f(a_i) + A_i a_i$ and $c_R = f(b_i) + A_i b_i$ respectively. Let x^* be the point at which these lines intersect. Then,

$$x^* = -\frac{f(x^*) - c_R}{A} = \frac{f(x^*) - c_L}{A}.$$

By substituting the expressions for c_R and c_L, we get the desired result.

Now, subadditivity gives us

$$\Pr\{\bigvee_{\theta \in \Theta} \sup\{\phi^*(\theta)\} \leq \alpha\} \leq \sum_{j=1}^{5} \Pr\{\bigvee_{\theta \in \Theta_j} \sup\{\phi^*(\theta)\} \leq \alpha\},$$

and, by the claim,

$$\Pr\{\bigvee_{\theta\in\Theta_j}\sup\{\phi^*(\theta)\}\leq\alpha\} = 1-\Pr\{\bigwedge_{\theta\in\Theta_j}\sup\{\phi^*(\theta)\}>\alpha\} =$$
$$= 1-\Pr\{\inf_{\theta\in\Theta_j}\sup\{\phi^*(\theta)\}>\alpha\}\leq\Pr\{\sup\{\phi^*(a_j)\}+\sup\{\phi^*(b_j)\}\leq$$
$$\leq 2\alpha + A_j(b_j-a_j)\}.$$

Since we have a finite number of points in the data set for each θ, we can obtain the following expression:

$$\Pr\{\sup\{\phi^*(a_j)\}+\sup\{\phi^*(b_j)\}\leq c_j\} =_D$$
$$\sum_{y,z\in D:y+z\leq c_j}\Pr\{\sup\{\phi^*(b_j)\}=y\}\Pr\{\sup\{\phi^*(a_j)\}=z\}.$$

We can now restrict attention to deriving an upper bound on $\Pr\{\sup\{\phi^*(\theta)\}=y\}$ for a fixed θ. To do this, observe that

$$\Pr\{\sup\{\phi^*(\theta)\}=y\}\leq_D\Pr\{\bigvee_{a\in D:\phi(a)=y}\epsilon(a)=0\}\leq\sum_{a\in D:\phi(a)=y}\Pr\{\epsilon(a)=0\}$$

by subadditivity and the fact that a profile a is a Nash equilibrium if and only if $\epsilon(a)=0$.

Putting everything together yields the desired result.

APPENDIX D

Proofs for Chapter 5

D.1 Proof of Theorem 5.1

Note that α is just the probability that the actual measure r of set B is above p if none of n i.i.d. samples X_i from the type distribution violated the constraint:

$$\alpha = \Pr\{r \geq p \mid \forall i = 1, \ldots, n, X_i \notin B\} = \frac{\Pr\{\forall i = 1, \ldots, n, X_i \notin B \& r \geq p\}}{\Pr\{\forall i = 1, \ldots, n, X_i \notin B\}}.$$

Since the samples are i.i.d.,

$$\Pr\{\forall i = 1, \ldots, n, X_i \notin B \mid r\} = (1 - r)^n,$$

and since we assumed a uniform prior on r, we get

$$\Pr\{\forall i = 1, \ldots, n, X_i \notin B\} = \int_0^1 (1 - r)^n dr = \frac{1}{n + 1}$$

and

$$\Pr\{\forall i = 1, \ldots, n, X_i \notin B \& r \leq p\} = \int_p^1 (1 - r)^n dr = \frac{(1 - p)^{n+1}}{n + 1}.$$

Consequently, we obtain the following relationship between α, p, and n:

$$\alpha = (1 - p)^{n+1}.$$

Solving for n, we get

$$n = \frac{\log \alpha}{\log(1-p)} - 1.$$

D.2 Proof of Theorem 5.6

Suppose p is the probability measure of T_A and suppose we select the best θ_i of $\{\theta_1, \ldots, \theta_L\}$. Suppose further that we take n samples for each θ_j, and let T^n be the set of n type realizations. We will also use the notation $\theta \in G$ to indicate an event that for a particular θ, $\min_{t \in T^n} W(r, t, \theta) > \inf_{t \in T \setminus T_A} W(r, t, \theta)$.

We would like to compute the number of samples n for each of these samples such that $P\{\theta_i \notin G\} \geq 1 - \alpha$.

Note that

$$P\{\theta_i \notin G\} \geq P\{\theta_1 \notin G \& \cdots \& \theta_L \notin G\} = P\{\theta_j \notin G\}^L.$$

Now,

$$P\{\theta_j \in G\} = P\{t_1 \notin T_A \& \cdots \& t_n \notin T_A\} = P\{t_i \notin T_A\}^n = (1-p)^n.$$

Thus,

$$P\{\theta_i \notin G\} \geq (1 - (1-p)^n)^L = 1 - \alpha.$$

Solving for n, we obtain the desired answer.

D.3 Proof of Theorem 5.8

We show that for the two-player game with types $U[A, B]$ and payoff function

$$u(t, a, t', a') = \begin{cases} t - ha - ka' & \text{if } a > a' \\ \frac{t - ha - ka' + ha' + ka}{2} & \text{if } a = a' \\ ha' + ka & \text{if } a < a', \end{cases}$$

with $h, k \geq 0$ and $B \geq A + 1$ that the following is a symmetric Bayes-Nash equilibrium strategy:

$$\frac{t}{3(h+k)} + \frac{hA + kB}{6(h+k)^2}. \tag{D.1}$$

Consider first the special case that $h = k = 0$. Equation D.1 prescribes a strategy of bidding ∞ and it is clear that this is a dominant strategy in a game where the winner is the high bidder with no payments required.[1] We will now assume that $h + k > 0$.

Define $m \equiv \frac{1}{3(h+k)}$ and $c \equiv \frac{hA+kB}{6(h+k)^2}$ and let T be a random $U[A, B]$ variable giving the opponent's type. Noting that the tie-breaking case ($a = a'$) happens with zero probability given that (D.1) is a continuous function of a uniform random variable, we write the expected utility for an agent of type t playing action a as

$$\begin{aligned}
\mathrm{EU}(t, a) &= E_T[u(t, a, T, mT + c)] \\
&= E[t - ha - k(mT + c) \mid a > mT + c] \Pr(a > mT + c)] \\
&\quad + E[h(mT + c) + ka \mid a < mT + c] \Pr(a < mT + c) \\
&= E\left[t - ha - kmT - kc \;\middle|\; T < \frac{a-c}{m}\right] \Pr\left(T < \frac{a-c}{m}\right) \\
&\quad + E\left[hmT + hc + ka \;\middle|\; T > \frac{a-c}{m}\right] \Pr\left(T > \frac{a-c}{m}\right)
\end{aligned} \tag{D.2}$$

We consider three cases on the range of a and find the optimal action a_i^* for each case i.

Case 1: $a \leq Am + c$. $\quad (\implies \frac{a-c}{m} \leq A)$

The probabilities in (D.2) are zero and one, respectively, and so the expected utility is:

$$\mathrm{EU}(t, a) = hm\frac{A+B}{2} + hc + ka.$$

This is an increasing function in a, implying an optimal action at the right boundary:

[1] This assumes that the space of possible bids includes ∞. More generally, the dominant strategy is the supremum of the bid space but if this is not itself a member of the bid space (as is the case if the bid space is $\mathbb{R}$) then there is in fact no Nash equilibrium of the game.

$a_1^* = Am + c$. Thus the best expected utility for case 1 is

$$\mathrm{EU}(t, a_1^*) = \frac{2A + B}{6}.$$

Case 2: $a \geq Bm + c$. $\quad (\implies \frac{a-c}{m} \leq B)$

The probabilities in (D.2) are one and zero, respectively, and so the expected utility is:

$$\mathrm{EU}(t, a) = t - ha - km\frac{A + B}{2} - kc.$$

This is a decreasing function in a, implying an optimal action at the left boundary: $a_2^* = Bm + c$. Thus the best expected utility for case 2 is

$$\mathrm{EU}(t, a_2^*) = t - \frac{A + 2B}{6}.$$

Case 3: $Am + c < a < Bm + c$.

Knowing that $\frac{a-c}{m}$ is between A and B it is straightforward to compute the probabilities in (D.2) and the conditional expectation of T. So we write $\mathrm{EU}(t, a)$ as:

$$\begin{aligned}
&\left(t - ha - km\frac{A + \frac{a-c}{m}}{2} - kc\right)\left(\frac{a-c}{m} - A\right) \\
&+ \left(hm\frac{B + \frac{a-c}{m}}{2} + hc + ka\right)\left(B - \frac{a-c}{m}\right) \\
=&(-108a^2h^4 - 432a^2kh^3 - 648a^2k^2h^2 - 432a^2k^3h- \\
&- 108a^2k^4 + 36aAh^3 + 72ath^3 + A^2h^2 + 4B^2h^2+ \\
&+ 4ABh^2 + 72aAkh^2 + 36aBkh^2 - 36Ath^2 + 216akth^2+ \\
&+ 36aAk^2h + 72aBk^2h + 8A^2kh + 8B^2kh + 2ABkh+ \\
&+ 216ak^2th - 60Akth - 12Bkth + 36aBk^3 + 4A^2k^2 + B^2k^2 \\
&+ 4ABk^2 + 72ak^3t - 24Ak^2t - 12Bk^2t)/(24(h + k)^2).
\end{aligned}$$

Since this is a concave function of a the maximum is where the derivative with respect to

a is zero, that is (skipping the tedious algebra for which we used Mathematica):

$$\frac{\partial\,\mathrm{EU}(t,a)}{\partial a} = 0$$
$$\implies a_3^* = \frac{t}{3(h+k)} + \frac{hA+kB}{6(h+k)^2}.$$

Since $A \le t \le B \implies Am + c \le a_3^* \le Bm + c$, a_3^* is in fact in the allowable range for case 3. The expected utility for case 3 is then

$$\mathrm{EU}(t, a_3^*) = \frac{3t^2 + A^2 + B^2 + A(B-6t)}{6}.$$

It now remains to show that neither $\mathrm{EU}(t, a_1^*)$ nor $\mathrm{EU}(t, a_2^*)$ is greater than $\mathrm{EU}(t, a_3^*)$ for any t.

Since $t \ge A$ there exists a $\delta \ge 0$ such that $t = A + \delta$. And since $B \ge A + 1$ there exists an $\varepsilon \ge 0$ such that $B = A + 1 + \varepsilon$. First, $\mathrm{EU}(t, a_3^*) \ge \mathrm{EU}(t, a_2^*)$ because

$$\begin{aligned}
& (\delta-1)^2 \ge 0 \\
\implies & \delta^2 - 2\delta + 1 \ge 0 \\
\implies & \delta^2 + 1 \ge 2\delta \\
\implies & (A+\delta-A)^2 + 2A + 1 \ge 2A + 2\delta \\
\implies & (t-A)^2 + 2A + 1 \ge 2t \\
\implies & t^2 + A^2 + 2A + 1 \ge 2At + 2t \\
\implies & 3t^2 + 3A^2 + 6A + 3 + (3A\varepsilon + \varepsilon^2 + 4\varepsilon) \ge 6At + 6t \\
\implies & 3t^2 + A^2 + (A^2 + 2A + 2A\varepsilon + \varepsilon^2 + 2\varepsilon + 1) + \\
& \quad + (A^2 + A + A\varepsilon) - 6At \ge 6t - A - 2A - 2 - 2\varepsilon \\
\implies & 3t^2 + A^2 + (A+1+\varepsilon)^2 + A(A+1+\varepsilon) - 6At \\
& \quad \ge 6t - A - 2(A+1+\varepsilon) \\
\implies & 3t^2 + A^2 + B^2 + AB - 6At \ge 6t - A - 2B.
\end{aligned}$$

Finally, $\mathrm{EU}(t, a_3^*) \geq \mathrm{EU}(t, a_1^*)$ because

$$\begin{aligned}
&(t-A)^2 \geq 0 \\
\Longrightarrow\ & t^2 - 2At + A^2 \geq 0 \\
\Longrightarrow\ & t^2 + A^2 \geq 2At \\
\Longrightarrow\ & 3t^2 + 3A^2 \geq 6At \\
\Longrightarrow\ & 3t^2 + 3A^2 + (3A\varepsilon + \varepsilon^2 + \varepsilon) \geq 6At \\
\Longrightarrow\ & 3t^2 + 3A^2 + 3A + 3A\varepsilon + \varepsilon^2 + \varepsilon - 6At \geq 3A \\
\Longrightarrow\ & 3t^2 + (A^2 + A + \varepsilon) - 6At + \\
& \quad + (A^2 + 2A + 2A\varepsilon + \varepsilon^2 + 2\varepsilon + 1) + A^2 \geq 3A + \varepsilon + 1 \\
\Longrightarrow\ & 3t^2 + A(A + 1 + \varepsilon) - 6At + A^2 + (A + 1 + \varepsilon)^2 \\
& \quad \geq 2A + (A + \varepsilon + 1) \\
\Longrightarrow\ & 3t^2 + AB - 6At + A^2 + B^2 \geq 2A + B.
\end{aligned}$$

D.4 Proof of Theorem 5.9

It is direct from Theorem 5.8 that setting $h = 1/3$ and $k = 0$ yields a symmetric Bayes-Nash equilibrium $s(t) = t$ when $A = 0$. We now show that the best response to truthful bidding is only truthful under this parameter setting—i.e., that $\mathrm{SGA}(1/3, 0)$ is the only BNIC game in the SGA family, for $U[0, B]$ types.

Suppose that the opponent bids truthfully (i.e., $s(t) = t$ for one of the agents). First, assume that $a \in [0, B]$. The expected utility of an agent with type t from bidding a is then

$$\begin{aligned}
\mathrm{EU}(t, a) &= \int_0^a (t - ha - kT)dT + \int_a^1 (hT + ka)dT = \\
&= \frac{1}{2}\left(-3(h+k)a^2 + 2(Bk + t)a + B^2 h\right).
\end{aligned}$$

Since this function is strictly concave in a, we can use the first-order condition to find the optimum bid:

$$\frac{\partial\,\mathrm{EU}(t,a)}{\partial a} = t - 3(h+k)a + Bk = 0$$

yielding

$$a = \frac{t+Bk}{3(h+k)}, \tag{D.3}$$

which is truthful for every type t only when $h = 1/3$ and $k = 0$.

Now, if $a \leq 0$, it will always lose, and the expected utility is

$$\mathrm{EU}(t,a) = \int_0^B (hT + ka)dT = B^2h/2 + kBa,$$

which is maximized when $a = 0$. Consequently, there is no incentive to ever bid below 0. Similarly, if $a \geq B$, the agent will never lose, and

$$\mathrm{EU}(t,a) = \int_0^B (t - ha - kT)dT = -\frac{1}{2}B(2ah + Bk - 2t),$$

which is maximized when $a = B$. Thus, there is no incentive to ever bid above B. All incentive compatible mechanisms will thus induce bidding according to (D.3). It follows, then, that $\mathrm{SGA}(1/3, 0)$ is the only truthful mechanism for $U[0, B]$ ($B > 0$) types.

D.5 Proof of Theorem 5.10

The objective function in terms of h and k is

$$\min_{h,k} |E[t_w - 2h(\frac{t_w}{3(h+k)} + \frac{k}{6(h+k)^2}) - 2k(\frac{t_l}{3(h+k)} + \frac{k}{6(h+k)^2}) \mid t_w > t_l]|.$$

Since $E[t_w \mid t_w > t_l]$ is the expectation of the first order statistic of two $U[0, 1]$ random variables, it is $2/3$ (and $1/3$ for t_l). Thus, the objective function above reduces to

$$\min_{h,k} \left| \frac{2h+k}{9(h+k)} \right|.$$

We now show that this expression cannot be less than 1/9:

$$\begin{aligned} & h \geq 0 \\ \Longrightarrow\ & 2h \geq h \\ \Longrightarrow\ & 2h + k \geq h + k \\ \Longrightarrow\ & \frac{2h+k}{h+k} \geq 1 \\ \Longrightarrow\ & \frac{2h+k}{9(h+k)} \geq \frac{1}{9}. \end{aligned}$$

Since setting $h = 0$ yields the minimum of 1/9 for any $k > 0$ we conclude that all mechanisms $\mathrm{SGA}(0, k)$ minimize the objective function.

D.6 Proof of Theorem 5.11

Let R designate the expected revenue of the winner.

$$\begin{aligned} R &= E[t_w - h(t_w/3(h+k) + k/6(h+k)^2) - k(t_l/3(h+k) + k/6(h+k)^2)] = \\ &= E[t_w] - hE[t_w]/3(h+k) - hk/6(h+k)^2 - kE[t_l]/3(h+k) - k^2/6(h+k)^2 = \\ &= 2/3 - (4h+5k)/18(h+k) = 4/9 - k/18(h+k). \end{aligned}$$

D.7 Proof of Theorem 5.12

Let R designate the expected revenue of the loser.

$$\begin{aligned} R &= E[h(t_w/3(h+k) + k/6(h+k)^2) + k(t_l/3(h+k) + k/6(h+k)^2)] = \\ &= hE[t_w]/3(h+k) + hk/6(h+k)^2 + kE[t_l]/3(h+k) + k^2/6(h+k)^2 = \\ &= (4h+5k)/18(h+k) = 2/9 + k/18(h+k). \end{aligned}$$

D.8 Proof of Theorem 5.13

First, we obtain the expression to be minimized.

$$\sup_{t>t'}|t - 2h(\frac{t}{3(h+k)} + \frac{k}{6(h+k)^2}) - 2k(\frac{t'}{3(h+k)} + \frac{k}{6(h+k)^2})| =$$

$$= \sup_{t>t'}|t - \frac{2ht+2kt'}{3(h+k)} - \frac{k}{3(h+k)}| = \sup_{t>t'}|\frac{ht+3kt-2kt'-k}{3(h+k)}|.$$

Clearly, this is minimized when $t = 1$ and $t' = 0$, yielding

$$\frac{h+3k-k}{3(h+k)} = \frac{h+2k}{3(h+k)}.$$

Now, note that since $h, k \geq 0$,

$$\frac{h+2k}{3(h+k)} \geq \frac{h+k}{3(h+k)} = \frac{1}{3}.$$

Thus, the expression cannot be less than 1/3. Consequently, since setting $k = 0$ for any $h > 0$ results in the objective function value of 1/3, it describes a subset of optimal values.

D.9 Proof of Theorem 5.14

$$\inf_{t>t'}[t - h(\frac{t}{3(h+k)} + \frac{k}{6(h+k)^2}) - k(\frac{t'}{3(h+k)} + \frac{k}{6(h+k)^2})] =$$

$$= \inf_{t>t'}[t - \frac{ht'+kt}{3(h+k)} - \frac{k}{6(h+k)}] = \inf_{t>t'}\frac{h(t-t')}{3(h+k)} - \frac{k}{6(h+k)}.$$

The infimum is equivalent to setting $t = 0$ and $t' = 0$, and thus the expression is maximized if

$$\frac{k}{6(h+k)}$$

is minimized, which is effected by setting $k = 0$. The resulting optimal value is 0.

D.10 Proof of Theorem 5.15

$$\inf_{t>t'}[h(\frac{t}{3(h+k)}+\frac{k}{6(h+k)^2})+k(\frac{t'}{3(h+k)}+\frac{k}{6(h+k)^2})] =$$
$$=\inf_{t>t'}[\frac{ht'+kt}{h+k}+\frac{k}{6(h+k)}]=\inf_{t>t'}\frac{ht'+kt}{h+k}+\frac{k}{6(h+k)}.$$

The infimum is equivalent to setting $t=0$ and $t'=0$, and the expression is thus maximized when $h=0$ for any $k>0$, with the optimum of $1/6$.

D.11 Proof of Lemma 5.16

First, let us derive $Q(q,t)$ and $U(q,x,t)$, where q is the probability that player with the higher type wins the good and $x(t)$ is the expected payment by players [Myerson, 1981].

$$Q(q,t)=\int_0^t q dT+\int_t^1(1-q)dT=t(2q-1)-q+1.$$

$$U(q,x,t)=\int_0^t(tq-k_1t-k_2T-K_1)dT+\int_t^1((1-q)t-k_3t-k_4T-K_2)dT=$$
$$=(2q-k_1-0.5k_2+k_3+0.5k_4-1)t^2+$$
$$+(1-q-K_1-k_3+K_2)t-(0.5k_4+K_2).$$

The first constraint that must be satisfied according to Myerson [1981] is if $s\le t$ then $Q(q,s)\le Q(q,t)$. This constraint is always satisfied in our design space by inspection of the form of $Q(q,t)$ above.

Individual rationality constraint requires that $U(q,x,0)\ge 0$, implying in our setting that $0.5k_4+K_2\le 0$. Since all design parameters are constrained to be non-negative, this implies that $k_4=K_2=0$, and, consequently, $U(q,x,0)=0$.

The version of the final constraint in Myerson [1981] in our setting

$$U(q,x,t)=\int_0^1 Q(q,s)ds=(q-0.5)t^2+(1-q)t$$

implies that $K_1 = k_3 = 0$ and $q - k_1 - 0.5k_2 - 0.5 = 0$, completing the proof.

D.12 Proof of Theorem 5.17

The expected revenue to the designer is

$$U_0(q, x) = \int_0^1 \int_0^1 (x_1(t, T) + x_2(t, T))dtdT$$

which by symmetry and Lemma 5.16 is equivalent to

$$U_0(q, x) = 2\int_0^1 \int_0^t (k_1 t + k_2 T)dTdt = \frac{2}{3}k_1 + \frac{1}{3}k_2.$$

Rewriting the constraint from Lemma 5.16 to be $k_1 + 0.5k_2 = q - 0.5$, it is clear that the revenue is maximal when $q = 1$. Now, if we let $k = k_1$ and $k_2 = 1 - 2k$, the expected revenue becomes $(2/3)k + (1/3)(1 - 2k) = 1/3$. Thus, we can set any $k_1 \in [0, 0.5]$ and $k_2 \in [0, 1]$, respecting the constraint, to achieve optimal revenue of 1/3.

D.13 Proof of Proposition 5.18

We will use the equilibrium bids of $s(t) = 0.72t - 0.73$ in this proof. First, let us derive the expected payment of an agent with type t, which we designate by $m(t)$. We can simplify our task by taking advantage of strict monotonicity of the equilibrium bid function in t.

$$\begin{aligned} m(t) = \int_0^t &(0.95s(t) + 0.84s(T) + 0.78)dT + \int_t^1 (0.73s(t) + 0.53)dT = \\ &= 0.95t(0.72t - 0.73) + 0.84(0.36t^2 - 0.73t) + \\ &\quad + 0.78t + 0.73(0.72t - 0.73)(1 - t) + 0.53(1 - t) = \\ &= 0.4604t^2 + 0.0018t - 0.0029. \end{aligned}$$

By symmetry, the expected revenue is twice the expectation of $m(t)$:

$$R = 2\int_0^1 m(t)dt = 2\int_0^1 (0.4604t^2 + 0.0018t - 0.0029)dt > 0.3.$$

To confirm individual rationality, we need to compute the expected value to an agent with type t from this auction, which we label $v(t)$:

$$v(t) = \int_0^t 0.96tdT + \int_t^1 0.04tdT = 0.92t^2 + 0.04t.$$

The expected utility to an agent with type t is its expected value less expected payment:

$$\text{EU}(t) = v(t) - m(t) = 0.4596t^2 + 0.0382t + 0.0029.$$

Clearly, this is always positive. Furthermore, the designer can charge each agent an additional participation fee of 0.0029 and maintain individual rationality. Since this uniform fee will not affect agents' incentives, the designer will gain an additional 0.0058 in revenue without compromising the individual rationality constraint.

D.14 Proof of Theorem 5.19

The intuition for the proof is straightforward. Suppose that the equilibrium bid function is strictly increasing and $q = 1$. Then, since the high bidder always gets the good, and the higher type is always the high bidder, the good always goes to the agent that values it more. Consequently, this design yields optimal welfare. The reverse argument works in the other case.

Formally, expected welfare is

$$pE_{t,T}[t \mid t > T] + (1-p)E_{t,T}[t \mid t < T] + 0.5E_{t,T}[t \mid t = T],$$

where p is the probability that the high type gets the good. Since the probability that types of both agents are equal is 0, the third term is 0. Furthermore, $E_{t,T}[t \mid t > T] = 2/3$, since this is just the first order statistic of the type distribution, and $)E_{t,T}[t \mid t < T] = 1/3$ since it is the second order statistic of the type distribution. Consequently, expected welfare is $(2/3)p + (1/3)(1-p)$. This is maximized when $p = 1$, and the maximal value is 2/3. Now, if bid function is increasing in t, then $q = p = 1$ ensures optimality. If bid function is decreasing in t, on the other hand, $q = (1-p) = 0$ ensures optimality.

D.15 Proof of Proposition 5.20

We will work with the symmetric equilibrium bid of $s(t) = 0.645t - 0.44$. Since we have already shown the optimality of this mechanism, we just need to confirm individual rationality and compute the revenue from this auction.

As before, we start with computing the payment of an agent with type t:

$$
\begin{aligned}
m(t) = &\int_0^t (0.88s(t) + 0.23s(T) + 0.28)dT + \\
&+ \int_t^1 (0.06s(t) + 0.32s(T))dT = \\
&= 0.88t(0.645t - 0.44) + 0.23(0.3225t^2 - 0.44t) + \\
&\quad + 0.28t + 0.06(0.645t - 0.44)(1-t) + \\
&\quad + 0.32(-0.3225t^2 + 0.44t - 0.1175) = \\
&= 0.499875t^2 - 0.0025t - 0.064.
\end{aligned}
$$

By symmetry, the expected revenue is twice the expectation of $m(t)$:

$$
R = 2\int_0^1 (0.499875t^2 - 0.0025t - 0.064)dt = 0.20275.
$$

The expected value of an agent, $v(t)$ is just t^2, since the high type always gets the good. Consequently, expected utility to an agent is

$$
\text{EU}(t) = v(t) - m(t) = 0.50012t^2 + 0.0025t + 0.064.
$$

Since this is always nonnegative when $t \in [0,1]$, ex interim individual rationality constraint holds. Note also that it will hold weakly if we charge each participant 0.064 for entering the auction. Thus, the designer could gain an additional 0.128 in revenue without affecting incentives, welfare optimality, and individual rationality.

D.16 Proof of Theorem 5.21

Since we are assuming symmetry and the equilibrium bid function is increasing in t, the objective is equivalent to

$$\inf_{t>T}[k_1 s(t) + k_2 s(T) + k_3 s(T) + k_4 s(t)] = \inf_{t>T}[k_1 mt + k_2 mT + k_3 mT + k_4 mt] =$$
$$= m \inf_{t>T}[(k_1 + k_4)t + (k_2 + k_3)T] = 0.$$

D.17 Proof of Proposition 5.22

We will use the symmetric equilibrium bid of (approximately) $s(t) = 0.43t - 0.51$.

First we establish the robust revenue properties of the design. By symmetry, the robust objective is equivalent to

$$\inf_{t>T}(s(t) + 0.34s(T) + 0.69) = \inf_{t>T}(0.43t + 0.1462T + 0.0066) = 0.0066.$$

The expected utility of type t is

$$\int_0^t (t - s(t) - 0.34s(T) - 0.69)dT = 0.4969t^2 - 0.0066t,$$

which attains a minimum at $t = 0.0066412$, with the minimum value of just above -0.000022.

D.18 Proof of Proposition 5.23

We will use the symmetric equilibrium bid of $s(t) = (7/9)t + 2/9$. The expected payment of type t is

$$m(t) = \int_0^t (\frac{7}{9}T + \frac{2}{9})dT = \frac{7}{18}t^2 + \frac{2}{9}t.$$

The expected revenue is then

$$R = 2\int_0^1 (\frac{7}{18}t^2 + \frac{2}{9}t)dt = \frac{13}{27}$$

which is approximately 0.48.

Since the high bidder always gets the good, $v(t) = t^2$. The expected utility of an agent with type t is then

$$eu = \frac{11}{18}t^2 - \frac{2}{9}t,$$

which attains its minimum when $t = 2/11$, with the minimum value of $-44/2178$ (just under -0.02). Thus, it is not individually rational. To fix the mechanism, the designer could afford each agent 0.021 for participation, reducing his revenue to 0.438.

D.19 Proof of Proposition 5.24

We will use the symmetric equilibrium bid of $s(t) = 1.613t - 0.234$. First, we compute expected payment of type t:

$$\begin{aligned} m(t) &= \int_0^t (0.98s(T) + 0.09)dT + \int_t^1 0.33s(t)dT \\ &= 0.98(0.8065t^2 - 0.234t) + 0.09t + 0.33(1.613t - 0.234)(1 - t) \\ &= 0.25808t^2 + 0.47019t - 0.07722. \end{aligned}$$

The expected revenue is then

$$R = 2\int_0^1 (0.25808t^2 + 0.47019t - 0.07722)dt = 0.4878.$$

Since the high bidder always gets the good, $v(t) = t^2$, and the expected utility of type t is then

$$\text{EU}(t) = 0.74192t^2 - 0.47019t + 0.07722.$$

The function $\text{EU}(t)$ is always positive, and the minimum gain for any agent type is 0.00273. Thus, the designer could charge an entry fee of 0.0027 and gain an additional 0.0054 in revenue, for a total of 0.4932.

D.20 Proof of Proposition 5.25

In this case, we will use the symmetric equilibrium bid of $s(t) = 0.595t - 0.2$. The expected payment of type t is

$$\begin{aligned} m(t) = &\int_0^t (s(t) + 0.33s(T) + 0.22)dT + \int_t^1 (0.22s(t) + 0.12s(T))dT \\ &= 0.595t^2 - 0.2t + 0.33(0.2975t^2 - 0.2t) + 0.22t + \\ &\quad + 0.22(0.595t - 0.2)(1 - t) + 0.12(-0.2975t^2 + 0.2t + 0.0975) \\ &= 0.526575t^2 + 0.1529t - 0.0323. \end{aligned}$$

The expected revenue is then

$$R = 2\int_0^1 (0.526575t^2 + 0.1529t - 0.0323) \approx 0.44.$$

Since $q = 1$, $v(t) = t^2$, and, therefore

$$\text{EU}(t) = 0.473425t^2 - 0.1529t + 0.0323,$$

which we can verify is always positive. Thus, this design is ex interim individually rational. Since its minimum value is slightly above 0.0199, we can bill this amount to each agent for participating in the auction without affecting incentives or ex interim individual rationality. This adjustment will give the designer 0.0398 of additional revenue, for a total of about 0.4798.

D.21 Proof of Proposition 5.26

We use the symmetric equilibrium bid function $s(t) = -0.22t - 0.175$ here. Since the bids are strictly decreasing in types, the expected value of type t is

$$v(t) = \int_0^t 0.63t\,dT + \int_t^1 0.37t\,dT = 0.26t^2 + 0.37t.$$

By symmetry, the expected welfare is then

$$W = 2\int_0^1 v(t)\,dt = 0.543.$$

The expected payment of type t is

$$\begin{aligned}
m(t) = &\int_0^t (0.29s(t) + 0.67s(T) + 0.48)dT + \int_t^1 (0.8s(t) + s(T) + 0.49)dT \\
&= -0.29(0.22t + 0.175)t - 0.67(0.11t^2 + 0.175t) + \\
&\quad + 0.48t + 0.8(-0.22t - 0.175)(1-t) - 0.11t^2 + 0.175t - 0.285 + 0.49(1-t) \\
&= 0.1485t^2 - 0.004t + 0.065.
\end{aligned}$$

Thus, we can compute the expected revenue:

$$R = \int_0^1 (0.1485t^2 - 0.004t + 0.065)dt = 0.225.$$

The expected utility of type t is

$$\mathrm{EU}(t) = v(t) - m(t) = 0.1115t^2 + 0.374t - 0.065,$$

which attains its minimum at the lower type boundary of 0, with the minimum value of -0.065, and is negative over the range of types $[0, 0.17]$. Thus, the designer could make the mechanism completely ex interim individually rational at a loss of an additional 0.013 in revenue by offering each agent a participation gift of 0.065. With this gift, the revenue would fall to 0.095.

D.22 Proof of Proposition 5.27

We use the symmetric equilibrium bid function $s(t) = 0.935t - 0.18$ here.

The expected payment of an agent with type t is

$$\begin{aligned} m(t) &= \int_0^t (0.51s(t) + s(T) + 0.09)dT + \int_t^1 (0.34s(t) + 0.26s(T))dT \\ &= 0.51(0.935t^2 - 0.18t) + 0.4675t^2 - 0.18t + 0.09t + \\ &\quad + 0.34(0.935t - 0.18)(1-t) + 0.26(-0.4675t^2 + 0.18t + 0.2875) \\ &= 0.5049t^2 + 0.2441t + 0.01355. \end{aligned}$$

The expected revenue is thus

$$R = 2\int_0^1 (0.5049t^2 + 0.2441t + 0.01355)dt = 0.6078.$$

The expected utility of an agent with type t is

$$\text{EU}(t) = v(t) - m(t) = 0.4951t^2 - 0.2441t - 0.01355,$$

which is negative for a fairly broad range of types (although always above the tolerance level that we set). Type $t^* = 0.24652$ fairs the worst, incurring a loss of nearly 0.044. However, by compensating both agents this amount, we ensure ex interim individual rationality without affecting incentives. As a result, the designer will lose 0.088 in expected revenue, which will fall to 0.5198.

D.23 Proof of Proposition 5.28

By symmetry, the objective value is equivalent to

$$\inf_{t>T} s(T) = \inf_{t>T} (\frac{7}{9}T + \frac{2}{9}) = 2/9.$$

The rest follows by Proposition 5.23.

D.24 Proof of Proposition 5.29

The objective value is equivalent to

$$\inf_{t>T}(s(t) + 0.71s(T) + 0.14 + 0.09s(t)) =$$
$$= \inf_{t>T}(0.3t - 0.045 + 0.71(0.3T - 0.045) + 0.14 + 0.09(0.3t - 0.045)) = 0.059.$$

The expected utility of an agent is

$$\begin{aligned} eu(t) = &\int_0^t (0.86t - 0.3t + 0.045 - 0.71(0.3T - 0.045) - 0.14)dT + \\ &+ \int_t^1 (0.14t - 0.09(0.3T - 0.045))dT = \\ &= 0.56t^2 + 0.07695t - 0.1065t^2 - 0.14t + 0.14t - 0.14t^2 + 0.00405 - \\ &- 0.00405t - 0.0135(1 - t^2) = \\ &= 0.327t^2 + 0.0729t - 0.0135. \end{aligned}$$

which attains a minimum value of -0.0135. Thus, the participation award of 0.0135 to each agent is necessary to make this design individually rational, with the resulting robust revenue of 0.032.

APPENDIX E

Proofs for Chapter 7

E.1 Proof of Theorem 7.9

First, we will need the following fact:

Claim E.1 *Given a function* $f_i(x)$ *and a set* X*,* $|\max_{x\in X} f_1(x) - \max_{x\in X} f_2(x)| \leq \max_{x\in X} |f_1(x) - f_2(x)|$.

To prove this claim, observe that

$$
|\max_{x\in X} f_1(x) - \max_{x\in X} f_2(x)| =
\begin{cases}
\max_x f_1(x) - \max_x f_2(x) & \text{if } \max_x f_1(x) \geq \max_x f_2(x) \\
\max_x f_2(x) - \max_x f_1(x) & \text{if } \max_x f_2(x) \geq \max_x f_1(x)
\end{cases}
$$

In the first case,

$$
\max_{x\in X} f_1(x) - \max_{x\in X} f_2(x) \leq \max_{x\in X}(f_1(x) - f_2(x)) \leq
\leq \max_{x\in X} |f_1(x) - f_2(x)|.
$$

Similarly, in the second case,

$$
\max_{x\in X} f_2(x) - \max_{x\in X} f_1(x) \leq \max_{x\in X}(f_2(x) - f_1(x)) \leq
\leq \max_{x\in X} |f_2(x) - f_1(x)| = \max_{x\in X} |f_1(x) - f_2(x)|.
$$

Thus, the claim holds.

By the Strong Law of Large Numbers, $u_{n,i}(a) \to u_i(a)$ a.s. for all $i \in I, a \in A$. That is,

$$\Pr\{\lim_{n\to\infty} u_{n,i}(a) = u_i(a)\} = 1,$$

or, equivalently [2004], for any $\alpha > 0$ and $\delta > 0$, there is $M(i,a) > 0$ such that

$$\Pr\{\sup_{n \geq M(i,a)} |u_{n,i}(a) - u_i(a)| < \frac{\delta}{2|A|}\} \geq 1 - \alpha.$$

By taking $M = \max_{i\in I} \max_{a\in A} M(i,a)$, we have

$$\Pr\{\max_{i\in I} \max_{a\in A} \sup_{n\geq M} |u_{n,i}(a) - u_i(a)| < \frac{\delta}{2|A|}\} \geq 1 - \alpha.$$

Thus, by the claim, for any $n \geq M$,

$$\begin{aligned}
&\sup_{n\geq M} |\epsilon_n(s) - \epsilon(s)| \leq \\
&\max_{i\in I} \max_{a_i\in A_i} \sup_{n\geq M} |u_{n,i}(a_i, s_{-i}) - u_i(a_i, s_{-i})| + \\
&\qquad + \sup_{n\geq M} \max_{i\in I} |u_{n,i}(s) - u_i(s)| \leq \\
&\max_{i\in I} \max_{a_i\in A_i} \sum_{b\in A_{-i}} \sup_{n\geq M} |u_{n,i}(a_i, b) - u_i(a_i, b)| s_{-i}(b) + \\
&\qquad + \max_{i\in I} \sum_{b\in A} \sup_{n\geq M} |u_{n,i}(b) - u_i(b)| s(b) \leq \\
&\max_{i\in I} \max_{a_i\in A_i} \sum_{b\in A_{-i}} \sup_{n\geq M} |u_{n,i}(a_i, b) - u_i(a_i, b)| + \\
&\qquad + \max_{i\in I} \sum_{b\in A} \sup_{n\geq M} |u_{n,i}(b) - u_i(b)| < \\
&\max_{i\in I} \max_{a_i\in A_i} \sum_{b\in A_{-i}} (\frac{\delta}{2|A|}) + \max_{i\in I} \sum_{b\in A} (\frac{\delta}{2|A|}) \leq \delta
\end{aligned}$$

with probability at least $1 - \alpha$. Note that since $s_{-i}(a)$ and $s(a)$ are bounded between 0 and 1, we were able to drop them from the expressions above to obtain a bound that will be valid independent of the particular choice of s. Furthermore, since the above result can be obtained for an arbitrary $\alpha > 0$ and $\delta > 0$, we have $\Pr\{\lim_{n\to\infty} \epsilon_n(s) = \epsilon(s)\} = 1$ uniformly on S.

E.2 Proof of Lemma 7.11

We prove the result using uniform continuity of $u_i(s)$ and preservation of continuity under maximum.

Claim E.2 *A function $f : \mathbb{R}^k \to \mathbb{R}$ defined by $f(t) = \sum_{i=1}^k z_i t_i$, where z_i are constants in $\mathbb{R}$, is uniformly continuous in t.*

The claim follows because $|f(t) - f(t')| = |\sum_{i=1}^k z_i(t_i - t'_i)| \le \sum_{i=1}^k |z_i||t_i - t'_i|$. An immediate result of this for our purposes is that $u_i(s)$ is uniformly continuous in s and $u_i(a_i, s_{-i})$ is uniformly continuous in s_{-i}.

Claim E.3 *Let $f(a, b)$ be uniformly continuous in $b \in B$ for every $a \in A$, with $|A| < \infty$. Then $V(b) = \max_{a\in A} f(a, b)$ is uniformly continuous in b.*

To show this, take $\gamma > 0$ and let $b, b' \in B$ such that $\|b - b'\| < \delta(a) \Rightarrow |f(a, b) - f(a, b')| < \gamma$. Now take $\delta = \min_{a\in A} \delta(a)$. Then, whenever $\|b - b'\| < \delta$,

$$|V(b) - V(b')| = |\max_{a\in A} f(a, b) - \max_{a\in A} f(a, b')| \le$$
$$\max_{a\in A} |f(a, b) - f(a, b')| < \gamma.$$

Now, recall that $\epsilon(s) = \max_i[\max_{a_i\in A_i} u_i(a_i, s_{-i}) - u_i(s)]$. By the claims E.2 and E.3, $\max_{a_i\in A_i} u_i(a_i, s_{-i})$ is uniformly continuous in s_{-i} and $u_i(s)$ is uniformly continuous in s. Since the difference of two uniformly continuous functions is uniformly continuous, and since this continuity is preserved under maximum by our second claim, we have the desired result.

E.3 Proof of Theorem 7.12

Choose $\delta > 0$. First, we need to ascertain that the following claim holds:

Claim E.4 $\bar{\epsilon} = \min_{s \in S \setminus \mathcal{N}_\delta} \epsilon(s)$ *exists and* $\bar{\epsilon} > 0$.

Since $\mathcal{N}_\delta$ is an open subset of compact S, it follows that $S \setminus \mathcal{N}_\delta$ is compact. As we had also proved in Lemma 7.11 that $\epsilon(s)$ is continuous, existence follows from the Weierstrass theorem. That $\bar{\epsilon} > 0$ is clear since $\epsilon(s) = 0$ if and only if s is a Nash equilibrium of Γ.

Now, by Theorem 7.9, for any $\alpha > 0$ there is M such that

$$\Pr\{\sup_{n \geq M} \sup_{s \in S} |\epsilon_n(s) - \epsilon(s)| < \bar{\epsilon}\} \geq 1 - \alpha.$$

Consequently, for any $\delta > 0$,

$$\Pr\{\sup_{n \geq M} h(\mathcal{N}_n, \mathcal{N}_\delta) < \delta\} \geq \Pr\{\forall n \geq M \; \mathcal{N}_n \subset \mathcal{N}_\delta\} \geq$$
$$\Pr\{\sup_{n \geq M} \sup_{s \in \mathcal{N}} \epsilon(s) < \bar{\epsilon}\} \geq$$
$$\Pr\{\sup_{n \geq M} \sup_{s \in S} |\epsilon_n(s) - \epsilon(s)| < \bar{\epsilon}\} \geq 1 - \alpha.$$

Since this holds for an arbitrary $\alpha > 0$ and $\delta > 0$, the desired result follows.

E.4 Proof of Lemma 7.13

I use Claim E.1 to provide a bound on $|\epsilon(r) - \hat{\epsilon}(r)|$:

$$|\epsilon(r) - \hat{\epsilon}(r)| = |\max_{i \in I} \max_{a_i \in A_i} [\hat{u}(a_i, r_{-i}) - \hat{u}(r)] - \max_{i \in I} \max_{a_i \in A_i} [u_i(a_i, r_{-i}) - u_i(r)]| \leq$$
$$\max_{i \in I} \max_{a_i \in A_i} |[\hat{u}(a_i, r_{-i}) - \hat{u}(r)] - [u_i(a_i, r_{-i}) - u_i(r)]| =$$
$$\max_{i \in I} \max_{a_i \in A_i} |[\hat{u}(a_i, r_{-i}) - u_i(a_i, r_{-i})] + [u_i(r) - \hat{u}(r)]| \leq$$
$$\max_{i \in I} \max_{a_i \in A_i} |\hat{u}(a_i, r_{-i}) - u_i(a_i, r_{-i})| + |\hat{u}(r) - u_i(r)|.$$

The result now follows from the assumption that $|u_i(r) - \hat{u}_i(r)| \leq \delta$ for strategy profiles r and $\forall (a_i, r_{-i}) : a_i \in A_i$.

E.5 Proof of Theorem 7.14

Let $\epsilon_n(r)$ denote the empirical regret with respect to the empirical game with at least n payoff samples taken for every strategy profile, and let $\epsilon_i(r)$ denote the player i's contribution to regret (that is, the benefit player i can gain by deviating from r.

$$\begin{aligned}
&\Pr\{|\epsilon_n(r) - \epsilon(r)| \geq 2\gamma\} = \Pr\{|\max_{i\in I}\epsilon_{n,i}(r) - \max_{i\in I}\epsilon_i(r)| \geq 2\gamma\} \leq \\
&\quad \leq \Pr\{\max_{i\in I}|\epsilon_{n,i}(r) - \epsilon_i(r)| \geq 2\gamma\} \leq \\
&\quad \leq \Pr\{\bigcup_{i\in I}|\epsilon_{n,i}(r) - \epsilon_i(r)| \geq 2\gamma\} \leq \\
&\quad \leq \sum_{i\in I}\Pr\{|\epsilon_{n,i}(r) - \epsilon_i(r)| \geq 2\gamma\} \leq \\
&\quad \leq \sum_{i\in I}\Pr\{\bigcup_{q\in r\cup(A_i,r_{-i})}[|u_{n,i}(q) - u_i(q)| \geq \gamma]\} \leq \\
&\quad \leq \sum_{i\in I}(K+1)\delta \leq m(K+1)\delta.
\end{aligned}$$

E.6 Proof of Lemma 7.17

$$\begin{aligned}
&\Pr\{|u(s) - \hat{u}(s)| \geq \epsilon\} = \\
&\quad = \Pr\{|u(s) - \hat{u}(s)| \geq \epsilon \,|\, |u(a) - \hat{u}(a)| \leq \epsilon \;\forall a \in A\} \times \\
&\qquad \times \Pr\{|u(a) - \hat{u}(a)| \leq \epsilon \;\forall a \in A\} + \\
&\qquad + \Pr\{|u(s) - \hat{u}(s)| \geq \epsilon \,|\, \exists a \in A : |u(a) - \hat{u}(a)| > \epsilon\} \times \\
&\qquad \times \Pr\{\exists a \in A : |u(a) - \hat{u}(a)| > \epsilon\} \leq \delta.
\end{aligned}$$

E.7 Proof of Proposition 7.22

$$\begin{aligned}
&\Pr\left(\max_{i\in I}\max_{b\in R_i\setminus r_i} u_i(b,r_{-i})-u_i(r)\le\epsilon\right)= \\
&=\prod_{i\in I}\Pr(\max_{b\in R_i\setminus r_i} u_i(b,r_{-i})-u_i(r)\le\epsilon)= \\
&=\prod_{i\in I}E_{u_i(r)}\Pr(\max_{b\in R_i\setminus r_i} u_i(b,r_{-i})-u_i(r)\le\epsilon|u_i(r))= \\
&=\prod_{i\in I}E_{u_i(r)}\left[\prod_{b\in R_i\setminus r_i}\Pr(u_i(b,r_{-i})-u_i(r)\le\epsilon|u_i(r))\right]= \\
&=\prod_{i\in I}\int_{\mathbb{R}}\prod_{b\in R_i}\Pr(u_i(b,r_{-i})\le u+\epsilon)f_{u_i(r)}(u)du.
\end{aligned}$$

E.8 Proof of Proposition 7.23

Define $W_i(b)=u_i(b,s_{-i})$, $b\in A_i$, and let $W_i^*=u_i(s)$. Since I assumed that $u_i(\cdot)$ is a vNM utility function, we have

$$W_i(b)=\sum_{c\in A_{-i}}u_i(b,c)s_{-i}(c)=\sum_{c\in A_{-i}}\bar{u}_i(b,c)s_{-i}(c)-\sum_{c\in A_{-i}}\frac{\sigma_i(b,c)s_{-i}(c)Z_i(b,c)}{\sqrt{n_i(b,c)}}.$$

Since $Z_i(\cdot)$ are i.i.d. with distribution $N(0,1)$,

$$\left(\sum_{c\in A_{-i}}\frac{\sigma_i(b,c)s_{-i}(c)Z_i(b,c)}{\sqrt{n_i(b,c)}}\right)\sim N\left(0,\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}\right).$$

Then,

$$\left(\frac{\sum_{c\in A_{-i}}\left(\sigma_i(b,c)/\sqrt{n_i(b,c)}\right)s_{-i}(c)Z(b,c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right)\sim N(0,1).$$

As a result, we can derive

$$\Pr(W_i(b)\le u+\epsilon)=1-\Phi\left[\frac{\sum_{r\in Q_{-i}}\bar{u}_i(q,r)s_{-i}(r)-u-\epsilon}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right]$$

Similarly we can show that

$$W_i^* \sim N(\sum_{a \in A} \bar{u}_i(a)s(a), \sum_{a \in A} \frac{\sigma^2(a)s^2(a)}{n_i(a)}).$$

We now get the desired result by applying Proposition 7.22.

E.9 Proof of Proposition 7.24

$$\begin{aligned}
&\Pr\left(\max_{i \in I} \sup_{t \in R_i} u_i(t, r_{-i}) - u_i(r) \leq \epsilon\right) \geq \\
&\geq \Pr\left(\max_{i \in I} \max_{t \in R_{k,i}} u_i(t, r_{-i}) - u_i(r) \leq \epsilon - Bd\right) = \\
&= \prod_{i \in I} \int_{\mathbb{R}} \prod_{t \in R_{k,i}} \Pr(u_i(t, r_{-i}) \leq u + \epsilon - Bd) f_{u_i(r)}(u) du.
\end{aligned}$$

E.10 Proof of Lemma 7.25

First, we prove the following claim.

Claim E.5 *Let s be a mixed strategy profile. Then $\Pr\{W_i(b, s_{-i}) \leq d\}$ is continuous in s.*

Proof. Fix $s \in S$ and let s' be such that $\|s - s'\|_\infty \leq \epsilon$. Note that our choice of max-norm here is not significant since all norms on finite-dimensional vector spaces are equivalent.

$$\begin{aligned}
&|\Pr\{W_i(b, s_{-i}) \leq d\} - \Pr\{W_i(b, s'_{-i}) \leq d\}| \leq \\
&= |\Phi\left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b, c)s_{-i}(c) - d}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right] - \Phi\left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b, c)s'_{-i}(c) - d}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right]|.
\end{aligned}$$

Since $\Phi(\cdot)$ is continuous, we need only bound the absolute difference in the arguments to arrive at the desired result. To that end,

$$\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|\leq$$

$$\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s_{-i}(c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right|+$$

$$+\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|.$$

Let us now isolate the two parts of the sum. The left sum becomes

$$\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s_{-i}(c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right|\leq$$

$$\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right|\left|s_{-i}(c)-s'_{-i}(c)\right|\leq\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}\right|\epsilon.$$

This suffices so far, since s is fixed. Now we turn to the second summand above.

$$\left|\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{\sum_{c\in A_{-i}}\bar{u}_i(b,c)s'_{-i}(c)-d}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|\leq$$

$$\left|\sum_{c\in A_{-i}}\bar{u}_i(b,c)-d\right|\left|\frac{1}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{1}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|.$$

Now, we are left with the task of bounding the difference between the fractions on the right. Define $\sigma_L^2=\min_{c\in A_{-i}}\sigma_i^2(b,c)$; $\sigma_H^2=\max_{c\in A_{-i}}\sigma_i^2(b,c)$; $n_L=\min_{c\in A_{-i}}n_i(b,c)$;

and $n_H = \max_{c\in A_{-i}} n_i(b,c)$.

$$\left|\frac{1}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}-\frac{1}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right| =$$
$$\left|\frac{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}-\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|$$

By interpreting the root-sums of squares as norms and using the triangle inequality we obtain

$$\left|\frac{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}-\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right| \le$$
$$\left|\frac{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)}{n_i(b,c)}[s'_{-i}(c)-s_{-i}(c)]^2}}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right|.$$

By the equivalence of vector norms, there exists a constant K such that

$$\left|\frac{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)}{n_i(b,c)}[s'_{-i}(c)-s_{-i}(c)]^2}}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}}\right| \le$$
$$\frac{\epsilon\frac{\sigma_H^2}{n_L}K}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s'^2_{-i}(c)}{n_i(b,c)}}} \le \frac{\epsilon\frac{\sigma_H^2}{n_L}K}{\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}(c)s'_{-i}(c)}{n_i(b,c)}}},$$

where the last step was a consequence of the Cauchy-Schwartz inequality. Finally, we let $c_H = \arg\max_{c\in A_{-i}} s_{-i}$, noting that $s_{-i}(c_H) > 0$. For ϵ small enough, then, $s'_{-i}(c_H) \ge s_{-i}(c_H) - \epsilon > 0$, which will be the case as s' approaches s in max-norm. Consequently,

$$\sqrt{\sum_{c\in A_{-i}}\frac{\sigma_i^2(b,c)s_{-i}(c)s'_{-i}(c)}{n_i(b,c)}} \ge \sqrt{\frac{\sigma_i^2(b,c_H)s_{-i}(c_H)s'_{-i}(c_H)}{n_i(b,c_H)}} \ge$$
$$\sqrt{\frac{\sigma_i^2(b,c_H)s_{-i}(c_H)(s_{-i}(c_H)-\epsilon)}{n_i(b,c_H)}} > 0.$$

Thus, the denominator is a strictly positive constant for a given $s \in S$. This, therefore, gives us the desired result, since as ϵ decreases to 0, so will the difference in arguments to Φ, and by continuity of Φ, so will the difference in probabilities. □

We now use this result to prove the lemma. First, we note that the continuity of the density function in the expression for $\Pr\{\epsilon(r) = 0\}$ can be established in essentially the same way as the continuity of $\Pr\{W_i(b, s_{-i}) \leq d\}$, with the lone substantial exception that we in that case rely on the continuity of the normal density function rather than the normal distribution function. The remaining argument proceeds identically. Furthermore, the fact that a finite product of continuous functions is continuous produces an integrand (trivially integrable) which is continuous in s. The integral is then also a continuous function, and we complete the continuity argument by again noting that a finite product of continuous functions is continuous.

E.11 Proof of Theorem 7.29

First we demonstrate a few basic general facts about continous functions that are strictly increasing.

Claim E.6 *Let $f : \mathbb{R} \to \mathbb{R}$ be a continuous strictly increasing function and set $B \subset \mathbb{R}$ be open. Then $f(B)$ is open. Furthermore, if B is non-empty, so is $f(B)$.*

Proof. First, let's suppose that B is an open interval $I = (a, b)$. Since f is strictly increasing, we can consider the interval in its range $(f(a), f(b))$. Let $z \in (f(a), f(b))$. By the Intermediate Value Theorem, there is $c \in [a, b]$ such that $f(c) = z$, and since f is strictly increasing, $c \in (a, b)$. Thus, $(f(a), f(b)) \subset f(B)$. Note that for every $c \in (a, b)$, $f(a) \leq f(c) \leq f(b)$. Thus, $f(B) \subset (f(a), f(b))$ and, consequently, $f(B) = (f(a), f(b))$. Consequently, the image of an open interval I is an open interval. Trivially, since f is a function, $f(I)$ must be non-empty as long as $a < b$ (i.e., as long as I is non-empty).

Now consider an arbitrary open set B. Since $B \subset \mathbb{R}$, it is a countable union of disjoint open intervals. The image of each of these is an open interval, and these images are disjoint since f is strictly increasing. Thus, $f(B)$ must be a union of open intervals and, consequently, open. Furthermore, by above, as long as at least one of the open intervals is non-empty, so is $f(B)$. □

Claim E.7 *Let $f : \mathbb{R} \to \mathbb{R}$ be a continuous strictly increasing function and $B \subset \mathbb{R}$ be a set such that $\lambda(B) > 0$, where λ is the Lebesgue measure on $\mathbb{R}$. Then $\lambda(f(B)) > 0$.*

Proof. Since $\lambda(B) > 0$, $\lambda(int(B)) > 0$ and thus $int(B)$ must be non-empty. By the previous claim, $f(int(B))$ is a non-empty open subset of $\mathbb{R}$ and thus $\lambda(f(int(B))) > 0$. Since f is strictly increasing, $f(int(B)) \subset f(B)$ and therefore $\lambda(f(B)) > 0$. □

Claim E.8 *Let $f_i : \mathbb{R} \to \mathbb{R}$ for $i = 1, ldots, m$ be non-negative, continuous, and strictly increasing and let $f : \mathbb{R}^m \to \mathbb{R}^m$ be defined by $f(x) = (f_1(x), \ldots, f_m(x))$. Let $B_i \subset \mathbb{R}, i = 1, \ldots, m$, such that $\lambda(B_i) > 0\ \forall i$, where λ is the $\mathbb{R}$-Lebesgue measure. Define a measure m on $\mathbb{R}^m$ to be $m = \lambda \circ f$ for all $B \subset \mathbb{R}^m, B = B_1 \times \cdots \times B_m, B_i \subset \mathbb{R}$ and $m = \lambda$ for all other $B \subset \mathbb{R}^m$. Then $m \lessdot \lambda$ (i.e., m is absolutely continuous w.r.t. λ).*

Proof. The conclusion is trivial when B is not a product set. Suppose $B = B_1 \times \cdots \times B_m, B_i \subset \mathbb{R}$ and suppose $\lambda(B) > 0$. Then $\lambda_i(B_i) > 0$ for all i and, by the above claim $\lambda_i(f_i(B_i)) > 0$ for all i. Since λ on $\mathbb{R}^m$ is the product measure, $\lambda(f(B)) > 0$. □

Claim E.9 *Let $B \subset \mathbb{R}^m$ with $B = B_1 \times \cdots \times B_m$ and $\lambda(B) > 0$. Let $X = \{x \in B | \prod_{i=1}^m x_i = c\}$. Then $\lambda(X) = 0$, where λ is the $\mathbb{R}^m$-Lebesgue measure.*

Proof. Since $x_1 = c/\prod_{i=2}^m x_i$, X is an $m-1$-dimensional subset of B and thus $\lambda(X) = 0$. □

Now let us define $\bar{u}_i(s) = \sum_{a \in A} \bar{u}_i(a)s_i(a)$. Take a set of game payoffs of positive Normal measure. Since Normal distribution is absolutely continuous, this set also has positive Lebesgue product measure. Then for every $s \in S$ the set $U(s)$ of corresponding $\bar{u}_i(s)$ must have positive $\mathbb{R}$-Lebesgue measure. Since all payoffs are sampled independently, we can wlog fix the payoffs for all the deviations from s for the following argument.

Let

$$K_i(u) = \prod_{b \in A_i} \left[1 - \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c) s_{-i}(c) - u}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c) s_{-i}^2(c)}{n_i(b,c)}}} \right] \right]$$

and note that $K_i(u)$ is strictly increasing in u since Φ is a standard normal distribution and is strictly decreasing in u. Let $f_i(u, \mu)$ be a normal density function with a fixed variance and mean μ.

Claim E.10 *The function $\int_{\mathbb{R}} K_i(u) f_i(u, \mu) du$ is strictly increasing in μ.*

Proof. Let $m > m'$ and let $g(u) = f_i(u, m) - f_i(u, m')$. Then

$$\int_{\mathbb{R}} [K_i(u) f_i(u, m) - K_i(u) f_i(u, m')] du = \int_{\mathbb{R}} K_i(u) [f_i(u, m) - f_i(u, m')] du \geq$$
$$\geq \int_{-\infty}^{m'} K_i(u) g(u) du + \int_{m'}^{m} K_i(u) g(u) du + \int_{m}^{\infty} K_i(u) g(u) du.$$

Since $K_i(u)$ is strictly increasing in u,

$$\int_{m'}^{m} K_i(u) g(u) du > \int_{m'}^{m} K_i(m') g(u) du = K_i(m') \int_{m'}^{m} g(u) du = 0,$$

where the last equality follows from the symmetry of Normal density around the mean. Similarly,

$$\int_{-\infty}^{m'} K_i(u) g(u) du + \int_{m}^{\infty} K_i(u) g(u) du >$$
$$> \int_{-infty}^{m'} K_i(m') g(u) du + \int_{-infty}^{m} K_i(m) g(u) du =$$
$$= K_i(m') \int_{-infty}^{m} g(u) du + K_i(m) \int_{m'}^{\infty} g(u) du > 0,$$

where the last inequality follows from the symmetry of Normal and the fact that $K_i(u)$ is strictly increasing. It then follows that $\int_{\mathbb{R}} [K_i(u) f_i(u, m) - K_i(u) f_i(u, m')] du > 0$ and, thus, $\int_{\mathbb{R}} K_i(u) f_i(u, \mu) du$ is strictly increasing in μ. $\Box$

Now, the probability that two profiles s, s' yield $P(s) = P(s')$ is $E_{P(s')}[\Pr\{P(s) = c\}]$. Since $\Pr\{P(s) \in B\}$ is absolutely continuous with respect to the $\mathbb{R}^m$-Lebesgue measure by the above claims, $\Pr\{P(s) = c\} = 0$ and thus the probability that $P(s) = P(s')$ is 0, completing the proof.

E.12 Proof of Lemma 7.32

Claim E.11 *Suppose r is a Nash equilibrium. Then* $\Pr_n(\epsilon(r) = 0) \rightarrow \Pr(\epsilon(r) = 0)$.

Proof. For simplicity, we will assume (without loss of generality) that every player and profile has the same variance σ^2. Let $f_i(u)$ be the density function of the normal distribution $N(\bar{u}_i(r), \sigma^2(r))$, where $\bar{u}_i(r) = \sum_{a\in A} \bar{u}_i(a)r(a)$ and $\sigma^2(r) = (\sigma^2/n)\sum_{a\in A} r(a)^2$. Now, fix $\epsilon > 0$ and $\alpha > 0$ (and assume both to be below 1). By strong law of large numbers, payoffs converge almost surely for every player and pure strategy profile, and, consequently, for every mixture. Thus, for any $\delta > 0$ and $\gamma > 0$, there is N_1 finite such that $\Pr\{\sup_{n\geq N_1} |\bar{u}_{i,n}(r) - u_i(r)| < \delta\} \geq 1-\gamma$. We also have that

$$\sup_{n\geq N_1}[1 - \prod_{i\in I}(\int \prod_{b\in A_i\backslash r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u)du)] \leq$$

$$\leq \sup_{n\geq N_1}[1 - \prod_{i\in I}(\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b\in A_i\backslash r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u)du)].$$

Also, uniformly on S for any $\delta' > 0$ there is N_2 finite such that $\Pr\{\sup_{n\geq N_2} |\bar{u}_{i,n}(b, r_{-i}) - u_i(b, r_{-i})\|| < \delta'\} \geq 1 - \alpha$. Since by assumption of no ties $u_i(r) > u_i(b, r_{-i})$ for any $b \neq r_i$, there are $\delta > 0, \delta' > 0$ such that $u_i(r) - \delta' - \delta > u_i(b, r_{-i})$. Recall that $\Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} = 1 - \Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b, r_{-i}) - u))$. Since $u \geq u_i(r) - \delta$ and $\Phi(\cdot)$ is monotone increasing, $\Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b, r_{-i}) - u)) \leq \Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b, r_{-i}) - \bar{u}_{i,n}(r) + \delta + \delta'))$. Let $z_n = \frac{\bar{u}_{i,n}(b,r_{-i}) - \bar{u}_{i,n}(r)+\delta+\delta'}{\sigma^2}$ and note that $z = \sup_{n\geq \max\{N_1,N_2\}} z_n < 0$. Thus, by monotonicity of $\Phi(\cdot)$, for any $\gamma_i'' > 0$ there is N_3 finite such that $\sup_{n\geq \max\{N_1,N_2,N_3\}} \Phi(\sqrt{n}z_n) \leq \Phi(\sqrt{N_3}z) < \gamma_i''$. Let $N = \max\{N_1, N_2, N_3\}$, $\gamma = (0.5(1-(1-\epsilon)^{1/|I|})$, and $\gamma_i'' =$

$1 - [\frac{(1-\epsilon)}{0.5(1+(1-\epsilon)^{1/|I|}}]^{1/|A_i|}$. Then

$$\sup_{n \geq N}[1 - \prod_{i \in I}(\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b \in A_i \backslash r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du)] \leq$$
$$\leq \sup_{n \geq N}[1 - \prod_{i \in I}((1-\epsilon)^{1/|I|})] \leq$$
$$\leq 1 - 1 + \epsilon = \epsilon$$

with probability at least $1 - \alpha$. $\square$

Claim E.12 *Suppose r is not a Nash equilibrium. Then* $\Pr_n(\epsilon(r) = 0) \to \Pr(\epsilon(r) = 0)$.

Proof. Let $f_i(u)$ be as before and select $\epsilon > 0$ and $\alpha > 0$. We want to show that there is N finite such that $\Pr\{\sup_{n \geq N} \Pr\{\epsilon(r) = 0 | \Gamma_n\} < \epsilon\} \geq 1 - \alpha$. By the law of large numbers, for any $\delta > 0$ and $\gamma > 0$ there is N_1 finite such that $\Pr\{\sup_{n \geq N_1} |\bar{u}_{i,n}(r) - u(r)| < \delta\} \geq 1 - \gamma$. Consequently,

$$\sup_{n \geq N_1}(\prod_{i \in I}[\int \prod_{b \in A_i \backslash r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du]) \leq$$
$$\leq \sup_{n \geq N_1}(\prod_{i \in I}[\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b \in A_i \backslash r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du + \gamma]) \leq$$
$$\leq \sup_{n \geq N_1}[\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du + \gamma],$$

where the last inequality takes advantage of the fact that a product of probabilities is no greater than each taken separately. We thus select an arbitrary player i which has b_i, such that $u_i(b_i, r_{-i}) > u_i(r)$, which is possible by the no-ties assumption and the fact that r is not a Nash equilibrium. Applying the law of large numbers again, we have that for every $\delta' > 0$, there is N_2 finite such that $\Pr\{\sup_{n \geq N_2} |\bar{u}_{i,n}(b_i, r_{-i}) - u_i(r)| < \delta'\} \geq 1 - \alpha$. Again, by the no-ties assumption, we have $\delta > 0, \delta' > 0$ such that $u_i(b_i, r_{-i}) - u_i(r) - \delta - \delta' > 0$. Thus, with probability at least $1 - \alpha$, and letting

$\bar{N} = \max\{N_1, N_2\}$, we have

$$\sup_{n \geq N}[\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \Pr\{\bar{u}_{i,n}(b_i, r_{-i}) - u < 0\} f_i(u) du + \gamma] \leq$$
$$\leq \sup_{n \geq N}[(1 - \Phi(\frac{\sqrt{n}}{\sigma^2}(\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'))(1-\gamma) + \gamma].$$

Let $z_n = \frac{\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'}{\sigma^2}$ and note that $z = \sup_{n \geq N} z_n > 0$. By monotonicity of $\Phi(\cdot)$ for any $\gamma'' > 0$ we can find finite N_3 such that $\sup_{n \geq N_3} \Phi(\sqrt{n} z_n) \geq 1 - \gamma''$. Let $N = \max\{\bar{N}, N_3\}$, $\gamma = \epsilon/2$, $\gamma'' = 0.25\epsilon/(1 - \epsilon/2)$. Then

$$\sup_{n \geq N}[(1 - \Phi(\frac{\sqrt{n}}{\sigma^2}(\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'))(1-\gamma) + \gamma] \leq$$
$$\leq \gamma + (1-\gamma)(\gamma'') = \epsilon/2 + (1 - \epsilon/2)(0.25\epsilon/(1 - \epsilon/2) = 3\epsilon/4 < \epsilon$$

with probability at least $1 - \alpha$. $\square$

E.13 Proof of Theorem 7.33

First, note that since the game is generic, there are no ties with probability 1 and thus by Lemma 7.32 $\Pr_n(\epsilon(r) = 0)$ converges pointwise to 1 a.s. if r is a Nash equilibrium and to 0 otherwise. We now proceed to prove the result by contradition. Suppose that $\hat{r}_n$ is our maximum likelihood estimator of a Nash equilibrium when the number of samples taken of overy pure profile is n (without loss of generality, we assume that it is the same for all profiles). Let us suppose that $\epsilon(\hat{r}_n)$ does not converge to 0. Then there exists $\bar{\epsilon}$ such that for every $N < \infty$ there exists $\bar{n} \geq N$ such that $\epsilon(\hat{r}_{\bar{n}}) > \bar{\epsilon}$. Now, suppose that the player $\bar{i}$ and action $\bar{b}$ are most beneficial deviant and deviation respectively. Let

$U = \bar{u}_{\bar{i}}(\hat{r}_{\bar{n}})$ and let $K = \sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c)}{\bar{n}}}$. Then

$$\Pr(\epsilon(\hat{r}_{\bar{n}}) = 0) \leq \int_{\mathbb{R}} (1 - \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - u}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}} \right]) f_{W_{\bar{i}}^*}(u)du \leq$$

$$\leq 1 - \int_{U-\bar{\epsilon}}^{U+\bar{\epsilon}} \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - u}{\sqrt{\sum_{c \in A_{-i}} \frac{\sigma_i^2(b,c)s_{-i}^2(c)}{n_i(b,c)}}} \right] f_{W_{\bar{i}}^*}(u)du \leq$$

$$\leq 1 - \int_{U-\bar{\epsilon}}^{U+\bar{\epsilon}} \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - u}{K} \right] f_{W_{\bar{i}}^*}(u)du \leq$$

$$\leq 1 - \int_{U-\bar{\epsilon}}^{U+\bar{\epsilon}} \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - U - \bar{\epsilon}}{K} \right] f_{W_{\bar{i}}^*}(u)du,$$

where the last step is by monotonicity of Φ. Now, by our assumption,

$$\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - U - \bar{\epsilon} > 0.$$

Additionally, since we assumed that the game is generic, it has a finite number of Nash equilibria and thus by Lemma 7.32, for any $\delta > 0$ we can select N large enough such that $\Pr_n(\epsilon(r^*) = 0) \geq 1 - \delta$ for every $n \geq N$ uniformly on these profiles. Furthermore, by the Law of Large Numbers, for any $\alpha > 0$ and $\gamma > 0$ we can select N high enough such that $\Pr(|\bar{u}_{\bar{i}}(r) - u_{\bar{i}}(r)| < \gamma) \geq 1 - \alpha$. Let us select $\gamma = \delta$. Thus,

$$1 - \int_{U-\bar{\epsilon}}^{U+\bar{\epsilon}} \Phi \left[\frac{\sum_{c \in A_{-i}} \bar{u}_i(b,c)\hat{r}_{\bar{n}}(c) - U - \bar{\epsilon}}{K} \right] f_{W_{\bar{i}}^*}(u)du <$$

$$< 1 - 2\Phi(0)\gamma = 1 - \gamma = 1 - \delta$$

with probability at least $1 - \alpha$. But this is a contradiction since then $\Pr(\epsilon(\hat{r}_{\bar{n}}) = 0) < \Pr(\epsilon(r^*) = 0)$.

APPENDIX F

Proofs for Chapter 9

F.1 Proof of Lemma 9.6

Fix $\delta > 0$ and choose $\gamma > 0$ which ensures that for every $\|s - x\| < \gamma$, where $s, x \in S$, $|u_i(s) - u_i(x)| < \delta/2$ for any i. Then,

$$\begin{aligned}
|\epsilon(s) - \epsilon(x)| = |\max_{i\in I}\max_{a\in S_i}[u_i(a, s_{-i}) - u_i(s)] - \\
- \max_{i\in I}\max_{a\in S_i}[u_i(a, x_{-i}) - u_i(x)]| \le \\
\le \max_{i\in I}|\max_{a\in S_i} u_i(a, s_{-i}) - \max_{a\in S_i} u_i(a, x_{-i}) + \\
+ u_i(x) - u_i(s)| \le \\
\le \max_{i\in I}|\max_{a\in S_i} u_i(a, s_{-i}) - \max_{a\in S_i} u_i(a, x_{-i})| + \\
+ \max_{i\in I}|u_i(x) - u_i(s)| \le \\
\le \max_{i\in I}\max_{a\in S_i}|u_i(a, s_{-i}) - u_i(a, x_{-i})| + \max_{i\in I}|u_i(x) - u_i(s)|.
\end{aligned}$$

Since $\|s - x\| < \gamma$, so is $\|(a, s_{-i}) - (a, x_{-i})\|$, and, consequently,

$$|\epsilon(s) - \epsilon(x)| < \max_{i\in I}\delta/2 + \max_{i\in I}\delta/2 = \delta.$$

F.2 Proof of Lemma 9.7

Note that

$$
\begin{aligned}
|\epsilon(s) - \hat{\epsilon}_k(s)| = |\max_i [u_i^*(s_{-i}) - u_i(s)] - \\
- \max_i [\hat{u}_{i,k}(s_{-i}) - \hat{u}_{i,k}(s)]| \leq \\
\leq \max_i |u_i^*(s_{-i}) - u_i(s) - \hat{u}_{i,k}(s_{-i}) + \hat{u}_{i,k}(s)| \leq \\
\leq \max_i |u_i^*(s_{-i}) - \hat{u}_{i,k}(s_{-i})| + \max_i |\hat{u}_{i,k}(s) - u_i(s)|.
\end{aligned}
$$

Since the former converges in probability to 0 for every i by Theorem 9.4 and the latter converges in probability to 0 for every i by the law of large numbers, and since I is finite, we have the desired result.

APPENDIX G

Proofs for Chapter 10

G.1 Proof of Lemma 10.6

Let the deviant be bidder i with value v_i and suppose w.l.o.g. that in a truthful fixed point it gets slot i and in an untruthful fixed point it gets slot j. First, consider $j = i$, that is, its bidding did not affect the slot it received. Since its payment is b_{i+1} which would remain unchanged in a minimum symmetric equilibrium as it only depends on valuations of players ranked below bidder i, its utility will remain unchanged.

Suppose it bids as if its value were $v' < v_i$ and gets slot $j > i$. Since b_{j+1} is unaffected by its untruthfulness, as it only depends on values (and bidding behavior) of bidders ranked below bidder j, and since the bidder was ranked in slot i in a minimum symmetric equilibrium when it bid truthfully, deviation to slot j was unprofitable then and it remains so now.

Finally, suppose i bids as if its value were $v' > v_i$ and gets slot $j < i$. In a truthful minimum symmetric equilibrium,

$$b_{j+1} = v_i(x_{i-1} - x_i) + \sum_{t=j}^{i-1} v_{t+1}(x_{t+1} - x_t) + \sum_{t=i+1}^{K} v_{t+1}(x_{t+1} - x_t).$$

In a minimum symmetric equilibrium which obtains when i is untruthful in the above sense, its payment in slot j is

$$b'_{j+1} = v'_i(x_{i-1} - x_i) + \sum_{t=j}^{i-1} v'_{t+1}(x_{t+1} - x_t) + \sum_{t=i+1}^{K} v'_{t+1}(x_{t+1} - x_t),$$

where v'_t is the value of bidder that would get position t if i played untruthfully. Note that bidders ranked lower than i will retain the same ranking and, as we have already observed, will have the same bids as in a truthful fixed point. Thus,

$$\sum_{t=i+1}^{K} v'_{t+1}(x_{t+1} - x_t) = \sum_{t=i+1}^{K} v_{t+1}(x_{t+1} - x_t).$$

Furthermore, all the bidders in slots $j..i-1$ will now move down one slot. Thus,

$$\sum_{t=j}^{i-1} v'_{t+1}(x_{t+1} - x_t) = \sum_{t=j}^{i-1} v_t(x_{t+1} - x_t) \geq \sum_{t=j}^{i-1} v_{t+1}(x_{t+1} - x_t).$$

Finally, the bidder previously in slot $i-1$ will now be in slot i. Thus,

$$v'_i(x_{i-1} - x_i) = v_{i-1}(x_{i-1} - x_i) \geq v_i(x_{i-1} - x_i).$$

As a result, $b'_{j+1} \geq b_{j+1}$, and if bidder i had no incentive to switch to slot j in a truthful minimum symmetric equilibrium, it certainly will not now, since it would face at least as high a price (and probably higher). Note that these inequalities are strict when values are generic; thus, obtaining a higher slot than under a truthful fixed point yields strictly lower utility to i.

G.2 Proof of Theorem 10.7

Take a player s (recall that players are indexed according to the slots they occupy) and let the discount factor of that player be γ. First, note that if $s \geq K+1$, the player can only win a slot by paying more than v_s, and thus has no incentive to deviate.

Suppose that $s \leq K$. If the player s never deviates, it will accrue the payoff of

$u_s = c_s(v_s - (K-s)\epsilon - \frac{w_{K+1}v_{K+1}}{w_s}$ at every stage. With γ as the discount factor, the resulting total payoff would be $\sum_{i=0}^{\infty} \gamma^i u_s = \frac{u_s}{1-\gamma}$. For ϵ sufficiently small, there will be no incentive to deviate to an inferior slot, since it offers a strictly lower click-through-rate with negligible difference in payment. The one-shot payoff for deviating to $t \leq s$ is $u_s' = c_t(v_s - \frac{w_t}{w_s}(K-t+1)\epsilon - \frac{w_{K+1}v_{K+1}}{w_s})$. For all stages thereafter, the utility will be $u_s^p = c_s(v_s - \sum_{t=s+1}^{K+1} w_{t-1}v_{t-1}\frac{c_{t-1}-c_t}{c_s}) = c_s v_s - \sum_{t=s+1}^{K+1} w_{t-1}v_{t-1}(c_{t-1}-c_t) = c_s v_s - \sum_{t=s+1}^{K} w_{t-1}v_{t-1}(c_{t-1}-c_t) - w_K v_K c_K$. Since this utility will be played starting at the second stage, the total utility from deviating is $u_s' + \frac{\gamma u_s^p}{1-\gamma}$. For deviations to be unprofitable, it must be that for every $s \leq K$ and every $t \leq s$, $\frac{u_s}{1-\gamma} \geq u_s' + \frac{\gamma u_s^p}{1-\gamma}$, or, alternatively, $u_s \geq (1-\gamma)u_s' + \gamma u_s^p$. Plugging in the expressions for utilities and rearranging gives us the result.

G.3 Proof of Theorem 10.10

Since payoffs are discounted (and, thus, the discounted sum converges), for any $\delta > 1$, there is S large enough such that $\sum_{i=S+1}^{\infty} \gamma^i u_s < \delta$, where u_s is the stage payoff during the *COLLUSION* subgame. This is true for any discount factor and any u_s. As the payments are arbitrarily close to being identical during *COLLUSION*, they can be made close enough to identical to eliminate any incentive to reduce the price during collusion, since it would yield a strictly lower click-through-rate. Finally, by Lemma 10.9, all players strictly prefer to stay in their slot than to switch to a lower-numbered slot (lower-numbered is better, i.e., higher on the page). Thus, there exists $\delta > 0$ such that they still strictly prefer their current slot even if they get δ more from switching to the other. Thus, in particular, if their maximum payoff from the *COLLUSION* subgame does not exceed δ, they will have no incentive to switch.

APPENDIX H

Proofs for Chapter 11

H.1 Proof of Theorem 11.2

$$\Pr\left(r \text{ is } \epsilon\text{-dominant}\right) = \Pr\left(\max_{i\in I}\max_{t\in R_{-i}}\max_{r'\in R_i\setminus r_i}[u_i(r',t) - u_i(r_i,t)] \leq \epsilon\right) =$$
$$= \prod_{i\in I}\prod_{t\in R_{-i}}\int_{\mathbb{R}}\prod_{r'\in R_i\setminus r_i}\Pr\{u_i(r',t) \leq u+\epsilon\} f_{u_i(r_i,t)}(u)du.$$

H.2 Proof of Theorem 11.4

$$\Pr\left(r \text{ is } \epsilon\text{-undominated}\right) = \Pr\left(\max_{i\in I}\min_{t\in R_{-i}}\max_{r'\in R_i\setminus r_i}[u_i(r',t) - u_i(r_i,t) \leq \epsilon]\right) =$$
$$= \prod_{i\in I}\left[1 - \Pr\left(\max_{t\in R_{-i}}\min_{r'\in R_i\setminus r_i}[u_i(r',t) - u_i(r_i,t) > \epsilon]\right)\right].$$

Isolating the probability term, we get

$$
\begin{aligned}
&\Pr\left(\max_{t\in R_{-i}}\min_{r'\in R_i\setminus r_i}[u_i(r',t)-u_i(r_i,t)>\epsilon]\right)=\\
&=\prod_{t\in R_{-i}}\int_{\mathbb{R}}\Pr\left(\min_{r'}[u_i(r',t)-u_i(r_i,t)>\epsilon]\right)f_{u_i}(u)du=\\
&=\prod_{t\in R_{-i}}\int_{\mathbb{R}}\left(1-\Pr\{\max_{r'}[u_i(r',t)-u_i(r_i,t)>\epsilon]\}\right\}f_{u_i}(u)du.
\end{aligned}
$$

By substituting this expression into the one above, we obtain the desired result.

H.3 Proof of Theorem 11.7

$$
\begin{aligned}
&\Pr([\max_{i\in I}\max_{t\in R_{-i}}(u_i(r)-u_i(r_i,t))\le\delta]\&[\max_{i\in I}\max_{r'\in R_i}(u_i(r',r_{-i})-u_i(r))\le\epsilon])=\\
&=\prod_{i\in I}\int_{\mathbb{R}}\Pr\{[\max_{t\in R_{-i}}u_i(r_i,t)\ge u-\delta]\&[\max_{r'\in R_i}u_i(r',r_{-i})\le u+\epsilon]\}f_{u_i(r)}(u)du.
\end{aligned}
$$

The result then follows from independence.

H.4 Proof of Theorem 11.8

$$
\begin{aligned}
&\Pr\{\max_{i\in I}\max_{t\in R_{-i}}[u_i(r)-u_i(r_i,t)]\le\delta\;\&\max_{i\in I}\max_{t\in R_{-i}}\max_{r'\in R_i}[u_i(r',t)-u_i(r_i,t)]\le\epsilon\}=\\
&=\prod_{i\in I}\int_{\mathbb{R}}\prod_{t\in R_{-i}}\Pr\{[u_i(r_i,t)\ge u-\delta]\;\&[\max_{r'\in R_i}[u_i(r',t)-u_i(r_i,t)]\le\epsilon]\}f_{u_i(r)}du=\\
&=\prod_{i\in I}\int_{\mathbb{R}}\prod_{t\in R_{-i}}\Pr\{u_i(r_i,t)\ge u-\delta\}\times\\
&\qquad\times\Pr\{\max_{r'\in R_i}[u_i(r',t)-u_i(r_i,t)]\le\epsilon|u_i(r_i,t)\ge u-\delta\}f_{u_i(r)}du.
\end{aligned}
$$

Since

$$\Pr\{\max_{r' \in R_i}[u_i(r',t) - u_i(r_i,t)] \leq \epsilon | u_i(r_i,t) \geq u - \delta\} \geq \prod_{r' \in R_i} \Pr\{u_i(r',t) \leq u + \epsilon - \delta\},$$

the result follows.

H.5 Proof of Theorem 11.9

$$\Pr\{\max_{i \in I} \max_{t \in R_{-i} \backslash r_{-i}} [u_i(r) - u_i(r_i,t)] \leq \delta \,\&\, \max_{i \in I} \min_{t \in R_{-i}} \max_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] \leq \epsilon\} =$$

$$= \prod_{i \in I} \int_{\mathbb{R}} \Bigg[\Pr\{ \max_{t \in R_{-i} \backslash r_{-i}} u_i(r_i,t) \geq u - \delta \bigcap$$

$$\Bigg(\min_{t \in R_{-i} \backslash r_{-i}} \max_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] \leq \epsilon \bigcup \max_{r' \in R_i \backslash r_i} u_i(r', r_{-i}) \leq \epsilon + u \Bigg) \} \Bigg] dF_{u_i(r)}(u)$$

$$= \prod_{i \in I} \int_{\mathbb{R}} \Bigg[\Pr\{ \max_{t \in R_{-i} \backslash r_{-i}} [u_i(r_i,t)] \geq u - \delta\} \times$$

$$\times \Pr\{ \min_{t \in R_{-i} \backslash r_{-i}} \max_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] \leq \epsilon \bigcup$$

$$\max_{r' \in R_i \backslash r_i} u_i(r', r_{-i}) \leq \epsilon + u \mid \max_{t \in R_{-i} \backslash r_{-i}} u_i(r_i,t) \geq u - \delta\} \Bigg] dF_{u_i(r)}(u).$$

Now, let's isolate each of the multiplicand $\Pr\{\cdot\}$ parts above. The first is equivalent to

$$\prod_{t \in R_{-i} \backslash r_{-i}} \Pr\{u_i(r_i,t) \geq u - \delta\}.$$

The case of the second is more complicated:

$$\Pr\{ \min_{t \in R_{-i} \backslash r_{-i}} \max_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] \leq \epsilon \bigcup$$

$$\max_{r' \in R_i \backslash r_i} u_i(r', r_{-i}) \leq \epsilon + u \mid \max_{t \in R_{-i} \backslash r_{-i}} u_i(r_i,t) \geq u - \delta\} =$$

$$1 - \Pr\{ \max_{t \in R_{-i} \backslash r_{-i}} \min_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] > \epsilon \mid \max_{t \in R_{-i} \backslash r_{-i}} u_i(r_i,t) \geq u - \delta\} \times$$

$$\times \Pr\{ \min_{r' \in R_i \backslash r_i} u_i(r', r_{-i}) > \epsilon + u\}.$$

Again, isolating the two multiplicands for convenience, we have

$$1 - \prod_{r' \in R_i \backslash r_i} \Pr\{u_i(r', r_{-i}) > \epsilon + u\}$$

for the second. As for the first,

$$\Pr\{\max_{t \in R_{-i} \backslash r_{-i}} \min_{r' \in R_i \backslash r_i} [u_i(r', t) - u_i(r_i, t)] > \epsilon \mid \max_{t \in R_{-i} \backslash r_{-i}} u_i(r_i, t) \geq u - \delta\} =$$
$$\prod_{t \in R_{-i} \backslash r_{-i}} \Pr\{\min_{r' \in R_i \backslash r_i} [u_i(r', t) - u_i(r_i, t)] > \epsilon \, | u_i(r_i, t) \geq u - \delta\}.$$

At this point, I will proceed by first proving the following claim:

Claim H.1 *Let X, Y be random variables, A, B Borel sets, and suppose we have probability measures for X, Y, as well as the corresponding product measure (I designate each simply using $\Pr\{\cdot\}$, endowing it with the appropriate probability measure). Then*

$$\Pr\{f(X, Y) \in A | Y \in B\} = \frac{\int_B \Pr\{f(X, y) \in A\} f_Y(y) dy}{\int_B f_Y(y) dy},$$

where $F_Y(y)$ is the distribution function for Y.

To prove this, first observe that by Bayes rule,

$$\Pr\{f(X, Y) \in A | Y \in B\} = \frac{\Pr\{f(X, Y) \in A \bigcap Y \in B\}}{\Pr\{Y \in B\}}.$$

For the numerator, I apply smoothing with respect to Y to get

$$\begin{aligned}
&\Pr\{f(X, Y) \in A \bigcap Y \in B\} = \\
&\quad E_Y[\Pr\{f(X, Y) \in A \bigcap Y \in B\} | Y = y] = \\
&\quad \Pr\{f(X, y) \in A \bigcap y \in B\} = \\
&\quad \int_{\mathbb{R}} \Pr\{f(X, y) \in A \bigcap y \in B\} f_Y(y) dy.
\end{aligned}$$

Since

$$\Pr\{f(X,y) \in A \bigcap y \in B\} = \begin{cases} \Pr\{f(X,y) \in A\} & \text{if } y \in B, \\ 0 & \text{if } y \notin B \end{cases}$$

the claim follows.

Now I apply the claim to get

$$\Pr\{\min_{r' \in R_i \backslash r_i} [u_i(r',t) - u_i(r_i,t)] > \epsilon \,|u_i(r_i,t) \geq u - \delta\} = \frac{\int_{u-\delta}^{\infty} \Pr\{\min_{r' \in R_i \backslash r_i} u_i(r',t) > \epsilon + v\} f_{u_i(r_i,t)}(v)dv}{\int_{u-\delta}^{\infty} f_{u_i(r_i,t)}(v)dv}.$$

Finally,

$$\Pr\{\min_{r' \in R_i \backslash r_i} u_i(r',t) > \epsilon + v\} = \prod_{r' \in R_i \backslash r_i} [1 - \Pr\{u_i(r',t) \leq \epsilon + v\}].$$

Combining all the pieces gives us the desired result.

APPENDIX I

Proofs for Chapter 12

I.1 Proof of Theorem 12.1

Fix θ and choose $\delta > 0$. Since $W(s, \theta)$ is continuous at $s^*(\theta)$, given $\epsilon > 0$ there is $\delta > 0$ such that for every s' that is within δ of $s^*(\theta)$, $|W(s', \theta) - W(s^*(\theta), \theta)| < \epsilon$. By Theorem 7.12, we can find $M(\theta)$ large enough such that all $s' \in \mathcal{N}_n$ are within δ of $s^*(\theta)$ for all $n \geq M(\theta)$ with probability 1. Consequently, for any $\epsilon > 0$ we can find $M(\theta)$ large enough such that with probability 1 we have $\sup_{n \geq M(\theta)} \sup_{s' \in \mathcal{N}_n} |W(s', \theta) - W(s^*(\theta), \theta)| < \epsilon$.

Assume without loss of generality that there is a unique optimal choice of θ. Now, since the set Θ is finite, there is also the "second-best" choice of θ (if there is only one $\theta \in \Theta$ this discussion is moot anyway):

$$\theta^{**} = \arg\max_{\Theta \setminus \theta^*} W(s^*(\theta), \theta).$$

Suppose w.l.o.g. that θ^{**} is also unique and let

$$\Delta = W(s^*(\theta^*), \theta^*) - W(s^*(\theta^{**}), \theta^{**}).$$

Then if we let $\epsilon < \Delta/2$ and let $M = \max_{\theta \in \Theta} M(\theta)$, where each $M(\theta)$ is large enough such that $\sup_{n \geq M(\theta)} \sup_{s' \in \mathcal{N}_n} |W(s', \theta) - W(s^*(\theta), \theta)| < \epsilon$ a.s., the optimal choice of θ based on *any* empirical equilibrium will be θ^* with probability 1. Thus, in particular, given any probability distribution over empirical equilibria, the best choice of θ will be θ^* with probability 1 (similarly, if we take supremum or infimum of $W(\mathcal{N}_n(\theta), \theta)$ over the set of empirical equilibria in constructing the objective function).

I.2 Proof of Lemma 12.2

We start by proving the first result in the lemma. To do this, we arbitrarily select $\theta \in \Theta$.

We would like to show that for every $\gamma > 0$ and $\alpha > 0$, we can find M large enough such that the probability that

$$\sup_{n \geq M_r} |W_n(R, \theta) - W(R, \theta)| < \delta.$$

is at most $1 - \alpha$.

Let us select $\gamma > 0$ and $\alpha > 0$. We know from the condition in the theorem that for every r and for any $\delta > 0$ we can find M_r such that the probability that

$$\sup_{n \geq M_r} |P_n(r, \theta) - P(r, \theta)| < \delta.$$

is at most $1 - \alpha$. If we take $M = \max_{r \in R} M_r$, the above condition would hold uniformly on R. Now,

$$\sup_{n \geq M} |W_n(R, \theta) - W(R, \theta)| = \sup_{n \geq M} |\sum_{r \in R} W(r, \theta)(P_n(r, \theta) - P(r, \theta))| \leq$$
$$\leq \sup_{n \geq M} \sum_{r \in R} |W(r, \theta)||P_n(r, \theta) - P(r, \theta)|.$$

If we take $\overline{W}(\theta) = \max_{r \in R} |W(r, \theta)|$, we can rewrite this expression as

$$\sup_{n \geq M} \sum_{r \in R} |W(r, \theta)||P_n(r, \theta) - P(r, \theta)| \leq \overline{W}(\theta) \sup_{n \geq M} \sum_{r \in R} |P_n(r, \theta) - P(r, \theta)| \leq$$
$$\overline{W} \sum_{r \in R} \sup_{n \geq M} |P_n(r, \theta) - P(r, \theta)| < \overline{W}|R|\delta.$$

If we now set $\delta(\theta) = \frac{\gamma}{\overline{W}(\theta)|R|}$, the desired result follows, since γ, α and θ were arbitrarily chosen.

To obtain the second result, note that since $W(r, \theta)$ is continuous and Θ is compact, $\overline{W}(r) = \max_{\theta \in \Theta} W(r, \theta)$ exists by the Weierstrass theorem. Let us define $\overline{W} = \max_{r \in R} \overline{W}(r)$. Since we can find M large enough such that δ in the above proof does not depend on θ, we can replace $\delta(\theta)$ by δ and $\overline{W}(\theta)$ by $\overline{W}$ and the result follows independently of θ, thus showing uniformity.

I.3 Proof of Lemma 12.5

Fix $\alpha > 0, \gamma > 0$ and let M be large enough such that

$$\sup_{n \geq M} |W_n(R, \theta) - W(R, \theta)| < \delta$$

uniformly on Θ with probability at least $1 - \alpha$. Then,

$$\begin{aligned}
&\sup_{n \geq M} |W(R, \theta^*) - W(R, \hat{\theta})| = \sup_{n \geq M} (W(R, \theta^*) - W(R, \hat{\theta})) = \\
&\sup_{n \geq M} (W(R, \theta^*) - W(R, \hat{\theta}) - W_n(R, \theta^*) + W_n(R, \theta^*) + W_n(R, \hat{\theta}) - W_n(R, \hat{\theta})) \leq \\
&\sup_{n \geq M} (|W(R, \theta^*) - W_n(R, \theta^*)| + |W_n(R, \hat{\theta}) - W(R, \hat{\theta})| - (W_n(R, \hat{\theta}) - W_n(R, \theta^*))) \\
&< 2\delta
\end{aligned}$$

with probability at least $1 - \alpha$. If we let $\delta = \gamma/2$ above, we obtain the desired result.

I.4 Proof of Theorem 12.8

First, we will fix $\theta \in \Theta$. For convenience, we drop it as an argument. We would like to show that for every $\gamma > 0$ and $\alpha > 0$ there is M such that the probability that

$$\sup_{n \geq M} |P(r) - P_n(r)| < \gamma$$

is at least $1 - \alpha$.

Fix $\gamma > 0$ and $\alpha > 0$. It is given that for any δ and r we can find M_r such that the probability that

$$\sup_{n \geq M_r} |m(r) - m_n(r)| < \delta$$

is at least $1 - \alpha$. To get uniformity, set $M = \max_{r \in R} M_r$. Let $\bar{m} = \max_{r \in R} m(r)$, $S = \sum_{r \in R} m(r)$.

Consequently, with probability at least $1 - \alpha$ we have

$$\begin{aligned}
&\sup_{n \geq M} |P(r) - P_n(r)| = \sup_{n \geq M} |\frac{m(r)}{\sum_{t \in R} m(t)} - \frac{m_n(r)}{\sum_{t \in R} m_n(t)}| \leq \\
&\sup_{n \geq M} (|\frac{m(r) + \delta}{\sum_{t \in R} m(t)} - \frac{m(t)}{\sum_{t \in R} m_n(t)}| + |\frac{m(r)}{\sum_{t \in R} m(t)} - \frac{m(t) - \delta}{\sum_{t \in R} m_n(t)}|) \leq \\
&2 \sup_{n \geq M} m(r)(|\frac{1}{\sum_{t \in R} m(t)} - \frac{1}{\sum_{t \in R} m_n(t)}|) + \frac{\delta}{\sum_{t \in R} m_n(t)} \leq \\
&2 \sup_{n \geq M} \bar{m} |\frac{\sum_{t \in R}(m(t) - m_n(t))}{(\sum_{t \in R} m(t))(\sum_{t \in R} m_n(t))}| + \frac{\delta}{\sum_{t \in R} m_n(t)} \leq \\
&2 \sup_{n \geq M} \bar{m} |\frac{\sum_{t \in R}(m(t) - m_n(t))}{(\sum_{t \in R} m(t))(\sum_{t \in R} m_n(t))}| + \frac{\delta}{\sum_{t \in R} m_n(t)} \leq \\
&2 \frac{\bar{m}}{S} \sup_{n \geq M} |\frac{\sum_{t \in R}(m(t) - m_n(t))}{(\sum_{t \in R} m_n(t))}| + 2\delta \sup_{n \geq M} \frac{1}{\sum_{t \in R} m_n(t)}.
\end{aligned}$$

Since there is t' such that $m(t') > 0$, we can set $0 < \alpha < m(t')$ and find $M(\alpha)$ large enough such that $\inf_{n \geq M(\alpha)} m_n(t') = l(\alpha) > 0$. We can further observe that for any $\delta \leq \alpha$, $l(\delta) \geq l(\alpha)$.

Now,

$$\sup_{n \geq M} \frac{1}{\sum_{t \in R} m_n(t)} = \frac{1}{\inf_{n \geq M} \sum_{t \in R} m_n(t)} \leq \frac{1}{\inf_{n \geq M} m(t')} = \frac{1}{l(\delta)} \leq \frac{1}{l(\alpha)},$$

as long as $\delta \leq \alpha$.

Similarly,

$$\begin{aligned}\sup_{n\geq M}|\frac{\sum_{t\in R}(m(t)-m_n(t))}{\sum_{t\in R}m_n(t)}| &= \sup_{n\geq M}\frac{|\sum_{t\in R}(m(t)-m_n(t))|}{\sum_{t\in R}m_n(t)}\\ &\leq \sum_{t\in R}\sup_{n\geq M}\frac{|m(t)-m_n(t)|}{\sum_{t\in R}m_n(t)}\\ &\leq \sum_{t\in R}\frac{\sup_{n\geq M}|m(t)-m_n(t)|}{\inf_{n\geq M}\sum_{t\in R}m_n(t)}\\ &\leq \sum_{t\in R}\frac{\delta}{l(\delta)} = \frac{\delta|R|}{l(\delta)} \leq \frac{\delta|R|}{l(\alpha)}.\end{aligned}$$

so long as $\delta \leq \alpha$.

Putting these together gives us

$$\sup_{n\geq M}|P(r)-P_n(r)| \leq 2\delta(\frac{\bar{m}|R|}{l(\alpha)S}+\frac{1}{l(\alpha)})$$

with probability at least $1-\alpha$. To achieve the desired result, then, we need to set

$$\delta = \min\{\alpha, \frac{\gamma}{2(\frac{\bar{m}|R|}{l(\alpha)S}+\frac{1}{l(\alpha)})}\}.$$

.

I.5 Proof of Theorem 12.10

Claim I.1 *Suppose r is a Nash equilibrium. Then $m_n(r) \to m(r)$.*

Proof. For simplicity, we will assume (without loss of generality) that every player and profile has the same variance σ^2. Let $f_i(u)$ be the density function of the normal distribution $N(\bar{u}_i(r), \sigma^2)$. Now, fix $\epsilon > 0$ and $\alpha > 0$ (and assume both to be below 1). By strong law of large numbers, payoffs converge almost surely for every player and profile. Thus, for any $\delta > 0$ and $\gamma > 0$, there is N_1 finite such that

$\Pr\{\sup_{n\geq N_1} |\bar{u}_{i,n}(r) - u_i(r)| < \delta\} \geq 1-\gamma$. We also have that

$$\sup_{n\geq N_1}[1-\prod_{i\in I}(\int \prod_{b\in S_i\setminus r_i} \Pr\{u_i(b,r_{-i}) - u_i(r) < 0\} f_i(u)du)] \leq$$
$$\leq \sup_{n\geq N_1}[1-\prod_{i\in I}(\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b\in S_i\setminus r_i} \Pr\{u_i(b,r_{-i}) - u_i(r) < 0\} f_i(u)du)].$$

Also, uniformly on S for any $\delta' > 0$ there is N_2 finite such that $\Pr\{\sup_{n\geq N_2} |\bar{u}_{i,n}(b,r_{-i}) - u_i(b,r_{-i})\| < \delta'\} \geq 1-\alpha$. Since by assumption $u_i(r) > u_i(b,r_{-i})$ for any $b \neq r_i$, there are $\delta > 0, \delta' > 0$ such that $u_i(r) - \delta' - \delta > u_i(b,r_{-i})$. Recall that $\Pr\{u_i(b,r_{-i}) - u_i(r) < 0\} = 1 - \Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b,r_{-i}) - u))$. Since $u \geq u_i(r) - \delta$ and $\Phi(\cdot)$ is monotone increasing, $\Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b,r_{-i}) - u)) \leq \Phi(\frac{\sqrt(n)}{\sigma^2}(\bar{u}_{i,n}(b,r_{-i}) - \bar{u}_{i,n}(r) + \delta + \delta'))$. Let $z_n = \frac{\bar{u}_{i,n}(b,r_{-i}) - \bar{u}_{i,n}(r) + \delta + \delta'}{\sigma^2}$ and note that $z = \sup_{n\geq \max\{N_1,N_2\}} z_n < 0$. Thus, by monotonicity of $\Phi(\cdot)$, for any $\gamma_i'' > 0$ there is N_3 finite such that $\sup_{n\geq\max\{N_1,N_2,N_3\}} \Phi(\sqrt{n}z_n) \leq \Phi(\sqrt{N_3}z) < \gamma_i''$. Let $N = \max\{N_1,N_2,N_3\}$, $\gamma = (0.5(1-(1-\epsilon)^{1/|I|})$, and $\gamma_i'' = 1 - [\frac{(1-\epsilon)}{0.5(1+(1-\epsilon)^{1/|I|}}]^{1/|S_i|}$. Then

$$\sup_{n\geq N}[1-\prod_{i\in I}(\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b\in S_i\setminus r_i} \Pr\{u_i(b,r_{-i}) - u_i(r) < 0\} f_i(u)du)] \leq$$
$$\leq \sup_{n\geq N}[1-\prod_{i\in I}((1-\epsilon)^{1/|I|})] \leq 1-1+\epsilon = \epsilon$$

with probability at least $1-\alpha$. □

Claim I.2 *Suppose r is not a Nash equilibrium. Then $m_n(r) \to m(r)$.*

Proof. Let $f_i(u)$ be as before and select $\epsilon > 0$ and $\alpha > 0$. We want to show that there is N finite such that $\Pr\{\sup_{n\geq N} \Pr\{\epsilon(r) = 0|\Gamma_n\} < \epsilon\} \geq 1-\alpha$. By the law of large numbers, for any $\delta > 0$ and $gamma > 0$ there is N_1 finite such that $\Pr\{\sup_{n\geq N_1} |\bar{u}_{i,n}(r) - u(r)| <$

$\delta\} \geq 1 - \gamma$. Consequently,

$$\sup_{n \geq N_1} (\prod_{i \in I} [\int \prod_{b \in S_i \setminus r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du]) \leq$$

$$\leq \sup_{n \geq N_1} (\prod_{i \in I} [\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \prod_{b \in S_i \setminus r_i} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du + \gamma]) \leq$$

$$\leq \sup_{n \geq N_1} [\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \Pr\{u_i(b, r_{-i}) - u_i(r) < 0\} f_i(u) du + \gamma],$$

where the last inequality takes advantage of the fact that a product of probabilities is no greater than each taken separately. We thus select an arbitrary player i which has b_i, such that $u_i(b_i, r_{-i}) > u_i(r)$, which is possible by the no-ties assumption and the fact that r is not a Nash equilibrium. Applying the law of large numbers again, we have that for every $\delta' > 0$, there is N_2 finite such that $\Pr\{\sup_{n \geq N_2} |\bar{u}_{i,n}(b_i, r_{-i}) - u_i(r)| < \delta'\} \geq 1 - \alpha$. Again, by the no-ties assumption, we have $\delta > 0, \delta' > 0$ such that $u_i(b_i, r_{-i}) - u_i(r) - \delta - \delta' > 0$. Thus, with probability at least $1 - \alpha$, and letting $\bar{N} = \max\{N_1, N_2\}$, we have

$$\sup_{n \geq N} [\int_{\bar{u}_{i,n}(r)-\delta}^{\bar{u}_{i,n}(r)+\delta} \Pr\{\bar{u}_{i,n}(b_i, r_{-i}) - u < 0\} f_i(u) du + \gamma] \leq$$

$$\leq \sup_{n \geq N} [(1 - \Phi(\frac{\sqrt{n}}{\sigma^2}(\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'))(1 - \gamma) + \gamma].$$

Let $z_n = \frac{\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'}{\sigma^2}$ and note that $z = \sup_{n \geq N} z_n > 0$. By monotonicity of $\Phi(\cdot)$ for any $\gamma'' > 0$ we can find finite N_3 such that $\sup_{n \geq N_3} \Phi(\sqrt{n} z_n) \geq 1 - \gamma''$. Let $N = \max\{\bar{N}, N_3\}$, $\gamma = \epsilon/2$, $\gamma'' = 0.25\epsilon/(1 - \epsilon/2)$. Then

$$\sup_{n \geq N} [(1 - \Phi(\frac{\sqrt{n}}{\sigma^2}(\bar{u}_i(b_i, r_{-i}) - \bar{u}_{i,n}(r) - \delta - \delta'))(1 - \gamma) + \gamma] \leq$$

$$\gamma + (1 - \gamma)(\gamma'') = \epsilon/2 + (1 - \epsilon/2)(0.25\epsilon/(1 - \epsilon/2) = 3\epsilon/4 < \epsilon.$$

□

I.6 Proof of Theorem 12.13

Suppose $I(r) = 1$ (i.e., r is a strict $\bar{\epsilon}$-Nash). Then, r is also a strict ϵ'-Nash for some $\epsilon' < \bar{\epsilon}$.

Fix $\alpha > 0$, $\gamma > 0$. By Theorem 7.9, for any $\delta > 0$ there is an M such that with probability of at least $1 - \alpha$, $\sup_{n \geq M} |\epsilon(r) - \epsilon_n(r)| < \delta$, or, alternatively, $\inf_{n \geq M} \epsilon_n(r) > \epsilon(r) - \delta$, and $\sup_{n \geq M} \epsilon_n(r) < \epsilon(r) + \delta$.

Let $\delta < \bar{\epsilon} - \epsilon'$. Then

$$\sup_{n \geq M} \epsilon_n(r) < \epsilon(r) + \delta < \epsilon' + \delta < \bar{\epsilon}$$

with probability at least $1 - \alpha$. Thus, $\inf_{n \geq M} I_n(r) = 1$.

Similarly, we can prove the same if r is not $\bar{\epsilon}$-Nash by finding $\epsilon' > \bar{\epsilon}$ and allowing $\delta < \epsilon' - \bar{\epsilon}$.

BIBLIOGRAPHY

BIBLIOGRAPHY

Olivier Armantier and Jean-Francois Richard. Empirical game theoretic models: Computational issues. *Computational Economics*, 15(1-2):3–24, 2000.

Olivier Armantier, Jean-Pierre Florens, and Jean-Francois Richard. Approximation of Bayesian Nash equilibria. Working paper, 2007.

Raghu Arunachalam and Norman M. Sadeh. The supply chain trading agent competition. *Electronic Commerce Research and Applications*, 4:63–81, 2005.

Raghu Arunachalam, Joakim Eriksson, Niclas Finne, Sverker Janson, and Norman Sadeh. The TAC supply chain management game. Technical report, Swedish Institute of Computer Science, 2003.

Christopher G. Atkeson, Andrew W. Moore, and Stefan Schaal. Locally weighted learning. *Artificial Intelligence Review*, 11:11–73, 1997.

Lawrence M. Ausubel and Peter Cramton. Demand reduction and inefficiency in multi-unit auctions. Working paper, 2002.

Erik J. Balder. Remarks on Nash equilibria for games with additively coupled payoffs. *Economic Theory*, 9(1):161–167, 1996.

Aharon Ben-Tal and Arkadi Nemirovski. Robust optimization—Methodology and applications. *Mathematical Programming*, 92:453–480, 2002.

Michael Benisch, Amy Greenwald, Victor Naroditskiy, and Michael Tschantz. A stochastic programming approach to scheduling in TAC SCM. In *Fifth ACM Conference on Electronic Commerce*, pages 152–159, New York, 2004.

Navin Bhat and Kevin Leyton-Brown. Computing Nash equilibria of action-graph games. In *Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 35–42, 2004.

Felix Brandt and Gerhard Weiß. Antisocial agents and Vickrey auctions. In *Eighth International Workshop on Agent Theories, Architectures, and Languages*, volume 2333 of *Lecture Notes in Computer Science*, pages 335–347, Seattle, 2001. Springer.

Felix Brandt, Tuomas Sandholm, and Yoav Shoham. Spiteful bidding in sealed-bid auctions. In *Twentieth International Joint Conference in Artificial Intelligence*, pages 1207–1214, 2007.

Matthew Cary, Aparna Das, Ben Edelman, Ioannis Giotis, Kurtis Heimerl, Anna R. Karlin, Claire Mathieu, and Michael Schwarz. Greedy bidding strategies for keyword auctions. In *Eighth ACM Conference on Electronic Commerce*, pages 262–271, 2007.

Siu-Shing Chan. On using recursive least squares in sample-path optimization of discrete event systems. Working paper, 1995.

Yi-Ping Chang and Wen-Tao Huang. Generalized confidence intervals for the largest value of some functions of parameters under normality. *Statistica Sinica*, 10:1369–1383, 2000.

Xi Chen and Xiaotie Deng. Settling the complexity of two-player Nash equilibrium. In *Symposium on Foundations of Computer Science*, pages 261–272, 2006.

Shih-Fen Cheng and Michael P. Wellman. Iterated weaker-than-weak dominance. In *Twentieth International Joint Conference on Artificial Intelligence*, pages 1233–1238, 2007.

Shih-Fen Cheng, Daniel M. Reeves, Yevgeniy Vorobeychik, and Michael P. Wellman. Notes on equilibria in symmetric games. In *AAMAS-04 Workshop on Game-Theoretic and Decision-Theoretic Agents*, New York, 2004.

Edward H. Clarke. Multipart pricing of public goods. *Public Choice*, 2:19–33, 1971.

Dave Cliff. Evolution of market mechanism through a continuous space of auction-types. In *Congress on Evolutionary Computation*, pages 2029–2034, 2002.

Dave Cliff. Evolution of market mechanism through a continuous space of auction-types II: Two-sided auction mechanisms evolve in response to market shocks. In *Agents for Business Automation*, pages 682–688, 2002.

David Cohn, Zoubin Ghahramani, and Michael Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. In *Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 103–110, 2002.

Vincent Conitzer and Tuomas Sandholm. Applications of automated mechanism design. In *UAI-03 Bayesian Modeling Applications Workshop*, 2003.

Vincent Conitzer and Tuomas Sandholm. An algorithm for automatically designing deterministic mechanisms without payments. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 128–135, 2004.

Vincent Conitzer and Tuomas Sandholm. Self-interested mechanism design and implications for optimal combinatorial auctions. In *Fifth ACM Conference on Electronic Commerce*, pages 132–141, 2004.

Vincent Conitzer and Tuomas Sandholm. Incremental mechanism design. In *Twentieth International Joint Conference on Artificial Intelligence*, pages 1251–1256, 2007.

Vincent Conitzer and Tuomas Sandholm. New complexity results about Nash equilibria. *Games and Economic Behavior*, 2008. To appear.

Russell Cooper and Thomas W. Ross. Cooperation without reputation: experimental evidence from prisoner's dilemma games. *Games and Economic Behavior*, 12:187–218, 1996.

A. Corana, M. Marchesi, C. Martini, and S. Ridella. Minimizing multimodal functions of continuous variables with simulated annealing algorithm. *ACM Transactions on Mathematical Software*, 13(3):262–280, 1987.

Peter Cramton, Yoav Shoham, and Richard Steinberg, editors. *Combinatorial Auctions*. MIT Press, 2006.

Peter Cramton. Dissolving a partnership efficiently. *Econometrica*, 55(3):615–632, 1987.

Peter Cramton. Simultaneous ascending auctions. In Cramton et al. [2006].

Constantinos Daskalakis, Paul Goldberg, and Christos Papadimitriou. The complexity of computing a nash equilibrium. In *Symposium on Theory of Computing*, pages 71–78, 2006.

Claude d'Aspremont and Louis-André Gérard-Varet. Incentives and incomplete information. *Journal of Public Economics*, 11:25–45, 1979.

Quang Duong, Michael P. Wellman, and Satinder Singh. Knowledge combination in graphical multiagent models. In *Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, Helsinki, 2008.

Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 9(1):242–259, 2007.

Ido Erev and Alvin E. Roth. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *American Economic Review*, 88(4):848–881, 1998.

Juan Feng and Xiaoquan Zhang. Dynamic price competition on the Internet: Advertising auctions. In *Eighth ACM Conference on Electronic Commerce*, pages 57–58, 2007.

Mark Fleischer. Simulated Annealing: Past, present, and future. In *Winter Simulation Conference*, pages 155–161, 1995.

Daniel Friedman. Evolutionary games in economics. *Econometrica*, 59(3):637–666, 1991.

Drew Fudenberg and David K. Levine. *The Theory of Learning in Games*. MIT Press, 1998.

Drew Fudenberg and Jean Tirole. *Game Theory*. MIT Press, 1991.

D. Fudenberg, D. Kreps, and D. Levine. On the robustness of equilibrium refinements. *Journal of Economic Theory*, 44:354–380, 1988.

Archis Ghate and Robert L. Smith. Adaptive search with stochastic acceptance probabilities for global optimization. *Operations Research Letters*, 2008. to appear.

Itzhak Gilboa and Eitan Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1:80–93, 1989.

Srihari Govindan and Robert Wilson. Structure theorems for game trees. *National Academy of Sciences*, 99(13):9077–9080, 2002.

Srihari Govindan and Robert Wilson. A global Newton method to compute Nash equilibria. *Journal of Economic Theory*, 110(1):65–86, 2003.

Jerry R. Green and Jean-Jacques Laffont. *Incentives in Public Decision Making*. Amsterdam: North-Holland, 1979.

Gul Gukan, A. Yonca Ozge, and Stephen M. Robinson. Sample-path optimization in simulation. In *Winter Simulation Conference*, pages 247–254, 1994.

F. Gul, D. Pearce, and E. Stacchetti. A bound on the proportion of pure strategy equilibria in generic games. *Mathematical Methods for Operations Research*, 18:548–552, 1993.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.

M. Hirsch and S. Smale. On algorithms for solving $f(x) = 0$. *Communications of Pure Applied Mathematics*, 32:281–312, 1979.

Nathanael Hyafil and Craig Boutilier. Mechanism design with partial revelation. In *Twentieth International Joint Conference on Artificial Intelligence*, pages 1333–1340, 2007.

Nathanael Hyafil and Craig Boutilier. Partial revelation automated mechanism design. In *Twenty-Second National Conference on Artificial Intelligence*, pages 72–78, 2007.

Albert Xin Jiang and Kevin Leyton-Brown. A polynomial-time algorithm for action-graph games. In *Twenty-First National Conference on Artificial Intelligence*, pages 679–684, 2006.

Thorsten Joachims. Making large-scale SVM learning practical. In B. Scholkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods—Support Vector Learning*. MIT Press, 1999.

Patrick Jordan, Yevgeniy Vorobeychik, and Michael P. Wellman. Searching for approximate equilibria in empirical games. In *Seventh International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1063–1070, 2008.

Laura Kang and David C. Parkes. Passive verification of the strategyproofness of mechanisms in open environments. In *International Conference on Electronic Commerce*, pages 19–30, 2006.

Michael Kearns, Michael L. Littman, and Satinder Singh. Graphical models for game theory. In *Seventeenth Conference on Uncertainty in Artificial Intelligence*, pages 253–260, 2001.

Robert Keener. *Statistical Theory: A Medley of Core Topics*. University of Michigan Department of Statistics, 2004.

Philipp W. Keller, Felix-Olivier Duguay, and Doina Precup. RedAgent: Winner of TAC SCM 2003. *SIGecom Exchanges*, 4(3):1–8, 2004.

J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

Christopher Kiekintveld and Michael P. Wellman. Selecting strategies using empirical game models: an experimental analysis of meta-strategies. In *Seventh International Conference on Autonomous Agents and Multiagent Systems*, 2008.

Christopher Kiekintveld, Michael P. Wellman, Satinder Singh, Joshua Estelle, Yevgeniy Vorobeychik, Vishal Soni, and Matthew Rudary. Distributed feedback control for decision making on supply chains. In *Fourteenth International Conference on Automated Planning and Scheduling*, pages 384–392, 2004.

Christopher Kiekintveld, Yevgeniy Vorobeychik, and Michael P. Wellman. An analysis of the 2004 supply chain management trading agent competition. In *IJCAI-05 Workshop on Trading Agent Design and Analysis*, 2005.

Christopher Kiekintveld, Michael P. Wellman, and Satinder Singh. Empirical game-theoretic analysis of Chaturanga. In *AAMAS-06 Workshop on Game-Theoretic and Decision-Theoretic Agents*, 2006.

E. Kohlberg and J.-F. Mertens. On the strategic stability of equilibria. *Econometrica*, 54:1003–1039, 1986.

Daphne Koller and Brian Milch. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1):181–221, 2003.

David M. Kreps. *Game Theory and Economic Modelling*. Oxford University Press, 1990.

Vijay Krishna. *Auction Theory*. Academic Press, 1st edition, 2002.

Sebastien Lahaie and David M. Pennock. Revenue analysis of a family of ranking rules for keyword auctions. In *Eighth ACM Conference on Electronic Commerce*, pages 50–56, 2007.

Sebastien Lahaie. An analysis of alternative slot auction designs for sponsored search. In *Seventh ACM Conference on Electronic Commerce*, pages 218–227, 2006.

Averill M. Law and W. David Kelton. *Simulation Modeling and Analysis*. Tata McGraw-Hill, 3rd edition, 2000.

Pierre L'Écuyer. An overview of derivative estimation. In *Winter Simulation Conference*, pages 207–217, 1991.

Pierre L'Écuyer. Efficiency improvement and variance reduction. In *Winter Simulation Conference*, pages 122–132, 1994.

John Ledyard. Optimal combinatoric auctions with single-minded bidders. In *Eighth ACM Conference on Electronic Commerce*, pages 237–242, 2007.

C. Lemke and J. Howson. Equilibrium points of bimatrix games. *Journal of the Society of Industrial and Applied Mathematics*, 12:413–423, 1964.

David G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, Inc, 1969.

Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, 1995.

Richard D. McKelvey and Andrew McLennan. Computation of equilibria in finite games. In *Handbook of Computational Economics*, volume 1, chapter 2, pages 87–142. North Holland, 1996.

Richard D. McKelvey and Thomas R. Palfrey. Quantal response equilibria for normal form games. *Games and Economic Behavior*, 10:6–38, 1995.

Richard D. McKelvey, Andrew M. McLennan, and Theodore L. Turocy. Gambit: Software tools for game theory, version 0.2005.06.13, 2005.

Richard D. McKelvey. A Liapunov function for Nash equilibria. Working Paper, 1998.

Paul Milgrom and John Roberts. Rationalizability, learning, and equilibrium in games with strategic complementarities. *Econometrica*, 58(6):1255–1277, 1990.

Paul Milgrom. Putting auction theory to work: The simultaneous ascending auctions. *Journal of Political Economy*, 108(2):245–272, 2000.

Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economics Behavior*, 14:124–143, 1996.

John Morgan, Ken Steiglitz, and George Reis. The spite motive and equilibrium behavior in auctions. *Contributions to Economic Analysis and Policy*, 2(1), 2003.

Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

John Nash. Non-cooperative games. *Annals of Mathematics*, 54:286–295, 1951.

John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1980.

Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer, 2006.

Eugene Nudelman, Jennifer Wortman, Yoav Shoham, and Kevin Leyton-Brown. Run the GAMUT: A comprehensive approach to evaluating game-theoretic algorithms. In *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 880–887, New York, 2004.

Sigurdur Olafsson and Jumi Kim. Simulation optimization. In *Winter Simulation Conference*, pages 79–84, 2002.

Luis E. Ortiz and Michael Kearns. Nash propagation for loopy graphical games. *Advances in Neural Information Processing Systems*, 15:793–800, 2003.

Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

Anna Osepayshvili, Michael P. Wellman, Daniel M. Reeves, and Jeffrey K. MacKie-Mason. Self-confirming price prediction for bidding in simultaneous ascending auctions. In *Twenty-First Conference on Uncertainty in Artificial Intelligence*, pages 441–449, 2005.

David Pardoe and Peter Stone. TacTex-03: A supply chain management agent. *SIGecom Exchanges*, 4(3):19–28, 2004.

David Pardoe, Peter Stone, Mayal Saar-Tsechansky, and Kerem Tomak. Adaptive mechanism design: A metalearning approach. In *Eighth International Conference on Electronic Commerce*, 2006.

David C. Parkes. Iterative combinatorial auctions. In *Combinatorial Auctions*, pages 41–78. MIT Press, 2006.

David G. Pearce. Rationalizable strategic behavior and the problem of perfection. *Econometrica*, 52(4):1029–1050, 1984.

Steve Phelps, Simon Parsons, Peter McBurney, and Elizabeth Sklar. Co-evolution of auction mechanisms and trading strategies: towards a novel approach to microeconomic design. In *ECOMAS 2002 Workshop*, 2002.

Steve Phelps, Simon Parsons, Elizabeth Sklar, and Peter McBurney. Using genetic programming to optimise pricing rules for a double-auction market. In *Workshop on Agents for Electronic Commerce*, 2003.

Ryan W. Porter, Eugene Nudelman, and Yoav Shoham. Simple search methods for finding a Nash equilibrium. *Games and Economic Behavior*, 2006. To appear.

Daniel M. Reeves and Michael P. Wellman. Computing best-response strategies in infinite games of incomplete information. In *Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 470–478, 2004.

Daniel M. Reeves, Michael P. Wellman, Jeffrey K. MacKie-Mason, and Anna Osepayshvili. Exploring bidding strategies for market-based scheduling. *Decision Support Systems*, 39:67–85, 2005.

Daniel M. Reeves. *Generating Trading Agent Strategies: Analytic and Empirical Methods for Infinite and Large Games*. PhD thesis, University of Michigan, 2005.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

Robert W. Rosenthal. Bargaining rules of thumb. *Journal of Economic Behavior and Organization*, 22:15–24, 1993.

Robert W. Rosenthal. Rules of thumb in games. *Journal of Economic Behavior and Organization*, 22:1–13, 1993.

Sheldon M. Ross. *Simulation*. Academic Press, 3rd edition, 2001.

Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2nd edition, 2003.

Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial Intelligence*, 49:361–395, 1991.

Tuomas Sandholm, Andrew Gilpin, and Vincent Conitzer. Mixed-integer programming methods for finding Nash equilibria. In *Twentieth National Conference on Artificial Intelligence*, pages 495–501, 2005.

Tuomas Sandholm, Vincent Conitzer, and Craig Boutilier. Automated design of multistage mechanisms. In *Twentieth International Joint Conference on Artificial Intelligence*, pages 1500–1506, 2007.

Herbert Scarf. The approximation of fixed points of a continuous mapping. *SIAM Journal of Applied Mathematics*, 15:1328–1343, 1967.

Reinhard Selten and Joachim Buchta. Experimental sealed bid first price auctions with directly observed bid functions. Technical Report Discussion Paper B-270, University of Bonn, 1994.

Reinhard Selten. Evolution, learning, and economic behavior. *Games and Economic Behavior*, 3:3–24, 1991.

Patrick Siarry, Gerard Berthiau, Francois Durbin, and Jacques Haussy. Enhanced simulated annealing for globally minimizing functions of many continuous variables. *ACM Transactions on Mathematical Software*, 23(2):209–228, 1997.

Satinder Singh, Vishal Soni, and Michael P. Wellman. Computing approximate Bayes-Nash equilibria in games of incomplete information. In *Fifth ACM Conference on Electronic Commerce*, pages 81–90, 2004.

Vishal Soni, Satinder Singh, and Michael P. Wellman. Constraint satisfaction algorithms for graphical games. In *International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2007.

James C. Spall. *Introduction to Stochastic Search and Optimization*. John Wiley and Sons, Inc, 2003.

Ashish Sureka and Peter R. Wurman. Using tabu best-response search to find pure strategy Nash equilibria in normal form games. In *Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1023–1029, 2005.

Theodore L. Turocy. A dynamic homotopy interpretation of the logistic quantal response equilibrium correspondence. *Games and Economic Behavior*, 51:243–263, 2005.

Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

Hal Varian. Position auctions. *International Journal of Industrial Organization*, 25(6):1163–1178, 2007.

William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

Bernhard von Stengel. Computing equilibria for two-person games. In R.J. Aumann and S. Hart, editors, *Handbook of Game Theory with Economic Applications*, pages 1723–1759. Elsevier, 2002.

Yevgeniy Vorobeychik and Daniel M. Reeves. Equilibrium analysis of dynamic bidding in sponsored search auctions. *International Journal of Electronic Business*, 2008. To appear.

Yevgeniy Vorobeychik and Michael P. Wellman. Mechanism design based on beliefs about responsive play. In *Workshop on Alternative Solution Concepts in Mechanism Design*, 2006.

Yevgeniy Vorobeychik and Michael P. Wellman. Stochastic search methods for Nash equilibrium approximation in simulation-based games. In *Seventh International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1055–1062, 2008.

Yevgeniy Vorobeychik, Christopher Kiekintveld, and Michael P. Wellman. Empirical mechanism design: Methods, with an application to a supply chain scenario. In *Seventh ACM Conference on Electronic Commerce*, pages 306–315, 2006.

Yevgeniy Vorobeychik, Daniel M. Reeves, and Michael P. Wellman. Constrained automated mechanism design for infinite games of incomplete information. In *Twenty-Third Conference on Uncertainty in Artificial Intelligence*, pages 400–407, 2007.

Yevgeniy Vorobeychik, Michael P. Wellman, and Satinder Singh. Learning payoff functions in infinite games. *Machine Learning*, 67(2):145–168, 2007.

William E. Walsh, Rajarshi Das, Gerald Tesauro, and Jeffrey O. Kephart. Analyzing complex strategic interactions in multi-agent systems. In *AAAI-02 Workshop on Game Theoretic and Decision Theoretic Agents*, 2002.

William E. Walsh, David C. Parkes, and Rajarshi Das. Choosing samples to compute heuristic-strategy Nash equilibrium. In *Agent Mediated Electronic Commerce*, 2003.

Michael P. Wellman, Peter R. Wurman, Kevin O'Malley, Roshan Bangera, Shou de Lin, Daniel M. Reeves, and William E. Walsh. The 2001 trading agent competition. *IEEE Internet Computing*, 5(2):43–51, 2001.

Michael P. Wellman, Joshua J. Estelle, Satinder Singh, Yevgeniy Vorobeychik, Christopher Kiekintveld, and Vishal Soni. Strategic interactions in a supply chain game. *Computational Intelligence*, 21(1):1–26, 2005.

Michael P. Wellman, Daniel M. Reeves, Kevin M. Lochner, and Rahul Suri. Searching for Walverine 2005. *Lecture Notes in Artificial Intelligence*, 3937:157–170, 2006.

David H. Wolpert. Predictive game theory. Working paper, 2008.