Thesis Proposal:

Approximate Game Theoretic Analysis of Large Simulation-Based Games

Bryce Wiedenbeck

March 25, 2014
Contents

1 Introduction ................................................. 4
   1.1 Game Theory Notation ......................................... 4
   1.2 Simulation-Based Game Theory .......................... 5
   1.3 Application Domains ........................................ 6
   1.4 Methodological Contributions .......................... 7

2 Deviation-Preserving Reduction ......................... 9
   2.1 Player Reductions ........................................... 9
   2.2 Deviation-Preserving Reduction ........................ 10
   2.3 Impact and Extensions ..................................... 11

3 Bootstrap Methods for Statistical Confidence .......... 11
   3.1 The Need for Statistics in Games .................... 11
   3.2 Bootstrap Confidence Intervals for Regret ............. 12
   3.3 Proposed Extensions ...................................... 13

4 Strategic Formation of Credit Networks ............... 14
   4.1 Credit Network Formation Model ...................... 14
   4.2 Theoretical Results ...................................... 15
   4.3 Simulation-Based Game Analysis ..................... 16
   4.4 Proposed Extensions ..................................... 17

5 Proposed Method: Learning Game Models ............... 18
   5.1 Related Work ............................................. 19
   5.2 Model and Evaluation Criteria ......................... 19
   5.3 Preliminary Work ......................................... 20

6 Proposed Application: Optimal Stopping Games ...... 22
   6.1 Theoretical Results ...................................... 23
      6.1.1 Even Number of Candidates ..................... 24
      6.1.2 Odd Number of Candidates ..................... 26
   6.2 Theoretical Extensions .................................. 27
   6.3 Simulation-Based Game .................................. 28

7 Research Timeline .......................................... 28
A Scaling Simulation-Based Game Analysis through DPR 33

B Bootstrap Statistics for Empirical Games 42

C Strategic Formation of Credit Networks 51
1 Introduction

Multi-agent systems research raises many interesting questions about the actions of independent but interacting decision-makers. Game theory helps to answer some of those questions by providing a mathematical language for representing and reasoning about the incentives of self-interested autonomous agents. As is often the case with mathematical modeling, creating tractable game-theoretic models frequently requires abstracting away many aspects of the system under investigation. Simulation-based game theory constructs richer game models using agent-based simulation, but still suffers from tradeoffs between tractability and accuracy. My research agenda focuses first on improving this tradeoff by developing techniques for simulation-based game theory to conduct better analysis with limited simulation data and second on applying these techniques to understand complex multi-agent interactions. My thesis will present novel methods that improve the tools available for simulation-based game theory and demonstrate the application of these tools to uncover new results in multi-agent systems of interest.

I begin by introducing notation and background material that will be helpful throughout the remainder of the proposal. I then discuss some of the work I have done so far, including two primary contributions to the methodology of simulation-based game theory and an application of simulation-based game theory to a strategic network formation domain. I conclude with proposals for upcoming work in both methods for and applications of simulation-based game theory.

1.1 Game Theory Notation

A symmetric\(^1\) normal form game \(\Gamma = (N, S, u)\) consists of an integer \(N\) specifying the number of players, an ordered set of strategies \(S\) common to all players, and a utility function \(u\). A pure-strategy profile is a vector \(\vec{s} \in \mathbb{N}_0^{|S|}\) specifying the number of players playing each strategy. The expression \(s \in \vec{s}\) indicates that at least one player plays strategy \(s\) in profile \(\vec{s}\). The expression \(\vec{s} - _s\) refers to an \((N - 1)\)-player profile where the count of strategy \(s \in \vec{s}\) has been decreased by one. The set of all pure-strategy profiles \(\vec{S} = \{\vec{s} \in \mathbb{N}_0^{|S|} : ||\vec{s}||_1 = N\}\) in \(\Gamma\) is known as the profile space and has size \(\binom{N+|S|-1}{N}\). The utility function \(u : S \times \vec{S} \to \mathbb{R}\) maps a strategy \(s\) and a strategy profile \(\vec{s}\) to a payoff \(u(s, \vec{s})\) achieved by a player choosing strategy \(s\) when the players jointly choose \(\vec{s}\). The standard tabular representation of the

\(^{1}\)I will primarily consider ex-ante symmetric simulation-based games where the players select strategies before any differentiation of types occurs. This assumption simplifies exposition and is necessary for some of my methods, but is not a strict requirement of simulation-based game theory. I try to point out places where generalization to non-symmetric games is possible.
utility function is called a payoff matrix, and stores $|S|^{(N+|S|-2)}$ payoff values.

A *mixed strategy* $\sigma$ is a discrete probability distribution over the strategies; all players playing according to $\sigma$ is called a symmetric mixed strategy profile. I will use $\vec{\sigma}$ to represent a pure strategy profile or a symmetric mixed-strategy profile. I will overload $u$ to accommodate symmetric mixed strategy profiles as follows: $u(s, \sigma)$ is the expected utility of an agent playing pure strategy $s$ when all others play mixed strategy $\sigma$, and $u(\sigma)$ is the expected utility of any agent when all agents play $\sigma$. I will also overload $u(s', \vec{s}_{-s})$ to be the utility to an agent playing $s'$ when the $N - 1$ other agents play according to $\vec{s}_{-s}$.

The regret $\epsilon(\vec{s}) = \max_{s \in \vec{s}} \max_{s' \in S} u(s', \vec{s}_{-s})$ of a pure-strategy profile is the maximum gain any player could achieve by deviating from its strategy $s$ to any other strategy $s'$ with other players’ strategies held constant. Similarly, the regret $\epsilon(\sigma) = \max_{s \in S} u(s, \sigma)$ of a symmetric mixed strategy profile is the maximum gain a player could achieve by deviating from the mixed strategy $\sigma$ to any pure strategy $s$. A profile $\vec{\sigma}$ is a *Nash equilibrium* if $\epsilon(\vec{\sigma}) = 0$ and an $\epsilon$-Nash equilibrium if $\epsilon(\vec{\sigma}) \leq \epsilon$. A pure strategy $s$ is *dominated* by pure strategy $s'$ if for all profiles $\vec{s}$ in which $s$ appears, $s'$ is a beneficial deviation: $u(s', \vec{s}_{-s}) > u(s, \vec{s})$ for all $\vec{s} \in \vec{S}$ where $s \in \vec{s}$. The *social welfare* of a profile $\vec{\sigma}$ is the sum of the expected utilities to all agents under that profile. The *price of anarchy* of a game is the ratio of the social welfare of the best pure-strategy profile $\vec{s}$ to that of the worst Nash equilibrium.

### 1.2 Simulation-Based Game Theory

Multi-agent interactions of interest may extend over long time periods with repeated interaction between agents and may involve uncertain outcomes or hidden information. Rather than reasoning explicitly about an extensive game model with incomplete or imperfect information, practitioners of simulation-based game theory employ simulations of agent interactions to construct a normal form game [33]. In a simulation-based game, each agent selects a strategy corresponding to a program that governs its behavior in the simulated environment. Agent interactions can then be simulated for various combinations of strategies to construct a payoff matrix.

The fundamental building block of a simulation-based game model is the *environment simulator*. The simulator takes as input a strategy profile $\vec{s}$, simulates those agent programs interacting, and outputs a payoff for each strategy. For a fixed number of agents $N$ and set of strategies $S$, the simulator defines a *true game*, with $u(s, \vec{s})$ equal to the expected payoff of $s$ when profile $\vec{s}$ is simulated. In practice, this true game is unavailable and must be approximated using a finite set of observations $\Theta$ from the simulator.
A simulation-based game \( \Gamma = \mathcal{M}(\Theta) \) is a normal form game resulting from the application of a game model function \( \mathcal{M} \) to an observation set. In the most basic case, all profiles are simulated repeatedly and \( \mathcal{M} \) simply calculates sample averages for every payoff. Game-theoretic analyses such as computation of Nash equilibria performed on the simulation-based game are treated approximating such analyses in the true game.

1.3 Application Domains

My thesis will present results from applying simulation-based game theory to study two multi-agent interaction domains. Both studies arise from attempts to create mathematical models of real-world multi-agent interactions. In both cases, relatively simple game models can capture some interesting dynamics, but fall short of satisfactory descriptions of agent behavior. Unfortunately, solving richer game models that incorporate important factors in agents’ decisions has proven intractable. Even in the cases where some equilibria can be identified, they have relied critically on simplifying assumptions that lack behavioral justification. I will discuss theoretical results in both domains, but my main focus will be on broader insights derived from simulation-based game models. These applications also help to motivate the methodological contributions discussed in Sections 2, 3, and 5.

Section 4 describes completed work and planned extensions on a strategic credit network formation problem. A credit network is a formal model of trust relationships that was invented independently by researchers studying several distinct phenomena. With my collaborators, I studied a game model describing how strategic agents would issue credit as a function of their expectations about future transactions and other agents’ trustworthiness. When we reached the limits of tractable analysis in our explicit game models, I extended our analysis using simulation-based game theory to explore the sorts of equilibrium credit networks that arise in a much richer model. Initial simulation experiments found star-like equilibrium networks where credit relationships resembled a central currency; these findings fed back into theoretical analysis characterizing dynamics that could lead to similar networks. Further simulation experiments explored a range of settings and showed how network structure depends on the availability of information and the relative costs and benefits of participation. Future experiments will investigate coexistence of credit networks and explicit central currencies.

Section 6 describes initial work on a theoretical model of an optimal stopping game. The optimal stopping game models hiring managers faced with a decision whether to make a job offer or interview more candidates, subject to the risk that a competitor will make an offer.
first. With my collaborators, I derived equilibria in a very restricted subclass of optimal stopping games. I propose to explore a much broader set of instances that come closer to modeling real-world interactions using simulation-based game theory. My expectation is that, much like with credit networks, this application of simulation-based game theory can both inspire new theoretical results and offer independently valuable insight into richer strategic interactions.

1.4 Methodological Contributions

The problems I study using simulation-based game theory raise several questions about general methods for simulation-based game theory that I will address in my thesis. In the credit network formation game, questions about the structure of equilibrium networks can only be realistically addressed by simulations involving a very large number of agents. Section 2 discusses a method I proposed for analyzing simulated environments with many agents that has already been employed in multiple domains. In Section 5, I propose additional research to improve these methods. In the theoretical analysis of optimal stopping games presented in Section 6, small payoff differences between strategies can have large effects on agent behavior. If the same holds true in my proposed simulation study, random variation in observed payoffs could interfere with analysis. I address this problem in Section 3 by introducing new statistical methods for simulation-based game theory.

The problem with simulations involving large numbers of agents arises from the growth of the normal form game representation. As mentioned above, a symmetric game $\Gamma = (N, S, u)$ has $|\vec{S}| = (N + |S| - 1)\frac{1}{N}$ distinct profiles. Thus $|\vec{S}|$ grows exponentially in the smaller of $N$ and $|S|$, making simulation of all profiles in the game prohibitively expensive. For a sense of how great a burden this imposes, consider that a symmetric game with 15 agents and 15 strategies contains over 77,000,000 profiles, so if all required simulations of a profile could be collected in one second, constructing the full game would take more than two years. The largest simulation-based game study to date is the work on credit network formation games that I present in Section 4, in which roughly 166,000 profiles were simulated. The second largest study, investigating a routing protocol [32] simulated roughly 72,000 profiles.

A partial solution to the growth of the normal form representation can be achieved by searching only for equilibria in which a small number of strategies are played. It is well known that most large games have small-support equilibria [22], and both of the aforementioned studies employed an iterative strategy-exploration procedure exploiting this fact. This procedure simulates profiles restricted to a subset of strategies $S' \subset S$ and unilateral deviations
to other strategies $s \in S \setminus S'$, expanding $S'$ only if no Nash equilibria are found. However, because both studies relied on simulations of a large number of agents (approximately 60 for the credit network and 5,000 for the routing protocol), constraining the strategy space is insufficient to make analysis tractable. Such large populations of agents are often required for simulations to produce the dynamics the game model is intended to study. For example, a credit network is of particular interest when it enables non-local transactions, but with a small number of agents, network distances are always short. Simulation-based game studies of financial markets (e.g. [2]) may similarly require a large number of agents to achieve desired trade volumes.

Researchers have studied games with a very large number of agents using a set of techniques known as player reduction. Player reductions employ a modeling function that maps payoff values from a full game with a large number of agents to payoff values in a reduced game with a much smaller number of players. Constructing a reduced game model requires simulating a number of profiles proportional to the size of the reduced game, which is generally a tiny fraction of the size of the full game. Equilibria computed in the reduced game are then treated as approximate equilibria of the full game. In Section 2 I discuss a novel player reduction method called deviation-preserving reduction (DPR). Relative to prior player reduction methods, DPR improves the fidelity with which the reduced game approximates the full game.

I also plan to more-directly investigate the problem of choosing a modeling function for large games. While deviation-preserving reduction beats prior player reduction methods experimentally, it leaves room for improvement in multiple areas. Simulations called for by DPR produce payoff data that is ignored when constructing reduced games. In addition, all player reduction methods call for payoff data about a fixed set of full-game profiles, leading experimenters to repeatedly sample those profiles to accurately estimate their payoffs. I expect that game model creation functions that can make use of a broader set of profiles and incorporate all available payoff data could substantially improve analysis of large simulation-based games. In Section 5, I frame this general modeling question as a machine learning problem that I propose to investigate.

As a variety of experimental research, simulation-based game theory would benefit greatly from the use of statistical methods, but applying traditional statistics to the conclusions drawn from simulation-based games is extremely complex. As a result, practical methods for statistical inference in simulation-based games have not previously been developed. I have developed a new method, described in Section 3, that applies bootstrap resampling
techniques to estimate the true-game regret of equilibria found in simulation-based games. This technique enables researchers to report the probability that $\epsilon(\vec{\sigma}) < \delta$ for some relevant threshold $\delta$, or to give a 95% confidence bound on $\epsilon(\vec{\sigma})$. Such bounds facilitate statistical reasoning about the robustness of conclusions drawn from simulation-based games and support sampling procedures that make better use of computational resources.

2 Deviation-Preserving Reduction

2.1 Player Reductions

Player reduction is a method for simulation-based game theory under which simulated environments with a large number of agents are represented by games with a small number of players. Player reduction is employed when the number of agents necessary for accurate or interesting simulations is large enough to make simulating all full-game profiles infeasible. Player reduction assumes that the variations in payoffs that govern strategic behavior can be summarized after simulating only a small subset of those profiles. Methods for player reduction specify a set of $N$-agent full-game profiles to simulate, and a modeling function that maps payoffs from those full-game profiles to payoffs in a reduced game with $n \ll N$ players.

Prior to my work, two player reduction methods had been proposed in the literature. The first, hierarchical reduction ($HR$) [30], preserves the fraction of players choosing each strategy between the full and reduced games. Under $HR$, the utility values for reduced-game profile $\vec{s}$ all derive from the full game profile $\frac{N}{n} \vec{s}$, where if $c$ players select strategy $s$ in the reduced-game $\frac{N}{n}c$ agents select strategy $s$ in the full game. Hierarchical reduction implicitly assumes that the payoffs in the game are smooth in the sense that small perturbations in the number of agents playing a particular strategy will not have large effects on agents’ payoffs. Sensitivity to non-smoothness can to some extent be controlled by varying the number of players in the reduced game.

The second reduction method, twins reduction ($TR$) [11], when applied to symmetric games, requires $n = 2$. Under $TR$, the utility to a player in the reduced game choosing $s_i$ when the opponent chooses $s_j$ is equal to the full-game payoff to an agent choosing $s_i$ when all $N - 1$ other full-game agents choose $s_j$. I show that twins reduction successfully identifies any symmetric pure strategy Nash equilibria in a symmetric game, but it makes a much stronger smoothness assumption: that payoffs to each strategy are linear in the fraction of agents adopting it. Worse still, $TR$ has no means of representing payoff interactions among
more than two strategies.

2.2 Deviation-Preserving Reduction

The paper included in Appendix A appeared in AAMAS 2012 [34]. In it, I proposed a method called deviation-preserving reduction (DPR), which combines the best aspects of HR and TR by preserving the fraction of opponents who play all other strategies between the full and reduced games. Under DPR, the utility to a player choosing strategy $s$ in reduced game profile $\vec{s}$ is the full-game payoff $u(s, \frac{N-1}{n-1}\vec{s}_s)$. That is, each reduced game player views itself as controlling one full game agent (as in twins reduction), while the opponents in the reduced game summarize the opponents in the full game (similar to hierarchical reduction). DPR inherits from TR the ability to identify symmetric pure-strategy Nash equilibria. It also inherits from HR the potential to adjust granularity by changing the size of the reduced game.

Under deviation-preserving reduction, each reduced-game player has a slightly different view of any given profile. The payoffs to reduced-game players choosing different strategies therefore derive from different full-game profiles. Whereas HR requires simulating $N + |S| - 1 \choose N$ full-game profiles (one for every reduced-game profile), DPR requires simulating $|S| (n^{|S| - 2} \choose n-1)$ (one for every reduced-game payoff); TR always simulates a fixed set of profiles. Because 2-player DPR games are equivalent to TR games, the relevant comparison is whether adding additional players improves approximation quality. Comparing DPR and HR is harder: it is generally not possible to choose full and reduced game sizes that give divisibility for both methods ($N \choose n$ for HR and $\frac{N-1}{n-1}$ for DPR) and also comparable numbers of profiles simulated. My experiments therefore use game sizes that make $n$ divide $N$ (advantaging HR), and compare DPR games to HR games that simulate at least as many profiles.

All three reduction methods can be arbitrarily inaccurate if the smoothness assumptions are violated: if an unobserved profile has exceptionally high or low payoffs, reduced-game equilibrium analysis could be completely wrong. It is also possible to construct game instances either where HR or DPR out-performs the other, and little information with which to classify games is generally known in advance of simulation. I therefore compare the various reduction methods empirically by generating random instances of various classes of 12-agent, 6-strategy true games and constructing 2, 3, 4, 5, and 6-player reduced games. Across experiments on congestion games, local effect games, and credit network games, DPR with more than two players significantly out-performed TR in terms of full-game regret of reduced game symmetric mixed strategy Nash equilibria. Despite an experimental setup that
heavily favored HR, DPR consistently gave lower-regret reduced game equilibria. For both HR and TR, there was a game class where strategies that were not dominated in the full game frequently appeared to be dominated in the reduced game, while DPR avoided such mistakes. See Appendix A for more details.

2.3 Impact and Extensions

Since the publication of the paper in Appendix A, deviation-preserving reduction has been implemented as a part of the EGTA-online experiment management system [3]. It has also been employed in the study of credit network formation games described in Section 4 and Appendix C. A study of high-frequency trading in equity markets that uses deviation-preserving reduction in a role-symmetric\(^2\) game is forthcoming [29].

Characterizing the efficacy of DPR under specific assumptions about full-game payoff distributions could be worthwhile if it sees continued use. In particular, formalizing the smoothness assumption that underlies HR and DPR might yield bounds on approximation quality or tests for when DPR is most likely to work. However, all three player reduction methods provide ad hoc answers to the question of how to create an approximate game model from limited simulation data. I argue in Section 5 that it should be feasible to improve on DPR by directly addressing this modeling problem.

3 Bootstrap Methods for Statistical Confidence

3.1 The Need for Statistics in Games

Virtually all published simulation-based game studies have been based on simulated environments with substantial randomness. As a result, it has been common practice to gather many samples of each simulated profile. Despite this, statistical methods for simulation-based game theory are underdeveloped, and the few tools that exist are rarely applied. Some researchers have used variance reduction techniques [17, 25], and many have gathered extremely large numbers of samples, but few studies have attempted to quantify uncertainty or estimate confidence in simulated models. As a consequence, published results rarely reflect the underlying uncertainty of simulation-based game models.

\(^2\)In a role-symmetric game, players are indistinguishable within roles (e.g., buyers and sellers), but players in different roles may have different strategy sets and utility functions. My paper [34] shows how to extend all three reduction methods to role-symmetric games.
My co-authors and I attempt to remedy this situation in a paper to appear in AAMAS 2014 [35], which is included in Appendix B. In it, I describe the first method to combine bootstrap statistics and game theory. My method performs a simultaneous bootstrap over all payoffs in a game to derive statistical confidence on the true-game regret of equilibria computed in simulation-based games. This should aid in reasoning about equilibria in simulation-based games, especially in identifying profiles that are most likely to have low regret in the true game. My co-authors also go on to show that these bootstrap regret estimates can be used effectively in determining how many samples of each profile in a game are required.

Two other methods of estimating true-game regret of simulation-based game equilibria have been proposed in the literature. Reeves [23] proposed estimating the empirical distribution of the regret of a profile by sampling game matrices from the space of possible matrices induced by assuming every payoff is independent and distributed normally with mean and variance equal to its sample mean and sample variance respectively. He recommended using the probability that a profile’s regret is zero from the estimated empirical distribution of regret as a measure of the confidence that a profile is a Nash equilibrium. Vorobeychik [28] presented a Bayesian framework for determining the posterior probability that a profile is an $\epsilon$-Nash equilibrium of the true game from payoff sample data. He provided one set of probability bounds under the assumption that payoff observations are independent draws with Gaussian noise, and much weaker distribution-free bounds. Both of these works rely strongly on distributional assumptions about simulation data, and neither has been empirically validated or implemented in practice.

### 3.2 Bootstrap Confidence Intervals for Regret

The bootstrap [8] treats a sample set as representative of the population from which it was drawn for the purpose of computing distributional statistics. To derive a sampling distribution for a statistic, the bootstrap resamples the sample set to simulate drawing many samples from the population. If the original sample has size $k$, then each resample is a set of size $k$ drawn with replacement from that sample. The statistic is then computed on each resample set, giving a bootstrap distribution for the statistic that can be used in place of a sampling distribution for computing confidence intervals.

My method to compute bootstrap confidence intervals on regret constructs a large number of bootstrap games by simultaneously resampling every payoff in the observation set $\Theta$. Each bootstrap game is based on a resampled observation set $\hat{\Theta}$, where for each payoff, a resample
set is drawn with replacement from its samples to have equal size. The bootstrap game $M(\hat{\Theta})$ is then constructed by applying the game model creation function to the resampled observations. The regret of a profile of interest can then be computed in every bootstrap game, giving a bootstrap distribution for the regret statistic. The 95th percentile of this distribution is then used as a 95% confidence bound on the true-game regret of the profile. Note that this method would be perfectly applicable to non-symmetric simulation-based games. Section 3.1 of Appendix B describes the method in greater detail.

Section 4.1 of Appendix B gives experimental evidence demonstrating the validity of bootstrap confidence intervals for regret. Briefly, the experiments involve a large set of true games: randomly generated instances of the congestion game, uniform symmetric game, or credit network formation game classes, of varying numbers of players and strategies. For the congestion and uniform symmetric games, I emulated a simulator by adding artificial noise drawn from normal, uniform, multi-modal, or skewed distributions with widely varying magnitudes. For the credit network games, I treated a very large sample set as the true game and randomly subsampled that set to create each experimental $\Theta$. For all game classes, I then computed equilibria in simulation-based games constructed from the noisy data and compared the regret of those equilibria in the corresponding true games to the bootstrap distribution estimates. In congestion and uniform symmetric games, the bootstrap distribution proved to be a good estimator of true game regret. Results in credit network games were weaker, but I attribute this at least in part to the fact that I used only one true game across all experiments. The evidence supporting my method that appears in the conference paper was limited by available space; my thesis will provide data on additional experiments that the paper mentioned but did not show.

My co-authors went on to provide methods for sample control in simulation-based games based on bootstrap regret estimates. Section 3.2 of Appendix B describes the sample control algorithms, and section 4.2 gives experimental evidence for their validity, especially compared to pre-existing rules of thumb.

### 3.3 Proposed Extensions

Several interesting questions relating to bootstrap methods for simulation-based games remain open, and I intend to address at least two of them in my thesis. The first such question is whether my bootstrap method can be applied to other game-theoretic statistics of interest. Many simulation-based game studies have reported and drawn important conclusions from the social welfare of equilibria or other profiles of a game. I would like to show that valid con-
fidence intervals for social welfare and possibly an empirical notion of price of anarchy can be
derived using bootstrap methods. The second question is how much better distribution-free
bootstrap statistics perform than the methods that have been proposed previously. Such
an investigation could involve a theoretical treatment of the bootstrap method, and/or an
empirical evaluation of bounds offered by Reeves [23] and Vorobeychik [28].

4 Strategic Formation of Credit Networks

4.1 Credit Network Formation Model

A credit network is a formal model of trust among agents. In a credit network, nodes
represent agents, and edges represent credit relationships between them. A directed edge
with weight $c$ from nodes $i$ to node $j$ indicates that agent $i$ issues $c$ units of credit to agent $j$,
or equivalently, agent $i$ is willing to accept I.O.U.s from agent $j$ up to value $c$. The existence
of an edge from agent $i$ to $j$ allows agent $j$ to purchase goods or services from agent $i$ on
credit. To purchase something of price $p$ from $i$, agent $j$ issues an I.O.U. of value $p$, which
reduces the credit remaining on edge $(i,j)$ to $c - p$. However, because agent $j$ will accept
that I.O.U. back as payment from agent $i$, the transaction increases the capacity of edge
$(j,i)$ by an equivalent amount.

Of particular interest, and the reason for representing credit relationships as a network,
is the potential for long-distance transactions. Consider a the case illustrated in Figure 1,
where agent $k$ wishes to make a purchase at price $p$ from agent $i$. Agent $k$ has no credit
from agent $i$ and therefore cannot make the purchase directly, but due to the presence in
the network of a trusted intermediary, agent $j$, the two can transact. To complete the
transaction, agent $k$ issues an I.O.U. for $p$ units to agent $j$, and agent $j$ issues an I.O.U. for
$p$ units to agent $i$. As long as the initial capacities $c_{ij}$ and $c_{jk}$ are at least $p$, the transaction
succeeds. More generally, a transaction is possible in a credit network as long as the max-
flow from the seller to the buyer meets or exceeds the price. Note that after the transaction,
agent $j$ holds an I.O.U. from agent $k$, and agent $i$ holds an I.O.U. from agent $j$; even long-
distance transactions cause agents to hold I.O.U.s only from the neighbors to whom they
have issued credit. Note also that the total amount of credit available to the intermediary
remains unchanged; agent $j$ has only exchanged credit from agent $i$ for credit from agent $k$.

The credit network model has arisen independently in a number of applications; the
following list is non-exhaustive. DeFigueiredo and Barr [9] used credit networks to track
promises of collateral backing transactions in eBay-style auctions. Ghosh et al. [13] consider
the mechanism design problem of conducting multi-unit auctions when payments must be routed through a credit network. Mislove et al. [20] propose a variant on the classic idea of deterring spam by imposing a small cost on senders that uses credit networks to denominate willingness to accept messages. Karlan et al. [18] model several types of trust-based social interactions such as job recommendations as mediated through an implicit credit network. In addition, a few researchers have studied general properties of credit networks. Dandekar et al. [6] showed that decentralized credit networks have high liquidity and can support almost the same set of transactions as a centralized currency. Viswanath et al. [26] demonstrated that a large number of reputation systems designed for sybil-tolerance are in fact variations on credit networks.

My work, in collaboration with Pranav Dandekar, Ashish Goel, and Michael Wellman, studies strategic formation of credit networks. Our work has been published in the WWW 2012 conference [7], and has subsequently been updated and submitted for journal publication; the most recent version of our paper is included in Appendix C. This research investigates what criteria are important to strategic agents when deciding to issue credit, and what sorts of credit networks can arise in equilibrium when agents issue credit strategically. Issuing credit increases general liquidity in the network and increases the connectivity of the issuing agent, both of which potentially enable profitable transactions, but also entails risk that agents receiving credit will use it to make purchases and then default. The decision to issue credit is modeled as a one-shot game in which agents determine initial allocations of credit. Afterwards some agents randomly default and the remaining agents engage in repeated probabilistic transactions.

4.2 Theoretical Results

My co-authors began by studying an extremely simple instance of the credit network formation game, in which each agent is constrained to issue a fixed credit budget among a small set of neighbors and then transact only with those neighbors. In this model, agents never default, transaction probabilities are symmetric, and all transactions are of unit size.
Section 3.1 of Appendix C analyzes this setting, demonstrating that under these restrictions, the credit network formation game is a potential game in which all equilibria maximize social welfare and are equivalent in the set of transactions they support. Unfortunately, even slight extensions to this model lose these properties. In section 3.2 we lift the restriction that transactions happen exclusively between neighbors, and I demonstrate that equilibria may not exist (Theorem 3.8) and that the price of anarchy is unbounded (Theorem 3.9).

My initial experiments on a much less restrictive simulation-based credit network game model found that equilibrium networks were commonly empty, with no credit issued, or star-like, with a small number of the agents least likely to default receiving credit from everyone. This result inspired my co-authors to look for similar equilibria in a slightly richer version of the theoretical model. In Section 4 of Appendix C they consider a model in which agents have commonly known probabilities of defaulting, transactions between any pair of agents are equally likely, and each agent may issue credit to only one other. In this setting, they show that an empty network can arise in equilibrium when a star network would be socially optimal, giving unbounded price of anarchy. They also show that if agents make sequential credit issuing decisions, equilibria have a comb-like network structure.

4.3 Simulation-Based Game Analysis

Less restricted variants of the credit network formation game have frustrated our attempts at theoretical analysis. I have therefore employed simulation-based game theory to tackle more general scenarios. In the simulation-based model, 61 agents simultaneously choose how much credit to issue to each other agent. All transactions have cost and price of one unit, but the buyer has a randomly drawn valuation $v \geq 1$. Each agent has a default probability drawn from a known distribution, and each ordered pair of agents has a probability of transacting. After agents issue credit, some agents are randomly selected to exhaust all available credit and default, and remaining agents trade until 10,000 transactions have been attempted. The 61-agent simulations are used to construct a 6-player game by deviation-preserving reduction.

Across experiments, the distributions of buyer value and default probability were varied, as was the information available about default probabilities. Buyer values were drawn from one of two distributions, with average values of 1.1 (low surplus) and 1.5 (high surplus). Default probabilities were drawn from one of three distributions with average values of $\frac{1}{10}$ (low default), $\frac{1}{3}$ (medium default), and $\frac{1}{2}$ (high default). There were also two models of information about default probabilities. In the first (global risk), all agents’ default probabilities were common knowledge. In the second (graded risk), an Erdős-Rényi social network graph
was generated, and information about default probability decayed with distance in the social
network.

Simulated heuristic strategies included issuing no credit, and issuing $q$ units of credit to
each of the best $k$ other agents either randomly or according to some criterion. The criteria
for ranking other agents included: lowest (estimated) default probability, highest probability
of transacting as a buyer, highest probability of transacting as a seller, buy expected value
of attempted purchases, highest net expected value of attempted transactions, and lowest
index in an arbitrary (common) ordering. Note that ranking by default probability gives
very different outcomes in the two information settings: under global risk, all agents have
the same default ranking, while under graded risk, agents will generally issue credit to those
neighbors in the social network about whom they get positive signals.

In all low surplus experiments and all high default experiments, an equilibrium arose
where agents issued no credit. Empty networks support no transactions, resulting in low
social welfare; however in most of the environments considered, other equilibria exist. All
global risk settings included a Nash equilibrium where all agents issued credit according to the
same default probability heuristic. These equilibria result in star-like networks with multiple
central nodes to whom all other agents issue credit. Credit from these nodes effectively acts
as a central currency, with most transactions happening over paths of length two. Central
currencies give relatively high social welfare because many transactions are feasible and
costly defaults are rare. In graded risk experiments, neither the index- or default-based
strategies arise in equilibrium, indicating that the central currency outcome depends both
on coordination and on picking agents that are unlikely to default. Only in high surplus
and low default settings did heuristics involving transaction probabilities or values appear
in equilibrium. See Appendix C for further discussion of equilibria and social welfare in
simulation-based credit network formation games.

### 4.4 Proposed Extensions

In addition to revising the journal draft in Appendix C, I plan to conduct a few additional
experiments on the credit network model. In many of the domains that have been modeled
with credit networks, de jure central currencies are unavailable, but the experiments on the
global risk domain indicate that de facto central currencies could arise anyway. I would
like to investigate the follow-up question of what sorts of credit networks could arise on
top of a de jure central currency. Many real-world instances of credit issuing arise due to
liquidity constraints or time-shifting of purchases. Karlan et al. [18] even provide instances of
transactions between businesses along multi-step credit paths. I plan to conduct experiments that include an explicit bank that all agents trust and look for circumstances where agents still choose to issue credit. If I can find such scenarios, I will investigate the resulting credit networks and the incentives that lead to private credit relationships.

5 Proposed Method: Learning Game Models

One of the most basic methodological questions of simulation-based game theory is how to translate a simulator into a game whose equilibria are predictive of agent behavior. This translation process can be broken into two main steps: first selecting a set Θ of samples to request from the simulator, and second constructing a game model \( \mathcal{M}(\Theta) \) from the resulting observations. As described in Section 3, the bootstrap methods I have studied can apply to the first step, helping decide when to stop adding additional samples to Θ. Deviation-preserving reduction addresses both questions: prescribing a collection of full-game profiles to simulate and a procedure for constructing a reduced game from them. DPR improves on previous player reduction methods, but is clearly not an optimal solution either for selecting observations or for constructing a game model.

To see the shortcomings of DPR, first consider how it creates game models. When a profile \( \vec{s} \) is simulated, the simulator outputs a payoff value for every strategy \( s \in \vec{s} \). However, because each reduced-game payoff comes from a distinct full-game profile, DPR uses the payoff value from only one strategy of each simulated profile. This means that \(|S|\left(\begin{array}{c} n+|S|-2 \\ n-1 \end{array}\right)\) full-game profiles are simulated to construct an \( n \)-player reduced game with \( \left(\begin{array}{c} n+|S|-1 \\ n \end{array}\right) \) profiles. In the credit network study discussed in Section 4, approximately two thirds of all payoff data in Θ was ignored by \( \mathcal{M}(\Theta) \). A method for constructing a game model that makes use of all the data that is collected should perform no worse than DPR, and potentially much better.

There is also obvious potential to improve on the observation set Θ generated by DPR. Like other player reductions, DPR provides a mapping from full-game profiles to reduced-game profiles. To accurately estimate payoffs, the relevant full-game profiles are simulated repeatedly, while other full-game profiles are ignored. As more samples of a single profile are collected, the returns to additional samples of that profile diminish. If \( \mathcal{M} \) could make use of a broader set of observations, simulation resources could probably be allocated more effectively by taking fewer samples of a larger set of profiles.

In this section I describe a proposed method for learning payoff functions in symmetric games with a large number of players. The key insight I hope to exploit is that in symmetric
games, representing profiles as vectors of strategy counts provides sufficient structure to \( \vec{S} \) to support regression methods that learn utility functions. With large \( N \), there should be enough data available to apply machine learning. The primary goal of my proposed research is to find regression methods appropriate to the task of learning payoff functions for each strategy in a game from simulation data. The natural dimension of the data is \( |S| \), so appropriate methods may need to accommodate high-dimensional data. Further, any game model that is learned must be amenable to tractable game analysis: in particular, for computing symmetric mixed-strategy Nash equilibria, it must enable computation of expected values of symmetric mixed strategies. Because the full-game payoff matrix is generally too large to represent, the expected value of a symmetric mixed strategy must be computable in the model without enumerating all full-game profiles. New solutions to the modeling problem may also suggest or require improvements to the selection of profiles to simulate.

5.1 Related Work

The question of how to construct game models and its implications for sample control have been addressed previously. Of particular note, Jordan and Wellman [16] formalized a general problem of fitting a game model to an observation set and used it to answer questions relating to strategy design. The work of Vorobeychik et al. [27] is closest among prior research to what I propose. These authors test several regression methods for learning payoff functions over a strategy space parameterized by a single continuous variable. Their methods can extend to multi-dimensional inputs, but rely fundamentally on the assumption of continuity in the strategy space.

Both of these studies focus on regularities in the strategy space of simulation-based games, but in many applications, generalizing across strategies may not be feasible. In most simulation-based game studies, strategies are treated as black boxes during game model creation and analysis [7, 31, 32]. This has held true even in cases where the underlying programs have been related by a small number of continuous parameters. Other work on machine learning in games has focused on detecting graphical independence structure in observational data [10, 14].

5.2 Model and Evaluation Criteria

The objective of this project is to devise a new game model creation function for approximating true games with a very large number of agents. The important criteria for the
simulation-based game $\mathcal{M}(\Theta)$ are first that it enable tractable equilibrium analysis, and second that its equilibria are reasonable approximations of true-game equilibria. I propose to achieve these objectives by learning utility functions $u^*(\cdot) = u(s, \cdot)$ for each strategy $s \in S$. Each $u^*$ will the output of a a regression over the payoff data for strategy $s$. The domain of each regression is the profile space $\vec{S} = \{\vec{s} \in \mathbb{Z}_+^{\vert S \vert} : \|\vec{s}\|_1 = N\}$, which has dimension $\vert S \vert$. Together, the $u^*$ functions constitute a utility function of a simulation-based game.

Note that in contrast to the regression-based game models of Vorobeychik et al. [27], this approach treats strategies as black boxes, learning each strategy’s utility function independently and representing each strategy’s count with a separate input dimension. This black box approach is consistent with common practice of simulation-based game theory: both the credit network and routing protocol studies discussed in Section 1 include a search to explore a parameterized strategy space, but disregard information about common parameters between strategies categorically when performing game-theoretic analysis. Because the dimension of the problem grows with $\vert S \vert$, this approach is likely to scale poorly to large numbers of strategies. Iterative exploration of the strategy space, as discussed in Section 1.4, should ameliorate the problem.

I plan to evaluate various regression methods based on the quality of equilibrium analysis on the resulting game models. As in previous studies, the most important evaluation criterion is the true-game regret of equilibria computed under each candidate game model. Finding Nash equilibria requires computing expected values of mixed-strategy profiles, which depend on the payoffs to all profiles that occur with positive probability. If it were possible to represent all such payoffs, regression would often be unnecessary, so approximate expected values should somehow be computed without enumerating payoffs for all profiles. This could be achieved using appropriately re-weighted sums over estimated payoffs, or by approximating the discrete payoff function with a continuous one. Player reduction can be viewed as a special case of the first of these methods.

5.3 Preliminary Work

I have conducted two preliminary experiments, which have ruled out some of the simplest approaches to the modeling problem. Both of these experiments employed the same data set of 12-agent, 6-strategy games with no noise on which deviation-preserving reduction was tested. This data set provides a non-ideal test of modeling functions for multiple reasons. First, the number of players is not large enough to necessitate approximate modeling, meaning that methods relying on large numbers of agents may underperform. And second, the
lack of noise in the data could cause over-valuation of methods like player reduction that require accurate estimates of a small number of profiles. In future experiments, I intend to test on a data set more like the one used to evaluate the bootstrap methods in Section 3.

The first approach I have tried simply performs expected value computations ignoring profiles for which Θ has no data. This method approximates $E(u(s, σ)) = \sum_{\vec{s} \in \vec{S}} pr(\vec{s} | σ)u(s, \vec{s})$ by the same sum over just the profiles that have been observed: $\sum_{\vec{s} \in Θ} pr(\vec{s} | σ)u(s, \vec{s})$. To see why this approach might be intuitively plausible, consider gathering the same Θ required by hierarchical reduction. In this case, computing expected values under this model sums over the same payoffs used in computing an expected value in the reduced game, but uses weights corresponding to full-game probabilities rather than reduced game probabilities (in the reduced game, extreme profiles like all agents choosing the same strategy are far more probable). If Θ corresponds to the profiles required by deviation-preserving reduction, then this method makes use of the payoffs that DPR would ignore. Further, if Θ has data for most of the profiles in $\vec{S}$, then the only reasonable justification for more complicated regression methods would be noise reduction.

In initial experiments this straightforward approach performed poorly. Using the same data as hierarchical reduction, it found equilibrium candidates with higher average true game regret than the equilibria of the the hierarchically-reduced game. Comparison against deviation-preserving reduction produced the same outcome. The only moderate success of this method came when comparing it on the n-player DPR observation set to the n-player HR game, which gives it the unfair advantage of seeing many additional profiles.

The second approach I have tried involves filling in payoffs for missing profiles by locally-weighted linear regression, with weights decreasing in the deviation distance between profiles. This method does not achieve the compactness goal: straightforwardly computing expected values from the locally-linear regression requires instantiating every value in the payoff matrix. Worse still, under local regression, the estimate for $u(s, \vec{s})$ depends on every other profile in which $s$ appears. Despite this complexity, in initial experiments, locally weighted regression did not out-perform deviation-preserving reduction. However, the regret of equilibria computed in this model were much better than those from the first model, indicating that other regression methods, especially when working with bigger games and more-varied data sets, could perform well.

In future work, I plan to explore a number of regression methods in search of one or more that consistently outperform player reduction. I will pay particular attention to ensuring that Nash equilibria can be computed in time that grows with the number of simulations.
performed, not with the size of the full game. I also want to develop a more appropriate test suite, in which known structure can be exploited to compute regrets in very large games, but that structure is unknown to the regression methods being tested.

6 Proposed Application: Optimal Stopping Games

In collaboration with Jacob Baron and Yevgeniy Vorobeychik, I have done preliminary work studying optimal stopping games. These games are inspired by a classic problem [12], in which a manager interviews candidates from a finite set $C$ about whom she has no prior information. After each interview the manager knows the relative rank of all candidates seen so far and has the option to either hire the current candidate and stop or reject the current candidate and continue interviewing. The manager receives utility 1 if the single best candidate is hired or 0 otherwise, and therefore wishes to maximize the probability of hiring that candidate. The optimal policy for the manager is to interview and deterministically reject roughly $\frac{|C|}{e}$ candidates and then hire the next feasible candidate (one who is the best seen so far). Many variations on this single-manager problem have been proposed and studied, including richer utility functions [1], more-informative observations [21], costly interviews [19], and multiple hires [4].

The extension to multiple competing managers is known as the optimal stopping game, and has been studied by a number of groups [5, 15, 24]. Existing research on optimal stopping games provides a strategic dimension not present in the single-manager problem, but has preserved a number of modeling choices that make less sense in a multi-manager setting. For instance, the single-manager model requires that hiring decisions be made immediately following an interview. The justification for this requirement is that candidates will receive other offers and become unavailable. With multiple managers, pressure to hire immediately should arise strategically, yet prior work imposes it exogenously. In addition, with one manager and unknown candidates, the order of interviews is irrelevant and can be fixed in advance. With multiple managers, information may be gleaned from hiring decisions by other managers, and could lead to strategic reasons to alter the order of interviews, but prior research holds interview order fixed.

My collaborators formulated an alternative optimal stopping game that endogenizes hiring pressure and interview order. In the model I consider, $N$ managers choose candidates to interview from a common pool $C$ about which they have no prior information. The managers take turns in round-robin order, and on each turn the current manager may choose to
interview a candidate and learn the candidate’s rank relative to others she has interviewed. Managers observe which candidates other managers have interviewed, and whether those candidates were hired. After each interview the manager may hire or reject the current candidate, however rejection is not permanent. On her turn, a manager may, instead of choosing a new candidate to interview, hire a candidate that she has interviewed previously, as long as that candidate has not already been hired by another manager.

My collaborators and I have solved an extremely simple version of the game that adds the smallest reasonable extension to the classic problem: $N = 2$ managers who observe the relative rank of candidates they interview and receive utility 1 upon hiring the best candidate or 0 otherwise. In Section 6.1, I include our proof that with an even number of candidates, both players have probability $\frac{1}{2}$ of hiring the best candidate in equilibrium, while having an odd number of candidates gives the first mover an advantage, allowing her to hire the best candidate with probability slightly greater than $\frac{1}{2}$.

These results do not provide substantial new insight into how hiring decisions should be made, but solutions to a richer model could. In Section 6.2, I discuss some preliminary ideas for how to solve slightly more complicated models. In Section 6.3, I propose a simulation environment in which to study much more realistic scenarios. I hope that as with credit network formation, traditional and simulation-based game theory can complement one another and give insight into the decision processes modeled by the optimal stopping game.

### 6.1 Theoretical Results

In this section, I consider a simple optimal stopping game in which managers $M_0$ and $M_1$ alternate turns, indexed by $t \in \{0, 1, \ldots\}$; beginning with $M_0$. Letting $\tau = \text{mod} \ (t, 2)$, on turn $t$ manager $M_\tau$ moves. Each candidate $c \in C$ has a (hidden) value $v(c) \sim U[0, 1]$. Let $I_0, I_1 \subseteq C$ be the sets of candidates interviewed by each manager. Both sets are initially empty, and their contents and histories are always common knowledge. In addition, manager $M_i$ always knows $\text{arg max}_{c \in I_i} v(c)$. For each manager, the set $c(M_i)$ is empty until she exits the game; afterwards $c(M_i)$ contains the candidate claimed by $M_i$.

On turn $t$, manager $M_\tau$ generally has two available actions: interview and hire. $M_\tau$ can choose interview as long as $I_\tau \neq C$; if she does she must then select $c^t$, an unclaimed candidate she has not previously interviewed: $c^t \in C \setminus I_\tau \setminus c(M_{1-\tau})$. The interviewed candidate $c^t$ is immediately added to $I_\tau$, and $M_\tau$ learns immediately whether $c^t$ is the best candidate seen so far. If $c^t$ is in fact the best, $M_\tau$ must immediately choose accept or reject. If she chooses accept, she exits the game, setting $c(M_\tau) = \{c^t\}$. If she chooses reject, the she
remains in the game. \( M_r \) can choose hire as long as \( I_r \neq \emptyset \); if she does, she exits the game and claims the best unclaimed candidate she has seen: \( c(M_r) = \{ \arg\max_{c \in I_r \setminus c(M_{1-r})} v(c) \} \). After manager \( M_i \) exits all her subsequent turns are skipped. The game concludes when both managers have exited. Letting \( c^* = \arg\max_{c \in C} v(c) \), the utility to manager \( M_i \) is 1 (she wins) if \( c^* \in c(M_i) \) and 0 (she loses) otherwise.

### 6.1.1 Even Number of Candidates

**Proposition 6.1.** If \(|C|\) is even then the value of the game is \( \frac{1}{2} \).

**Proof.** Consider the following strategy for \( M_0 \):

On turns \( t < |C| \), choose interview, selecting an arbitrary candidate, and always reject. On turns \( t \geq |C| \), if \( M_1 \) has not exited or \( I_0 \supseteq C \setminus c(M_1) \) then choose hire; otherwise, interview an arbitrary candidate and reject.

This strategy has expected utility at least \( \frac{1}{2} \); to see this, consider the strategy’s two cases. If \( M_1 \) exits first, \( M_0 \) wins with probability \( \frac{|I_1|}{|C|} \geq \frac{1}{2} \), because if \( t \leq |C| \) then \( |I_1| \leq \frac{|C|}{2} \), so \( M_1 \) claims the best of at most half of the candidates, while \( M_0 \) claims the best of the rest. If \( M_0 \) exits first, \( M_0 \) wins with probability exactly \( \frac{1}{2} \), because \( |I_0| = \frac{|C|}{2} \), so \( M_0 \) claims the best out of half the candidates.

It remains to show that \( M_1 \) can guarantee a \( \frac{1}{2} \) probability of winning. Consider the following strategy for \( M_1 \):

On turns \( t < |C| \), choose interview, select the unique candidate \( c \in I_0 \setminus I_1 \), and reject. On turns \( t > |C| \), if \( M_0 \) has not exited or \( I_1 \supseteq C \setminus c(M_0) \) then choose hire; otherwise, interview an arbitrary candidate and reject.

Showing that \( M_1 \)’s strategy guarantees expected utility \( \frac{1}{2} \) is analogous to the argument for \( M_0 \) above. \( \square \)

That the value of the game is \( \frac{1}{2} \) is not particularly surprising, but the strategies that achieve this outcome might be. Even though the players are not constrained to follow the same interview order, in equilibrium they do. It is in fact possible to show that this equilibrium is unique.

**Lemma 6.1.** If on turn \( t = |C| \), \( I_0 \neq I_1 \) then \( M_0 \) wins with probability greater than \( \frac{1}{2} \).

**Proof.** Consider the following partial strategy for \( M_0 \):
On turn \( t = |C| \), choose interview, selecting a candidate \( c \in I_1 \setminus I_0 \). If \( c \) is feasible, choose accept. On turns \( t > |C| \), if \( M_1 \) has not exited or \( I_0 \supseteq C \setminus c(M_1) \) then choose hire; otherwise, interview an arbitrary candidate and reject.

Consider again the two cases. If \( M_0 \) exits first, she gets expected utility \( \frac{1}{2} + \frac{1}{|C|} \). If \( M_1 \) exits first, \( M_0 \)'s payoff depends on the actions taken by \( M_1 \) at turn \( t = |C| + 1 \), for which there are three possibilities:

1. \( M_1 \) could choose hire
2. \( M_1 \) could interview \( c \in C \setminus I_0 \setminus I_1 \)
3. \( M_1 \) could interview \( c \in I_0 \setminus I_1 \)

In case 1, \( M_0 \) wins if \( c^* \in C \setminus I_1 \), which has size \( \frac{|C|}{2} \) or if she interviewed \( c^* \) on turn \( t = |C| \). In case 2, \( M_0 \) wins if \( c^* \in I_0 \) on turn \( t = |C| + 1 \), at which point \( |I_0| = \frac{|C|}{2} + 1 \). In case 3, \( M_0 \) wins if
   a. she interviewed \( c^* \) on turn \( t = |C| \),
   b. \( c^* \in I_0 \), but is not the candidate interviewed by \( M_1 \) on turn \( t = |C| + 1 \), or
   c. \( M_1 \) chose accept on turn \( t = |C| + 1 \), but at the time \( c^* \in C \setminus I_0 \setminus I_1 \).

Cases a–c are disjoint, and cases a and b have total probability \( \frac{1}{2} \), while case c has positive probability. Thus regardless of \( M_1 \)'s response, \( M_0 \)'s strategy wins more than half of the time.

If \( M_1 \) ever fails to interview the same candidate that \( M_0 \) just interviewed, it is possible for \( M_0 \) to force \( I_0 \neq I_1 \) on turn \( t = |C| \), which means that \( M_1 \) must “follow-the-leader” in equilibrium. Note that in equilibrium, neither manager will ever choose the action accept, but lemma 6.1 requires it. The presence of the accept action serves to enforce follow-the-leader interviews.

Note also that the partial strategy for \( M_0 \) given in lemma 6.1 is not optimal in the event that \( M_1 \) rejects \( \arg \max_{c \in I_0} v(c) \) on turn \( t = |C| + 1 \), because \( M_0 \)'s favorite candidate has been proven infeasible. However, this can be easily remedied by allowing \( M_0 \) to interview again on turn \( t = |C| + 2 \). This modification can only increase the payoff to \( M_0 \) and therefore doesn’t change the uniqueness result: \( M_1 \) should avoid this situation altogether by following \( M_0 \)'s interviews.
6.1.2 Odd Number of Candidates

With an odd number of candidates, clearly either manager can guarantee expected utility \( \frac{|C|-1}{2|C|} \) using the following strategy:

On turn \( t < |C| \), choose interview, selecting an arbitrary candidate and reject. On turns \( t \geq |C| - 1 \), if \( M_{1-r} \) has not exited or \( I_t \supseteq C \setminus c(M_{1-r}) \) then choose hire; otherwise, interview an arbitrary candidate and reject.

The question then is how the remaining \( \frac{1}{|C|} \) “surplus” over what either player can guarantee gets split. If both players follow the above strategy, \( M_0 \) will hire first, having interviewed \( \frac{|C|-1}{2|C|} \) candidates, leaving the entire surplus to \( M_1 \). This is clearly not an equilibrium because \( M_0 \) has an obvious beneficial deviation to interview one more candidate before she considers choosing hire. The following proposition shows that in equilibrium, \( M_0 \) is able to capture nearly, but not quite, all of the surplus.

**Proposition 6.2.** If \( |C| \mod 2 = 1 \), then the value of the game is \( \frac{|C|-1}{2|C|} + \left( \frac{1}{|C|} \right) \left( \frac{|C|-1}{|C|+1} \right) \).

**Proof.** Consider the following pair of strategies: \( M_0 \) plays

On turns \( t \leq |C| \), choose interview and reject. If \( I_1 \subseteq I_0 \), then select an arbitrary candidate; otherwise select the unique \( c \in I_1 \setminus I_0 \). On turns \( t > |C| \), if \( M_1 \) has not exited or \( I_0 \supseteq C \setminus c(M_1) \) then choose hire; otherwise, interview an arbitrary candidate and reject.

while \( M_1 \) plays

On turns \( t < |C| \), choose interview selecting a candidate \( c \in C \setminus I_0 \setminus I_1 \) and reject. On turn \( t = |C| \), choose interview selecting a candidate \( c \in I_0 \setminus I_1 \) and accept if feasible. On turn \( t > |C| \), if \( M_0 \) has not exited or \( I_1 \supseteq C \setminus c(M_0) \) then hire; otherwise, interview an arbitrary candidate and reject.

Under these strategies, \( M_0 \) wins if on turn \( t = |C| \), \( c^* \in I_0 \) but is not the candidate \( c' \) that \( M_1 \) has the chance to claim on turn \( t \), which occurs with probability \( \frac{|C|-1}{2|C|} \). \( M_0 \) additionally wins in the disjoint event that \( M_1 \) erroneously claims \( c' \), which happens when \( c' = \arg \max_{c \in I_1} v(c) \) (probability \( \frac{2}{|C|+1} \)), but \( c' \neq c^* \) (probability \( \frac{|C|-1}{2|C|} \)). Thus the expected utility to \( M_0 \) under the proposed pair of strategies is \( \frac{|C|-1}{2|C|} + \left( \frac{1}{|C|} \right) \left( \frac{|C|-1}{|C|+1} \right) \).

It remains to show that these strategies are mutual best-responses. Consider first a deviation by \( M_0 \). If she chooses hire on any turn \( t < |C| \), then \( |I_0| \leq \frac{|C|-1}{2} \), so she wins with
probability at most $\frac{|C|-1}{2|C|}$. If she allows $M_1$ to choose hire first, $M_1$, then $|I| = \frac{|C|+1}{2}$, so $M_0$ wins with probability $\frac{|C|-1}{2|C|}$. $M_0$ could also choose accept (if feasible) on turn $t = |C| - 1$, but doing so offers no benefit, because $M_1$’s strategy will never claim this candidate. There is also no benefit to altering the order of interviews: because $M_1$ selects a candidate $c \notin I_0$ on all turns $t < |C|$, it will always be possible for her to select a candidate $c \in I_0$ on turn $t = |C|$, thereby capturing a $\frac{2}{|C|+1}$-fraction of the surplus.

The case of a deviation by $M_1$ is much simpler. Given $M_0$’s strategy, the only way that $M_1$ can capture any of the surplus is to choose interview on turn $t = |C|$, selecting a candidate $c^t \in I_0$ and choose accept if feasible.

Having an odd number of candidates results in several interesting differences from the even case. Most importantly, there is a first-mover advantage. In addition, the ability to choose accept is payoff-relevant: without it, $M_0$ would capture the entire surplus. Follow-the-leader is no longer optimal for $M_1$, but the managers are indifferent among all other interview orders. Finally, $M_0$ is indifferent between choosing accept on round $t = |C| - 1$ and waiting to choose hire on round $t = |C| + 1$, but this is not the case under all interview orders.

6.2 Theoretical Extensions

Several extensions to the simple model in Section 6.1 may be amenable to similar analysis methods. My collaborators and I have made some progress on the following 1 variations:

- utility functions where each manager gets $v(c)$ for the candidate she claims,
- allowing managers to see $v(c)$, not just relative rank, and
- $N > 2$ managers.

The most important effect of giving managers utility $v(c)$ is that they are willing to hire much later because the cost of having the opponent hire $\arg\max_{c \in I} v(c)$ is much smaller. The gain from hunting for the single best candidate is also smaller, but this effect is much less significant. We can solve for the turn at which the managers are indifferent between choosing hire and interview but have not nailed down the precise behavior around this turn or the extent to which interview order matters.

If the managers observe $v(c)$, threshold strategies seem plausible. A strategy with threshold function $\bar{v}(t)$ chooses hire when $\max_{c \in I_t} v(c) > \bar{v}(t)$; thresholds are clearly weakly-decreasing in $t$. If a manager chooses interview, it conveys that all candidates in her interview set have
$v(c) < \bar{v}(t)$. If $\bar{v}(t) < 1$, this upper bound lowers the expected value to the opponent of any candidate the manager has interviewed. Thus, non-trivial thresholds make candidates in $C \setminus \mathcal{I}_0 \setminus \mathcal{I}_1$ more attractive, but as long as $\mathcal{I}_0$ and $\mathcal{I}_1$ are disjoint each manager should keep her threshold at 1.

Returning to 0/1 utilities, with $N > 2$ managers, only $M_0$ can unilaterally guarantee a $\frac{1}{N}$ probability of winning. In addition, interviewing in a follow-the-leader order is clearly a bad choice that gives a huge advantage to any deviator. It also seems plausible that non-perfect Nash equilibria involving coordinated punishment could arise.

### 6.3 Simulation-Based Game

While the game presented in Section 6.1 is of interest as a multi-manager extension that endogenizes interview ordering and hiring pressure, it clearly falls short in modeling actual hiring decisions. The prediction that one manager may need to interview twice as many candidates as the other seems implausible. An obvious remedy would be to make interviews costly. Real managers may also evaluate the same candidates differently. This could be modeled using a private value component that differs across candidate-manager pairs. Relatedly, the information managers get from interviews could be noisy. Another interesting extension would be for each manager to consider hiring multiple candidates. A number of researchers have considered multiple-hire variants of the single-manager problem, but it is unclear how their results would extend to the optimal stopping game.

Some of these extensions might be addressable analytically, but some, as well as some of the extensions in Section 6.2, may prove intractable. I intend to investigate models incorporating these additional aspects using simulation-based game theory. If the payoff differences between profiles in the resulting games are as small as those in Section 6.1, bootstrap methods are likely to be especially important. I hope that simulations and theory will each propel the other forward.

### 7 Research Timeline

In Table 1, I have set optimistic goals for the completion of the projects I propose in Sections 5 and 6. I believe that, if successful, the work on learning game models would be appropriate for the Symposium on Algorithmic Game Theory, which has a deadline of May 1st. I would also like to submit the work on optimal stopping games to the Web and Information Economics conference, which has a deadline of August 1st. Because either of these projects could take
longer than the allotted time, I have deliberately built in some slack by allocating more time than I think will be necessary for extensions to the credit network and bootstrap projects.

<table>
<thead>
<tr>
<th>Proposed Task</th>
<th>Goal Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game model learning</td>
<td>May 2014</td>
</tr>
<tr>
<td>Optimal stopping games</td>
<td>August 2014</td>
</tr>
<tr>
<td>Credit network and bootstrap extensions</td>
<td>December 2014</td>
</tr>
<tr>
<td>Defense</td>
<td>April 2015</td>
</tr>
</tbody>
</table>

Table 1: Thesis Timeline

References


A Scaling Simulation-Based Game Analysis through DPR
Scaling Simulation-Based Game Analysis through Deviation-Preserving Reduction

Bryce Wiedenbeck and Michael P. Wellman
University of Michigan
{btwied,wellman}@umich.edu

ABSTRACT
Multiagent simulation extends the reach of game-theoretic analysis to scenarios where payoff functions can be computed from implemented agent strategies. However, this approach is limited by the exponential growth in game size relative to the number of agents. Player reductions allow us to construct games with a small number of players that approximate very large symmetric games. We introduce deviation-preserving reduction, which generalizes and improves on existing methods by combining sensitivity to unilateral deviation with granular subsampling of the profile space. We evaluate our method on several classes of random games and show that deviation-preserving reduction performs better than prior methods at approximating full-game equilibria.

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

General Terms
Algorithms, Economics

Keywords
empirical game theory, simulation-based game theory, game reduction

1. INTRODUCTION
Game-theoretic analysis plays an increasingly prominent role in research on understanding and designing multiagent systems. Agent-based simulation offers the potential to increase the scope of applicability for game theory, beyond those game scenarios that can be described straightforwardly and solved analytically. In the simulation-based approach, rather than directly express all payoffs for a game, the analyst describes an environment procedurally and then computes payoffs by simulation of agent interactions in that environment.

Simulation enables analysis of many rich strategic environments, but determining payoffs for a large game in this way may be prohibitively expensive. Straightforward estimation of a payoff function requires simulation of every possible combination, or profile, of agent strategies. If the environment is stochastic, then many simulation runs may be necessary to obtain a reasonable estimate of even a single profile. For multiagent interactions that extend over time, or are otherwise complex, the computational cost of simulation may severely limit the number of profiles—and therefore the size of the game—that can be considered in such an analysis.

We focus for most of this paper on symmetric games, in which all agents have the same set of available strategies and payoffs depend only on the number of agents playing each strategy, not on the specific identities of those agents. Formally, a symmetric game is a tuple $\Gamma = (N, S, u)$, where $N$ is the number of agents, $S$ is the set of strategies available to all agents, and the utility function $u(s, \bar{s})$ gives the payoff to any agent playing strategy $s$ in profile $\bar{s}$. To conduct a complete analysis of $\Gamma$, we require that $u$ specifies payoffs for all possible profiles. A symmetric game with $N$ agents and $|S|$ strategies contains $\binom{N+|S|-1}{N}$ profiles. For a sense of how great a burden this imposes, consider that a symmetric game with 15 agents and 15 strategies contains over 77 million profiles, so if estimating a profile’s payoff through simulation required one second, constructing the full game would take more than two years.

We seek to combat this exponential growth using a technique broadly known as player reduction. Player reductions approximate games with many agents by constructing smaller games that aggregate over those agents in some way. Equilibria of the reduced game can then be viewed as approximate equilibria of the full game.

As an example, consider trading in continuous double auctions (CDAs)—a problem of agent strategy that has been extensively investigated through simulation. We review the coverage of several studies that employed simulation to estimate payoff functions for purposes of game-theoretic or evolutionary analysis. In the first empirical game analysis of CDA strategy, Walsh et al. [15] analyzed a 20-player game with three strategies. The 231 distinct profiles were within their simulation budget, whereas adding just one


Copyright © 2012, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

1To see this, note that we can describe a profile in terms how many agents play each strategy. Suppose an ordering of strategies, and consider a representation that indicates players by one symbol (.) and partitions by another (|). For instance, with $S = \{s_1, s_2, s_3, s_4\}$ and $N = 6$, the profile $...[1]$ has three agents playing $s_1$, two $s_2$, and one $s_4$. The representation contains $N + |S| - 1$ total symbols, and the choice of which $N$ of them to make players (or equivalently, choice of partitions) uniquely defines a profile.
more strategy would have entailed estimating 1540 more. Vytelingum et al. [14] likewise considered a 20-player game, though one that imposed symmetry only within the subgroups of 10 buyers and 10 sellers. Their study also compared three strategies, but for tractability and to facilitate visualization of their evolutionary traces, they limited analysis to two strategies at a time, which requires 121 profiles for each strategy pair and scenario combination. Phelps et al. [8] covered up to four strategies, in a 12-agent simulation employing another form of double auction mechanism (455 profiles). The study of Tesauro and Bredin [11] considered as many as 44 trading agents with three different strategies, but their analysis evaluated only profiles where agents were evenly divided across two strategies. This selection is similar to a two-player reduction according to the hierarchical method, as discussed below. Tesauro and Das [12] covered five strategies for a 20-agent scenario, this time with a mix of evenly-divided profiles and profiles where only one agent deviates from a homogeneous profile. As we see below, this is suggestive of the deviation-preserving reduction method we introduce here. In [54x260](1) this was rendered feasible only by virtue of their reduction to a four-player game, comprising 2380 profiles as opposed to 68 million in the unreduced game. Overall, we see that many-agent simulation studies either adopt player reductions, or make do with very narrow strategy exploration.

In this paper, we propose and study deviation-preserving reduction, which renders reduced-game equilibria more informative with respect to the full game. We start in Section 2 by reviewing existing methods for player reduction. Section 3 introduces deviation-preserving reduction, and explains how our new method is designed to combine the best aspects of its predecessors. Section 4 evaluates the reductions, and Section 5 shows how both our reduction and previous ones can be extended to games that are symmetric only with respect to a partition of players into roles.

2. BACKGROUND: PLAYER REDUCTION

Two methods for player reduction have been proposed in the literature: hierarchical reduction [16], and twins reduction [4]. Both methods are defined with respect to symmetric games. They define a subset of the profiles in the given (full) game, and map the payoffs of these profiles to a payoff function defined over a game with fewer players (the reduced game). Analyses of the reduced game are then interpreted as approximately applying to the full game.

2.1 Hierarchical Reduction

Of the two existing methods, hierarchical reduction is the more extensively used [1, 5, 10]. Hierarchical reduction works by grouping agents into coalitions that are constrained to act together. One player in the reduced game selects an action to be played by all agents in a coalition, and receives the payoff to any agent playing that strategy. To capture this formally, we introduce the following notation: a strategy profile \( \hat{s} = \langle c_1 \times s_1, \ldots, c_{|S|} \times s_{|S|} \rangle \) of game \( \Gamma \) consists of strategies \( s_i \in S \) and integer counts \( c_i \geq 0 \) for each strategy such that \( \sum_{s_i \in S} c_i = N \). When \( c_i = 0 \), we may omit it from the expression.

The hierarchical reduction of \( \Gamma \) to \( n < N \) players is defined as \( HR_n(\Gamma) = (n, S, u^{HR}) \), where

\[
u^{HR}(s, \langle c_1 \times s_1, \ldots, c_{|S|} \times s_{|S|} \rangle) =\]

\[
u(s, \langle \frac{N}{n} c_1 \times s_1, \ldots, \frac{N}{n} c_{|S|} \times s_{|S|} \rangle)].\]

This definition follows previous applications of hierarchical reduction in assuming that \( N \) is an integer multiple of \( n \). In our evaluation we employ a generalized version (described in Section 4.1) that allows reduction to numbers of players that do not evenly divide the number of agents in the full game.

For illustration, consider a full game with \( N = 25 \) agents, and a hierarchical reduction to \( n = 5 \) players. The action of each reduced-game player is played by five agents in the full game, so the reduced-game profile \( (2 \times s_1, 1 \times s_2, 2 \times s_3) \) corresponds to the full-game profile \( (10 \times s_1, 5 \times s_2, 10 \times s_3) \).

The main idea behind hierarchical reduction is that though the payoff to a particular strategy generally varies with the number of agents that play each strategy, it often can be expected to do so smoothly. Kearns and Mansour [6] formalize a related condition called bounded influence to define a class of compactly representable and solvable games. Whereas it is easy to construct games that violate this assumption, in many natural symmetric games, the payoffs are smooth in this way.

However, \( HR_n(\Gamma) \) lacks crucial information relevant to Nash equilibria of \( \Gamma \). In a Nash equilibrium of \( \Gamma \), no individual agent can gain by deviating to another strategy, but the hierarchical reduction contains no information about unilateral deviations. In an equilibrium of \( HR_n(\Gamma) \), no \( \frac{N}{n} \)-agent coalition can gain by all deviating to the same strategy, but there are many cases in which these conditions differ substantially.

Consider a network formation game [3] in which agents create links to one another. Agents gain from being in a connected network, but incur a cost for each link they create. In such a game we can envision a full-game equilibrium that is not an equilibrium of the reduced game, as well as a reduced-game equilibrium that is not an equilibrium of the full game. If network effects are large, but cost of creating links is high, there could be an equilibrium of the full game where agents create no links. However, if several players are allowed to deviate together they may create a sufficiently dense network to overcome the link-creation cost: this would be a beneficial deviation in the reduced game, so the full-game equilibrium would not be found. Under different parameters, there may be a spurious equilibrium in the reduced game where all players contribute links to the network, and no player can gain by deviating because if all agents represented by one reduced-game player changed strategies simultaneously, the network would collapse. On the other hand, a unilaterally deviating agent in the full game might have a much smaller impact and still receive the network benefits while avoiding the link creation cost.

2.2 Twins Reduction

The natural solution to this problem is to incorporate information about the value of unilateral agent deviations into the payoffs of the reduced game. Ficici et al. [4] propose a method called twins reduction that takes a first step in this
direction. The twins reduction of a symmetric game is a 2-player game, \( TR(\Gamma) = (2, S, u^{TR}) \), where each player views itself as controlling one agent in the full game, and the opponent as controlling all remaining agents:

\[
u^{TR}(s, \langle 1 \times s, 1 \times s' \rangle) = u(s, \langle 1 \times s, (N - 1) \times s' \rangle).
\]

Note that the payoffs for the two strategies in a twins reduction profile \( \langle 1 \times s, 1 \times s' \rangle \), \( s \neq s' \), correspond to two different profiles in the full game:

- \( \langle 1 \times s, (N - 1) \times s' \rangle \) and
- \( \langle (N - 1) \times s, 1 \times s' \rangle \),

but that the reduced game is still symmetric.

Ficici et al. [4] advocate constructing twins reduction games not by explicitly simulating these full-game profiles, but by sampling random profiles from the full game and determining the payoffs by linear regression on the number of agents playing each strategy. We refer to this approach as \( TR-R \), where the second “R” stands for “regression”. We consider the direct simulation approach a more appropriate benchmark, but evaluate both methods in our experiments.

The advantage of the twins reduction is that it captures information about individual agents’ incentives to deviate. Its major disadvantage is that it is limited to two players, and can therefore give only an extremely coarse-grained view of the game. In general, a reduced-game representation will have difficulty capturing equilibria of the full game that have support size (number of distinct strategies played with positive probability) larger than \( n \), the reduced number of players, as no profiles of the reduced game capture the interaction of all strategies in the support set. Since the twins reduction (\( n = 2 \)) never contains profiles where more than two strategies are played, it is particularly restrained by this limitation.

### 3. Deviation-Preserving Reduction

We propose a new game reduction method that combines the sensitivity to unilateral deviation afforded by twins reduction with the profile-space granularity of hierarchical reduction. We call this method deviation-preserving reduction. In a deviation-preserving reduction game, each player views itself as controlling a single agent in the full game, but views the profile of opponent strategies in the reduced game as an aggregation of all other agents in the full game. Formally, \( DPR_n(\Gamma) = (n, S, u^{DPR}) \), where

\[
u^{DPR}(s, \langle c_1 \times s_1, \ldots, c_n \times s_n \rangle) = \nu(s, \langle N - 1 \times c_1 \times s_1, \ldots, \frac{N - 1}{n - 1} (c_n - 1) + 1 \times s_n \rangle).
\]

In a hierarchical reduction, the proportion of agents playing each strategy is the same in the full and reduced games. Under deviation-preserving reduction, analogously, the proportion of opponents playing a strategy in the full and reduced games is the same from each player’s perspective. And as in a twins reduction, each player in a deviation-preserving

\(^3\)The original definition [4] applies to a somewhat broader class: role-symmetric games with identical strategies. In Section 5 we describe how to extend player reductions, including twins reductions, to the entire class of role-symmetric games.

The reduction game is sensitive to the payoffs of exactly one agent in the full game. As a consequence of this sensitivity to single agents, the deviation-preserving reduction game can identify exact symmetric pure strategy equilibria of the full game if they exist.

**Proposition 1.** A profile \( (n \times s) \) is a Nash equilibrium of \( DPR_n(\Gamma) \) if and only if the profile \( (N \times s) \) is a Nash equilibrium of \( \Gamma \).

**Proof.** The profile \( (n \times s) \) is a NE when \( u^{DPR}(n \times s) \geq u^{DPR}(n - 1 \times s, 1 \times s') \) for all \( s' \in S \). This is the case exactly when \( u(s, (N \times s)) \geq u(s, ((N - 1) \times s, 1 \times s')) \) for all \( s' \in S \). \( \square \)

This property also holds for twins reduction games, because the deviation preserving reduction is a strict generalization of the directed-sampling twins reduction: \( TR(\Gamma) = DPR_2(\Gamma) \).

To construct each profile’s payoffs in a deviation-preserving reduction game, several profiles from the full game must be simulated. Returning to the example of a 25-agent full game and a 5-player reduced game, the profile \( (2 \times s_1, 1 \times s_2, 2 \times s_3) \) in the deviation-preserving reduction game employs payoff values from several profiles in the full game:

\[
u^{DPR}(s_1, (2 \times s_1, 1 \times s_2, 2 \times s_3)) = u(s_1, (7 \times s_1, 6 \times s_2, 12 \times s_3))
\]

\[
u^{DPR}(s_2, (2 \times s_1, 1 \times s_2, 2 \times s_3)) = u(s_2, (12 \times s_1, 1 \times s_2, 12 \times s_3))
\]

\[
u^{DPR}(s_3, (2 \times s_1, 1 \times s_2, 2 \times s_3)) = u(s_3, (12 \times s_1, 6 \times s_2, 7 \times s_3))
\]

Note that we again assume divisibility: in this case, \( n - 1 \) has to divide \( N - 1 \) for the aggregation of opponents to be precise. As with hierarchical reduction, we can extend the definition to reduced games with any number of players, as described in Section 4.1. We quantify the number of profile simulations required for deviation-preserving reduction in the following proposition.

**Proposition 2.** Constructing \( DPR_n(\Gamma = (N, S, u)) \) requires simulating \( |S|\left(\frac{n + |S| - 2}{n - 1}\right) \) full-game profiles.
Proof. In each profile of the deviation-preserving reduction game, \( n-1 \) of the players each control \( N-1 \) players each control \( S^{(n+\vert S\vert-2)}/n-1 \) of them. Each of these profiles must be paired with each \( s \in S \), so \( S^{(n+\vert S\vert-2)}/n-1 \) profiles must be simulated.

Proposition 2 shows that constructing \( DPR_n(\Gamma) \) requires simulating strictly more profiles than \( HR_n(\Gamma) \), but by a factor of at most \( \vert S \vert \). As we show in Section 4, the extra profiles comprising this constant factor can contribute to significantly improved accuracy. Even so, we would like to minimize the number of simulations required when possible, and therefore also consider a variant of the deviation-preserving reduction, which we call \( DPR' \).

The idea behind \( DPR' \) is that most of the profiles simulated to construct a deviation-preserving reduction game are quite similar. For example, in the 5-player deviation-preserving reduction of the 25-agent game, the payoff to strategy \( s_1 \) in the full-game profile \( s_{\bar{1}} = (7 \times s_1, 6 \times s_2, 12 \times s_3) \) is employed in the reduced-game profile \( (2 \times s_1, 1 \times s_2, 2 \times s_3) \). The payoff for \( s_2 \) in reduced-game profile \( (1 \times s_1, 2 \times s_2, 2 \times s_3) \) is derived from \( s_{\bar{2}} = (6 \times s_1, 7 \times s_2, 12 \times s_3) \), which differs from \( s_{\bar{1}} \) only in that a single agent has switched from \( s_1 \) to \( s_2 \), both of which are played by many other agents. If we believe our assumption—derived from hierarchical reduction—that payoffs vary smoothly in the number of agents playing each strategy, we should expect the payoffs to strategy \( s_1 \) in profiles \( s_{\bar{1}} \) and \( s_{\bar{2}} \) to be very similar (likewise for \( s_2 \)).

Formally, \( DPR'(\Gamma) = (n, S, u^{DPR'}) \), where \( u^{DPR'} \) is defined as follows. Let \( \bar{s} = (c_{\min} \times s_{\min}, \ldots, c_s \times s, \ldots) \), where \( s_{\min} \) is the first strategy played by at least one agent. If \( c_s = 1 \) or \( c_s = c_{\min} \), then \( u^{DPR'}(s_{\bar{s}}) = u^{DPR'}(s_{\bar{1}}, s_{\bar{2}}) \). Otherwise, \( u^{DPR'}(s_{\bar{s}}) \) is given by

\[
u(s_{\bar{s}}) = \left( \frac{N-1}{n-1} c_{\min} + 1 \right) \times s_{\min} \ldots \left( \frac{N-1}{n-1} (c_s - 1) \times s \ldots \right).
\]

The result is that when \( DPR \) would prescribe simulation of several profiles that differ only by deviation of a single agent (and no strategy is played by only one agent), \( DPR' \) requires that only one be simulated. From among these profiles, \( DPR' \) selects the one in which the lowest-numbered strategy by which they differ is played most. In the example above, payoffs

- \( u^{DPR'}(s_1, (2 \times s_1, 1 \times s_2, 2 \times s_3)) \) and
- \( u^{DPR'}(s_2, (1 \times s_1, 2 \times s_2, 2 \times s_3)) \)

both come from full-game profile \( (7 \times s_1, 6 \times s_2, 12 \times s_3) \). The savings in terms of profiles sampled are illustrated in Figure 1. In that graph, the curve for \( DPR' \) follows the formula \( \vert \Gamma \vert = \vert S^{(n+\vert S\vert-2)/n-1} \vert - (n-2) \vert S^{(n+\vert S\vert-2)/n-1} \vert \). \( DPR' \) always requires more full-game profiles than \( HR \), and fewer than \( DPR \), except when \( n = 2 \), where both are equivalent to \( TR \).

4. EMPIRICAL EVALUATION

The goal of a player reduction is to replace a full game that is too large to effectively analyze with a more manageable reduced game. To compare reduction methods, we therefore need to evaluate how well analysis performed on a reduced game translates back to the full game. This presents a problem for evaluation, in that full games of interest are too big to effectively analyze. For example, in the simulated credit network games discussed below, we construct 12-agent, 6-strategy full games; we would like to analyze reductions of 60-agent games, but even with just six strategies, the full game would consist of 8,259,888 profiles. We therefore compromise by reducing several types of medium-sized games to very small ones. If one reduction consistently performs better in such cases, we take it as an indication that the same will hold for reductions of very large games.

4.1 Regret of Reduced Game Equilibria

Numerous methods for analyzing games exist, but the most important is finding Nash equilibria. Because player reductions work with symmetric games, we evaluate them primarily by how well symmetric mixed strategy Nash equilibria computed in the reduced game approximate symmetric mixed strategy equilibria of the full game. Our primary measure for the quality of reduced-game equilibria is regret. The regret \( \epsilon(\bar{\sigma}) \) of a symmetric mixed strategy profile \( \bar{\sigma} \), in which all players play mixed strategy \( \sigma \) is the maximum gain any player could achieve by deviating to a pure strategy:

\[
\epsilon(\bar{\sigma}) = \max_{s \in S} u(s, \bar{\sigma}_{-i}),
\]

where \( u(s, \bar{\sigma}_{-i}) \) is the expected payoff to a player playing \( s \) when all others play \( \sigma \). A Nash equilibrium has zero regret, but a symmetric mixed profile \( \bar{\sigma} \) that is an equilibrium of the reduced game will generally have positive regret with respect to the full game. Such a \( \bar{\sigma} \) can be viewed as an approximate, or \( \epsilon(\bar{\sigma}) \)-Nash equilibrium of the full game, where the lower the regret, the better the approximation.

However, we cannot simply compare the regret of equilibria from \( k \)-player reduced games under each method. The first problem is that the number of players in a twins reduction game is not scalable. Moreover, since the goal of player reduction is to simulate fewer profiles, the relevant comparison is not the number of players in the reduced game, but the number of profiles required to construct it, and \( DPR_k \) always requires sampling strictly more profiles than \( HR_k \). For example, in the 12-agent 6-strategy game instances below, \( |DPR_3| = 126 = |HR_4| \). Because the directed-simulation twins reduction always requires \( |S^{(n+\vert S\vert-2)/2}| \) profiles, we compare to \( TR \) by addressing the question of whether lower regret can be achieved by a method that samples more profiles. Twins reduction with regression can use any set of profiles; in our experiments we varied the size of the set over a range similar to that required to construct the various reduced games.

Hierarchical reduction as defined by Wellman et al. [16] requires that the number of reduced-game players \( n \) divide the number of full-game agents \( N \). By analogy, in our definition of deviation preserving reduction above, we assume that \( n - 1 \) divides \( N - 1 \). In our experiments, we perform reductions of both varieties where these conditions do not hold. We extend the definition of \( HR_n \) to allow indivisibility as follows:

\[
u^{HR}(s_{\bar{1}}, (c_1 \times s_1, \ldots, c_{\vert S \vert} \times s_{\vert S \vert})) =
\]

\[
u(s, \left( \frac{N}{n} c_1 + 1 \right) \times s_1, \ldots, \left( \frac{N}{n} c_{j+1} \right) \times s_{j+1}, \ldots),
\]

where \( j = N - \sum \left( \frac{N}{n} c_j \right) \). That is, the number of opponents...
playing strategy \(s_i\) is the integral part of \(\frac{N}{n}c_i\), with extra player slots allocated one each to strategies with lower indices. For example, when we construct \(HR_5\) of a game with 12 players, the reduced-game profile \((1 \times s_1, 3 \times s_2, 1 \times s_3)\) corresponds to the full-game profile \((3 \times s_1, 7 \times s_2, 2 \times s_3)\). We extend the definition of \(DPR_n\) to handle indivisibility in a similar manner.

We evaluate the reductions using two classes of random games: congestion games [9], in which agents select a fixed-size subset of available facilities and payoffs are decreasing in the number of agents choosing a facility; and local effect games [7], which have a graph over actions and each action’s payoff is a function of the number of agents choosing it and adjacent actions. We randomly varied the payoff function parameters of these games to create 250 game instances for each test described below. We also evaluate on one simulated game class, based on a scenario of credit network formation [2]. In the model of credit networks employed in this scenario, directed links represent credit issued to other agents: agents wish to transact with one-another, but issuing credit bears risk in that debtors may default. In the credit network formation game, payoffs are determined by simulating a sequence of transactions and defaults on the network induced by agent strategies. We sampled each profile of a 12-agent, 6-strategy credit network game 100 times, and randomly recombined these samples to create 250 game instances.

The first finding of note is that twins reduction performs very poorly with linear regression. The top line in Figure 2a shows the regret of equilibria found in \(TR-R\) games with random sampling of profiles, which is an order of magnitude worse than the \(HR\), \(TR\), \(DPR\), and \(DPR'\). Two observations led us to try the method labeled \(TR-DPR\): first, that sampling profiles according to uniform agent play leads to a very low likelihood of observing payoffs for profiles where most agents play the same strategy, and these are exactly the profiles whose payoffs the regression estimates. Second, simulating all the profiles for \(DPR\) or \(DPR'\) makes available a substantial amount of payoff data that goes unused in constructing the reduced game. We therefore thought to try using all of the profiles simulated for the deviation-preserving reduction as input to the linear regression of \(TR-R\). As is clear from Figure 2a, this improves very little on random sampling.

In retrospect, it is not particularly surprising that approximating payoffs by linear regression performs so poorly: all of our example games and most games requiring simulation have nonlinear payoffs. A better regression model could potentially alleviate this problem, but choosing one requires knowledge of the game’s payoff function that may not be available when payoffs are determined by simulation. We also ran \(TR-R\) and \(TR-DPR\) on each of the other game classes, but the results are similarly poor, and are excluded from subsequent figures.

![Figure 2: Average full-game regret of reduced-game equilibria in local effect games.](image)

Figure 3: Full-game regret of reduced-game equilibria in congestion games. \(N = 100, |S| = 2, 2 \leq n \leq 10\).

Figures 2b, 3, and 4 show that deviation-preserving reduction outperforms hierarchical reduction and twins reduction in a wide variety of settings. In 12-agent, 6-strategy local effect games, \(DPR\) is clearly better than \(HR\), but the comparison to \(DPR'\) is less conclusive. We were surprised to find that hierarchical reduction would perform worse with increased reduced-game size, which corresponds to increased abstraction granularity. We note, however, that the 5, 7, and 8-player reduced games where \(HR\) performs poorly are exactly the cases where \(n\) does not divide \(N = 12\). This also leads us to observe that because 11 is prime, the deviation-preserving reduction never has the advantage \(n - 1\) dividing \(N - 1\), and yet consistently performs well. The results from 12-agent, 6-strategy congestion games (not shown) are broadly similar.
In an attempt to get at the effect of very substantial player reductions, we created 100-agent, 2-strategy congestion games. The results in Figure 3 show clear separation between hierarchical reduction and both variants of deviation-preserving reduction, suggesting that as the number of players grows, the relative difference between DPR and DPR' may be smaller. Results for the 12-agent, 6-strategy credit network game appear in Figure 4. Here again, DPR and DPR' perform similarly, and better than HR.

Across all game classes and sizes examined (including those not shown) deviation-preserving reduction of any given size outperforms the hierarchical reduction with at least as many profiles that is closest in size. This means that for any size hierarchical reduction, there exists a better deviation-preserving reduction that requires simulating fewer profiles. Virtually all of these differences are significant at $p < 0.05$; the only exceptions are 4-player DPR versus 6-player HR in the 12-player congestion game (Figure 3) and 12-player credit network game (Figure 4). In addition, DPR$_3$ outperforms TR across all game classes; the difference is significant at $p < 0.05$ in all cases except the credit network game. The difference between DPR$_4$ and TR is significant in all cases.

### 4.2 Comparison to Full-Game Equilibria

We also compared reduced-game equilibria under HR and DPR to equilibria from 12-player, 6-strategy full games using two metrics: similarity of support sets, and $L^2$ distance between distributions. Table 1 shows the number of strategies by which the support sets of full and reduced-game equilibria differ. Here, we consider a strategy to be in the support of a symmetric $\epsilon$-Nash equilibrium if it is played with probability $0.01$ or greater. In nearly all cases support sets of DPR$_n$ match match those of full-game equilibria significantly ($p < 0.05$) better than both HR$_n$ and HR$_{n+1}$. In addition, for congestion games and local effect games, DPR$_{>2}$ significantly outperforms TR, whereas in credit network games, there is no significant difference between TR and DPR.

Table 2 presents a similar message, but in terms of the $L^2$ distances between the mixed strategy distributions in full and reduced-game equilibria. Again, DPR is significantly better than HR and TR for congestion and local effect games, while performing similarly on credit network games. As in Section 4.1, in these experiments, we compute one symmetric mixed-strategy Nash equilibrium per game by running replicator dynamics initialized to the uniform mixture.

### 4.3 Dominated Strategies

Another useful operation in the analysis of simulation-based games is to check for dominated strategies. A dominated strategy is one that no agent should ever play because there is an alternative strategy that is at least as good in response to any profile of opponent strategies. We ran experiments on 12-agent, 6-strategy congestion and credit network games (250 each), comparing the set of strategies that remain after iterated elimination of strictly dominated strategies in the full game against those that remain in 2, 4, and 6-player reduced games. We observed that DPR and DPR' produced very similar results, and that both improved over hierarchical and twins reduction. Figures 5 and 6 show histograms of the number of strategies eliminated in reduced games but not eliminated in full games.

In congestion games (Figure 5), twins reduction and both forms of deviation-preserving outperform hierarchical reduction, eliminating fewer strategies in the reduced game that survive in the full game, even when hierarchical reduction samples vastly more profiles. These congestion games often exhibit dominated strategies in the full game, but we almost never observed strategies surviving in reduced games that are dominated in the full game.

In credit network games (Figure 6), no strategies are dominated in the full game, but in the twins reduction game, many strategies are eliminated. Moving to DPR$_n$ or DPR' solves this problem almost entirely. These experiments also confirm that for all reduction types, increasing the number of players in the reduced game reduces the number of strategies erroneously found to be dominated.

### 5. ROLE-SYMMETRIC GAMES

We can smoothly relax the constraint that games be fully
symmetric by assigning agents to roles, and enforcing symmetry only within these roles. Across roles, agents’ strategy sets and payoffs can differ, but within a role, they are symmetric. Formally, a role-symmetric game is a tuple $\Gamma = (\{N_i\}, \{S_i\}, u)$, where the number of agents with role $i$ is $N_i$, and agents with role $i$ have strategy set $S_i$. Role-symmetric games provide a natural model for many settings where agents can be partitioned into meaningful categories, such as buyers and sellers in a market, or attackers and defenders in a security game. Role symmetry imposes no loss of generality on normal-form games, spanning the spectrum from complete asymmetry (each player has its own role) to full symmetry (a single role for everyone).

All of the player reduction methods discussed here can be straightforwardly extended to role-symmetric games. Consider for example the 20-agent continuous double auction study of Vytelingum et al. [14] with $N_1 = 10$ buyers and $N_2 = 10$ sellers. Instead of choosing $n$, the number of players in the reduced game, we must choose each $\{n_i\}$, the number of players with each role in the reduced game.

To perform a hierarchical reduction, a natural choice would be $n_1 = n_2 = 2$. This would involve simulating all profiles where 0, 5, or 10 agents play each buyer strategy, and a multiple of five agents likewise play each seller strategy.

With twins reduction, there are two players per role. Each player views itself as controlling a single agent, and the other player with the same role as controlling nine agents. It views the two other-role players as each representing half the ten agents with that role, so in the reduced-game profile $\langle 1 \times s_1.1, 1 \times s_2.1, 1 \times s_2.2 \rangle$, the payoff to buyer 1, who plays $s_1.1$, comes from full-game profile $\langle 1 \times s_1.1, 9 \times s_1.2, 5 \times s_2.1, 5 \times s_2.2 \rangle$.

The deviation-preserving reduction extends the twins reduction to more than two reduced-game players per role, maintaining the view that a reduced-game player controls a single agent, while the other players with the same role aggregate over the rest of the agents with that role, and players with another role aggregate over all agents with their role. With either hierarchical reduction or deviation-preserving reduction, it would be possible to choose $n_i \neq n_j$ if different granularity of reduction were desired for different roles.

This extension to role-symmetric games encompasses the broader class over which Ficici et al. [4] define the twins reduction. The clustering method by which they aggregate agents induces a role-symmetric game that restricts all roles to have the same strategy set (but allows different payoffs). They mention but do not develop the idea that the twins reduction might extend to role-symmetric games. To our
knowledge, hierarchical reduction has not been applied to role-symmetric games.

6. CONCLUSIONS

Our new player reduction method, deviation-preserving reduction, combines the most appealing aspects of hierarchical reduction and twins reduction. It also performs better than both prior methods experimentally: equilibria from DPR games have lower full-game regret and more closely resemble full-game equilibria, even when sampling fewer full-game profiles. In addition, performing iterated elimination of dominated strategies on deviation-preserving reduction games stays more faithful to the full game compared to other player reductions. Our alternative DPR’ formulation performs reasonably well in the same tests. The simulation savings from DPR’ are greatest when the reduced game has many players but few strategies, so DPR’ may prove useful in such cases. Though it may not be obvious how to choose between DPR and DPR’, the evidence is quite compelling that deviation-preserving reduction is the best available player reduction method for analyzing large simulation-based games.

7. REFERENCES

B Bootstrap Statistics for Empirical Games
Bootstrap Statistics for Empirical Games

Bryce Wiedenbeck
University of Michigan
btwied@umich.edu

Ben-Alexander Cassell
University of Michigan
bcassell@umich.edu

Michael P. Wellman
University of Michigan
wellman@umich.edu

ABSTRACT
Researchers often use normal-form games to model multiagent interactions. When a game model is based on observational or simulated data about agent payoffs, we call it an empirical game. The payoff matrix of an empirical game can be analyzed like any normal-form game, for example, by identifying Nash equilibria or instances of other solution concepts. Given the game model’s basis in sampled data, however, empirical game analysis must also consider sampling error and distributional properties of candidate solutions. Toward this end, we introduce bootstrap techniques that support statistical reasoning as part of the empirical game-theoretic analysis process. First, we show how the bootstrap can be applied to compute confidence bounds on the regret of reported approximate equilibria. Second, we experimentally demonstrate that applying bootstrapped regret confidence intervals can improve sampling decisions in simulation-based game modeling.

1. MOTIVATION

Traditional game-theoretic analysis is often confined to what is mathematically tractable for the game theorist, and as such is often restricted in scale (numbers of players, strategies) and complexity (dynamics, strategic interaction, information structures). When the extremes of abstraction required for analytic solution are too constraining, we may instead describe a strategic scenario in terms of an agent-based model (ABM), and attempt to build a game form by simulating that ABM [5, 6, 12, 18]. Alternatively, we may wish to build game models by observing real world play [4, 10]. In either case, payoff observations may be noisy, leading to uncertainty as to whether the game model accurately describes the true game. As such, we would like to quantify this uncertainty when conducting game-theoretic analysis on such games.

The approach of building and analyzing game models from data is known as empirical game-theoretic analysis (EGTA) [17]. Though some EGTA studies report statistical information (e.g., error bars on payoff estimates or significance of particular comparisons), there has been limited work to date on general methods for statistical reasoning about conclusions from empirical games. Instead, studies rely on extremely large sample size [18], or variance reduction techniques [12], to bolster confidence in results. Whereas more samples and less variance does increase confidence, we must quantify this confidence if results from EGTA studies are to be taken as serious scientific evidence for propositions of interest. In the absence of statistical guidance, we cannot tell whether practitioners are taking insufficient or excessive observations, or whether complex variance reduction measures are worth their cost, or most importantly, whether published results reflect fundamental properties of the games studied or artifacts of sampling variation.

The general dearth of statistical analysis in EGTA is understandable, given the difficulty of evaluating complex game-solution hypotheses in a traditional statistical framework. Little is typically known about payoff distributions prior to observing play, rendering parametric approaches inapplicable. Moreover, determining whether a profile satisfies a solution concept such as Nash equilibrium generally requires evaluating multiple statistical hypotheses about comparison of corresponding payoffs across potential deviations. This raises multiple-testing concerns [3], as well as other complexities due to the interdependencies among these comparisons.

This paper represents a first systematic effort to develop statistical methods generally applicable to game models derived from empirical data. Our approach is based on bootstrap techniques, which leverage the additional information available in observation data to characterize distributions over game-theoretic conclusions. We apply this approach to two problems. First, we present a bootstrap method to calculate statistical confidence bounds on reported equilibria. Second, we exploit this statistical information in an algorithm to improve sampling decisions.

2. BACKGROUND

2.1 Simulation-Based Games

In most applications of EGTA, a simulator acts as an oracle for player utility functions, taking as input a strategy profile—an assignment of one strategy to every player—and returning an observation of the payoff each player accrued from that profile in simulation. Since ABMs typically incorporate stochastic factors (uncertainty in environment and agent private information), the payoff-vector observation is actually a sample from some underlying distribution of payoffs. Using the ABM simulator, we collect an observation set Θ by repeatedly sampling each strategy profile. We can then construct a game model $M(\Theta)$ from the observation set. The most common way to construct a game
model from simulation data is to build a normal-form payoff matrix, where each entry in the matrix corresponds to the mean of payoff observations of the corresponding profile. Researchers adopting the EGTA approach have used such models to perform standard game-theoretic calculations, such as derivation of Nash equilibria or finding dominant strategies [2, 12, 14]. They have also employed results of these game computations to estimate features of ABM outcomes in equilibrium, such as the average price of a security in a financial market [5].

Due to its statistical nature, it may be costly or impossible to identify exact Nash equilibria of the game underlying the simulator (hereafter referred to as the true game). Instead, it is often helpful to identify profiles that are close approximations to Nash equilibria. Quality of an approximate equilibrium is measured by regret, the largest gain any player can achieve through unilateral deviation. Formally, regret of a (potentially mixed-strategy) profile $\vec{\sigma}$ in a symmetric game is given by:

$$\epsilon(\vec{\sigma}) = \max_{s \in S} \max_{\vec{\sigma}'} u(s, \vec{\sigma}') - u(\sigma, \vec{\sigma}),$$

where $S$ is the set of pure strategies available to each player, $u(\cdot)$ is the utility function, $\sigma \in \vec{\sigma}$ indicates that in profile $\vec{\sigma}$ some player is playing strategy $\sigma$ and $\vec{\sigma}' - \sigma$ is the partial profile where the player using $\sigma$ has been removed. By definition, a profile is a Nash equilibrium if and only if it has zero regret. We say a profile approximates Nash equilibrium at the level $\epsilon$ if the profile has regret less than or equal to $\epsilon$.

Since empirical games are constructed from limited sampling of a noisy payoff-generating process, even exact equilibria of the empirical game may have non-negligible regret in the true game. Therefore, in order to draw conclusions about the true game, we wish to estimate the regret of a profile in the true game from payoff sample data. More specifically, we would like to state with statistical confidence whether or not a profile is an $\epsilon$-Nash equilibrium of the true game through limited sampling of the simulator.

### 2.2 Bootstrap Statistics

The bootstrap is a computational method for estimating distributional information about a statistic computed on sample data [7]. Unlike classical statistical tests, it does not rest on explicit assumptions about the shape of the distribution. This feature is useful in reasoning about game solutions derived from sample data since nothing is initially known about the true payoff distributions. Furthermore, since the primary measure of interest, regret, is computed through taking a maximum rather than a sum or average, we cannot rely on the central limit theorem to fit a parametric statistical model to our sampling distribution.

The bootstrap treats a sample set as representative of the population and resamples the sample set to simulate drawing many samples from the population. If the original sample has size $n$, then each resample is a set of size $n$ drawn with replacement from that sample. The statistic is then computed on each resample set, giving a bootstrap distribution for the statistic that can be used in place of a sampling distribution for the statistic. Of particular interest, the bootstrap distribution can be used to estimate confidence intervals for the statistic: the interval between the 2.5th and 97.5th percentile of the bootstrap distribution gives a two-sided 95% confidence interval, while the 5th or 95th percentiles give one-sided 95% confidence bounds.

### 2.3 Related Work

There has been limited work using bootstrapped statistics to analyze an agent-based model, let alone to analyze games. In contrast, discrete-event systems modeling has seen adoption of the bootstrap to analyze the output of simulation [8]. Axtell, et al. [1] suggested that a bootstrap approach may be necessary for determining if two agent-based models are equivalent, due to the complicated nature of such a hypothesis.

While the bootstrap has not previously been used in analyzing games, two methods for estimating true game regret from payoff sample data have been introduced. Reeves [15] proposed estimating the empirical distribution of regret of a profile by sampling game matrices from the space of possible matrices induced by assuming every payoff is independent and distributed normally with mean and variance equal to its sample mean and sample variance respectively. He recommended using the probability that a profile’s regret is zero from the estimated empirical distribution of regret as a measure of the confidence that a profile is a Nash equilibrium. However, it is easy to demonstrate that mixed-strategy profiles, even true-game Nash equilibria, will have estimated regret of zero with negligible probability. Vorobeychik [16] presented a Bayesian framework for determining the posterior probability that a profile is an $\epsilon$-Nash equilibrium of the true game from payoff sample data. The author proved tight probability bounds under the assumption that payoff observations are independent draws with Gaussian noise, and much weaker distribution-free bounds. Both of these works ignore the possibility that payoffs within a single observation of a profile may be correlated, and rely on distributional assumptions that cannot be guaranteed for small samples. No empirical evaluations have been presented for these methods, leaving open the question of their usefulness under more varied simulators and sampling designs.

### 3. METHODS

We propose two methods that apply bootstrapping to improve empirical game-theoretic analysis. First, we show how to estimate confidence intervals for the regret of strategy profiles, and in particular approximate Nash equilibria. We then show how such confidence intervals can be used to improve sample control decisions in simulation-based games.

#### 3.1 Bootstrap Regret Estimates

Because empirical games are constructed from data, it is often unclear whether approximate Nash equilibria calculated in an empirical game correspond to Nash equilibria of the true game. Vorobeychik [16] proved that in the infinite sample limit, empirical-game and true-game equilibria are identical, however, this does not rule out reporting spurious equilibria in empirical games. The bootstrap estimate of regret allows us to report confidence intervals for the true-game regret of empirical-game equilibria.

To compute a bootstrap distribution for the regret of a (mixed- or pure-strategy) profile $\vec{\sigma}$, we construct a large number of bootstrap games by simultaneously resampling...
the observations of every entry in the payoff matrix. For each bootstrap game, we construct a resampled observation set \( \Theta \) where for each payoff, we draw with replacement from its samples a resample set of equal size. We then construct a bootstrap game model \( M(\Theta) \) by applying the same procedure used to construct the empirical game (usually setting payoffs equal to sample averages).

Frequently, the payoffs for all strategies in a profile are observed in a single simulation. In this case, correlation between samples can be preserved by applying common indexing, where the resample is computed over indices \( i \in \{1, \ldots, n\} \). If \( i \) is drawn, then the \( i^{th} \) observation of each payoff is included in \( \Theta \). In the event that all payoffs in a game were sampled with common random numbers, correlations can similarly be preserved by common indexing across all payoffs in the game.

In each bootstrap game, we compute the regret of profile \( \bar{\sigma} \), and across many bootstrap games, these values constitute a bootstrap distribution for the regret statistic. If \( \bar{\sigma} \) is an equilibrium of the empirical game, we are generally interested in an upper bound on its true-game regret. The 95\(^{th}\) percentile of the bootstrap distribution gives us a 95\%-confidence upper bound for \( \epsilon(\bar{\sigma}) \). In Algorithm 1, we will also be interested in a two-sided confidence interval for \( \epsilon(\bar{\sigma}) \), which is estimated by the 2.5\(^{th}\) and 97.5\(^{th}\) percentiles of the bootstrap distribution.

### 3.2 Sampling Control

When game data is generated by a simulator, the practitioner may exert control over how many observations to gather of each profile. Historically, researchers have interwoven sampling and interim game-theoretic analysis to minimize the number of observations gathered in unproductive regions of profile space [13]. Similarly, it may be possible to reduce the number of observations taken even of relevant profiles while still delivering a baseline of statistical confidence by interweaving evaluation of regret confidence intervals with sampling. At the conclusion of a stage of sampling, a \( \gamma \)-level two-sided confidence interval on the regret of a profile expresses with greater than \( \gamma \) confidence that the profile is an \( \epsilon \)-Nash equilibrium of the true game when the interval falls below \( \epsilon \); however, when the interval includes \( \epsilon \), we are unable to distinguish between having insufficient evidence that the claim is true and having sufficient evidence that the claim is false. Rather than report that we are uncertain whether or not a profile is an \( \epsilon \)-equilibrium, we would often prefer to continue sampling until we have sufficient confidence to make a determination. To address this issue, we propose the Confidence-Interval-Based Stopping Rule (CIBSR) presented in Algorithm 1.

CIBSR is similar to the repeated confidence interval approach to terminating clinical trials proposed by Jennison and Turnbull [11], but utilizes the bootstrap to construct confidence intervals in place of parametric assumptions. The algorithm takes as arguments a candidate profile \( \bar{\sigma} \), and any observations taken thus far \( \Theta_{\text{init}} \), and samples sequentially until there is sufficient evidence to decide whether or not the candidate profile is an \( \epsilon \)-equilibrium of the true game. CIBSR is parameterized by the acceptable regret threshold \( \epsilon \), the confidence interval level \( \gamma \), and the number of observations to gather of each relevant profile in each step \( \Theta \). CIBSR decides if a candidate is an \( \epsilon \)-equilibrium by comparing the boundaries of a two-sided confidence interval\(^2\) to \( \epsilon \), accepting the hypothesis that \( \bar{\sigma} \) is an \( \epsilon \)-equilibrium when the interval falls entirely within \([0, \epsilon]\), rejecting it when the interval falls entirely within \((\epsilon, \infty)\), and otherwise requesting further observations. Sampling under CIBSR is restricted to profiles that can affect the estimated regret distribution of the candidate. These profiles belong to either \( S(\bar{\sigma}) \), the set of pure-strategy profiles that are realized with positive probability under the profile \( \bar{\sigma} \), or \( \bigcup_{\bar{\sigma} \in D} S(\bar{\sigma}) \), the set of pure-strategy profiles that are realized with positive probability under some profile reachable from \( \bar{\sigma} \) through a unilateral deviation to a pure strategy.

Rather than simply wishing to determine if a particular profile is an equilibrium, practitioners may begin with no specific candidates, but sample from a simulator with the purpose of finding one or more equilibria of the true game. At any point in the sampling process, equilibria may be computed in the empirical game induced from the observations gathered thus far. As nothing is known about the payoff distributions prior to sampling, nor even which payoff distributions will be relevant for identifying equilibria of the game, practitioners typically sample sequentially according to rules of thumb, such as taking observations until the set of equilibria of the empirical game does not change with further sampling. Existing rules of thumb may reduce uncertainty in an indirect manner, but when such procedures terminate, no direct evidence can be provided of the regret of playing equilibria of the empirical game in the true game. Furthermore, since these stopping rules are typically very coarse heuristics, they may actually require more sampling than is necessary to have sufficient confidence that a profile is an \( \epsilon \)-Nash equilibrium of the true game. We propose incorporating bootstrap confidence intervals into a sequential equilibrium finding procedure as in the Confidence-Interval-Based Equilibrium Finding (CIBEF) algorithm, presented in Algorithm 2.

At each step, CIBEF requests \( x \) additional observations of each profile and finds equilibria of the updated empirical game. For each equilibrium of the empirical game, a one-sided regret confidence interval at the \( \gamma \)-level is constructed.

\(^2\)For all experiments we present, ONE-SIDED-REGRET-CI and TWO-SIDED-REGRET-CI implement the confidence interval methods described in Section 3.1.

---

**Algorithm 1 Confidence-Interval-Based Stopping Rule**

\[
\text{(} \bar{\sigma}, \Theta_{\text{init}}, \epsilon, \gamma, x \text{)}
\]

**Require:** \( \bar{\sigma} \), the profile to evaluate
**Require:** \( \Theta_{\text{init}} \), the observations used to identify \( \bar{\sigma} \) as a candidate
**Require:** \( \epsilon \), the acceptable approximation threshold
**Require:** \( \gamma \), the confidence level to use
**Require:** \( x \), the number of observations to take of each profile in each step

\[
\Theta_{\text{seq}} \leftarrow \Theta_{\text{init}}
\]

\[
[\epsilon_{\text{left}}, \epsilon_{\text{right}}] \leftarrow \text{TWO-SIDED-REGRET-CI}(\bar{\sigma}, \Theta_{\text{seq}}, \gamma)
\]

**while** \( \epsilon_{\text{left}} < \epsilon \) and \( \epsilon_{\text{right}} > \epsilon \) **do**

\[
\text{Append} \ x \text{ observations of each profile } s \in S(\bar{\sigma}) \cup (\bigcup_{\bar{\sigma} \in D} S(\bar{\sigma})) \text{ to } \Theta_{\text{seq}}
\]

\[
[\epsilon_{\text{left}}, \epsilon_{\text{right}}] \leftarrow \text{TWO-SIDED-REGRET-CI}(\bar{\sigma}, \Theta_{\text{seq}}, \gamma)
\]

**end while**

**return** \( \epsilon_{\text{right}} \leq \epsilon \)
Algorithm 2 Confidence-Interval-Based Equilibrium Finding \((\epsilon, \gamma, x)\)

 Require: \(\epsilon\), the acceptable approximation threshold
 Require: \(\gamma\), the confidence level to use
 Require: \(x\), the number of observations to take of each profile in each step

\[
\Theta_{seq} \leftarrow \{\}
\]

\[
E \leftarrow \{\}
\]

while \(E = \{\}\) do

Append \(x\) observations of each profile to \(\Theta_{seq}\)

for \(\bar{\sigma} \in \text{EQUILIBRIA}(\Theta_{seq})\) do

If one-sided-regret-CI\((\bar{\sigma}, \Theta_{seq}, \gamma)\) \(\leq \epsilon\) then

Append \(\bar{\sigma}\) to \(E\)

end if

end for

end while

return \(E\)

and if the right-hand side of this interval is not greater than \(\epsilon\), the profile is appended to the set of equilibria. When one or more equilibria of the empirical game meet this criterion, sampling is terminated and the candidates meeting the criterion are returned.

4. EXPERIMENTS

4.1 Regret Bootstrap Experiments

For our bootstrap regret confidence intervals to be useful, we need to show that they are well-calibrated. We hypothesize that the 95\(^{th}\) percentile of the sample-game bootstrap distribution of the regret of a candidate equilibrium provides an accurate 95\(^{th}\) confidence bound for the true-game regret of that candidate. We are not aware of sufficient theoretical foundations to prove this hypothesis, so we test it experimentally.

4.1.1 Experimental Setup

Our hypothesis yields several testable predictions: most importantly, the confidence bound should be well-calibrated, namely the true-game regret of an equilibrium candidate should fall below the 95\(^{th}\) percentile of the bootstrap distribution 95\(^{th}\) of the time. We would also expect the other quantiles of the bootstrap distribution to be well-calibrated. In addition, confidence bounds should grow tighter as data is acquired, so the 95\(^{th}\) percentile of the bootstrap distribution should shrink as the number of payoff observations grows. We also expect confidence bounds to be wider when data are more noisy, so the 95\(^{th}\) percentile should grow as the variance of payoff samples grows.

We test these hypotheses by artificially generating true games and drawing samples from known noise distributions centered around each true game payoff. We then compute pure-strategy Nash equilibria and symmetric mixed-strategy Nash equilibria in the resulting empirical games\(^3\), and use our bootstrap method to estimate the distribution of each candidate equilibrium’s regret. These bootstrap estimates are compared against the true-game regrets of the equilibrium candidates. Across a large number of randomly generated true games, our hypothesis predicts that \(k\%\) of

\[\text{true-game regret values will fall below the } k^{\text{th}} \text{ percentile of the empirical game’s bootstrap regret distribution, especially when } k = 95.\]

Our experiments employ two classes of synthetic games: uniform symmetric games (uSym) and congestion games (Cgst), as well as one class of simulated game: credit network games (CredNet). To generate a true game from the uSym class, we draw a value from the distribution \(U[0, 100]\) for each unique payoff in a symmetric game with \(p \in \{2, 4\}\) players and \(s \in \{2, 4, 6\}\) strategies. All results presented here use uSym games with 4 players and 4 strategies; results for other combinations of players and strategies are similar. To generate a true game from the Cgst class, we use 5 players and 3 strategies; each strategy \(s\) has a base value \(u_b(s) \sim U[0, 3]\), a linear congestion cost \(v_1(s) \sim U[0, 1]\), and a quadratic congestion cost \(v_2(s) \sim [0, 1]\). The payoff to a player choosing strategy \(s\) is a function of the total number \(n(s)\) of players choosing that strategy: \(u(s) = u_b(s) - v_1(n(s)) - v_2(n(s))^2\). CredNet games are generated based on data from a simulator described by Dandekar, et al. [6]. In our initial experiments, we generated a CredNet game with 6 players, 6 strategies, and 2644 samples of each payoff, but found that it had particularly high variance; therefore we also generated a second data set with the same players and strategies called CredNet agg., where each of 1000 samples comes from 20 pre-aggregated runs of the simulator. The true game in our CredNet experiments is always the empirical game constructed using the full set of samples. To facilitate comparison of regret values across classes, we applied an affine transformation to rescale each uSym and Cgst payoff matrix to match range \([0, 100]\), which closely matches the payoff range of the CredNet true game.

<table>
<thead>
<tr>
<th>Game, Noise</th>
<th>Size</th>
<th>.95 Frac.</th>
<th>.95(\epsilon)</th>
<th>.95 Frac.</th>
<th>.95(\epsilon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>uSym, normal</td>
<td>10</td>
<td>0.924</td>
<td>34.4</td>
<td>0.951</td>
<td>25.9</td>
</tr>
<tr>
<td>uSym, normal</td>
<td>100</td>
<td>0.947</td>
<td>1.5</td>
<td>0.955</td>
<td>6.3</td>
</tr>
<tr>
<td>uSym, bimodal</td>
<td>10</td>
<td>0.949</td>
<td>71.6</td>
<td>0.957</td>
<td>50.1</td>
</tr>
<tr>
<td>uSym, bimodal</td>
<td>100</td>
<td>0.935</td>
<td>13.5</td>
<td>0.949</td>
<td>12.7</td>
</tr>
<tr>
<td>Cgst, normal</td>
<td>10</td>
<td>0.928</td>
<td>20.6</td>
<td>0.966</td>
<td>18.1</td>
</tr>
<tr>
<td>Cgst, normal</td>
<td>100</td>
<td>0.972</td>
<td>0</td>
<td>0.941</td>
<td>1.8</td>
</tr>
<tr>
<td>CredNet, agg.</td>
<td>10</td>
<td>0.981</td>
<td>1.51</td>
<td>0.997</td>
<td>1.04</td>
</tr>
<tr>
<td>CredNet, agg.</td>
<td>100</td>
<td>0.971</td>
<td>0</td>
<td>0.927</td>
<td>0.23</td>
</tr>
</tbody>
</table>

Table 1: Bootstrap confidence intervals for regret.

\(^3\)In all of our experiments, mixed-strategy equilibria are computed using replicator dynamics [9].
4.2 Bootstrap in Sample Control

Frequently, rules of thumb are used to decide when to terminate sampling in iterative applications of EGTA, such as sampling until the uncertainty in payoff estimates is below some threshold; however, the theoretical underpinnings of the bootstrap assume a fixed-sample statistical estimate, and as such may not deliver reliable inference when applied to EGTA applications where the sampling process is terminated based upon a feature of the gathered data. To empirically evaluate the usefulness of bootstrapping regret for iterative EGTA approaches, including its use in sample control, we conduct two sets of experiments. In the first set of experiments, we evaluate the accuracy of using bootstrapped regret confidence intervals to decide whether a mixed-strategy profile is an approximate equilibrium in the algorithm CIBSR. In the second set, we test the accuracy of bootstrapped regret confidence intervals at the termination of sampling with CIBEF and two commonly-applied rules of thumb to determine (i) whether the bootstrap approach is reliable when working with sequentially gathered data and (ii) whether CIBEF provides any reduction in sample costs or expected regret of candidate profiles when compared to existing rules of thumb.

4.2.1 Labeling Equilibria

To test the efficacy of CIBSR at labeling profiles as either \( \epsilon \)-Nash equilibria (hereafter referred to as Eq) or not (hereafter referred to as Not-Eq), we measure the frequencies at which the algorithm correctly labels candidate profiles from synthetic or simulated game data, as in Section 4.1. For each game type considered, 1000 trials were run, where each trial consisted of labeling a single candidate profile, which may be either Eq or Not-Eq with respect to the true game. Similar to the experiments in Section 4.1, for synthetic games, each trial corresponds to a different randomly generated game, while a trial with simulation game data corresponds to a different random ordering of observations of a fixed data set, due to the cost of gathering additional simulation data. This means that for the simulated game trials, the set of approximate equilibria of the true game remains fixed across trials.

Selecting profiles randomly for evaluation would be an insufficiently stringent test for the algorithm, since Not-Eq profiles with high regret can be labeled with confidence with very few observations. Furthermore, such profiles would not merit application of statistical tools in practice, as a mixed-strategy profile is typically only of interest if it is believed to be a close approximation of equilibrium. We therefore construct candidate profiles by taking a small number of observations of each profile and computing an equilibrium of the current empirical game. These observations are then passed to the algorithm as \( \Theta_{init} \). With this procedure we are
able to generate candidate profiles of either type, and Not-Eq instances will frequently be low regret, making correct labeling appropriately difficult.

For the synthetic games, candidates are selected after taking 5 observations of each profile. On each trial the algorithm is parameterized with a regret threshold $\epsilon = 0.05$ and step size of $x = 5$ observations. Each trial also has an observation cap of 1000 observations per profile, at which point, if the algorithm has not terminated, it labels the point as Eq if the median of the bootstrap distribution is below $\epsilon$. For the credit network simulator, we explored both the unaggregated and aggregated data sets described in Section 4.1. Due to relatively high level of noise in the credit network data, candidates were chosen after 100 observations for the unaggregated data, and after 5 observations for the aggregated data. Similarly, the algorithm is parameterized with $x = 100$ for the unaggregated data and $x = 5$ for the aggregated data, with observation caps set at the size of the full data set, 2644 and 1000 respectively. Experiments on these games were conducted for $\epsilon \in \{0.05, 0.2, 0.5\}$, as different settings of $\epsilon$ lead to a different distribution of Eq and Not-Eq instances, and potentially change the difficulty of correct labeling. For all experiments presented here, the algorithm is parameterized with $\gamma = 0.95$. As the algorithm uses a two-sided $\gamma$-confidence interval and the final decision only depends on one of the boundaries of the interval, the algorithm terminates with a confidence level of 0.975. Table 2 presents selected findings from these experiments. With the exception CredNet instances which use simulation data, all experiments reported here used normally distributed noise, with standard deviation given by $\sigma$ in the table, to generate payoff observations. As in the prior experiments, for synthetic games $\sigma \in \{0.1, 1, 10, 100\}$ were evaluated, but only $\sigma = 1$ and $\sigma = 100$ are presented due to the similarity of the results. “Inst. Type” specifies whether the row refers to instances where the ground truth is Eq or Not-Eq. “#” specifies the number of instances out of the 1000 trials that were of the named type, while “Und. #” gives the number of trails for that type where the algorithm was undecided after reaching the observation cap imposed in the experiment. “Accuracy” lists first the fraction of labelings that were correct for the trials where the algorithm terminated with a confident decision, and second the fraction correct when the algorithm was forced to decide at the observation cap. “Agg. Acc.” specifies the the fraction of labelings that were correct across all 1000 trials, including both Eq and Not-Eq instances.

Despite our specific choice of instances that were difficult to correctly label, CIBSR using the bootstrap method to construct confidence intervals delivered high levels of accuracy across all game types and parameter settings, with the lowest accuracy observed over a full set of trials being 0.922. On synthetic data, CIBSR delivered accuracy of at least 0.97 across all games and instance types. The credit network data proved more difficult, and despite overall high levels of accuracy, accuracy varied considerably between experiments and instance types. In particular, low regret thresholds made the algorithm less likely to label candidates as equilibria; this is reflected in the high accuracy for Non-Eq, low accuracy for Eq instances, and higher incidence of inconclusive results.

4Since each aggregated observation averages 20 observations, the candidate profiles for both experiments are constructed after the same number of simulations.

<table>
<thead>
<tr>
<th>Game</th>
<th>$\epsilon$</th>
<th>Inst. Type</th>
<th>#</th>
<th>Und. #</th>
<th>Accuracy</th>
<th>Agg. Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>uSym</td>
<td>.05</td>
<td>Eq</td>
<td>624</td>
<td>22</td>
<td>.99, .77</td>
<td>.978</td>
</tr>
<tr>
<td>$\sigma = 1$</td>
<td>.05</td>
<td>Not</td>
<td>376</td>
<td>21</td>
<td>.98, .76</td>
<td></td>
</tr>
<tr>
<td>uSym</td>
<td>.05</td>
<td>Eq</td>
<td>240</td>
<td>7</td>
<td>.97, 1</td>
<td>.972</td>
</tr>
<tr>
<td>$\sigma = 1$</td>
<td>.05</td>
<td>Not</td>
<td>760</td>
<td>11</td>
<td>.98, .36</td>
<td></td>
</tr>
<tr>
<td>Cgst</td>
<td>.05</td>
<td>Eq</td>
<td>911</td>
<td>6</td>
<td>1, .67</td>
<td>.995</td>
</tr>
<tr>
<td>$\sigma = 1$</td>
<td>.05</td>
<td>Not</td>
<td>89</td>
<td>1</td>
<td>.98, 1</td>
<td></td>
</tr>
<tr>
<td>Cgst</td>
<td>.05</td>
<td>Eq</td>
<td>610</td>
<td>4</td>
<td>.99, .25</td>
<td>.984</td>
</tr>
<tr>
<td>$\sigma = 1$</td>
<td>.05</td>
<td>Not</td>
<td>390</td>
<td>6</td>
<td>.98, .83</td>
<td></td>
</tr>
<tr>
<td>CredNet</td>
<td>.05</td>
<td>Eq</td>
<td>96</td>
<td>84</td>
<td>0, .83</td>
<td>.974</td>
</tr>
<tr>
<td></td>
<td>Not</td>
<td>904</td>
<td>250</td>
<td>1</td>
<td>1, 1</td>
<td></td>
</tr>
<tr>
<td>CredNet</td>
<td>.5</td>
<td>Eq</td>
<td>642</td>
<td>558</td>
<td>.75, .90</td>
<td>.922</td>
</tr>
<tr>
<td></td>
<td>Not</td>
<td>358</td>
<td>152</td>
<td>1</td>
<td>.91, 1</td>
<td></td>
</tr>
<tr>
<td>CredNet,</td>
<td>.05</td>
<td>Eq</td>
<td>428</td>
<td>122</td>
<td>.86, .98</td>
<td>.953</td>
</tr>
<tr>
<td>agg.</td>
<td>Not</td>
<td>572</td>
<td>64</td>
<td>1</td>
<td>.99, 1</td>
<td></td>
</tr>
<tr>
<td>CredNet,</td>
<td>.2</td>
<td>Eq</td>
<td>774</td>
<td>156</td>
<td>.96, .95</td>
<td>.965</td>
</tr>
<tr>
<td>agg.</td>
<td>Not</td>
<td>226</td>
<td>18</td>
<td>1</td>
<td>.98, 1</td>
<td></td>
</tr>
<tr>
<td>CredNet,</td>
<td>.5</td>
<td>Eq</td>
<td>997</td>
<td>128</td>
<td>.98, 1</td>
<td>.983</td>
</tr>
<tr>
<td>agg.</td>
<td>Not</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1, 1</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Sequential classification performance.

for Eq instances. The observation that the entire data set for the credit network game was frequently insufficient to have high confidence in declaring that a candidate profile was an $\epsilon$-equilibrium reaffirms the difficulty of working with the CredNet data set.

4.2.2 Sequential Equilibria Finding

The previous experiment demonstrated that the bootstrap method of constructing confidence intervals can successfully be used as terminating condition for a sequential statistical experiment without drastically biasing inferences drawn at the conclusion of the experiment. In contrast to the rules of thumb typically used to terminate iterative applications of EGTA, using a confidence-interval-based stopping rule yields statements about the likely values of the underlying regret; however, nothing precludes using the bootstrap method to estimate the regret distribution at the conclusion of an application of EGTA that used a rule of thumb to determine when to terminate sampling. In this section we conduct an experiment to evaluate the accuracy of the bootstrap method applied to the conclusion of sampling using two common rules of thumb, as well as CIBEF. Furthermore, we compare the performance of these rules in terms of average regret and number of observations requested.

The first rule of thumb considered is to cease sampling when all relevant payoff estimates demonstrate low variability; specifically, the stopping rule labeled SEM will request $x$ further observations be made of each profile in each step until the estimated standard error in mean of each payoff is below a specified threshold $\xi$. The intuition behind this stopping rule is that reliable estimates of payoffs should lead to reliable inferences about the true game. The second rule we consider ceases sampling when the set of equilibria of the empirical game does not change with additional observations. This stopping rule, labeled EQC (for equilibrium comparison), will request $x$ further observations be made for each profile until the sets of empirical game equilibria found in successive steps are equivalent. Distributions used
in mixed-strategy equilibria are considered to be equivalent if their Euclidean distance is below some threshold $\Delta$. Both rules of thumb prescribe a stopping point at which equilibria of the empirical game are considered approximate equilibria of the true game, but provide no guarantee that the profiles that they identify are $\epsilon$-equilibria for any particular $\epsilon$.

In this experiment, a trial consists of the specified algorithm requesting observations from a synthetic or simulation-based game model until its stopping condition is triggered or an observation cap is reached, at which point it returns one or more equilibrium candidates. If the observation cap is reached, CIBEF returns the equilibrium of the empirical game with the lowest right-hand one-sided confidence bound, while SEM and EQC return all equilibria of the empirical game. For each candidate we record the regret of each candidate profile in the true game, as well as the number of observations taken of each profile prior to terminating the sampling procedure.\(^5\) For each game model and algorithm, we ran 1000 trials, with each trial corresponding to a new random game for the synthetic game data, and new random reordering of the data for the simulated game data. In addition, metrics about the number of observations and regret of the selected profiles, we measure the average fraction of regret of the true games captured by the 95th-percentile of bootstrapped regret distributions calculated at the termination of each trial, similar to Section 4.1. This measure gives an indication of the accuracy of using the bootstrap approach to give a confidence interval at the termination of sampling, when sampling is guided by these sample control algorithms.

Table 3 shows selected results from these experiments. “Mean Obs.” refers to the average number of observations taken of each profile when the algorithm terminated. All trials were conducted with the same step sizes and observation caps as in Section 4.2.1. For the SEM stopping rule, the threshold $\xi$ was set to 1.0, while for the EQC rule, a distance less than $\Delta = 0.01$ was considered sufficient to call two equilibrium candidates identical. For all experiments conducted with the CIBEF stopping rule, $\epsilon$ was set to 0.5 and $\gamma$ was set to 0.95.

These results emphasize the general applicability of the bootstrap method of generating confidence intervals, even for sequential sampling experiments. The only experiments that resulted in substantial overconfidence on average, that is experiments in which the fraction of true game regrets captured by 95th percentile of the bootstrap regret distribution is less than 95%, were those settings in which sampling typically halted at the first opportunity. This outcome mirrors the results from Section 4.1, where we noted that for very small numbers of observations, the bootstrap method could be poorly calibrated. In contrast, some combinations of game models and stopping rules led to overly large confidence intervals, with greater than 95% of true game regrets being captured by the confidence interval method. In such cases the bootstrap method may be conservative in declaring candidates as equilibria, occasionally ruling out more profile candidates than is warranted by the expected confidence level; however, using CIBEF an equilibrium meeting the confidence requirements will eventually be found.

In comparing the equilibrium-finding characteristics of the three stopping rules, our experiments show that CIBEF is typically comparable to and often an improvement over existing rules of thumb. For every game model considered, the median regret of the profiles returned by CIBEF was considerably below the approximation threshold of $\epsilon = 0.5$. Similarly, the mean regret was below the threshold for all but the uniform symmetric game model with the highest variance. Our data suggests that in most scenarios when CIBEF misidentifies a profile as an approximate Nash equilibrium it is still likely to have regret close to the threshold, but for high noise settings, profiles that are incorrectly returned may have significant regret. EQC nearly always yielded higher regret profiles than CIBEF, and performed particularly poorly in the synthetic game models with high variance. SEM frequently performed at the same level or better than CIBEF in terms of regret, but almost always terminated either immediately or at the observation cap. In terms of the number of observations taken prior to stopping, CIBEF was similar to SEM for low noise settings, but required many fewer observations for high noise settings. In contrast, CIBEF required fewer observations than EQC in low noise settings, in part due to EQC requiring two sampling steps prior to terminating, but required slightly more samples in noisier settings.

CIBEF outperformed EQC and SEM in terms of mean regret of returned candidates and the average number of observations taken prior to stopping on the aggregated credit network data. All rules performed excellently in terms of median regret, meaning that either CIBEF returned fewer non-equilibrium candidates or that the non-equilibrium candidates that it returned were closer approximations to equilibrium than the other two stopping rules. Here, CIBEF may benefit from often returning one candidate that is highly likely to be an equilibrium, rather than returning multiple candidates that may vary greatly in how well they approximate equilibria, as in EQC or SEM. As such, CIBEF can deliver significant savings in terms of sampling costs when finding only one equilibrium is acceptable. We were, however, unable to present results for the credit network game with unaggregated data, as this experiment proved too costly, particularly for CIBEF, as it must find equilibria and calculate confidence intervals for them in every sampling step. Though a potential detriment to CIBEF, in real applications of EGTA the cost of sampling will typically outweigh

<table>
<thead>
<tr>
<th>Game</th>
<th>Rule</th>
<th>Mean Obs.</th>
<th>Mean Regret</th>
<th>Median Regret</th>
<th>.95 Frac.</th>
</tr>
</thead>
<tbody>
<tr>
<td>uSym</td>
<td>EQC</td>
<td>17.07</td>
<td>.0807</td>
<td>.0107</td>
<td>.94</td>
</tr>
<tr>
<td></td>
<td>SEM</td>
<td>5.02</td>
<td>.0073</td>
<td>.0014</td>
<td>.90</td>
</tr>
<tr>
<td></td>
<td>CIBEF</td>
<td>5.08</td>
<td>.0081</td>
<td>.0018</td>
<td>.90</td>
</tr>
<tr>
<td>σ = 100</td>
<td>SEM</td>
<td>78.66</td>
<td>2.641</td>
<td>.6669</td>
<td>.96</td>
</tr>
<tr>
<td></td>
<td>CIBEF</td>
<td>100.00</td>
<td>.5176</td>
<td>.1168</td>
<td>.96</td>
</tr>
<tr>
<td>Cgst</td>
<td>EQC</td>
<td>20.06</td>
<td>.899</td>
<td>1.44e-6</td>
<td>.98</td>
</tr>
<tr>
<td></td>
<td>SEM</td>
<td>100.00</td>
<td>.0771</td>
<td>2.11e-6</td>
<td>.94</td>
</tr>
<tr>
<td></td>
<td>CIBEF</td>
<td>26.94</td>
<td>.1703</td>
<td>1.58e-6</td>
<td>.97</td>
</tr>
<tr>
<td>Cbst</td>
<td>EQC</td>
<td>66.71</td>
<td>.0468</td>
<td>1.03e-7</td>
<td>.98</td>
</tr>
<tr>
<td></td>
<td>SEM</td>
<td>116.05</td>
<td>.0386</td>
<td>1.96e-7</td>
<td>.95</td>
</tr>
<tr>
<td></td>
<td>CIBEF</td>
<td>11.10</td>
<td>.0295</td>
<td>1.70e-7</td>
<td>.99</td>
</tr>
</tbody>
</table>

Table 3: Stopping rule performance.

\(^5\)All algorithms considered here sample evenly from all profiles in a game’s profile space, so “10 observations” refers to 10 observations taken of every value in the payoff matrix.
the cost of calculating confidence intervals, thus making the overhead of using CIBEF over EQC negligible.

5. CONCLUSION

Our experimental evidence demonstrates that the bootstrap method of confidence interval generation is approximately accurate for bounding the true-game regret of candidate equilibria in empirical games. Despite possible repeated testing concerns, our bootstrap method also proves approximately accurate in constructing confidence intervals on regret at the conclusion of sequential sampling procedures. Accuracy is lower in our experiments on credit network games than on randomly-generated games, but we believe that our random game experiments may be more representative of EGTA in practice, due to limitations of conducting experiments with costly simulation data. Because our experiments show bootstrap confidence intervals to be accurate across multiple synthetic game classes, as well as across noise distributions and magnitudes, we recommend that practitioners of empirical game theory employ bootstrap methods to give regret bounds for reported equilibria. This recommendation stands even if bootstrap regret estimates are used to determine when to conclude sampling.

In addition, we provide evidence that the Confidence-Interval-Based Equilibrium Finding algorithm improves over previous EGTA experiment designs. Relative to rules of thumb for sample control, CIBEF can more consistently identify low regret equilibrium candidates, and often requires fewer observations. However, in games with particularly noisy payoffs, where misclassifications can be particularly egregious in terms of regret, we recommend that practitioners err on the side of caution and collect extra observations. Given the savings in observations CIBEF demonstrated, we believe that using bootstrapped confidence intervals on regret is a promising tool for sample control in this domain.

This work constitutes a first systematic effort to develop practical statistical methods for EGTA; future work could focus on developing theoretical foundations of applying the bootstrap to empirical games, and characterizing games for which the bootstrap approach is reliable. Additionally, there are other measures of interest to EGTA practitioners, such as social welfare, that may also benefit from using the bootstrap for statistical analysis. Other avenues of research include evaluating different bootstrap designs, and using information obtained through the bootstrap to guide more sophisticated sampling, such as profile exploration [13].

6. REFERENCES


C Strategic Formation of Credit Networks
Strategic Formation of Credit Networks

Pranav Dandekar∗ Ashish Goel† Michael P. Wellman‡ Bryce Wiedenbeck§

February 28, 2014

Abstract

Credit networks are an abstraction for modeling trust among agents in a network. Agents who do not directly trust each other can transact through exchange of IOUs (obligations) along a chain of trust in the network. Credit networks are robust to intrusion, can enable transactions between strangers in exchange economies, and have the liquidity to support a high rate of transactions. We study the formation of such networks when agents strategically decide how much credit to extend each other. When each agent trusts a fixed set of other agents, and transacts directly only with those it trusts, the formation game is a potential game and all Nash equilibria are social optima. Moreover, the Nash equilibria of this game are equivalent in a very strong sense: the sequences of transactions that can be supported from each equilibrium credit network are identical. When we allow transactions over longer paths, the game may not admit a Nash equilibrium, and even when it does, the price of anarchy may be unbounded. Hence, we study two special cases. First, when agents have a shared belief about the trustworthiness of each agent, the networks formed in equilibrium have a star-like structure. Though the price of anarchy is unbounded, simple greedy dynamics quickly converge to a social optimum. Similar star-like structures are found in equilibria of heuristic strategies found via simulation. In addition, we simulate a second case where agents may have varying information about each others’ trustworthiness based on their distance in a social network. Empirical game analysis of these scenarios suggests that star structures arise only when defaults are relatively rare, and otherwise, credit tends to be issued over short social distances conforming to the locality of information.

∗Department of Management Science & Engineering, Stanford University, Stanford, CA. Email: ppd@stanford.edu.
†Departments of Management Science & Engineering and (by courtesy) Computer Science, Stanford University, Stanford, CA. Email: ashishg@stanford.edu.
‡Division of Computer Science & Engineering, University of Michigan, Ann Arbor, MI. Email: wellman@umich.edu.
§Division of Computer Science & Engineering, University of Michigan, Ann Arbor, MI. Email: btwied@umich.edu.
1 Introduction

The study of strategic network formation seeks to understand the emergent behavior and properties of a network when self-interested agents establish connections to one another based on their local information. In general, establishing a connection incurs a cost but also yields some benefit to connected agents. The agents are deemed to be utility-maximizing, that is, they make decisions in order to maximize the difference between their total benefit and their total cost. This problem has been studied in many different settings [Jackson and Wolinsky, 1996; Bala and Goyal, 2000; Fabrikant et al., 2003; Corbo et al., 2006; Anshelevich and Hoefer, 2012]. One can ask interesting questions about the emergent properties of the networks formed in each setting: What network topologies are feasible in equilibrium? How do equilibrium networks differ from socially optimal ones? How does this depend upon the cost of forming an edge and the benefit derived from having a connection? If there are multiple equilibria, how can agents select among them? This paper is an investigation into some of these questions in the context of credit networks, an abstraction for modeling trust among agents.

1.1 Credit Networks

Internet-based content sharing systems (e.g., BitTorrent, YouTube, Wikipedia, email) often use virtual currencies, known as scrip, to limit abuse such as spam and free-riding [Vishnumurthy et al., 2003; Aperjis and Johari, 2006; Reeves et al., 2007]. A scrip system is a centralized currency infrastructure where a single entity issues currency. Credit networks, by contrast, represent a model of decentralized currency. Agents in the network print their own currency, and commit to accept a certain amount of each other’s currency. This allows agents to pay for goods and services using their own credit, instead of using a common currency.

A credit network represents these credit relationships through a directed graph with edge capacities. Nodes in this graph correspond to agents, and edges correspond to credit relationships between them. An edge of capacity $c$ from node $u$ to node $v$ indicates that agent $u$ extends $c$ units of credit to agent $v$, or equivalently, $u$ is committed to accept IOUs (obligations) issued by $v$ up to value $c$. It is possible that in the future $v$ refuses to honor his outstanding obligations, thereby leaving $u$ with irredeemable IOUs. Therefore, the capacity $c$ can be viewed as a measure of $u$’s trust in $v$.

**Illustrative Example** Credit commitments between trusting nodes also enable remote transactions, as illustrated in Fig. 1. Say node $w$ wants to buy a good worth $p$ units from node $u$. Nodes $u$ and $w$ can transact—even though $u$ does not directly trust $w$—via the trusted intermediary $v$. Assuming $p \leq \min\{c_1, c_2\}$, the payment proceeds by $w$ issuing an IOU to $v$ worth $p$ units, and $v$ issuing an IOU to $u$ worth $p$ units. If, however, $p > \min\{c_1, c_2\}$, the transaction fails. As a result of a successful transaction, the credit capacities $c_{uv}$ and $c_{vw}$ decrease by $p$, representing the remaining credit commitments. In addition, the capacities $c_{vu}$ and $c_{wv}$ both increase to $p$ from zero, since $v$ and $w$ will both accept the return of their own IOUs as payment.

Thus arbitrary payments can be routed through a credit network by passing IOUs along a chain of trusting agents. Observe that routing payments in credit networks is identical to routing residual flows in single-commodity flow networks. Also note that payment flows in the opposite direction of credit, so a payment merely results in a redistribution of credit: buyers expend credit, sellers gain it, and intermediaries exchange credit between their neighbors, with the total credit in the network unchanged.
1.1.1 Origins of the Model

The credit network model was invented independently by (at least) four distinct groups of researchers, motivated by somewhat different issues and applications, but arriving at the same essential elements.

- DeFigueiredo and Barr [2005] sought a reputation system with bounded loss from coalitions of malicious users.
- Ghosh et al. [2007] aimed to support distributed payment and multi-user credit checking for multi-item auctions.
- Mislove et al. [2008] were concerned with deterring spam.

A common thread in the objectives of these researchers was to capture a notion of pairwise trust, representable in quantified terms. In each case, the trust measure is grounded by interpreting the quantity as a capacity for transaction. That is, the degree of trust in one agent for another is measured by how much it is willing to expose itself to transactions with that counterpart. In other words, the model operationalizes trust as an extension of credit, in a framework where a credit balance entitles an agent to transact with the agent granting credit. The common underlying credit model of these four proposals was first noticed by Dandekar et al. [2011], who introduced the unifying term “credit network” and its formal definition. By introducing suitable definitions of transaction, credit networks can support a wide variety of applications. For example, the inventors enumerated above interpret transactions respectively as obtaining references guaranteeing good behavior [DeFigueiredo and Barr, 2005], paying for auction winnings [Ghosh et al., 2007], borrowing an asset [Karlan et al., 2009], and communicating messages [Mislove et al., 2008]. Subsequent authors proposed using this framework to support networked asynchronous bilateral trading [Liu et al., 2010], and bartering of tutorial services [Limpens and Gillet, 2011].

1.1.2 Properties

Routing payments along chains of trust ensures that agents hold IOUs issued only by other agents that they directly trust. As a result, if an agent defaults on his outstanding obligations, the only agents that incur a loss are those that extended credit to the defaulting agent. Thus losses from default are localized. Moreover, the total loss incurred is bounded by the total credit extended to the defaulting agent. These properties make credit networks robust against two attacks that centralized currency systems are vulnerable to: whitewashing [Friedman and Resnick, 2001], and Sybil attacks [Friedman et al., 2007]. Viswanath et al. [2012] argue that in fact all reputation schemes designed for Sybil tolerance have essentially been versions of the credit network idea.
They propose an approximation to the max-flow calculation that enables scalability to very large networks.

The effectiveness of credit networks in supporting distributed transactions was most powerfully demonstrated by Dandekar et al. [2011]. Their analysis posits that nodes repeatedly transact with each other according to a known probability distribution. In particular they showed analytically and via simulations that for several classes of graphs and with symmetric transaction probabilities, the long-term transaction failure probability in credit networks is comparable to that in equivalent centralized currency systems. Thus, in addition to being robust against attacks by malicious agents, credit networks also provide a high degree of liquidity: the ability to sustain long sequences of transactions.

1.2 Formation of Credit Networks

Extending credit to other agents increases liquidity in the network, enabling more profitable transactions to go through. However, it also entails risk, since a counterparty might default on its outstanding obligations. This raises the natural question: if agents rationally weigh these risks and benefits, what kinds of networks will they form? In order to use credit networks for practical applications, it is critical to understand the structural and economic properties of the credit networks formed by strategic agents. We address this question in this paper.

1.2.1 Our Setting

In our model, agents play a one-shot game where they determine how much credit to extend other agents, and then engage in repeated probabilistic transactions over the formed credit network. Agents derive value from successful transactions. Extending credit to other agents increases transaction success probability, thus contributing to utility. On the other hand, if an agent defaults on its creditors, those agents who extended credit to the defaulter suffer a loss. Thus, an agent’s net utility is its total value from successful transactions minus the utility loss from extending credit to untrustworthy agents.

We study the formation of credit networks under various models of risk analytically as well as through simulations. Our simulations employ an approach known as empirical game-theoretic analysis (EGTA) [Wellman, 2006]. Network formation in the presence of risk was recently studied by Blume et al. [2011] in a model motivated by financial contagion and epidemic diseases. In their setting, nodes derive utility only from direct edges, whereas risk is contagious (i.e., failure of distant nodes is also a source of risk). The credit network model flips this: nodes derive benefit from transactions along direct as well as multi-hop paths, whereas only direct edges are sources of risk.

We start with a model of dichotomous risk: agents are embedded in a social network represented by an undirected graph. Agents trust their neighbors in the social network and may extend credit to them. However, they associate a very high loss of utility with extending credit to non-neighbors, and consequently, never extend credit to them. This setting captures situations illustrated by the following examples where directly transacting with a stranger may have grave consequences.

- During a disease epidemic within a human population, high-risk groups will limit their interactions to those who belong to similar social circles. Evidence of this has been found, for example, in the setting of HIV/AIDS [Jacquez et al., 1988; Barnard, 1993].
- Users trying to circumvent Internet censorship and evade network surveillance in repressive regimes make use of Internet proxies [Mahdian, 2010]. If caught, penalties may be severe.
Thus, users rely on their friends and acquaintances to distribute proxy addresses.

- Members of covert organizations face the prospect of severe harm at the hands of the enemy if their identity is compromised. As a result, they may rely on longstanding relationships and assets built over time to conduct their business.

We also study a model of global risk, which represents the other extreme with respect to the dichotomous risk model. In this model, each node has a publicly known risk of default. This corresponds to situations involving small, densely interacting social groups, or where there are organizations such as credit-reporting agents that systematically gather and disseminate relevant risk information.

Finally, we study a model of graded risk that helps bridge the gap between global and dichotomous risk. Under this model, each agent has a private default probability. Agents receive noisy signals about each other’s probability of defaulting, and these signals are more informative for neighbors in the social network.

### 1.2.2 Our Results

**Dichotomous Risk** Under dichotomous risk, when we allow only bilateral transactions (*i.e.*, transactions only between adjacent nodes in the social network, and payments routed only along the direct edge between nodes), we show that the formation game is a potential game (Theorem 3.1). This implies that best-response dynamics always converge to a Nash equilibrium.* Moreover, for a large, natural class of transaction size distributions, we show that agents’ utilities are concave in their credit allocations. This allows us to prove that every Nash equilibrium of the game maximizes social welfare (Theorem 3.2). More interestingly, we show that the Nash equilibria are equivalent in a much stronger sense: any two Nash equilibria are cycle-reachable from each other (Theorem 3.5), which means that they support the same sequences of transactions [Dandekar et al., 2011].

With non-bilateral transactions, the game becomes significantly less well-behaved: the game may not admit a Nash equilibrium (Theorem 3.8), and even when it does, the price of anarchy in this setting can be unbounded (Theorem 3.9).

**Global Risk** Under global risk, we analyze several scenarios. First, we investigate the price of anarchy and the structure of equilibria when each agent is limited to extend credit to at most one other agent. We prove that if we disallow the empty network as an outcome, the price of anarchy of the formation game is unbounded (Theorem 4.4), even though all Nash equilibria have a star-like structure (Theorem 4.3). Instead we focus on the structure of equilibria under two simple dynamics: sequential arrival and greedy dynamics. When nodes arrive sequentially and create a single link, we show that a node $u$ always extends credit to either the node $v$ that arrived immediately before $u$ or to the node that $v$ extends credit to (Theorem 4.6). Thus the resulting network has a comb-like structure. Under greedy dynamics, nodes extend their entire credit budget to the node that has the lowest risk of default. If the default risks are unique, this results in a star-like network structure which is also the optimal structure in terms of social welfare (Theorem 4.5). Thus, even though the price of anarchy can be unbounded, nodes can easily find the optimal network using greedy dynamics.

We also use empirical game-theoretic simulations to investigate a richer model of global risk, in which agents are not constrained to a single link or a fixed budget, and transaction probabilities and values may be asymmetric. Under global risk, we find star-like equilibrium networks under

---

*In our theoretical analysis, the term Nash Equilibrium always refers to a pure Nash equilibrium. Our empirical game analysis (Section 5) considers mixed equilibria as well.*
all conditions, but settings with high default rates or low transaction surpluses also have empty network equilibria. In addition, when default rates are low, transaction-dependent credit issuing strategies can appear in equilibrium.

**Graded Risk** We address graded risk exclusively through empirical game simulations. In this setting, the star-like equilibria disappear, highlighting the importance of ensuring that central nodes are unlikely to default. Empty network equilibria are present for exactly the same settings as under global risk, indicating that the conditions under which agents issue no credit may not depend as strongly on the information structure. Transaction-dependent equilibria arise under the same conditions as for global risk as well as those with high transaction surplus.

## 2 Model and Definitions

Let $V$ denote the set of $n$ agents. Each agent $u \in V$ has a budget $B_u \geq 0$ representing the total credit that $u$ can extend to other agents in $V$. Agents play a one-shot game where they choose credit allocations to form an initial network $G$. Agents represent nodes of the formed network. An edge from node $u$ to node $v$ of capacity $c_{uv}(G)$ represents the credit extended by agent $u$ to agent $v$ in the network $G$. Thus $G$ is a directed graph with edge capacities. A strategy for agent $u$ is a set of feasible credit allocations $\{c_{uv}(G), v \in V : c_{uv}(G) \geq 0 \text{ and } \sum_{v \in V} c_{uv}(G) \leq B_u\}$.

### 2.1 Transaction Model

Once a network $G$ is formed, agents engage in repeated probabilistic transactions with each other. At each time step $t = 1, 2, \ldots$, a pair of transacting agents $(u, v)$, with $u$ being the payer (buyer) and $v$ the payee (seller), is chosen with probability $\lambda_{uv}$. The transaction rate matrix $\Lambda = \{\lambda_{uv} : u, v \in V\}$ is public, and satisfies the following properties: (i) $\lambda_{uu} = 0$, (ii) $\lambda_{uv} \geq 0$, and (iii) $\sum_{u,v} \lambda_{uv} = 1$.

Suppose agents $(u, v)$ are chosen to transact at time $t$. Then the transaction size, $x^t_{uv}$, between $u$ and $v$ is drawn from a transaction size distribution over $[0, \infty)$ with a probability density function (pdf) $X^t_{uv}(\cdot)$ and a corresponding cumulative distribution function (cdf) $X^t_{uv}(\cdot)$. We assume that the pdfs $X^t_{uv}(\cdot)$ are public. Let $X := \{X^t_{uv}(\cdot) : u, v \in V\}$ be the pdf matrix.

Given a transaction size $x$, a feasible path in the network $G$ from node $v$ to node $u$ is a set of directed edges $P = \{(v, u_1), (u_1, u_2), \ldots, (u_k-1, u_k), (u_k, u)\}$ such that for all $(w, y) \in P$, $c_{wy}(G) \geq x$. We route payments along the shortest feasible path in the network. Let $P^t_{vu}$ be the shortest feasible path in the credit network from $v$ to $u$ at time $t$. A successful transaction of size $x_{uv}^t$ results in a change of credit capacities along edges in $P^t_{vu}$ as follows. Let $G^t$ denote the state of the network $G$ at time $t = 0, 1, 2, \ldots$, where $G^0 = G$. Then, for $w, y \in V$ and a successful transaction at time $t > 0$,

$$c_{wy}(G^t) = \begin{cases} c_{wy}(G^{t-1}) - x_{uv}^t, & \text{if } (w, y) \in P^{t-1}_{vu} \\ c_{wy}(G^{t-1}) + x_{uv}^t, & \text{if } (y, w) \in P^{t-1}_{vu} \\ c_{wy}(G^{t-1}), & \text{otherwise} \end{cases}$$

So, in order for a payment $x_{uv}^t$ from $u$ to $v$ to succeed, there must exist a feasible path in the credit network from the payee $v$ to the payer $u$. If no such path exists, the transaction fails, in which case all credit capacities remain unchanged. Thus, for all $t > 0$, and for all $u, v \in V$, $c_{uv}(G^t) + c_{vu}(G^t) = c_{uv}(G) + c_{vu}(G)$.

The repeated probabilistic transactions induce a Markov chain over the states of the network, which we denote by $\mathcal{M}(G, \Lambda, X)$. A transaction regime is defined as the tuple $(\Lambda, X)$. We say a transaction regime $(\Lambda, X)$ is symmetric if the transaction rate matrix $\Lambda$ is symmetric: for all nodes
For most of the analysis, we will consider symmetric transaction regimes where, additionally, the Markov chain is ergodic.

2.2 Utility

Agents choose credit allocations to maximize their utility. Agents derive value from successful transactions, but they risk loss of utility when they extend credit to potentially untrustworthy agents. We model this risk in several ways, but denote the expected loss of utility to \( u \) associated with the prospect of default by \( v \) by \( \Delta_{uv}(G) \), with the constraints that \( \Delta_{uv}(G) \geq 0 \) and \( \Delta_{uv}(G) > 0 \) only if \( c_{uv}(G) > 0 \).

Let us assume that an agent derives value from a successful transaction when he is the payer, because it expends credit to buy something of value. In our analytical work, we characterize the value derived by an agent from a credit network in terms of the steady-state success probability of all transactions where the agent is a payer. Let \( f_{uv}(G) \) be the steady-state success probability of the transactions from \( u \) to \( v \) (i.e., where \( u \) is the payer) when the initial network is \( G \). Then, the total utility of an agent \( u \) when the initial network is \( G \) is given by

\[
U_u(G) = \gamma \sum_{w \in V} f_{uw}(G) - \sum_{v \in V : c_{uv}(G) > 0} \Delta_{uv}(G)
\]

where \( \gamma \) is a constant that converts transaction success probability into equivalent utility units. The overall social welfare in network \( G \) is simply the sum of utilities of all nodes in \( G \): \( U(G) := \sum_{u \in V} U_u(G) \).

It is difficult to characterize the steady-state transaction success probabilities for arbitrary networks and transaction regimes. However, for the settings we analyze, we are able to characterize the transaction success probabilities using results established by Dandekar et al. [2011].

2.3 Risk Model

In order to model variation in \( \Delta_{uv}(G) \), we assume that the agents are embedded in an exogenously defined social network represented by a simple undirected graph \( H = (V, E) \). The social network \( H \) influences how \( \Delta_{uv}(G) \) for an agent \( u \) varies across agents \( v \in V \). We consider three specific models of how risk changes as a function of distance between \( u \) and \( v \) in \( H \).

**Dichotomous Risk.** In this model, an agent \( u \) partitions the set of agents \( V \) into two sets using \( H \): neighbors in \( H \) and non-neighbors in \( H \). For any network \( G \), agent \( u \) estimates risk exposure to be:

\[
\Delta_{uv}(G) = \begin{cases} 
0, & \text{if } (u,v) \in E \\
\infty, & \text{otherwise} 
\end{cases}
\]

This model assumes agents are willing to interact only with their neighbors in \( H \). For any credit network \( G \) formed under this model, \( c_{uv}(G) = 0 \) if \( (u,v) \notin E \).

**Global Risk.** In this model, we assume that each agent \( v \) has a default probability \( \delta_v \in (0, 1] \) which is public. If \( v \) defaults, a node \( u \) that extended credit \( c_{uv}(G) \) to \( v \) loses \( c_{uv}(G) \) units. Thus, \( \Delta_{uv}(G) = \delta_v c_{uv}(G) \).

**Graded Risk.** Here, as in the Global Risk model, each agent \( v \) has default probability \( \delta_v \), but this information is not publicly known. Instead, each agent \( u \) receives a signal \( \delta_{uv} \) about the default probability of each other agent \( v \). These signals are decreasingly informative with distance in \( H \), so agents know much more about the default probabilities of their neighbors in the social network.
network than about distant nodes. In our simulations, we implement this by drawing agents’
default probabilities from a beta distribution: \( \delta_u \sim \text{Beta}(\alpha, \beta) \). Agent \( u \) then receives a signal in
the form of some number of samples \( \partial_{uv} \) drawn from the binomial distribution on \( \delta_v \), where \( \partial_{uv} \)
decreases exponentially with social network distance.

3 Network Formation under Dichotomous Risk

Recall that under dichotomous risk, \( \Delta_{uv}(G) \) is defined by (2.2); as a result nodes only extend credit to their neighbors in \( H \).

3.1 Symmetric Bilateral Transactions

We call a transaction between nodes \( u \) and \( v \) bilateral if \( (u, v) \in E \) and the payment is routed along the edge \( (u, v) \). Here we allow only bilateral transactions: if a payment between adjacent nodes \( u \) and \( v \) cannot be routed along the direct edge \( (u, v) \), we fail the transaction. As a result, if \( (u, v) \notin E \), the steady-state success probability \( f_{uv}(G) = f_{vu}(G) = 0 \). Moreover, the steady-state transaction success probabilities along an edge \( e = (u, v) \) in a network \( G \) are governed only by the credit allocations, \( c_{uv}(G), c_{vu}(G) \), along \( e \) in \( G \). We also assume that the transaction regime \( \langle \Lambda, X \rangle \) is symmetric and that \( \lambda_{uv} > 0 \) if \( (u, v) \in E \). As a result, for all nodes \( u \) and \( v \), \( f_{uv}(G) = f_{vu}(G) \).

In our analysis of the symmetric bilateral transaction regime, for an edge \( e = (u, v) \in E \), we will use \( \lambda_{uv}, X_{uv}(\cdot), X_{vu}(\cdot) \) and \( f_{e}(\cdot) \) to denote \( \lambda_{uv}, X_{uv}(\cdot), X_{vu}(\cdot) \), and \( f_{uv}(\cdot) \), respectively. We first show that in this setting, the network formation game is a potential game.

**Theorem 3.1.** The network formation game under a symmetric bilateral transaction regime is a potential game.

**Proof.** Consider the function \( \Phi(G) \) defined as

\[
\Phi(G) := \frac{U(G)}{2} = \frac{1}{2} \sum_{u \in V} U_u(G) = \frac{\gamma}{2} \sum_{e \in E} \sum_{u,v \in V} f_{uv}(G)
\]

Since we are in a symmetric bilateral transaction regime, \( f_{uv}(G) = f_{vu}(G) \) for all \( (u, v) \in E \), and \( f_{uv}(G) = 0 \) if \( (u, v) \notin E \). Therefore,

\[
\sum_{u \in V} \sum_{v \in V} f_{uv}(G) = 2 \sum_{e \in E} f_e(G)
\]

This implies \( \Phi(G) = \gamma \sum_{e \in E} f_e(G) \). We will show that \( \Phi(G) \) is a potential function. Fix a node \( u \in V \). Consider a network \( G' \) which differs from \( G \) only in the credit allocation of \( u \). Formally, for all \( w, y \in V \),

\[
c_{wy}(G') = \begin{cases} 
  c_{wy}(G), & \text{if } w \neq u \\
  c'_{wy}, & \text{if } w = u \text{ and } (u, y) \in E
\end{cases}
\]

where \( \{c'_{wy} : (u, y) \in E\} \) is any feasible allocation of \( u \)'s credit. Let \( E_u \subseteq E \) be the set of edges incident upon \( u \) in \( E \). Note that for all \( e' = (u', v') \notin E_u \), \( c_{u'v'}(G') = c_{u'v'}(G) \). As a result, \( f_{e'}(G) = f_{e'}(G') \). It follows that

\[
\Phi(G) - \Phi(G') = \gamma \sum_{e \in E_u} (f_e(G) - f_e(G')) = U_u(G) - U_u(G')
\]

Thus the network formation game is a potential game with \( \Phi(G) \) as the potential function. \( \square \)
Theorem 3.1 implies that in this setting, a Nash equilibrium always exists, best-response dynamics always converge to a Nash equilibrium, and finally, because the potential function is given by \( \Phi(G) = U(G)/2 \), the price of stability is 1. Next we will show that for a large, natural class of transaction size distributions, agents’ utilities are concave, and consequently, the price of anarchy is also 1, i.e., every Nash equilibrium of the formation game maximizes social welfare.

### 3.1.1 Nash Equilibria Maximize Social Welfare

Next we show that under certain technical conditions on the transaction size distributions, every Nash equilibrium under a symmetric bilateral transaction regime maximizes social welfare.

**Theorem 3.2.** Assume that for every edge \( e \in E \), (i) \( X_e(\cdot) \) is non-increasing, (ii) \( X_e(\cdot) \) has support over \([0, \infty)\), and (iii) \( X_e(\cdot) \) is twice differentiable. Let \( G \) be an arbitrary Nash equilibrium of the network formation game under a symmetric bilateral transaction regime \( \langle \Lambda, X \rangle \). Then \( G \) maximizes social welfare \( U(G) \).

We establish this theorem via two lemmas. First we derive an expression for \( f_e(G) \) in terms of the credit allocations \( c_{uv}(G) \) and \( c_{vu}(G) \) along edge \( e \).

**Lemma 3.3.** For nodes \( u, v \in V \) such that \( e = (u, v) \in E \), the steady-state transaction success probability, \( f_e(G) \) is given by

\[
f_e(G) = f_e(c_e(G)) = \begin{cases} \frac{X_e(0)}{c_e(G)} \int_0^{c_e(G)} X_e(y)dy, & \text{if } c_e(G) > 0 \\ 0, & \text{if } c_e(G) = 0 \end{cases}
\]  

(3.1)

where \( c_e(G) = c_{uv}(G) + c_{vu}(G) \) is the total credit allocated along edge \( e \) in \( G \).

See Appendix A.1 for a proof. Observe from (3.1) that \( f_e(G) \) depends only on the total credit capacity \( c_e(G) \) along the edge \( e = (u, v) \). Therefore, for the rest of this section, instead of thinking of \( f_e \) as a function of \( c_{uv}(G) \) and \( c_{vu}(G) \), we will think of \( f_e \) as the function \( f_e : \mathbb{R}_+ \rightarrow [0, 1] \). That is, \( f_e(x) \) is the steady-state transaction success probability along edge \( e \) when the total credit allocated along it is \( x \). We write \( f_e(G) \) to mean \( f_e(c_e(G)) \) when there is no ambiguity. Next we prove some properties of the functions \( f_e(\cdot) \) that enable us to establish that every Nash equilibrium maximizes social welfare.

**Lemma 3.4.** For an edge \( e \in E \), the steady-state transaction success probability, \( f_e(\cdot) \), is continuously differentiable, strictly increasing, and concave.

See Appendix A.2 for a proof. As a corollary, if \( X_e(\cdot) \) is strictly decreasing instead of non-increasing, \( f_e(\cdot) \) is strictly concave. Recall from Theorem 3.1 that the formation game under a symmetric bilateral transaction regime is a potential game and \( \Phi(G) = U(G)/2 + \gamma \sum_{e \in E} f_e(G) \) is a potential function. From Lemma 3.4, we know that \( f_e(\cdot), e \in E \), are concave and continuously differentiable, which implies \( \Phi(\cdot) \) is concave and continuously differentiable. It was shown by Neyman [1997] that any Nash equilibrium of a potential game with a concave and continuously differentiable potential is also a potential maximizer. Therefore, \( G \) maximizes \( \Phi(G) \), or equivalently, \( U(G) \). This completes the proof of Theorem 3.2.

Note that the conditions imposed upon the transaction size distributions \( X_e(\cdot) \) in the theorem are satisfied by many natural distributions, including exponential, mean-zero normal, and power-law distributions.
3.1.2 Nash Equilibria are Cycle-Reachable

Theorem 3.2 implies an equivalence between the Nash equilibria of the game; any two Nash equilibria $G$ and $G'$ have the same social welfare, $U(G) = U(G')$. Next we will show that if the transaction size distributions $X_e(\cdot)$ are strictly decreasing, instead of non-increasing as in Theorem 3.2, the Nash equilibria of this game are equivalent in a much stronger sense: any two Nash equilibria $G$ and $G'$ are cycle-reachable.

**Definition 3.1 (Dandekar et al. [2011]).** Let $G$ and $G'$ be two credit networks. We say that $G'$ is cycle-reachable from $G$ if $G$ can be transformed into $G'$ by routing a sequence of payments along feasible cycles (i.e., from a node to itself along a feasible path).

The significance of this property, established by Dandekar et al. [2011], is that the sequences of transactions that succeed starting from $G$ and starting from $G'$ are identical. Observe that this equivalence between Nash equilibria is stronger than that implied by Theorem 3.2.

**Theorem 3.5.** Assume that for every edge $e \in E$, (a) $X_e(\cdot)$ is strictly decreasing, (b) $X_e(\cdot)$ has support over $[0, \infty)$, and (c) $X_e(\cdot)$ is twice differentiable. Let $G$ and $G'$ be two Nash equilibria of the network formation game under the symmetric bilateral transaction regime $\langle \Lambda, X \rangle$. Then $G$ and $G'$ are cycle-reachable from each other.

In order to prove this theorem, we first show that the total credit capacity of any edge in $E$ is identical in any Nash equilibrium.

**Lemma 3.6.** For all edges $e \in E$, $c_e(G) = c_e(G')$.

**Proof.** We will state here, without proof, the propositions that prove this lemma. The proofs of the propositions can be found in Appendix A.3. First, observe that for an edge $e \in E$, the steady-state transaction success probability, $f_e(\cdot)$ is strictly concave, strictly increasing and continuously differentiable (by Lemma 3.4). Let us define the marginal utility of an edge $e \in E$.

**Definition 3.2.** The marginal utility of an edge $e \in E$ is the function $r_e : \mathbb{R}_+ \to \mathbb{R}_+$ given by

$$r_e(x) = f_e'(x) = \frac{df_e(x)}{dx}$$

We first show that for any edge $e \in E$, $r_e(G) = r_e(G')$.

Since $f_e(\cdot)$ is strictly concave, strictly increasing and continuously differentiable, $r_e(\cdot)$ is continuous, strictly decreasing and strictly positive. In network $G$, the marginal utility on an edge $e \in E$ is given by $r_e(c_e(G))$. We denote it by $r_e(G)$ when there is no ambiguity. Let $E_u$ be the set of edges in $E$ incident upon node $u$.

**Definition 3.3.** For a node $u \in V$ and a network $G$, we define $\rho_u(G) := \max_{e \in E_u} r_e(G)$ and $E_u^*(G) \subseteq E_u$ as the set of edges $e \in E_u$ such that $r_e(G) = \rho_u(G)$.

In words, $E_u^*(G)$ is the set of edges incident on node $u$ that have the highest marginal utility in network $G$ among all edges in $E_u$. We show that in any Nash equilibrium $G$, each node $u$ exhausts its entire budget and allocates non-zero credit only along edges in $E_u^*(G)$.

**Proposition 3.1.** Let $G$ be a Nash equilibrium. Then, for all nodes $u \in V$, both (1) and (2) are true:

1. $\sum_{v: (u,v) \in E} c_{uv}(G) = B_u$. 

10
2. For each $e = (u, v) \in E$, if $e \notin E^*_u(G)$ then $c_{uv}(G) = 0$.

Next we define a slack edge.

**Definition 3.4.** Let $G$ be a Nash equilibrium. We call an edge $e = (u, v) \in E$ a slack edge in $G$ if $e \notin E^*_u(G)$ or $e \notin E^*_v(G)$ or both.

Note that by Proposition 3.1, if edge $e = (u, v)$ is a slack edge in Nash equilibrium $G$, either $c_{uv}(G) = 0$ or $c_{vu}(G) = 0$ or both $c_{uv}(G) = c_{vu}(G) = 0$.

**Definition 3.5.** Let $G$ be a credit network. We define

1. $r_G^{\text{min}} := \min_{e \in E} r_e(G)$ to be the minimum marginal utility of any edge $e \in E$ in $G$,
2. the set $E_G^{\text{min}} := \{e \in E \mid r_e(G) = r_G^{\text{min}}\}$,
3. the set $V_G^{\text{min}} := \{u \in V \mid u \text{ is incident on some edge in } E_G^{\text{min}}\}$, and
4. the set $V^+_G := \{u \in V \mid u \text{ is incident both upon some edge in } E_G^{\text{min}} \text{ and upon some edge in } E - E_G^{\text{min}}\}$

The minimum marginal utility in any two Nash equilibria is identical.

**Proposition 3.2.** Let $G$ and $G'$ be two Nash equilibria. Then $r_G^{\text{min}} = r_G'^{\text{min}}$.

Moreover, in any two Nash equilibria $G$ and $G'$, the set of edges with the minimum marginal utility in $G$ is identical to that in $G'$.

**Proposition 3.3.** Let $G$ and $G'$ be two Nash equilibria. Then $E_G^{\text{min}} = E_G'^{\text{min}}$.

**Corollary 3.1.** Let $G$ and $G'$ be two Nash equilibria. Then $V_G^{\text{min}} = V_G'^{\text{min}}$ and $V^+_G = V^+_G$.

Thus, we have established that for any two Nash equilibria $G$ and $G'$, $r_e(G) = r_e(G')$ for all edges $e \in E_G^{\text{min}}$. We show using an inductive argument that this is true of all edges in $E$.

**Definition 3.6.** Given an instance $I : H = (V, E); f_e, e \in E; B_u, u \in V$ of the network formation game under a symmetric bilateral transaction regime, a credit network $G$, and an arbitrary set of edges $F \subseteq E$, we define the $(G, F)$-restriction of $I$, denoted $I_{(G,F)}$, as follows: $H^{(F)} := (V, E \setminus F)$, $f^{(F)}_e := f_e, e \in E \setminus F$, and

$$B^{(G,F)}_u := \begin{cases} 0 & \text{if } E_u \subseteq F \\ B_u - \sum_{(u,w) \in F} c_{uw}(G) & \text{otherwise} \end{cases}$$

Note that for a node $u$, if $E_u \subseteq F$, then the value of $B^{(G,F)}_u$ is immaterial since $u$ has no incident edges in $I_{(G,F)}$ along which to allocate its budget.

**Definition 3.7.** Given a credit network $G$ and an arbitrary set of edges $F \subseteq E$, we define an $F$-restriction of $G$, denoted $G_{(F)}$, as follows: for all edges $e = (u, v) \in E \setminus F, c_{uv}(G_{(F)}) = c_{uv}(G)$ and $c_{vu}(G_{(F)}) = c_{vu}(G)$.

**Proposition 3.4.** If $G$ is a Nash equilibrium for instance $I$ of the network formation game in the symmetric bilateral transaction setting, then $G_{(F)}$ is a Nash equilibrium for $I_{(G,F)}$ for any set $F \subseteq E$. 


Proposition 3.5. Let $G$ and $G'$ be two Nash equilibria for instance $I$ of the network formation game under a symmetric bilateral transaction regime. Then for all edges $e \in E$, $r_e(G) = r_e(G')$.

Observe that since $f_e(\cdot)$ is strictly concave, $r_e(\cdot)$ is strictly decreasing. Therefore, Proposition 3.5 implies that for all $e \in E$, $c_e(G) = c_e(G')$, completing the proof of Lemma 3.6.

Lemma 3.6 allows us to show that any two Nash equilibria are cycle-reachable. First we define the generalized score vector of a credit network $G$.

Definition 3.8 (Dandekar et al. [2011]). Given a credit network $G$ of $n$ nodes, the generalized score vector of $G$ is the vector $D(G) = \langle d_u(G) : u \in V \rangle \in \mathbb{R}_+^n$ where for all $u \in V$, $d_u(G) := \sum_{v \in V} c_{vu}(G)$.

Next we show that any two Nash equilibria have the same generalized score vector.

Proposition 3.6. Let $G$ and $G'$ be two Nash equilibria. Then, $D(G) = D(G')$.

Proof. Fix a node $u \in V$. Recall from Proposition 3.1 that

$$\sum_{v : (u,v) \in E} c_{uv}(G) = \sum_{v : (u,v) \in E} c_{uv}(G') = B_u \quad (3.2)$$

Also, from Lemma 3.6, we know that for all edges $e \in E$,

$$c_e(G) = c_e(G') \quad (3.3)$$

Let $E_u$ be the set of edges in $E$ incident upon $u$. It follows from (3.2) and (3.3) that

$$d_u(G) = \sum_{v \in V} c_{vu}(G) = \sum_{v : (u,v) \in E} c_{uv}(G) = \sum_{e \in E_u} (c_e(G) - c_{uv}(G))$$

$$= \sum_{e \in E_u} c_e(G) - B_u = \sum_{e \in E_u} c_e(G') - \sum_{e \in E_u} c_{uv}(G') = d_u(G')$$

\[\square\]

Proposition 3.7 (Dandekar et al. [2011]). Two credit networks $G$ and $G'$ are cycle-reachable if and only if $D(G) = D(G')$.

From Propositions 3.6 and 3.7, it follows that $G$ and $G'$ are cycle-reachable. This completes the proof of Theorem 3.5.

3.2 Symmetric Transactions

Here we lift the restriction that transactions be bilateral, allowing transactions between nodes that are not neighbors in $H$. We also allow payments between neighboring nodes to be routed along paths other than the direct edge between them. The analytical results we prove in this section and the next assume a unit transaction regime that we define below.

Definition 3.9. A unit transaction regime over credit network $G$ is a transaction regime $(\Lambda, X)$ where, for all $u, v \in V$ and for all $t > 0$, the transaction size $x_{uv}^t = 1$, the transaction rate matrix $\Lambda$ is symmetric and the Markov chain $\mathcal{M}(G, \Lambda, X)$ is ergodic.

12
When the network \( G \) is acyclic (ignoring directionality), Dandekar et al. [2011] characterize the steady-state success probabilities under a unit transaction regime, which we will make use of in proving our results.

**Lemma 3.7** (Dandekar et al. [2011]). Consider a credit network \( G \). Assume that \( G \) is acyclic if we ignore the directions of the edges in \( G \). Let \( P_{uv} \) be the set of (undirected) edges along the path between nodes \( u \) and \( v \). Then, in a unit transaction regime over \( G \), the steady-state transaction success probability, \( f_{uv}(G) \), from node \( u \in V \) to node \( v \in V \) is given by:

\[
f_{uv}(G) = \lambda_{uv} \prod_{e=(w,y) \in P_{uv}} \frac{|c_{wy}(G)| + |c_{yw}(G)|}{|c_{wy}(G)| + |c_{yw}(G)| + 1}
\]

Using this characterization, we show that the network formation game in this setting is not a potential game.

**Theorem 3.8.** There exists an instance of the network formation game under a symmetric transaction regime that does not admit a Nash equilibrium.

**Proof.** We construct an instance of network formation game and show that it does not admit a Nash equilibrium. Consider a game with six agents: \( V = \{a, b, d, e, h, j\} \). The graph \( H \) is a line graph over \( V \), with edges \((a,b), (b,d), (d,e)\), and \((d,e)\). The graph \( G \) is given by \( \lambda_{ab} = \lambda_{ba} = \lambda_{de} = \lambda_{ed} = \lambda_{hj} = \lambda_{jh} = 0.001 \), \( \lambda_{ae} = \lambda_{ea} = \lambda_{hj} = \lambda_{jh} = 0.2435 \), \( \lambda_{ej} = \lambda_{je} = 0.01 \). All other entries in the transaction rate matrix \( \Lambda \) are zero. All transactions are of size one. Observe that this is a unit transaction regime, so we can use Lemma 3.7 to compute the steady-state transaction success probabilities between nodes.

Let \( G \) be a Nash equilibrium. Then, it must be that \( c_{ab}(G) = c_{de}(G) = c_{hj}(G) = c_{jh}(G) = 1 \). Let \( c_{bd}(G) = x \) and \( c_{bd}(G) = 1 - x \). Similarly, let \( c_{eh}(G) = y \) and \( c_{ed}(G) = 1 - y \). Observe that since all transactions are of size one, and \( G \) is a Nash equilibrium, it must be that \( x, y \in \{0,1\} \) (i.e., \( x \) and \( y \) cannot be strictly between \( 0 \) and \( 1 \)). We can easily verify that for each of the four combinations of \( (x,y) \), namely, \((0,0), (0,1), (1,0) \) and \((1,1)\), either \( b \) or \( e \) has an improving unilateral deviation. In fact, the four combinations form a best-response cycle. Hence, there is no assignment of \( x, y \in [0,1] \) that will ensure that \( G \) is a Nash equilibrium.

![Figure 2: Example of a formation game that does not admit a Nash equilibrium.](image-url)

Next we show that even if agents reach a Nash equilibrium, it may be arbitrarily bad in terms of social welfare compared to a social optimum.

**Theorem 3.9.** The price of anarchy of the network formation game under a symmetric transaction regime is unbounded.

**Proof.** We will construct an instance of the game and show that it has an unbounded price of anarchy. Consider a game with four agents: \( V = \{a, b, d, e\} \). The graph \( H \) is a line graph over nodes in \( V \) with edges \((a,b), (b,d) \) and \((d,e)\). For each node \( u \in V \), \( B_u = 1 \). The non-zero
transaction rates are given by: \( \lambda_{ab} = \lambda_{ba} = \lambda_{de} = \lambda_{ed} = \lambda_1 > 0, \lambda_{ae} = \lambda_{ea} = \lambda_2 \gg \lambda_1 \). All other entries in the transaction rate matrix \( \Lambda \) are zero. All transactions are of size one.

Consider the network \( G \) shown in Fig. 3a. Observe that we can use Lemma 3.7 to compute the steady-state transaction success probabilities between nodes and verify that \( G \) is a Nash equilibrium. On the other hand, since \( \gamma \lambda h + h^2 \), the empty network is a Nash equilibrium, and the price of anarchy is \( \infty \).

Now consider the socially optimal network \( G^* \) in Fig. 3b. The social welfare \( U(G^*) \) is given by

\[
U(G^*) = \sum_{u \in V} U_u(G^*) = 2 \left( \lambda_1 \frac{2}{3} + \lambda_1 \frac{1}{2} + \lambda_2 \frac{1}{6} \right)
\]

In the limit as \( \lambda_1 \to 0 \), the ratio \( U(G^*) / U(G) = \infty \). \( \square \)

## 4 Network Formation under Global Risk: Single-Minded Agents

Recall that in the global risk model, each agent \( v \) has a public default probability \( \delta_v \in (0,1] \). If \( v \) defaults, a node \( u \) that extended credit \( c_{uv}(G) \) to \( v \) loses \( c_{uv}(G) \) units. Thus, \( \Delta_{uv}(G) = \delta_v c_{uv}(G) \).

We analyze the setting where agents may issue credit to at most one counterpart.

**Definition 4.1.** We say that agent \( u \in V \) is single-minded if in any credit network \( G \), either \( c_{uv}(G) = 0 \) for all \( v \in V \), or there exists a single agent \( w \in V \) such that \( c_{uw}(G) = B_u \).

Further, we assume that (i) the transaction rate matrix \( \Lambda \) is uniform: for all \( u, v \in V \), \( \lambda_{uv} = \lambda = 1/(n(n-1)) \), (ii) all transactions are size one: for all \( u, v \in V \), and for all \( t > 0 \), \( x_{uv}^t = 1 \), and (iii) for all agents \( u \in V \), the credit budget \( B_u = c > 0 \), where \( c \) is an integer.

First we illustrate using a simple example that if the default probabilities are in a certain range, the empty network is a Nash equilibrium, and the price of anarchy is \( \infty \).

**Example** Consider a set of \( n \) agents. Further suppose that for all \( u \in V \), \( \gamma \lambda (h+h^2) > \delta_u c > \gamma \lambda h \), where \( h = c/(c+1) \). Let \( G \) be the empty network. Observe that by Lemma 3.7, the utility to a node \( u \) from extending to any node \( v \) in \( G \) is \( \gamma \lambda h \), which by assumption is less than \( \delta_u c \). Thus \( G \) is a Nash equilibrium. On the other hand, since \( \gamma \lambda (h+h^2) > \delta_u c \) for all \( u \in V \), the social optimum is a star network where every node extends credit to the root, while the root extends no credit. As a result, the price of anarchy is \( \infty \).

For the rest of this section, we assume that extending zero credit is not part of the agents’ strategy set. This assumption, coupled with the fact that agents are single-minded, implies that any credit network formed in this setting will have exactly \( n \) directed edges each of capacity \( c \), where \( n \) is the number of agents playing the game. Since an agent extends credit to exactly one
agent in any network, we define the following notation to denote the agent that has been extended credit by an agent \( u \) in network \( G \): for a network \( G \), we define \( \tau_s : V \to V \) to be the “trustee function”: \( \tau_G(u) = v \) implies \( c_{uv}(G) = c \).

We use the following observation to prove our results; the observation follows from the analysis by Dandekar et al. [2011] of the steady-state success probability in trees under a unit transaction regime.

**Lemma 4.1** ([Dandekar et al., 2011]). Consider a network \( G \). Let \( u \in V \) be a node such that no node extends credit to \( u \) in \( G \) and let \( \tau_s(G) = v \). Assume the transaction rate matrix \( \Lambda \) is uniform and \( G \) is under a unit transaction regime. Then, for any node \( w \in V \setminus \{u, v\} \), \( f_{uw}(G) = hf_{vw}(G) \), where \( h = c/(c+1) \).

### 4.1 Price of Anarchy and Structure of Equilibria

It is easy to see that any socially optimal network will have a star-like structure where the root is a node with the minimum default probability.

**Lemma 4.2.** Let \( v^* \in \arg \min_{v \in V} \delta_v \) be a node with the minimum default probability. Let \( u^* \in \arg \min_{v \in V \setminus \{v^*\}} \delta_v \) be a node with the minimum default probability among nodes other than \( v^* \). Consider a network \( G^* \) such that for all nodes \( u \in V \setminus \{v^*\} \), \( \tau_{G^*}(u) = v^* \), and \( \tau_{G^*}(v^*) = u^* \). Then, \( G^* \) maximizes social welfare. Moreover, \( G^* \) is also a Nash equilibrium.

Next we show that all Nash equilibria have a star-like structure.

**Theorem 4.3.** For a sufficiently large \( n \), in any Nash equilibrium \( G \) there exists a node \( u^* \) such that for all nodes \( v \in V \setminus \{u^*\} \), \( \tau_G(v) = u^* \).

Next we show that despite ruling out the empty network as a Nash equilibrium, the price of anarchy in this setting can be unbounded.

**Theorem 4.4.** The price of anarchy of the network formation game with single-minded agents is unbounded.

**Proof.** Consider a set of \( n \) agents. Assume, without loss of generality, that for nodes \( u_1, \ldots, u_n \in V \), \( \delta_{u_1} \leq \ldots \leq \delta_{u_n} \). Let \( \delta_{u_1} = c = \gamma(n-3)h^2/(2c+1) \), and \( \delta_{u_2} = \delta_{u_3} = \gamma(n-3)h^2 \), where recall that \( h = c/(c+1) \). Consider the network \( G^* \) in Fig. 4a. It follows from Lemma 4.2 that \( G^* \) is a socially optimal network. Consider the network \( G_1 \) in Fig. 4b. Observe that Lemma 3.7 can be used to compute the steady-state transaction success probabilities and, hence, the utilities, of all nodes in \( G_1 \). Since \( c(\delta_{u_3} - \delta_{u_1}) \leq (n-3)\gamma\lambda h^2/(2c+1) \), nodes in \( G_1 \) cannot benefit from extending...
credit to \( u_1 \) or \( u_2 \) instead of \( u_3 \). Thus, \( G_1 \) is a Nash equilibrium. Note that since \( G^* \) and \( G_1 \) are structurally identical

\[
\sum_{u,v} f_{uv}(G^*) = \sum_{u,v} f_{uv}(G_1) \\
= \lambda(n-2) \left( (n-3)h^2 + 2h \frac{2c}{2c+1} + 2h \right) + 2\lambda \frac{2c}{2c+1} \\
= \lambda(n-2)(n-3)h^2 + \Theta(n)
\]

Thus, the total social welfare in \( G^* \) is given by

\[
U(G^*) = \gamma \sum_{u,v} f_{uv}(G^*) - (n-1)\delta_{u_1} c - \delta_{u_2} c \\
= \gamma\lambda(n-3)h^2 \left( (n-2) - (n-1) \frac{2c}{2c+1} \right) + \Theta(n) = \Theta(n^2)
\]

On the other hand,

\[
U(G_1) = \gamma \sum_{u,v} f_{uv}(G_1) - (n-1)\delta_{u_3} c - \delta_{u_1} c \\
= \gamma \sum_{u,v} f_{uv}(G_1) - \gamma\lambda(n-1)(n-3)h^2 - \delta_{u_1} c = \Theta(n)
\]

Since the price of anarchy is lower-bounded by \( U(G^*)/U(G_1) \), we have that \( \text{PoA} = \Omega(n) \). \(\square\)

### 4.2 Dynamics of Network Formation

Despite the fact that the price of anarchy in this setting can be arbitrarily high, we demonstrate that greedy dynamics can quickly converge to a socially optimal network.

**Greedy Response** For network \( G \), and an agent \( u \), we define greedy response by \( u \) as follows: let \( v^* \in \arg\min_{v \in V \setminus \{u\}} \delta_v \) be a node with the lowest default probability among all nodes except \( u \). Then, \( u \)'s greedy response is to extend credit to \( v^* \), i.e., \( \tau_{G^*}(u) = v^* \), where \( G' = \{c_{uv}(G') : u, v \in V\} \) defined below is the network resulting from \( u \)'s greedy response in \( G \). For nodes \( w, y \in V \),

\[
c_{wy}(G') := \begin{cases} 
    c_{wy}(G), & \text{if } w \neq u \\
    0, & \text{if } w = u \text{ and } y \neq v^* \\
    c, & \text{if } w = u \text{ and } y = v^*
\end{cases}
\]

**Theorem 4.5.** Assume that the default probabilities, \( \delta_v, v \in V \), are all distinct. Consider a network \( G \). Let \( G^* \) be the network obtained after all agents have played greedy response, starting from \( G \). Then \( G^* \) maximizes social welfare.

**Proof.** Since the default probabilities are all distinct, there exists a unique node, say \( v^* \), with the lowest default probability, and another node \( u^* \) with the second lowest default probability. Then, observe that for all \( u \in V \setminus \{v^*\} \), \( \tau_{G^*}(u) = v^* \) and \( \tau_{G^*}(v^*) = u^* \). The optimality of \( G^* \) follows from Lemma 4.2. \(\square\)

**Sequential Arrival** We consider a model where agents arrive sequentially, and strategically decide which one of the agents in the network to extend credit to. Let \( G_0 \) be a network of two agents, say \( u_0 \) and \( v_0 \), such that \( \tau_{G_0}(u_0) = v_0 \) and \( \tau_{G_0}(v_0) = u_0 \). At each time \( t = 1, 2, \ldots, \) an
agent $u_t$ arrives and extends credit to one of agents in the network $G_{t-1}$ in order to maximize $U_{u_t}(G_t)$ where $G_t$ is the resulting network. We denote by $V_t$ the set of agents that have arrived up to and including time $t$. We show that the agent $u_t$ arriving at time $t$ always extends credit either to $u_{t-1}$ or to $\tau_{G_{t-1}}(u_{t-1})$.

**Theorem 4.6.** For all $t \geq 1$, $\tau_{G_t}(u_t) \in \{u_{t-1}, \tau_{G_{t-1}}(u_{t-1})\}$.

Since the node $u_t$ arriving at time $t$ always extends credit to either $u_{t-1}$ or $\tau_{G_{t-1}}(u_{t-1})$, the resulting network has a *comb-like* structure, where a chain of nodes forms the spine of the network, and each node in that chain is trusted by a number of leaf nodes.

## 5 Simulation Analysis of Credit Network Formation

Covering more relaxed scenarios, such as those with more flexible $\Lambda$, unconstrained budgets, multiple credit issuance, or graded risk has thus far proved elusive for analytic treatments. Several factors contribute to the difficulty of game-theoretic analysis of this problem. First, the strategy space is combinatorial and multi-dimensional. Strategies for this game are mappings from all the information an agent has about the environment (default probabilities for all other agents, probabilities and values of transactions with all other agents) to all possible credit assignments to the other agents. Second, the expected value to an agent of a credit assignment is defined in terms of the outcome of a stochastic transaction sequence, intermixed with adjustments of credit balances that have important but indirect effects on the probabilities of downstream transactions. For this reason, we employ simulation to analyze environments that relax the conditions for which we have theorems.

### 5.1 Empirical Game-Theoretic Analysis

Our investigation of the credit network formation game employs an approach called *empirical game-theoretic analysis* (EGTA) [Wellman, 2006]. In EGTA, techniques from simulation, search, and statistics combine with game-theoretic concepts to characterize strategic properties of a domain.

#### 5.1.1 Iterative EGTA Process

A high-level view of the EGTA process is presented in Fig. 5. We start with an enumerated set of strategies, typically heuristics derived from domain knowledge or experience, often parametrized by meaningful strategy features. The basic EGTA step is simulation of a strategy profile, determining a payoff observation (i.e., a sample drawn from the outcome distribution generated by the simulation environment), which gets added to the database of payoffs. Based on the accumulated data, we induce an empirical game model. On this model we may perform any of the standard computations applied to game forms (e.g., identifying dominated strategies, finding equilibria). Based on the results, we may choose to refine the model by considering more strategies or strategy profiles, or obtaining more samples of profiles already evaluated.

The most straightforward way to define the empirical game is simply to estimate payoffs for evaluated profiles by their sample mean.$^\dagger$ We employ this method for the baseline game model,

$^\dagger$More sophisticated approaches may generalize from simulation data using regression or other machine learning techniques [Vorobeychik et al., 2007; Jordan and Wellman, 2009].
but then produce an approximate *reduced-game* model as well, by the technique described in Section 5.1.2.

When games (such as our version of credit network formation) exhibit significant symmetry, this can be exploited in representation and reasoning. Even for a fully symmetric game, however, the the profile space grows exponentially with the lesser of number of players \(n\) and number of strategies \(m\). There are \(\binom{n+m-1}{n}\) distinct profiles, to be precise. For even moderate \(n\) and \(m\), therefore, we generally cannot afford to evaluate every profile through simulation. We thus require analysis techniques that operate on incompletely specified games.

The EGTA process we followed in this study can be described in terms of two key procedures. The first, termed the *EGTA inner loop*, searches for an equilibrium within a fixed strategy set \(S\). The second, *outer loop*, extends the strategy set through local search, implementing the selection of more strategies depicted in Fig. 5.

The EGTA inner loop starts by performing an initial set of simulations, covering all profiles over a small subset \(S_0 \subset S\). It then iterates the following steps.

1. Identify the *maximal complete subgames*, \(\{S^1, \ldots\}\), where a complete subgame is defined as a set of strategies for which we have simulated all profiles.

2. For each maximal complete subgame \(S^i\), search for symmetric mixed-strategy Nash equilibria (SMSNE). We employ replicator dynamics for this purpose, from a diverse set of starting points. Let \(\sigma^i_j\) denote the \(j\)th SMSNE found for subgame \(S^i\). These subgame SMSNE are candidate equilibria for the full game over \(S\).

3. For each \(\sigma^i_j\), check the strategies \(s' \in S \setminus S^i\) such that we have evaluated all profiles where one player plays \(s'\) and the other \(n-1\) play strategies in the support of \(\sigma^i_j\). For any pair \((\sigma^i_j, s')\) where \(s'\) is a beneficial deviation, the candidate \(\sigma^i_j\) is refuted. For instances \(\sigma^i_j\) such that all possible deviating strategies have been evaluated without refutation, we say that \(\sigma^i_j\) is confirmed.

4. If there remains an SMSNE candidate \(\sigma^i_j\) that is neither refuted nor confirmed, simulate the profiles necessary to check another strategy \(s'' \in S \setminus S^i\) not yet fully evaluated in context \(\sigma^i_j\), and repeat from Step 1.

5. If there exists a refuted SMSNE candidate \(\sigma^i_j\), such that the support of \(\sigma^i_j\) plus its best-response refuting strategy is not subsumed by any complete subgame, simulate the profiles

---

\[\text{Figure 5: Iterative procedure for empirical game-theoretic analysis.}\]
6. If there exists at least one confirmed SMSNE candidate $\sigma^i_j$, return. Otherwise, choose a subgame $i$, extend it with some strategy $s' \in S \setminus S^i$, and repeat from Step 1.

On termination, the empirical game is in a state where all SMSNE candidates are confirmed, and all maximal subgame best-responses are themselves in a completed subgame. As long as the operation of identifying subgame equilibria is complete, the procedure is guaranteed to identify at least one confirmed SMSNE candidate.

The outer loop takes as input a confirmed SMSNE from the inner loop, and attempts to find a new strategy $s' \not\in S$ that refutes that SMSNE. It assumes a parametrically structured strategy space, and performs local search in that space, from a given starting point and subject to given constraints. We describe how this was implemented for the credit network formation game below; the general procedure is presented in detail elsewhere [Wellman et al., 2013].

5.1.2 Deviation-Preserving Reduction

One of the virtues of credit networks is their ability to support transactions among nodes only indirectly related by paths of credit. This property is particularly advantageous for large populations, where directly connecting all pairs that might transact would be too unwieldy. Our analysis of strategic network formation, therefore, requires a sufficiently large number of agents to reap the benefits of distributed credit allocation.

Increasing the number of agents, however, tends to blow up the profile space. For example, with 61 players (the number of nodes considered in this study), even a subgame of three strategies requires 1953 profiles to complete, and four strategies requires 41,664. It would not be feasible to explore very many subgames at this population size. We therefore seek to approximate the 61-player game by a smaller game. We call this approach player reduction, and in prior work employed a hierarchical approach where each player in the reduced game controls a proportional number of players in the full game [Wellman et al., 2005].

In the current study, we employ a recently introduced technique called deviation-preserving reduction (DPR) [Wiedenbeck and Wellman, 2012]. DPR is motivated by the assumption that an agent’s payoff is sensitive to its own choice of strategy and to the strategies of its opponents in the aggregate, but that small numbers of opponents changing strategy can be ignored. To calculate the payoff of a player $i$ for a profile in the reduced DPR game, we consider the full-game profile where one player plays $i$’s designated strategy, and the remaining players are divided proportionally among the other strategies in the reduced-game profile.

In the current study, we focused our analysis on a six-player reduced game derived from simulations on 61-agent credit networks, depicted in fig. 6. For example, we construct the six-player DPR profile $\langle 1 \times s_1, 3 \times s_2, 2 \times s_3 \rangle$ where one player plays strategy $s_1$, three play $s_2$, and two play $s_3$, by simulating three 61-agent profiles. The payoff to the player playing $s_1$ comes from the full-game profile $\langle 1 \times s_1, 36 \times s_2, 24 \times s_3 \rangle$, the payoff for $s_2$ from $\langle 12 \times s_1, 25 \times s_2, 24 \times s_3 \rangle$, and for $s_3$ from $\langle 12 \times s_1, 36 \times s_2, 13 \times s_3 \rangle$. In effect, each reduced-game player views itself as controlling one full-game agent, while its reduced-game opponents represent the fraction of full-game opponents playing each strategy. By this description, we see that deviation-preserving reduction applies most straightforwardly when the reduced game size divides $n - 1$ (hence our choice of $n = 61$ for this study). The DPR technique, however, is defined more generally for non-divisible reduction factors, as well as for games that are symmetric only within roles [Wiedenbeck and Wellman, 2012].
Figure 6: Deviation-preserving reduction example. Each of the six reduced-game players views itself as controlling one of the 61 full-game agents while each opponent controls an equal fraction (12) of the remaining full-game agents. The payoff to the player playing $s_1$ in the reduced-game profile comes from the payoff to $s_1$ in the full-game profile depicted in (a); $s_2$ from (b); $s_3$ from (c).

5.2 Credit Network EGTA Study: Setup

As noted above, we consider a population of 61 agents. Each run of the scenario comprises 10,000 transaction request events. The transaction rate $\lambda_{uv}$ for each pair of agents $u \neq v$ is drawn uniformly and then normalized. All transaction requests from $u$ the buyer to $v$ the seller are for a single unit. The value to $u$ of a successful transaction with $v$ is drawn uniformly, $x_{uv} \sim U[1, \bar{x}]$, with $\bar{x}$ set to either 1.2 (low value) or 2 (high value). Cost to the seller is constant: one. The average surplus per transaction is thus either 0.1 or 0.5. Default probabilities $\delta_v$ for each agent are drawn from a Beta distribution: Beta(1, 9) (average default probability $\frac{1}{10}$) in the low default setting, Beta(1, 2) (average $\frac{1}{3}$) in the medium default setting, and Beta(1, 1) (average $\frac{1}{2}$) in the high default setting. In the global risk environment these default probabilities are revealed to all, whereas in the graded risk environment each agent gets sample data from the default distribution of others, with the number of samples $\partial_{uv}$ determined by the social network distance between $u$ and $v$. The social network itself is an Erdős-Rényi graph. We take $\partial_{uv} = 100$ if $u$ and $v$ are neighbors, $\partial_{uv} = 10$ if they are linked through one other node, $\partial_{uv} = 1$ if they have a shortest-path of length three, and $\partial_{uv} = 0$ otherwise.

We explored environments with high, medium, or low default, and high or low value, for each of global and graded risk. The twelve environments are listed in Table 1, along with the number of profiles and strategies we ended up simulating, in both the full and reduced games. Three-letter environment names are coded by risk model (C[omplete information] for global risk, I[ncomplete] for graded risk), default probability (L[ow]/M[edium]/H[igh]), and buyer value (L[ow]/H[igh]). These numbers of profiles and strategies are broken down by two stages (I and II) of search, as described below.

We considered a range of heuristic strategies available to agents. A strategy is defined by three parameters: (i) a criterion for ranking the other agents, (ii) the number $k$ of agents to issue credit (the best $k$ according to the ranking criterion), and (iii) the number of units $q$ of credit to issue to each of these chosen agents. The criteria we included in heuristics along with the $(k,q)$ values we considered in this study are enumerated below, defined from the perspective of agent $u$’s evaluation of credit prospect $v$:

- **Default** probability: lowest known default ($\delta_v$) for global risk, or lowest estimated default based on samples $\partial_{uv}$ for graded risk.
- **Buy rate**: highest probability of transacting ($\lambda_{uv}$).
Table 1: Exploration performed by the iterative EGTA process under various environment settings. Strategies gives the number of strategies added by the outer loop. Full-game profiles gives the number of 61-agent profiles sampled by the inner loop. DPR profiles gives the number of 6-player profiles in the empirical game model.

<table>
<thead>
<tr>
<th>name</th>
<th>Risk model</th>
<th>Default prob</th>
<th>buyer surplus</th>
<th>Full-game profiles (I/II)</th>
<th>DPR profiles (I/II)</th>
<th>Strategies (I/II)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLL</td>
<td>Global</td>
<td>low</td>
<td>low</td>
<td>4619 11695</td>
<td>1497 3946</td>
<td>17 32</td>
</tr>
<tr>
<td>CLH</td>
<td>Global</td>
<td>low</td>
<td>high</td>
<td>2179 9861</td>
<td>765 3359</td>
<td>17 32</td>
</tr>
<tr>
<td>CML</td>
<td>Global</td>
<td>med</td>
<td>low</td>
<td>3557 9425</td>
<td>1036 3213</td>
<td>8 32</td>
</tr>
<tr>
<td>CMH</td>
<td>Global</td>
<td>med</td>
<td>high</td>
<td>8619 20192</td>
<td>2622 6474</td>
<td>15 32</td>
</tr>
<tr>
<td>CHL</td>
<td>Global</td>
<td>high</td>
<td>low</td>
<td>3134 5901</td>
<td>1045 2090</td>
<td>17 32</td>
</tr>
<tr>
<td>CHH</td>
<td>Global</td>
<td>high</td>
<td>high</td>
<td>3202 6155</td>
<td>1101 2196</td>
<td>17 32</td>
</tr>
<tr>
<td>ILL</td>
<td>Graded</td>
<td>low</td>
<td>low</td>
<td>991 9322</td>
<td>394 3148</td>
<td>17 32</td>
</tr>
<tr>
<td>ILH</td>
<td>Graded</td>
<td>low</td>
<td>high</td>
<td>5377 28721</td>
<td>1824 8786</td>
<td>17 32</td>
</tr>
<tr>
<td>IML</td>
<td>Graded</td>
<td>med</td>
<td>low</td>
<td>1766 8927</td>
<td>565 3109</td>
<td>8 32</td>
</tr>
<tr>
<td>IMH</td>
<td>Graded</td>
<td>med</td>
<td>high</td>
<td>24612 39728</td>
<td>6818 11356</td>
<td>18 32</td>
</tr>
<tr>
<td>IHL</td>
<td>Graded</td>
<td>high</td>
<td>low</td>
<td>656 5080</td>
<td>261 1881</td>
<td>17 32</td>
</tr>
<tr>
<td>IHH</td>
<td>Graded</td>
<td>high</td>
<td>high</td>
<td>430 10529</td>
<td>201 3516</td>
<td>17 32</td>
</tr>
</tbody>
</table>

- **Sell rate**: highest probability of serving a transaction ($\lambda_{vu}$).
- **Trade value**: highest expected value of transaction per event ($\lambda_{uv}x_{uv}$).
- **Trade profit**: highest difference, expected value of transaction minus expected value of served transaction ($\lambda_{uv}x_{uv} - \lambda_{vu}$).
- **Index**: lowest node number (arbitrary global labeling).
- **Random**: uniform choice.

In addition, we included the no-credit strategy, Zero, which issues no credit to anyone.

Observe that the Default strategies behave qualitatively differently in the global and graded risk environments. Under global risk, all agents have the same information about default probabilities. Therefore, when agents issue credit to the least-likely defaulters, they are creating edges to the same target agents. This leads to a centralized or star-like credit network, as illustrated in Fig. 7. Such coordination on credit targets has potential advantages. If everyone including $u$ offers credit to $v$, then once $v$ transacts with $u$, $u$ enjoys credit paths to everyone in the network. This coordination does not result, in contrast, from mutual application of Default in the graded risk model. Under graded risk, agents have different information based on their positions in the social network. The counterparts judged to have lowest default probably are invariably those with whom the agent has had most positive experience. Since there is little experience of any kind with social strangers, these are unlikely to be judged most trustworthy (this happens only if one is unlucky enough to have only very untrustworthy friends). Indeed, under graded risk we find that 95% of the top five least likely defaulters are one or two hops away on the social network. Finally, note that the Index strategies do coordinate on a star-like network, in either the global or graded risk model. By comparing Default and Index strategies we can separate the pure benefits of coordination from the benefits of avoiding defaulters.

In Stage I of the analysis, we considered a fixed set of 17 strategies, and ran the inner loop on eight of the twelve environments: those with high or low (not medium) default probabilities.
The 17 predefined strategies were selected based on exploration in a preliminary study, and are enumerated in Table 2.

Table 2: Strategies included in the EGTA study: predefined for Stage I, and automatically generated in the outer loop.

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Predefined ((k, q))</th>
<th>Automatically Generated ((k, q))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Buy</td>
<td>((1, 1), (2, 2), (3, 2), (5, 2))</td>
<td>((3, 1), (4, 1), (5, 1), (6, 1), (8, 1), (9, 1))</td>
</tr>
<tr>
<td>rate</td>
<td>((1, 1), (1, 2), (2, 2))</td>
<td>((2, 1), (4, 1), (8, 1), (10, 1))</td>
</tr>
<tr>
<td>Sell rate</td>
<td>((2, 2), (5, 2))</td>
<td>((6, 1))</td>
</tr>
<tr>
<td>Trade value</td>
<td>((2, 2), (6, 2), (8, 1))</td>
<td>((2, 1), (3, 1))</td>
</tr>
<tr>
<td>Trade profit</td>
<td>((1, 1), (2, 2))</td>
<td>((3, 2), (5, 1))</td>
</tr>
<tr>
<td>Index</td>
<td>((2, 2))</td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>((0, 0))</td>
<td></td>
</tr>
<tr>
<td>Zero</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For the four medium default environments, we started with a smaller set of eight predefined strategies (the first listed for each criterion), and employed the automated strategy generation procedure (EGTA outer loop) to extend the set. On each iteration, we searched for refutations of an equilibrium \(\sigma\) confirmed for the existing strategy set, employing local search from a particular existing strategy. The search algorithm simply hill-climbs from the existing strategy, holding its credit criterion fixed but incrementing or decrementing its \(k\) and \(q\) parameters. The search halted upon reaching a local maximum in payoff, assuming all other nodes in the network play according to \(\sigma\). If that local maximum exceeds the equilibrium payoff, the new strategy is added to the set and we proceed with another round of the EGTA inner loop. If instead we cycle through the strategy categories (in this case, defined by credit criterion) without finding a beneficial deviation, the entire process concludes.

As indicated in Table 1, the CML and IML environments found no new strategies, whereas the CMH environment added seven strategies to the original eight, and IMH added ten. Together, there were 15 automatically generated strategies not included among the 17 predefined Stage I
strategies. These are listed in the final column of Table 2. For Stage II, we constructed the combined set of 32 strategies, and ran the EGTA inner loop for each environment with this set.

With 17 strategies, there are $1.4 \times 10^{16}$ distinct strategy profiles for the full 61-player game, and 74,613 for the six-player DPR game. These numbers grow to $3.0 \times 10^{21}$ (61-player) and 2,324,784 (six-player DPR) for the Stage II set of 32 strategies. As indicated in Table 1, our EGTA process evaluated only a very small fraction of these profiles at each stage. Nevertheless it was able to identify equilibria in each environment.

Altogether we evaluated 165,536 full-game strategy profiles across the twelve credit network environments, from which we estimated payoffs for 53,074 DPR profiles. Each full-game profile evaluated was simulated at least 1,000 and usually upwards of 2,000 times. Our simulations were performed on a computing cluster operated by the University of Michigan, using an experiment management facility designed expressly for EGTA studies [Cassell and Wellman, 2013].

5.3 Results

Through the process described in Section 5.1.1, we successfully derived equilibria for each of the twelve credit network games. Specifically, we identified between one and six SMSNEs for the reduced six-player DPR games corresponding to each environment. All candidate subgame equilibria were either confirmed or refuted by the process, and the subgames covering best responses to all candidates were completed.

The strategies Sell rate, Index, and Random are not supported in any equilibria. To characterize the equilibria qualitatively, we partition the remaining strategies as follows. Class D represents Default, Z represents Zero, and T groups together strategies based on criteria related to transaction probability and value: Buy rate, Trade value, and Trade profit. The SMSNEs identified are summarized in Fig. 8. In the figure, there is one cell for each environment, displaying class labels for strategies supported in some equilibrium. A class letter circled means that a strategy in that class was confirmed as a pure strategy Nash equilibrium. Interestingly, whereas many of the equilibria found were mixed, and several environments had equilibria in multiple classes, in no case did a single SMSNE mix across the class partitions defined above.

From the figure, we see that there is a no-credit equilibrium in eight of the twelve environments: all but those with low or medium default and high buyer value. The two least favorable environments—graded risk with high or medium default and low value—have only this equilibrium, whereas all the others have some equilibrium where credit is provided. All of the global risk environments have an equilibrium where everybody plays Default, but this strategy does not appear in equilibrium for any graded risk environments. Indeed, there is a one-to-one correspondence between the equilibria for the two risk classes, except that the graded risk environments omit these Default equilibria, and when buyer value is high, these are replaced by transaction-based equilibria. The weakened information about defaults plus the lack of coordinating power render this a poor credit-issuing criterion in graded risk environments.

For completeness, we list the equilibria found. Groups in brackets with probabilities represent SMSNEs, and ungrouped strategies indicate pure equilibria.

**CLL** Default(1,1); Default(3,1); Default(4,1); Zero; [Buy rate(2,1), 0.899; Trade value(3,1), 0.101]; [Trade value(2,1), 0.806; Trade value(3,1), 0.194]

**CLH** Default(3,2); Trade profit(5,1); [Default(6,1), 0.951; Default(8,1), 0.049]; [Default(5,1), 0.744; Default(8,1), 0.256]

**CML** Default(1,1); Default(3,1); Zero

23
Figure 8: Equilibria found for the twelve credit network environments. Letters denote the strategy classes represented in equilibrium, with circled letters indicating pure-strategy equilibria.

CMH Default(2,2); [Default(2,2), 0.014; Default(6,1), 0.986]; [Default(3,2), 0.880; Default(4,1), 0.120]; [Default(5,1), 0.821; Default(6,1), 0.179]

CHL Default(1,1); Default(3,1); Zero

CHH Default(2,2); Zero; [Default(3,2), 0.081; Default(4,1), 0.919]

ILL Zero; [Trade value(2,1), 0.637; Trade value(3,1), 0.363]

ILH [Buy rate(4,1), 0.229; Trade profit(5,1), 0.771]

IML Zero

IMH [Buy rate(4,1), 0.172; Trade profit(5,1), 0.785; Trade value(3,1), 0.042]

IHL Zero

IHH Buy rate(4,1); Zero

Whereas the set of equilibria evolved as the empirical game was refined from Stage I to Stage II, the qualitative categories of strategy profiles represented in equilibrium (as depicted in Fig. 8, ignoring the circle designations) remained constant.

We next turn to the question: How well do the credit networks generated in equilibrium perform? Fig. 9 compares the welfare (sum of agent utility) of equilibrium outcomes to that of an estimated social optimum. Our estimate is actually a lower bound, equal to the greatest social welfare seen in any full-game profile simulated. Equilibrium welfare varies across equilibria, hence we present the best and worst of those identified. In eight of twelve environments, the worst is the Zero equilibrium, which supports no transactions and thus yields zero welfare. What we find
overall is that when there is a substantial amount of welfare possible (i.e., the most favorable environments), equilibrium network formation does a good job of obtaining most of it. For less favorable environments, a network if it forms at all tends to produce little utility.

We can also observe directly the amount of credit issued in equilibrium networks, as compared to the social optimum—which is not necessarily the credit-maximizing network. As seen in Fig. 10, the comparison mirrors that for welfare, but with lower ratios of equilibrium to social optimum across the board. This is due to the diminishing returns to credit, once the network has ample credit capacity. In other words, we can achieve a substantial fraction of available social welfare without issuing this same fraction of the credit that a social planner would.

All of these results are of course relative to the particular strategy space included in the empirical game analysis. Our choice was driven by an effort to span a diverse space, and to include strategies successful in preliminary studies or otherwise representing plausible prospects for refuting initial equilibrium candidates. The fact that adding strategies in Stage II based on automated exploration of parametric variations on the original strategies did not change the qualitative character of equilibria lends support to the robustness of these results.


6 Conclusion

Our investigation of strategic issues in the formation of credit networks characterizes, in various settings, the nature and efficiency of credit networks that are formed by self-interested agents autonomously choosing how to issue credit among available counterparts. The analysis employs game-theoretic solution concepts, employed in theoretical examination of analytic models, as well as simulation-based exploration of extended environments.

In the most restrictive case of dichotomous risk with only bilateral transactions permitted, we show that the formation game is a potential game, and under many transaction size distributions every Nash equilibrium of the game maximizes social welfare. More interestingly, we showed that the Nash equilibria are equivalent in a much stronger sense: all Nash equilibria are cycle-reachable from each other, which implies that the sequences of transactions that can be supported from each equilibrium network are identical. However, when we allow transactions over longer paths, best-response dynamics may not converge, and the price of anarchy is unbounded.

Under a model of global risk, if agents are limited to extend credit to at most one other agent, we prove that the networks formed in equilibrium have a star-like structure. Although the price of anarchy is unbounded, simple greedy dynamics quickly converge to a social optimum. Even when agents are allowed to extend credit to multiple agents, we show using empirical game simulation that non-empty equilibria tend to be star-like.

Our empirical game simulations confirm the finding of star-like equilibrium networks under global risk, even in less-restrictive scenarios. In addition, we study a graded risk model, where agents have partial information about default risks. We find that star-like equilibria disappear because agents are unable to coordinate on highly trustworthy central nodes. We also find that whether empty networks can occur in equilibrium depends primarily on the relative profitability of transactions, and not on the structure of information about default probabilities.

Acknowledgments

Tae Hyung Kim assisted in the implementation of the automated EGTA process, and Ben-Alexander Cassell extended our EGTA infrastructure to facilitate this work. This research was supported in part by NSF award IIS-0904325, and by the Army Research Laboratory under Cooperative Agreement W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References


27


