Guarantees for Self-Play via Polymatrix Decomposability

by

Revan MacQueen

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

Self-play is a technique for machine learning in multi-agent systems where a learning algorithm learns by interacting with copies of itself. Self-play is useful for generating large quantities of data for learning, but has the drawback that agents the learner will face post-training may have dramatically different behaviour than the learner came to expect by interacting with itself. For the case of two-player constant-sum games, self-play that reaches Nash equilibrium is guaranteed to produce strategies that cannot lose utility from their equilibrium value against any post-training opponent; however, no such guarantee exists for multi-player games.

We show that in games that approximately decompose into a set of two-player constant-sum games (called polymatrix games) where global ϵ-Nash equilibria are boundedly far from Nash-equilibria in each subgame (called subgame stability), any no-external-regret algorithm that learns by self-play will produce a strategy with bounded loss of utility against new agents, which we call vulnerability. In approximate subgame stable constant sum polymatrix (SS-CSP) games, the strategies produced by self-play are also exchangeable and have values that fall into a bounded range. We extend these results to extensive-form games and give an efficient representation and algorithm for such a decomposition. We demonstrate our findings through experiments on Kuhn and Leduc poker. Finally, we extend our results to games which are strategically equivalent to SS-CSP games. For the first time, our results identify a structural property of multi-player games that enable performance

guarantees for the strategies produced by a broad class of self-play algorithms.

Preface

Parts of this thesis are accepted for publication at NeurIPS 2023 [39]. This includes the vulnerability bounds from Chapter 3 and Chapter 4 and the Leduc Poker experiments in Chapter 5.

In a game, for once in my life, I know exactly what it is that I'm supposed to be doing.

– C. Thi Nguyen

Acknowledgements

First and foremost, a huge thank you goes to my brilliant supervisor James Wright for his mentorship and teaching over the past three years. I have improved so much as a researcher because of James' guidance and belief in me and this project.

I am so grateful for the friends and collaborators I've have the pleasure of meeting at the University of Alberta and Amii. I'd like to thank my research collaborators Erfan Miahi, Abbas Masoumzadeh and Martha White for their persistence with the resmax project. Thank you to Dustin Morrill for answering many of my questions over Slack. A further thank you goes out to Daniel Chui, Niko Yasui, Rohini Das, Alireza Masoumian and all the members of the ABGT group for all the interesting discussions over the years. Thank you to all the Amii staff for the support outside of my studies (and coffee). In addition to all the friends mentioned so far, thank you to Aidan Bush, Maliha Sultana, Justin Stevens, Alex Lewandowski, Daniela Teodorescu, Andrew Jacobson, and all the Amii beers goers for the many memories from the past few years. To Sacha Davis, nothing has meant more to me than our time together, thank you for your love and support.

Starting a master's during a world-wide pandemic was made so much more manageable because of my roommates and friends at the McKernan house. I'd especially like to thank Ben Freeman for the many baked goods, cycling adventures and good company during the pandemic. Thank you to all my friends from high school, undergrad, and before for the frequent camping excursions, parties and pizza at Steel Wheels while I completed this degree.

Finally, thank you to my family and to my parents, Sonya and Jason MacQueen for their love, support and seemingly endless patience while I figured things out over the past three years (not to mention the countless Coke zeroes and use of their sunroom, where much of this document was written). I don't think I would be anywhere close to where I am without your support and love from the very start.

Contents

List of Figures

Chapter 1

Introduction

As intelligent decision makers, we act within the world to achieve our goals. Along the way, we will interact with other agents in situations that are competitive, cooperative, or anywhere in-between. Optimal behaviour depends on the behaviour of these other agents; for example one should drive on the left side of the road in England but not Canada. Since the world contains other agents, we naturally want machine learning algorithms to perform well in interactions with those agents. While we ideally want learning agents to adapt to other agents quickly and continually, oftentimes this is difficult in practice since current machine learning requires an enormous amount of experience.

The solution is train these learning algorithms offline first so they reach a desired level of performance before interacting within the real world. However, offline learning in multi-agent systems is an under-specified problem: what should be used as the behaviour of other agents in the environment? How should agents learn offline to interact with other agents with unknown behaviour?

Self-play is a common approach for machine learning in multi-agent systems that addresses this problem. In self-play, a learner interacts with copies of itself to produce data that will be used for training. Some of the most noteworthy successes of AI in the past decade have been based on self-play; by employing the procedure, algorithms have been able to achieve super-human abilities in various games, including Poker [42, 9, 10], Starcraft [65], Diplomacy [48], and Stratego [49], Go and Chess [56, 57].

	a	b
a	$1, 2$	$0, 0$
b	$0, 0$	$2, 1$

Figure 1.1: A simple game, commonly called the battle of the sexes.

Self-play has the desirable property that unbounded quantities of training data can be generated, assuming access to a simulator. However, using self-play necessarily involves a choice of agents for the learner to train with: namely copies of itself. Strategies that perform well during training may perform poorly in deployment against new agents, whose behaviour may differ dramatically from that of the agents that the learner trained against.

The problem of learning strategies during training that perform well against new agents is a central challenge in algorithmic game theory and multi-agent reinforcement learning (MARL) [41, 36]. In particular, even opponents from an independent self-play instance, differing only by random seed, can lead to dramatically worse performance than the agent came to expect during training [36].

There are special classes of environments where the strategies learned through self-play generalize well to new agents. In two-player, constant-sum games there exist strong theoretical results guaranteeing the performance of a strategy learned through self-play. No-regret self-play will converge to a Nash equilibrium, whose component strategies guarantee at least the value of that equilibrium against any opponent. Equilibria are also *exchangeable*: a selection of equilibrium strategies from different equilibria for each agent is itself an equilibrium. Finally, all equilibria yield the same amount of utility, called the *value of the game*.

We lose these guarantees outside of two-player constant-sum games. Fundamentally, no-regret self-play algorithms are no longer guaranteed to produce Nash equilibria. These algorithms instead converge to a *mediated equilibrium*, where a mediator recommends actions to each player [67, 18, 17, 45]. The mediator can represent an external entity that makes explicit recommendations, such as traffic lights mediating traffic flows. More commonly in machine

learning, correlation can arise through the shared history of learning agents interacting with each other [26]. In this second scenario, new agents may not have access to the actions taken by other agents during training, so players would no longer be able to correlate their actions. In fact, even if all agents play a decorrelated strategy from *the same* mediated equilibrium, the result may not be an equilibrium (please refer to Appendix A.3 for an example).

Even when algorithms find Nash equilibria, these equilibrium strategies are not as desirable as in two-player constant-sum games, when deployed against new players. For example, consider the simple two-player game of Figure 1.1. The row player prefers (b, b) over (a, a), while the column player prefers (a, a) over (b, b). However both players prefer to choose the same strategy over miscoordinating.

Suppose the row player learned in self-play to choose a (which performs well against another a-player). Similarly, column learned to play b. Upon the introduction of a new agent who did not train with a learner, despite a and b being optimal strategies during training, they fail to generalize to new agents. As this example demonstrates, equilibrium strategies in general are *vulnerable*: agents are not guaranteed the equilibrium's utility against new agents.

Beyond a lack of vulnerability guarantees, equilibrium strategies, in general, suffer from the well-known *equilibrium selection problem* [25, 51, 59, 40]. Even if all agents chose equilibrium strategies, they may *still* nonetheless regret their choices. For example, in Figure 1.1 a and b are both equilibrium strategies, but (a, b) is not an equilibrium. Lastly, equilibria do not yield the same amount of utility in general-sum and multi-player games. For example (a, a) gives the row players a lower utility than (b, b).

Put another way, two-player constant-sum games are a rare case where—in addition to *descriptive* value—equilibrium strategies have *prescriptive* value. It is a *good* idea to play an equilibrium strategy against new opponents [55].

The failure of equilibrium strategies has led some approaches to reject the paradigm of playing equilibrium strategies altogether. *Opponent modeling* tailors play to maximize utility with respect to an agent's internal model of opponents. With a good model, the agent may outperform equilibrium

strategies; however, an incorrect model can reduce performance [61, 60]. For example, the max^n algorithm [38]—which is guaranteed to produce Nash equilibria in extensive-form games of perfect information [59]—is outperformed by the opponent modeling-based prob-max^n in the game of Spades [61]. Alternatively, a learning agent could assume that other agents will choose actions that will minimize their utility—regardless of what the other agents desires are in actuality. These *paranoid* agents [62] learn robust strategies, but are likely sub-optimal. Agents may also choose strategies using learning rules that are maximally robust against symmetries present in the game [27]. This approach, however, still requires shared knowledge of the learning rule in order to be effective—essentially resulting in a "meta-equilibrium selection problem".

Despite these problems, self-play has shown promising results outside of two-player constant-sum games. For example, algorithms based on self-play have outperformed professional poker players in multi-player Texas hold 'em, despite the lack of theoretical guarantees [10]. This hints at the existence of classes of games, somewhere between two-player constant-sum and multi-player general sum, where self-play will perform well. Indeed, there has been much work generalizing the strategic properties of two-player constant-sum games to other two-player games [3, 46, 22, 28]; however, multi-player games have received less attention [12, 11].

What structural properties of multi-player games allow for the strategies learned in self-play to perform well (i.e. generalize) against novel *post-training* agents? We identify a class of multi-player games—called *subgame-stable constant-sum polymatrix games* (SS-CSP)—where self-play will converge to a strategy that is not vulnerable to a loss of utility against arbitrary changes in other agents from what the self-play agent came to expect during training. These games bring into the n-player setting many of the nice properties of two player constant-sum games. We show that any game can be approximately decomposed into this class of games and the underperformance against new agents is bounded by a function of the level of approximation. SS-CSP games also have approximate exchangeability and bounded equilibrium values. We decompose multi-player variants of Kuhn and Leduc poker into subgame-stable

constant-sum polymatrix games with a low degree of approximation in the decomposition, thereby elucidating *why* self-play performs well in multi-player poker.

Throughout this work, we take an algorithm-agnostic approach by assuming only that self-play is performed by a regret minimizing algorithm. This is accomplished by analyzing directly the equilibria that no-regret algorithms converge to—namely coarse correlated equilibria. As a result, our analysis applies to a broad class of game-theoretically-inspired learning algorithms but also to MARL algorithms that converge to coarse correlated equilibria [40, 37, 29], since any policy can be transformed into a mixed strategy with Kuhn's Theorem [33].

This thesis is outlined as follows. Chapter 2 defines two models of multi-agent decision making, normal form and extensive-form games. Learning takes place in the hindsight rationality framework. We describe two representative learning algorithms called Regret-Matching and Counterfactual Regret Minimization (CFR). Chapter 3 defines two properties for normal-form games—subgame stability and constant-sum polymatrix—that are sufficient to guarantee that self-play will produce a desirable strategy in the normal-form setting. We show that games that meet a relaxed version of these properties behave in much the same way as two player constant-sum games. Chapter 4 extends our definitions and theory to extensive form games via the poly-EFG representation and provides an efficient algorithm for computing these properties. In Chapter 5, we apply our theory to multi-player Kuhn and Leduc poker to elucidate why self-play performs well in multi-player poker. Our results suggest that regret-minimization techniques converge to a subset of the game's strategy space that is approximately SS-CSP. We also show that a toy Hanabi game where self-play performs poorly is not approximately SS-CSP for any reasonable degree of approximation. In Chapter 6 we consider games that are not themselves approximately SS-CSP, but are instead *strategically equivalent* to SS-CSP games, thereby tightening our bounds for many games. Chapter 7 presents additional results that give a class of games that are subgame stable, called aligned games; and shows that SS-CSP games are an example of a

broader class of games that generalize the minimax theorem to multi-player games

Chapter 2

Background

Games serve as models of multi-agent strategic situations. We use two formulations of games in this thesis, namely normal-form and extensive-form. In this chapter, we define normal-form and extensive-form games, their solution concepts, and the framework for learning we concentrate upon—hindsight rationality/no-regret learning.

2.1 Normal Form Games

The simplest form of game is a normal-form game. Here, a set of agents simultaneously choose strategies and receive a payoff depending on this selection.

Definition 2.1.1 (Normal form game)**.** A normal-form game G is a 3-tuple $G = (N, \mathrm{P}, u)$ where N is a set of *players*, $\mathrm{P} = \times_{i \in N} \mathrm{P}_i$ is a joint pure strategy space where P_i is a set of *pure strategies* for player i. $u = (u_i)_{i \in N}$ is a tuple of *utility functions* where $u_i : \mathrm{P} \to \mathbb{R}$.

Pure strategies are deterministic choices of actions in the game. In a normal-form game, each agent simultaneously chooses their pure strategy $\rho_i \in \mathrm{P}_i$; we call a joint selection of pure strategies $\rho \in \mathrm{P}$ a *pure strategy profile*. The pure strategy profile determines the payoff (or utility) to player via the utility function. For example, if the players jointly choose ρ, then each player $i \in N$ will receive utility $u_i(\rho)$. The dependence of the utility function on the strategies of other players is a fundamental aspect of multi-agent strategic situations.

	c	d
c	$1, 1$	$-1, 2$
d	$2, -1$	$0, 0$

Figure 2.1: The prisoner's dilemma. We show one player's pure strategies as rows and the other's as columns. Each entry of the matrix first lists the row player's utility followed by the column player's utility.

Players may also randomize their actions through the use of a *mixed strategy*: a probability distribution s_i over i's pure strategies. Let $S_i = \Delta(\mathrm{P}_i)$ be the set of player i's mixed strategies (where $\Delta(X)$ denotes the set of probability distributions over a domain X), and let $S = \times_{i \in N} S_i$ be the set of mixed strategy profiles. We overload the definition of utility function to accept mixed strategies as follows:

$$u_i(s) = \sum_{\rho \in \mathrm{P}} \left(\prod_{i \in N} s_i(\rho_i) \right) u_i(\rho),$$

where $s_i(\rho_i)$ denotes the probability mass placed on ρ_i by s_i.

Let $-i$ be a shorthand for the set $N \backslash \{i\}$; if $|-i| = 1$ then we overload $-i$ to refer to the single agent in $-i$. Let ρ_{-i} and s_{-i} to denote a joint assignment of pure (resp. mixed) strategies to all players except for i. Thus $s = (s_i, s_{-i})$. We distinguish the set of players N from the number of players $n = |N|$.

An important category of games are *zero-sum games*.

Definition 2.1.2 (Zero-sum)**.** A game $G = (N, \mathrm{P}, u)$ is zero-sum if

$$\sum_{i \in N} u_i(\rho) = 0 \quad \forall \rho \in \mathrm{P}.$$

However, a strategically identical set of games are *constant-sum* games.

Definition 2.1.3 (Constant-sum)**.** A game $G = (N, \mathrm{P}, u)$ is constant-sum if for $c \in \mathbb{R}$

$$\sum_{i \in N} u_i(\rho) = c \quad \forall \rho \in \mathrm{P}.$$

Thus, constant-sum and zero-sum games may be referred to interchangeably. *General-sum* games are games where the utility of players do not necessarily add up to a constant. *Multi-player* games are games with any number of players; this term is used to emphasize that there may be more than 2 players.

2.1.1 Solution Concepts and Equilibria

Since an agent's utility depends on the strategies of other players, we cannot generally define a notion of "optimality", which is present in single-agent decision making scenarios. What is optimal for an agent will change depending on what other agents do. Instead, *solution concepts* are criteria for specifying interesting or desirable criteria in games.

First, let us define "optimally" for player i *with respect to* a selection of strategies for the opponents $-i$

Definition 2.1.4 (Best response)**.** Given s_{-i}, the *best response* for i is a strategy s_i^* such that $s_i^* \in \arg\max_{s_i' \in S_i} u_i(s_i', s_{-i})$.

There is always a pure strategy best response to any s_{-i} [54]. Thus

$$u_i(s_i', s_{-i}) - u_i(s_i, s_{-i}) \leqslant 0 \ \forall s_i' \in S_i \iff u_i(\rho_i, s_{-i}) - u_i(s_i, s_{-i}) \leqslant 0 \ \forall \rho_i \in \mathrm{P}_i.$$

The canonical solution concept in game theory is a *Nash equilibrium*. In a Nash equilibrium, no agents wish to deviate from their equilibrium strategy to a different strategy:

Definition 2.1.5 (Nash equilibrium)**.** A strategy profile s is a Nash equilibrium if $\forall i \in N$, we have that s_i is a best response to s_{-i}, or equivalently,

$$u_i(s_i', s_{-i}) - u_i(s_i, s_{-i}) \leqslant 0 \quad \forall s_i' \in S_i.$$

An ϵ-Nash is a relaxation where a player can gain at most ϵ by deviating from their equilibrium strategy.

Definition 2.1.6 (ϵ-Nash equilibrium)**.** A strategy profile s is an ϵ-Nash equilibrium if $\forall i \in N$, we have

$$u_i(s_i', s_{-i}) - u_i(s_i, s_{-i}) \leqslant \epsilon \quad \forall s_i' \in S_i$$

Mixed strategy Nash equilibria are guaranteed to exist for any game [47]. However, they are not the only solution concept we are interested in. Rather than equilibrium, an agent might be interested in guaranteeing themselves some amount of utility regardless of the strategies of other players. The strategy that maximally accomplishes this objective is called a *maxmin strategy*.

Definition 2.1.7 (Maxmin strategy)**.** A strategy s_i is a maxmin strategy if:

$$s_i \in \arg\max_{s'_i \in S_i} \min_{s'_{-i} \in S_{-i}} u_i(s'_i, s'_{-i}).$$

furthermore, the maxmin value for player i is $\max_{s'_i \in S_i} \min_{s'_{-i} \in S_{-i}} u_i(s'_i, s'_{-i})$.

The dual of a maxmin strategy is a *minmax* strategy, where agents seek to minimize the utility of some other agent.

Definition 2.1.8 (Minmax strategy)**.** A strategy s_i is a minmax strategy against player $j \neq i$ if s_i is i's component of s_{-j} where

$$s_{-j} \in \arg\min_{s'_{-j} \in S_{-j}} \max_{s'_j \in S_j} u_j(s'_j, s'_{-j}).$$

The minmax value for player j is $\min_{s'_{-j} \in S_{-j}} \max_{s'_j \in S_j} u_j(s'_j, s'_{-j})$.

In two two-player zero-sum games, Nash equilibria are related to maxmin and minmax strategies by the famous minimax theorem.

Theorem 2.1.9 (von Neumann [66])**.** *In any finite two-player zero-sum game in any Nash equilibrium, each player receives utility equal to their minmax and maxmin values.*

Thus, in two player zero-sum games, Nash equilibrium strategies have the additional properties that they are both maxmin and minmax strategies. This means an agent cannot lose any utility from their equilibrium utility. Moreover, all equilibria yield the same utility, called the *value of the game*. Hence, Nash equilibrium strategies have *prescriptive* value: they have guarantees against other strategies—not just those from the same equilibrium. For ϵ-Nash equilibria, a player can lose at most ϵ from the utility of an ϵ-Nash equilibrium against a worst-case opponent.

Proposition 2.1.10. *In two-player constant-sum games, for any ϵ-Nash equilibrium s and player i, we have*

$$u_i(s) - \min_{s'_{-i} \in S_{-i}} u_i(s_i, s'_{-i}) \leqslant \epsilon.$$

In two-player constant-sum games the value of any ϵ-Nash equilibrium is bounded less than the value of the game.

Proposition 2.1.11. *In a two-player constant sum game, let v_i be the value of the game for player i. Suppose that s is a ϵ-Nash equilibrium. Then*

$$v_i - u_i(s) \leqslant \epsilon.$$

Proof. Let s^* be any Nash equilibrium. Then

$$v_i - u_i(s) = u_i(s^*) - u_i(s) \leqslant u_i(s_i^*, s_{-i}) - u_i(s) \leqslant \epsilon.$$

□

Equilibria in two-player constant-sum games are also exchangeable: two equilibrium strategies from *different* equilibria still form an equilibrium themselves.

Definition 2.1.12 (Exchangeable)**.** Let S' be some set of strategy profiles where S'_i is the set of i's strategies in S'. S' is ϵ-exchangeable if $\forall s \in \times_{i \in N} S'_i$, s is an ϵ-Nash equilibrium.

Proposition 2.1.13 (Fudenberg and Tirole [20])**.** *In two-player zero-sum games, the set of ϵ-Nash equilibria are ϵ-exchangeable.*

However, in games in with more than 2 players, a zero-sum game does not necessarily have the same interesting properties of two-player zero-sum games. This is commonly shown via the *dummy player* argument. We can take any general-sum n player game and produce a strategically identical zero-sum $n+1$ player game by adding a dummy player d which has only one pure strategy and receives utility equal to the negative sum of all other players utilities:

$$u_d(\rho) = -\sum_{i \neq d} u_i(\rho) \quad \forall \rho \in \mathrm{P}.$$

For example, we can turn the coordination game in Figure 1.1 into the zero-sum game in Figure 2.2. Note that this example also shows that in n-player zero-sum games, Nash equilibrium strategies carry no prescriptive value. Playing a Nash equilibrium strategy does not guarantee a player the same utility that they had received in the equilibrium. (a, a) is an equilibrium in Figure 2.2, but if the column player deviates to b, the row player may still lose utility.

	a	b
a	$1, 1, -2$	$0, 0, 0$
b	$0, 0, 0$	$1, 1, -2$

Figure 2.2: A 3 player "zero-sum" game with a dummy player.

2.1.2 Dominated Strategies

If a strategy s_i is always worse than some other strategy s'_i *regardless* of what the other players choose, then we say that s_i is dominated by s'_i

Definition 2.1.14 (Strict domination)**.** Let $s_i, s'_i \in S_i$, if $u_i(s_i, s_{-i}) < u_i(s'_i, s_{-i})$ $\forall s_{-i}$, we say s_i is *strictly dominated* by s'_i

Consider a strictly dominated pure strategy ρ_i. No equilibrium in mixed strategies would put any probability on ρ_i, thus, we can remove ρ_i from the set of i's pure strategies and still preserve the set of equilibria [54]. We may then iteratively repeat this process for i and other players as well. This process is called *iterative removal of strictly dominated strategies*, any pure strategy removed along the way is an *iteratively strictly dominated strategy.*

2.1.3 Mediated Equilibria

Nash equilibria require that players' strategies are uncorrelated; *mediated equilibria* are a generalization to settings where player's strategies may be be correlated. In mediated equilibria, a mediator recommends strategies for each player from some joint distribution over strategy profiles. Player's decide to either follow this recommendation, or instead deviate. A deviation $\phi \in \Phi$ is a mapping $\phi : S_i \to S_i$ that transforms a learner's strategy into some other strategy. *Regret* measures the amount the learner would prefer to deviate to $\phi(s_i)$:

$$R_i(\phi, s) \doteq u_i(\phi(s_i), s_{-i}) - u_i(s_i, s_{-i}).$$

Let $\mu \in \Delta(\mathrm{P})$ be a distribution over pure strategy profiles and $(\Phi_i)_{i \in N}$ be a choice of deviation sets for each player.

Definition 2.1.15 (ϵ-Mediated Equilibrium [45])**.** We say $m = (\mu, (\Phi_i)_{i \in N})$ is an *ϵ-mediated equilibrium* if $\forall i \in N, \phi \in \Phi_i$ we have

$$\mathbb{E}_{\rho \sim \mu} \left[R_i(\phi, s) \right] \leqslant \epsilon.$$

A *mediated equilibrium* is a 0-mediated equilibrium.

Different sets of deviations determine the strength of a mediated equilibrium. For normal-form games, the set of *swap* deviations, Φ_{SW}, are all possible mappings $\phi : \mathrm{P}_i \to \mathrm{P}_i$. We may apply a swap deviation ϕ to a mixed strategy s_i by taking its pushforward measure:

$$[\phi(s_i)](\rho_i) = \sum_{\rho_i' \in \phi^{-1}(\rho_i)} s_i(\phi(\rho_i')),$$

where $\phi^{-1}(\rho_i) = \{\rho_i' \in \mathrm{P}_i \mid \phi(\rho_i') = \rho_i\}$.

Internal deviations, which exchange a particular action recommended by the mediator with another, offer the same strategic power as swap deviations [19]. The set of external deviations Φ_{EX} is even more restricted: $\phi \in \Phi_{EX}$ maps all (mixed) strategies to some particular pure strategy; i.e. $\Phi_{EX} = \{\phi \in \Phi_{SW} \mid \exists \rho_i, \forall s_i', \phi(s_i') = \rho_i\}$. Since each deviation only maps to a single pure strategy, we write $R_i(\rho_i, s)$ as a shorthand for $R_i(\phi, s)$ when ϕ maps to ρ_i.

Note that a special case of mediated equilibria are Nash equilibria. If some mediated equilibrium μ is a product distribution (i.e. $\mu = \bigotimes_{i \in N} s_i$ for $s_i \in S_i$) and $\Phi_i \supseteq \Phi_{EX}$ $\forall i \in N$ then m is a Nash equilibrium. Similarly an ϵ-mediated equilibrium is an ϵ-Nash equilibrium if μ is a product distribution and all players have no regret w.r.t. Φ_{EX}. A *coarse correlated equilibrium* (CCE) [46] is a mediated equilibrium where no agent wants to deviate according to any Φ_{EX}. When referring to CCE, we identify them by the distribution μ alone, since $(\Phi)_{i \in N}$ is implicit. We may equivalently write the definition of an ϵ-CCE as follows

$$\mathbb{E}_{\rho \sim \mu} \left[u_i(\phi(\rho_i), \rho_{-i}) - u_i(\rho) \right] \leqslant \epsilon \quad \forall i \in N, \phi \in \Phi_{EX}. \tag{2.1}$$

Since elements in Φ_{EX} only map to a single pure strategy, (2.1) is equivalent to

$$\mathbb{E}_{\rho \sim \mu} \left[u_i(\rho_i', \rho_{-i}) - u_i(\rho) \right] \leqslant \epsilon \quad \forall i \in N, \rho_i' \in \mathrm{P}_i.$$

	b	s
b	$1, 2$	$0, 0$
s	$0, 0$	$2, 1$

Figure 2.3: Battle of the Sexes

For example, Figure 2.3 shows a games called "Battle of the Sexes". A distribution μ where $\mu(B, B) = 0.5$ and $\mu(S, S) = 0.5$ is a CCE. The expected utility under this distribution is $\mathbb{E}_{\rho\sim\mu}[u_i(\rho)] = 0.5 \cdot 2 + 0.5 \cdot 1 = 1.5$. However, any deviation would strictly decrease a player's utility if the other player continues to play according to μ's recommendations. Without loss of generality, consider the row player. Deviating to B would give an expected utility of $\mathbb{E}_{\rho\sim\mu}[u_i(B, \rho_{-i})] = 0.5 \cdot 2 + 0.5 \cdot 0 = 1$ and deviating to S would give an expected utility of $\mathbb{E}_{\rho\sim\mu}[u_i(S, \rho_{-i})] = 0.5 \cdot 0 + 0.5 \cdot 1 = 0.5$. Hence, neither player has an incentive to deviate from their recommendations.

The strategies in a mediated equilibrium are potentially correlated with each other. This means that in order for a player to play a strategy from a mediated equilibrium against agents with whom the player cannot correlate, the player must first extract it by marginalizing out other agent's strategies.

Definition 2.1.16 (Marginal strategy)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let s_i^μ be the *marginal strategy* for i, where

$$s_i^\mu(\rho_i) \doteq \sum_{\rho_{-i}\in \mathrm{P}_{-i}} \mu(\rho_i, \rho_{-i}).$$

Definition 2.1.17 (Marginal strategy profile)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let s^μ be a *marginal strategy profile*, where $s_i^\mu(\rho_i) \doteq \sum_{\rho_{-i}\in \mathrm{P}_{-i}} \mu(\rho_i, \rho_{-i}) \forall i \in N$.

In two-player constant-sum games, a well-known fact is that the marginal strategy profiles of CCE are Nash equilibria. We show a proof of this fact extending to CCE in Appendix A.2.

Proposition 2.1.18. *If μ is an ϵ-CCE of a two-player constant-sum game G, then s^μ is a 2ϵ-Nash equilibrium.*

2.2 Hindsight Rationality

Where does the recommendation distribution μ come from? Mediated equilibria can be specified by some designer; for example, a traffic light recommends actions to be taken by drivers. Alternatively, mediated equilibria can arise naturally via the behaviour of *learning* agents.

The hindsight rationality framework [45] conceptualizes the goal of an agent as finding a strategy that minimizes regret with respect to a set of deviations Φ_i. An agent is *hindsight rational* with respect to a set of deviations Φ_i if the agent does not have positive regret with respect to any deviation in Φ_i.

Learning takes place in an online learning environment. At each iteration t, a learning agent i chooses a strategy s_i^t while all other agents choose a strategy profile s_{-i}^t. The *cumulative regret* is the total amount of regret experienced with respect to a deviation ϕ, summed over iterations. This is expressed formally as

$$R_i^T(\phi) \doteq \sum_{t=1}^{T} R_i(\phi, s^t).$$

Let $\mathcal{A}$ be an algorithm that selects s_i^t at each iterations. $\mathcal{A}$ is a no-Φ-regret learning algorithm if the maximum average positive regret tends to 0.

$$\lim_{T\to\infty} \max_{\phi\in\Phi} \left(\frac{1}{T} R_i^T(\phi)\right) \to 0.$$

If each agent uses a no-Φ_i-regret learning algorithm w.r.t. a set of deviations Φ_i, the *empirical distribution of play* $\hat{\mu}$ converges to a mediated equilibrium. Formally, let $\hat{\mu} \in \Delta(\mathrm{P})$ where $\hat{\mu}(\rho) \doteq \sum_{t=1}^{T} \left(\prod_{i\in N} s_i^t(\rho_i)\right)$ be the empirical distribution of play. As $T \to \infty$, $\hat{\mu}$ nears μ of a mediated equilibrium $(\mu, (\Phi_i)_{i\in N})$.

The choice of $(\Phi_i)_{i\in N}$ determines the nature of the mediated equilibrium—provided the learning algorithm for player i is no-Φ_i-regret [24]. For example, if all players are hindsight rational w.r.t. Φ_{EX}, then $\hat{\mu}$ converges the set of CCE and if all players are hindsight rational w.r.t. Φ_I then $\hat{\mu}$ converges to a correlated equilibrium [4]. Theorem 2.2.1 makes this connection.

Theorem 2.2.1 (Greenwald and Jafari [23]). *If all players play a no-Φ_i-regret learning algorithm, then the empirical distribution of play $\hat{\mu}$ converges to μ of the mediated equilibrium $(\mu, (\Phi_i)_{i \in N})$.*

Corollary 2.2.2. *If all players play a no-Φ_{EX}-regret learning algorithm, then the empirical distribution of play $\hat{\mu}$ converges to a CCE μ. Moreover, in two player zero-sum games, the marginal strategy profile s^{μ} is a Nash equilibrium and s_i^{μ} is equal to the* average strategy *across iterations; i.e.*

$$s_i^{\mu}(\rho_i) = \bar{s}_i^T(\rho_i) \doteq \sum_{t=1}^{T} s_i^t(\rho_i) \quad \forall \rho_i \in \mathrm{P}_i.$$

Suppose the utility of players to add up in a way that is boundedly close to 0 [22]. We call this property δ-zero-sum:

$$|u_i(\rho) - u_{-i}(\rho)| \leqslant \delta \quad \forall \rho \in \mathrm{P}.$$

In two-player δ-zero-sum games, regret-minimization produces approximate Nash equilibrium.

Theorem 2.2.3 (Gibson [22]). *Let $s^1, ..., s^T$ be a sequence of strategy profiles produced by a no–external-regret algorithm. If $\max_{\phi \in \Phi_{EX}} \frac{1}{T} R_i^T(\phi) \leqslant \epsilon$ for all $i \in \{1, 2\}$ and the game is δ-zero-sum, then the profile $\bar{s}$ of average strategies $\bar{s}_i \doteq \sum_{t=1}^{T} s_i^t$ is a $2(\delta + \epsilon)$-Nash equilibrium.*

2.2.1 Removal of Strictly Dominated Strategies by No-Regret Algorithms

The empirical distribution of play of no-external-regret algorithms converges to CCE. Additionally, the algorithm will prune out iteratively strictly dominated strategies from the support of that CCE so long as the algorithm assigns zero probability to actions with negative regret.

Theorem 2.2.4 (Gibson [22]). *Let $s^1, ...s^n$ be a sequence of strategy profiles in a normal-form game where all players strategies are computed by a no-external-regret algorithm where $\forall i \in N, \rho_i \in \mathrm{P}_i$, $T \geqslant 0$ if $R_i^T(\rho_i) < \max_{\rho}$ and $R_i^T(\rho_i) < \max_{\rho_i' \in \mathrm{P}_i R_i^T(\rho_i')}$ then $s_i^{T+1} = 0$. If s_i is an iteratively strictly dominated strategy, then there exists an integer T_0 such that for all $T \geqslant T_0$, $supp(s_i) \nsubseteq supp(s_i^T)$.*

Regret-Matching

One no-external-regret algorithm is called *Regret-Matching* [26]. Let ρ_i^ϕ be the pure strategy that $\phi \in \Phi_{EX}$ maps all mixed strategies to (i.e. $\phi(s_i) = \rho_i^\phi \ \forall s_i \in S_i$). We overload the cumulative regret function to accept a pure strategy as argument: $R_i^T(\rho_i^\phi) = R_i^T(\phi)$.

Regret-Matching works by maintaining a cumulative regret $R_i^T(\rho_i)$ for each $\rho_i \in \mathrm{P}_i$. Upon completing iteration T, s_i^{T+1} plays ρ_i with probability proportional to the positive cumulative regret:

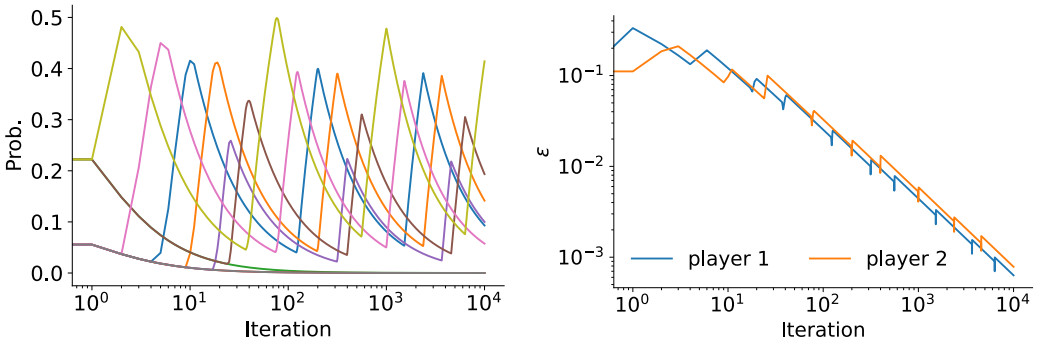
$$s_i^{T+1}(\rho_i) \leftarrow \frac{\max(R_i^T(\rho_i), 0)}{\sum_{\rho_i' \in \mathrm{P}_i} \max(R_i^T(\rho_i'), 0)}$$

By Theorem 2.2.1, if all players employ Regret-Matching, the empirical distribution of play will converge to the *set of CCE*. However, we want to note that while the empirical distribution will converge to the set of CCE, it may not converge to a *particular CCE*, and may instead cycle amongst a set of CCE. To illustrate this point, consider the modified version of the Shapley Game in Figure 2.4.

	l	c	r
t	$1, 0$	$0, 1$	$0, 0$
m	$0, 0$	$1, 0$	$0, 2$
b	$0, 1$	$0, 0$	$1, 0$

Figure 2.4: The modified Shapley Game

We ran Regret-Matching on this game and found that the empirical distribution of play follows in cycles with exponentially-increasing lengths; in Figure 2.5a we show the probability of each pure strategy profile and see that the probability cycles as the number of iterations increases. However, as shown in Figure 2.5b, as the number of iterations increases, the empirical distribution of play approaches being a CCE, since the maximum value of a deviation ϵ decreases with time. Thus, the empirical distribution of play does not necessarily converge to a particular CCE, but cycles while at any iteration nearing being a CCE.



(a) (b)

Figure 2.5: (a) The empirical distribution of play over iterations. Each line shows the probability of a different pure strategy profile under the empirical distribution of play. (b) ϵ-CCE convergence in the modified Shapley Game. We show the maximum deviation incentive ϵ for each player.

2.3 Extensive-Form Games

Next we consider games where players take actions *sequentially*. We use the imperfect information extensive-form game (EFG) as a model for sequential multi-agent strategic situations. An imperfect information extensive-form game is a 10-tuple $(N, \mathcal{A}, H, Z, A, P, u, \mathcal{I}, c, \pi_c)$ where N is a set of players; $\mathcal{A}$ is a set of actions; H is a set of sequences of actions, called *histories*; $Z \subseteq H$ is a set of terminal histories; $A : H \to \mathcal{A}$ is a function that maps a history to available actions; $P : H \to N$ is the player function, which assigns a player to choose an action at each non-terminal history; u $= \{u_i\}_{i \in N}$ is a set of utility functions where $u_i : Z \to \mathbb{R}$ is the utility function for player i; $\mathcal{I} = \{\mathcal{I}_i\}_{i \in N}$ where $\mathcal{I}_i$ is a partition of the set $\{h \in H : P(h) = i\}$ such that if $h, h' \in I \in \mathcal{I}_i$ then $A(h) = A(h')$. We call an element $I \in \mathcal{I}_i$ an information set. The chance player c has a function $\pi_c(a, h)$ $\forall h : P(h) = c$ which returns the probability of random nature events $a \in \mathcal{A}$. Let $N_c = N \cup \{c\}$ be the set of players including chance.

For some history h, the jth action in h is written h_j. A sub-history of h from the jth to kth actions is denoted $h_{j:k}$ and we use $h_{:k}$ as a short-hand for $h_{0:k}$. If a history h' is a prefix of history h, we write $h' \sqsubseteq h$ and if h' is a proper prefix of h, we write $h' \sqsubset h$.

A *pure strategy* in an EFG is a deterministic choice of actions for the player at every decision point. We use $\rho_i : \mathcal{I}_i \to \mathcal{A}$ to denote a pure strategy of player i, and the set of all pure strategies as P_i. Likewise, $s_i \in \Delta(\mathrm{P}_i) = S_i$ is a *mixed strategy*, where $\Delta(X)$ denotes the set of probability distributions over a domain X.

There are an exponential number of pure strategies in the number of information sets. A *behaviour strategy* is a compact representation of the behaviour of an agent that assigns a probability distribution over actions to each information set. We use $\pi_i \in \Pi_i = (\Delta(A(I)))_{I \in \mathcal{I}_i}$ to denote a behaviour strategy of player i and $\pi_i(a, I)$ as the probability of playing action a at I. Let $I(h)$ be the unique information set such that $h \in I$. We overload $\pi_i(a, h) = \pi_i(a, I(h))$. We use $\rho \in \mathrm{P}, s \in S$ and $\pi \in \Pi$ to denote pure, mixed and behaviour strategy profiles, respectively. Note that P is a subset of both Π and S.

Given a behaviour strategy profile, let

$$\begin{aligned} p_i(h_1, h_2, \pi_i) &\doteq \prod_{h_1 \sqsubseteq ha \sqsubseteq h_2, P(ha)=i} \pi_i(a, h) \\ p(h_1, h_2, \pi) &\doteq \prod_{i \in N_c} p_i(h_1, h_2, \pi_i) \\ p_{-i}(h_1, h_2, \pi_{-i}) &\doteq \prod_{j \in N_c \setminus \{i\}} p_j(h_1, h_2, \pi_j) \end{aligned}$$

be the probability of transitioning from history h_1 to h_2 according to π_i, π and π_{-i}, respectively. Let $p_i(z, \pi_i), p_{-i}(z, \pi_i)$ and $p(z, \pi_i)$ be short-hands for $p_i(\varnothing, z, \pi_i), p_{-i}(\varnothing, z, \pi_{-i})$ and $p(\varnothing, z, \pi)$, respectively, where $\varnothing$ is the empty history.

We define the utility of a behaviour strategy as:

$$u_i(\pi) \doteq \mathbb{E}_{z \sim \pi}\left[u_i(z)\right] = \sum_{z \in Z} p(z, \pi) u_i(z) = \sum_{z \in Z} \left(\prod_{i \in N_c} p_i(z, \pi_i) \right) u_i(z).$$

Perfect recall is a common assumption made on the structure of information sets in EFGs that prevents players from forgetting information they once possessed. Formally, for any $h \in I$ let $X_i(h)$ denote the set of (I, a) s.t. $I \in \mathcal{I}_i$ and $\exists h' \in I$ and $h'a \sqsubseteq h$. Let $X_{-i}(h)$ be defined analogously for $-i$ and $X(h)$ for all players.

Definition 2.3.1 (Perfect recall)**.** If $\forall I \in \mathcal{I}_i, \forall h, h' \in I, X_i(h) = X_i(h')$ then i has perfect recall. If all players possess perfect recall in some EFG G, we call G a game of perfect recall.

In games of perfect recall, the set of behaviour strategies and mixed strategies are equivalent: any behaviour strategy can be converted into a mixed strategy which is outcome equivalent over the set of terminal histories (i.e. has the same distribution over Z) and vice-versa.

Theorem 2.3.2 (Kuhn [33])**.** *In games of perfect recall, any behaviour strategy π_i has an equivalent mixed strategy s_i (and vice versa), such that*

$$p_i(z, \pi_i) = \mathbb{E}_{\rho_i \sim s_i} \left[p_i(z, \rho_i) \right].$$

Theorem 2.3.2 establishes a connection between equilibria in behaviour and mixed strategies: a Nash equilibrium behaviour strategy profile implies the equivalent mixed strategy profile is also a Nash equilibrium in mixed strategies and vice-versa.

We may also reduce any extensive-form game into an equivalent normal-form game.

Definition 2.3.3 (Induced normal-form)**.** The *induced normal-form* of an extensive-form game G (with utility functions u_i) is a normal-form game $G' = (N, \mathrm{P}, u')$ such that $u'_i(\rho) = u_i(\rho)$

The induced normal-form of an EFG has players making all decisions up-front. It is not always practical to construct an induced normal-form, but the concept is useful for proving things about EFGs.

2.4 Hindsight Rationality in Extensive-Form Games

In sequential decision making scenarios, the set of deviations is even more rich [45]. Figure 2.6 gives a sense of the deviation landscape in extensive-form games. We refer the reader to [45] and [44] for an in-depth description. For

our purposes, it suffices to say that all of these deviation classes—with the exception of action deviations [53]—are stronger than external deviations[1]. This means that the equilibria of any algorithm that minimizes regret w.r.t. a class of deviations that is at least as strong as external deviations still inherit all the properties of CCE. This includes internal counterfactual regret minimization (ICFR) [14], extensive-form regret minimization (EFR) [44] (so long as EFR is instantiated with a sufficiently strong deviation class) as well as deep regret minimization approaches, such as deep counterfactual regret minimization (Deep CFR) [8] and DREAM [58].

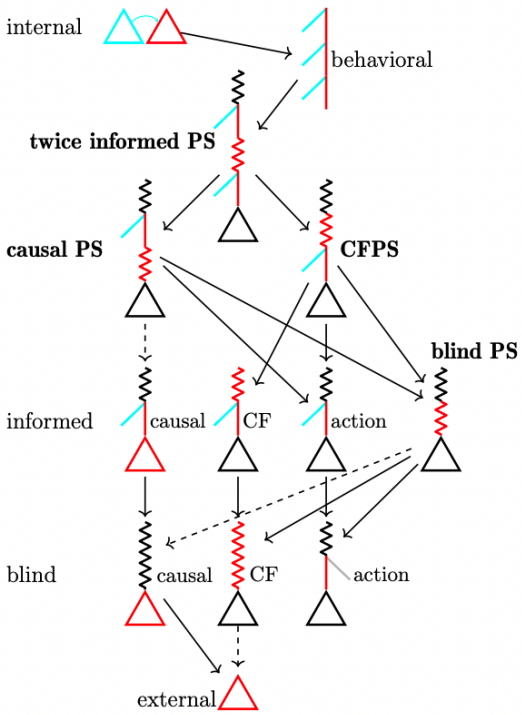


Figure 2.6: The deviation landscape for EFGs. Taken from Morrill [43] with permission.

[1]Action deviations are so weak they do not even imply Nash equilibria in two-player constant-sum games, see Appendix A.1.

2.5 Regret Minimization in Extensive Form Games

The size of extensive-form games makes regret minimization with algorithms like Regret-Matching infeasible. A number of algorithms exploit the structure of extensive-form games in order to minimize regret efficiently. Central to all of these approaches is CFR.

2.5.1 Counterfactual Regret Minimization

Counterfactual Regret Minimization (CFR) is an algorithm that exploits the structure of EFGs in order to efficiently minimize regret with respect to external deviations [69]. CFR works by training instances of Regret-Matching at each information set, but uses an alternate version of regret called *counterfactual regret*. The *counterfactual value function* $v_I : A(I) \times \Pi \to \mathbb{R}$ is the expected utility of player $i = P(I)$ given they played to reach I, take action a, then continue to play π_i, all the while $-i$ play π_{-i}.

$$v_I(a, \pi) \doteq \sum_{h \in I} \sum_{z \sqsupset ha} p_{-i}(z, \pi_{-i}) p_i(ha, z, \pi_i) u_i(z).$$

Additionally, we overload v_I to allow randomization at I:

$$v_I\left(\pi_i'(I), \pi\right) \doteq \sum_{a \in A(I)} \pi_i'(a, I) v_I(a, \pi).$$

and if $\pi_i' = \pi_i$, we simply write $v_I\left(\pi\right)$ The *immediate counterfactual regret* at I is

$$R_I(a, \pi) = v_I\left(a, \pi\right) - v_I\left(\pi\right).$$

At iteration T, CFR accumulates immediate counterfactual regret at each information set I, and updates $\pi_i^{T+1}(I)$ to be proportional to the positive part of the cumulative immediate counterfactual regret $R_I^T(a) = \sum_{t=1}^T R_I(a, \pi^t)$:

$$\pi_i^{T+1}(a, I) \leftarrow \frac{\max\left(R_I^T(a), 0\right)}{\sum_{a' \in A(I)} \max\left(R_I^T(a'), 0\right)}.$$

The *average strategy* is a behaviour strategy averaged across all iterations, defined as

$$\bar{\pi}_i^T(a, I) = \sum_{t=1}^{T} \pi_i^T(a, I).$$

CFR minimizes external regret [69], so the *average strategy profile* of CFR $\bar{\pi}$ approximates a Nash equilibrium in 2 player zero-sum games. However, note that these average strategies do not necessarily converge to CCE themselves in general; computing a CCE requires additional machinery. For example, CFR-JR [13] uses the iterates of CFR to construct the empirical distribution of play at each iteration by first converting these iterates into mixed strategies. CFR-JR then averages across iterations to produce a distribution over pure strategies that indeed converges to a CCE. This distribution can get very large for moderately-sized EFGs.

2.5.2 Behaviour of CFR

Can we give any additional characterizations to CFR's behaviour? Indeed we can. The empirical distribution of play generated by CFR in self-play implicitly converges to a class of mediated equilibrium called a *counterfactual coarse correlated equilibrium* (CF-CCE). This is mediated equilibrium where the recommendation distribution μ is hindsight rational with respect to *blind counterfactual deviations* Φ_{CF}. Given a strategy π_i, A blind counterfactual deviation $\phi_{a^\odot}^{I^\odot} \in \Phi_{CF}$ plays to reach a target information set $I^\odot$, plays action $a^\odot$, then returns to playing π_i

$$\left[\phi_{a^\odot}^{I^\odot}(\pi_i)\right](I) = \begin{cases} a^\odot & \text{if } I = I^\odot \\ a^{\rightarrow I^\odot} & \text{if } I \prec I^\odot \\ \pi_i(I) & \text{o.w.} \end{cases}$$

Where $a^{\rightarrow I^\odot}$ is the action at $I \prec I^\odot$ that plays to reach $I^\odot$. $a^{\rightarrow I^\odot}$ is well-defined in games of perfect recall.

Corollary 2.5.1 (Morrill *et al.* [45])**.** *CFR is no-regret/hindsight rational with respect to* Φ_{CF}*.*

Having no regret w.r.t. blind counterfactual deviations *on its own* is not sufficient to guarantee having no regret w.r.t. external deviations (which are sufficient for Nash equilibria in 2-player zero-sum games), as we show next. Consider the 1-player game in Figure 2.7. The player's strategy is shown in blue; clearly, this strategy does not minimize external regret, since instead playing R at I_1 and l at both I_2 and I_3 would increases expected utility to 2. However, no counterfactual deviation increases the agent's utility, so it has no regret with respect to this set.

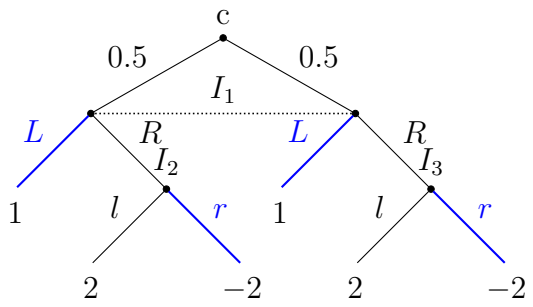


Figure 2.7: Hindsight Rational w.r.t. Φ_{CF} does not imply Hindsight Rational w.r.t. Φ_{EX}

However, CFR is hindsight rational *at each* information set. CFR is *observably sequentially rational* w.r.t. Φ_{CF} and this is sufficient to ensure CFR is no-external regret [45] .

Additionally, CFR prunes *strictly dominated actions*.

Definition 2.5.2 (Strictly dominated action)**.** An action $a \in A(I)$ of an EFG is a *strictly dominated action* if there exists a strategy π_i' such that $\forall \pi \in \Pi$ such that $\sum_{h \in I} p_{-i}(h, \pi_{-i}) > 0$ we have $v_I(a, \pi) < v_I(\pi_i', \pi_{-i})$.

An *iteratively strictly dominated action* is an action that becomes a strictly dominated action after the removal of other iteratively strictly dominated actions.

Theorem 2.5.3 (Gibson [22] (informal))**.** *CFR will play iteratively strictly dominated action with vanishing probability.*

2.5.3 Extensions of CFR

Regret minimization with CFR can become infeasible in very large extensive-form games, hence many extensions of CFR have been proposed in order to allow regret-minimization in these EFGs. We detail one extension, CFR+ and include a brief survey of other extensions.

CFR+ [63, 64] is an algorithm that modifies CFR in three imporant ways. First, CFR+ only accumulates positive regret-like values. Formally, CFR+ tracks a value $Q_I^T(a)$ for each information set I and action a. At iteration T, this quantity is updated by $Q_I^{T+1}(a) \leftarrow \max(Q_I^T(a) + R_I(a, \pi^T), 0)$. The probability given to action a at the next iteration is

$$\pi_i^{T+1}(a, I) \leftarrow \frac{Q_I^{T+1}(a)}{\sum_{a' \in A(I)} Q_I^{T+1}(a')}.$$

Second, CFR+ alternates updates to player's Q values, rather than performing them simultaneously; this speeds up convergence. Lastly, CFR+ returns a weighted average strategy, rather than the uniformly-weighted average strategy of CFR. At iteration T, the weighted average strategy strategy is

$$\bar{\pi}_i^T \doteq \frac{2}{T^2 + T} \sum_{t=1}^{T} t \cdot \pi_i^T.$$

Taken together, these modifications greatly improve the efficiency of CFR+—to the point where CFR+ was able to compute approximate Nash equilibria of heads-up limit texas hold 'em [7], a game with approximately 10^{14} information sets.

Other Extensions

Monte Carlo CFR [35] samples a subset of the full game tree when computing updates, rather than performing full tree traversals as CFR does. A number of variants of this algorithm have been proposed [21, 30, 52]. Regression CFR [68] uses regression tree function approximation using features of of the information sets to approximate the regret of CFR. Deep CFR [8] and DREAM [58] use neural networks as the function approximator.

A number of CFR-inspired algorithms have achieved expert-level performance in the game of heads-up no-limit Texas hold'em (HUNL). Deepstack was the first program to outplay human professionals at HUNL; it combines continual resolving at each decision points with limited look ahead with neural networks as function approximators. Libratus [9] reached superhuman levels by instead constructing a *blueprint* strategy on an *abstraction* of HUNL—a smaller game which is easier to compute strategies on. Based on play in HUNL, these blueprint strategies are refined as needed. Pluribus [10] outperformed professionals at 6 player no-limit Texas hold'em using a similar approach to Libratus.

Chapter 3

Self-Play and Polymatrix Games

Are there n-player games where self-play will compute a desirable strategy? We show that in games that approximately decompose into a set of two-player constant-sum games (called constant-sum polymatrix games) where global ϵ-Nash equilibria are boundedly far from Nash-equilibria in each subgame (called subgame stability), any no-external-regret algorithm that learns by self-play will produce a strategy with bounded vulnerability, exchangeability and value.

In this chapter, we interleave theoretical results with algorithmic results, where for each property we give an algorithm showing how to compute the relevant values on normal-form games.

3.1 Self-Play and Vulnerability

Given an environment containing agents with unknown behaviour, how can a learning algorithm produce a strategy that will perform well in this environment? A learning algorithm could choose some behaviour of the other agents during training, and then use data generated from these simulated agents to train itself. However, the choice of other agents during learning affects the strategy that is learned.

Self-play has a learning algorithm train with copies of itself as the other agents. If the algorithm is a no-Φ_i-regret algorithm for each agent i, the learned behaviour will converge to a mediated equilibrium; this gives a nice characterization of the convergence behaviour of the algorithm. For the remainder of this thesis when we say "self-play" we are referring to self-play

using a no-Φ-regret algorithm.

However, the strategies in a mediated equilibrium are correlated with each other. This means that in order to play a strategy learned in self-play, an agent must first extract it by marginalizing out other agent's strategies.[1] This new *marginal strategy* can then be played against new agents with whom the agent did not train (and thus correlate).

Once a strategy has been extracted via marginalization, learning can either continue with the new agents (and potentially re-correlate), or the strategy can remain fixed. We focus on the case where the strategy remains fixed. In doing so we can guarantee the performance of this strategy if learning stops, but also show guarantees about the initial performance of a strategy that continues to learn; this is especially important in safety-critical domains.

Given a marginal strategy s_i^μ, we can bound its underperformance against new agents that behave differently from the (decorrelated) training opponents by its vulnerability.

Definition 3.1.1 (Vulnerability)**.** The *vulnerability* of a strategy profile s for player i with respect to $S'_{-i} \subseteq S_{-i}$ is

$$\mathrm{Vul}_i\left(s, S'_{-i}\right) \doteq u_i(s) - \min_{s'_{-i} \in S'_{-i}} u_i(s_i, s'_{-i}).$$

Vulnerability gives a measure of how much worse s will perform with new agents compared to its training performance under pessimistic assumptions—that $-i$ play the strategy profile in S'_{-i} that is worst for i. We assume that $-i$ are not able to correlate their strategies.

Thus, if a marginal strategy profile s^μ is learned through self-play and $\mathrm{Vul}_i\left(s^\mu, S'_{-i}\right)$ is small, then s_i^μ performs roughly as well against new agents $-i$ playing some strategy profile in S'_{-i}. S'_{-i} is used to encode assumptions about the strategies of opponents. $S'_{-i} = S_{-i}$ means opponents could play *any* strategy, but we could also set S'_{-i} to be the set of strategies learnable through self-play if we believe that opponents would also be using self-play as a training procedure.

[1] Please refer to Definition 2.1.16 for the definition of a marginal strategy.

Some games have properties that make the vulnerability of strategies learned in self-play low with respect to S_{-i}. For example, in two-player constant-sum games the marginal strategies learned in self-play generalize well to new opponents since any Nash equilibrium strategy is also a maxmin strategy [66].

We may also want to know how much an agent would prefer to deviate to a different strategy when facing new opponents—i.e. how far are they from the optimal strategy against these new opponents? Suppose that each agent $j \in N$ learns in self-play, and computes the marginal strategy of a CCE μ^j. Note that μ^j is not necessarily the same as μ^i for $i \neq j$. Agents then jointly select a strategy profile $s = (s_i^{\mu^i})_{i \in N}$, where each plays a marginal strategy from *their* CCE. We wish to determine $\max_{i \in N} \max_{s_i^* \in S_i} u_i(s_i^*, s_{-i}) - u_i(s)$. This is equal to the *exchangeability*[2] of $S^\mu = \{s^\mu \mid \mu \text{ is a CCE}\}$, which we denote as $\text{Ex}(S^\mu)$. In two-player constant-sum games, $\text{Ex}(S^\mu) = 0$.

Summarizing this discussion and the background in Chapter 2, two-player constant-sum games have the following desirable properties

1. The marginal strategies of a CCE are Nash equilibria. A corollary is that no-external regret learning algorithms will converge to Nash equilibria.

2. There is a unique value of the game.

3. A Nash equilibrium strategy guarantees itself that value against any opponent; i.e., the vulnerability with respect to S_{-i} is 0.

4. The set of Nash equilibria are exchangeable.

We seek to identify classes of multi-player games that satisfy these properties.

3.2 Polymatrix Games

Multi-player games are fundamentally more complex than two-player constant-sum games [16, 15]. However, certain multi-player games can be decomposed into a graph of two-player games, where a player's payoffs depend only on their

[2]Please refer to Definition 2.1.12.

strategy and the strategies of players who are neighbours in the graph [6]. In these *polymatrix* games (a subset of graphical games [31]) Nash equilibria can be computed efficiently if player's utilities sum to a constant [12, 11].

Definition 3.2.1 (Polymatrix game)**.** A *polymatrix game* $G = (N, E, \mathrm{P}, u)$ consists of a set N of players, a set of edges E between players, a set of pure strategy profiles P, and a set of utility functions $u = \{u_{ij}, u_{ji} \mid \forall (i,j) \in E\}$ where $u_{ij}, u_{ji} : \mathrm{P}_i \times \mathrm{P}_j \to \mathbb{R}$ are utility functions associated with the edge (i,j). The *global utility function* $u_i : \mathrm{P} \to \mathbb{R}$ is a sum across subgames: $u_i(\rho) = \sum_{(i,j)\in E_i} u_{ij}(\rho_i, \rho_j)$ for each player where we use $E_i \subseteq E$ to denote the set of edges where i is a player.

We refer to the normal-form *subgame* between (i,j) as $G_{ij} = (\{i,j\}, \mathrm{P}_i \times \mathrm{P}_j, (u_{ij}, u_{ji}))$. When denoting all players except for i and j, we write $-ij$. We write $|E_i|$ to denote the number of subgames for which i is a player.

Definition 3.2.2 (Constant-sum polymatrix)**.** We say a polymatrix game G is *constant-sum* if for some constant c we have that $\forall \rho \in \mathrm{P}$, $\sum_{i\in N} u_i(\rho) = c$.

Constant-sum polymatrix (CSP) games have the desirable property that all CCE factor into a product distribution; i.e., are Nash equilibria [11]. We give a relaxed version of this property. First, consider the following useful proposition.

Proposition 3.2.3 (Cai *et al.* [11])**.** *In CSP games, for any CCE μ, if i deviates to s_i, then their expected utility if other players continue to play μ is equal to their utility if other players were to play the marginal strategy profile s^{μ}_{-i}:*

$$\mathbb{E}_{\rho\sim\mu}[u_i(s_i, \rho_{-i})] = u_i(s_i, s^{\mu}_{-i}) \ \forall s_i \in S_i.$$

Proposition 3.2.4. *If μ is an ϵ-CCE of a CSP game G, s^{μ} is an $n\epsilon$-Nash of G.*

Proof. Since μ is an ϵ-CCE, $\forall i \in N$, we have

$$\max_{\rho'_i \in \mathrm{P}_i} \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho'_i, \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant \epsilon$$

which implies (by Proposition 3.2.3) that $\forall i \in N$,

$$\begin{aligned} &\max_{\rho_i' \in \mathrm{P}_i} u_i(\rho_i', s_{-i}^{\mu}) - \mathbb{E}_{\rho \sim \mu}\left[u_i(\rho)\right] \leqslant \epsilon \\ \implies &\max_{\rho_i' \in \mathrm{P}_i} u_i(\rho_i', s_{-i}^{\mu}) \leqslant \epsilon + \mathbb{E}_{\rho \sim \mu}\left[u_i(\rho)\right]. \end{aligned}$$

Summing over N, we get

$$\begin{aligned} \sum_{i \in N} \max_{\rho_i' \in \mathrm{P}_i} u_i(\rho_i', s_{-i}^{\mu}) &\leqslant \sum_{i \in N} \left(\epsilon + \mathbb{E}_{\rho \sim \mu}\left[u_i(\rho)\right]\right) && (3.1) \\ &= \sum_{i \in N} \epsilon + \sum_{i \in N} \mathbb{E}_{\rho \sim \mu}\left[u_i(\rho)\right] && (3.2) \\ &= \sum_{i \in N} \epsilon + \mathbb{E}_{\rho \sim \mu}\left[\sum_{i \in N} u_i(\rho)\right] && (3.3) \\ &= n\epsilon + c && (3.4) \\ &= n\epsilon + \sum_{i \in N} u_i(s^{\mu}). && (3.5) \end{aligned}$$

Where (3.4) and (3.5) use the fact that $\forall \rho \in \mathrm{P}, \sum_{i \in N} u_i(\rho) = c$ for some constant. The above inequalities give us

$$\sum_{i \in N} \max_{\rho_i' \in \mathrm{P}_i} u_i(\rho_i', s_{-i}^{\mu}) \leqslant n\epsilon + \sum_{i \in N} u_i(s^{\mu}).$$

Rearranging, we get

$$\sum_{i \in N} \underbrace{\max_{\rho_i' \in \mathrm{P}_i} u_i(\rho_i', s_{-i}^{\mu}) - u_i(s^{\mu})}_{\geqslant 0} \leqslant n\epsilon.$$

All terms in the sum are are non-negative because ρ_i' is a best-response to s_{-i}^{μ}. Then any particular term in the summation is upper bounded by $n\epsilon$. □

This means no-external-regret learning algorithms will converge to Nash equilibria, and thus do not require a mediator to enable the equilibrium. However, they do not necessarily have the property of two-player constant-sum games that all (marginal) equilibrium strategies are maxmin strategies [11]. Thus Nash equilibrium strategies in CSP games have no vulnerability guarantees (we provide an example in Figure 3.1). Moreover, Nash equilibria are not exchangeable and they do not have a unique value [11].

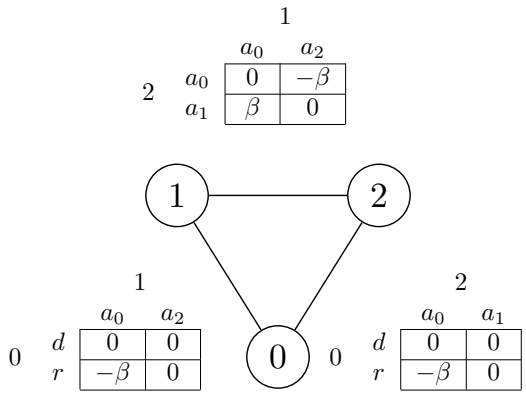


Figure 3.1: Offense-Defense, a simple CSP game. We only show payoffs for the row player, the column player's payoffs are zero minus the row player's payoffs.

Polymatrix games that are constant sum in each subgame are no more or less general than polymatrix games that are constant sum globally, since there exists a payoff preserving transformation between the two sets [11]. For this reason we focus on polymatrix games that are constant sum in each subgame without loss of generality. Note that the constant need not be the same in each subgame.

3.2.1 Vulnerability on a Simple Polymatrix Game

We next demonstrate why constant-sum polymatrix games do not have bounded vulnerability on their own without additional properties. Consider the simple 3-player constant-sum polyamtrix game called Offense-Defense (Figure 3.1). There are 3 players: 0, 1 and 2. Players 1 and 2 have the option to either attack 0 (a_0) or attack the other (e.g. a_1); player 0, on the other hand, may either relax (r) or defend (d). If either 1 or 2 attacks the other while the other is attacking 0, the attacker gets β and the other gets $-\beta$ in that subgame. If both 1 and 2 attack 0, 1 and 2 get 0 in their subgame and if they attack each other, their attacks cancel out and they both get 0. If 0 plays d, they defend and will always get 0. If they relax, they get $-\beta$ if they are attacked and 0 otherwise. Offense-Defense is a CSP game, so any CCE is a Nash equilibrium.

Note that $\rho = (r, a_2, a_1)$ is a Nash equilibrium. Each $i \in \{1, 2\}$ are attacking the other $j \in \{1, 2\} \backslash \{i\}$, so has expected utility of 0. Deviating to attacking 0 would leave them open against the other, so a_0 is not a profitable deviation, as it would also give utility 0. Additionally, 0 has no incentive to deviate to d, since this would also give them a utility of 0.

However, ρ is not a Nash equilibrium of the subgames—all $i \in \{1, 2\}$ have a profitable deviation in their subgame against 0, which leaves 0 vulnerable in that subgame. If 1 and 2 were to both deviate to a_0, and 0 continues to play their Nash equilibrium strategy of r, 0 would lose 2β utility from their equilibrium value; in other words, the vulnerability of player 0 is 2β.

3.3 Subgame Stability

However, some polymatrix games *do* have the aforementioned desirable properties of two-player constant-sum games; we call these *subgame stable games*. In subgame stable games, global equilibria imply equilibria at each pairwise subgame.

Definition 3.3.1 (Subgame stable profile)**.** Let G be a polymatrix game with global utility functions $(u_i)_{i \in N}$. We say a strategy profile s is γ-*subgame stable* if and only if $\forall (i, j) \in E$, we have (s_i, s_j) is a γ-Nash of G_{ij}; that is

$$u_{ij}(\rho_i, s_j) - u_{ij}(s_i, s_j) \leqslant \gamma \quad \forall \rho_i \in \mathrm{P}_i,$$
$$u_{ji}(\rho_j, s_i) - u_{ji}(s_j, s_i) \leqslant \gamma \quad \forall \rho_j \in \mathrm{P}_j.$$

For example, in Offense-Defense, (r, a_2, a_1) is β-subgame stable; it is a Nash equilibrium but is a β-Nash of the subgame between 0 and 1 and the subgame between 0 and 2.

Definition 3.3.2 (Subgame stable game)**.** Let G be a polymatrix game. We say G is (ϵ, γ)-*subgame stable* if for *any* ϵ-Nash equilibrium s of G, s is γ-subgame stable.

For example, Offense-Defense is $(0, \beta)$-subgame stable.

Subgame stability connects the global behaviour of play (ϵ-Nash equilibrium in G) to local behaviour in a subgame (γ-Nash in G_{ij}). If a polymatrix game is both constant-sum and is $(0, \gamma)$-subgame stable then we can bound the vulnerability of any marginal strategy.

Theorem 3.3.3. *Let G be a CSP game. If G is $(0, \gamma)$-subgame stable, then the following hold for any player $i \in N$:*

1. *For any CCE μ of G, we have* $\mathrm{Vul}_i\left(s^\mu, S_{-i}\right) \leqslant |E_i|\gamma$.

2. $\mathrm{Ex}(S^\mu) \leqslant |E_i|\gamma$.

Proof. First we show 1. Any marginal strategy s^μ of a CCE μ is a Nash equilibrium of G [11]. Then,

$$\begin{aligned}
\mathrm{Vul}_i\left(s^\mu, S_{-i}\right) &\doteq u_i(s^\mu) - \min_{s_{-i} \in S_{-i}} u_i(s_i^\mu, s_{-i}) \\
&= \sum_{(i,j)\in E_i} u_{ij}(s_i^\mu, s_j^\mu) - \min_{s_{-i}\in S_{-i}} \left(\sum_{(i,j)\in E_i} u_{ij}(s_i^\mu, s_j) \right) \\
&= \sum_{(i,j)\in E_i} u_{ij}(s_i^\mu, s_j^\mu) - \sum_{(i,j)\in E_i} \min_{s_j \in S_j} u_{ij}(s_i^\mu, s_j).
\end{aligned}$$

Where the last line uses the fact that $-i$ minimize $i's$ utility, so can do so without coordinating since G is polymatrix. Continuing,

$$\begin{aligned}
&= \sum_{(i,j)\in E_i} \left(u_{ij}(s_i^\mu, s_j^\mu) - \min_{s_j \in S_j} u_i(s_i^\mu, s_j) \right) \\
&\leqslant \sum_{(i,j)\in E_i} \gamma \\
&\leqslant |E_i|\gamma,
\end{aligned}$$

where by $(0, \gamma)$-subgame stability of each G_{ij}, (s_i^μ, s_i^μ) is a γ-Nash of G_{ij}. By Proposition 2.1.10, we have $u_{ij}(s_i^\mu, s_j^\mu) - \min_{s_j \in S_j} u_i(s_i^\mu, s_j) \leqslant \gamma$.

Next we show 2. Let s be a strategy profile such that $\forall i \in N$, s_i is the marginal strategy from some CCE μ^i. Then for any $i \in N$,

$$\begin{aligned}
\max_{s_i^* \in S_i} u_i(s_i^*, s_{-i}) - u_i(s_i, s_{-i}) &= \max_{s_i^*} \sum_{(i,j)\in E_i} \left(u_{ij}(s_i^*, s_j) - u_{ij}(s_i, s_j)\right) \\
&\leqslant \sum_{(i,j)\in E_i} \max_{s_i^*} u_{ij}(s_i^*, s_j) - u_{ij}(s_i, s_j).
\end{aligned}$$

Since G is $(0, \gamma)$-subgame stable, s_i and s_j are γ-Nash-equilibrium strategies of each subgame G_{ij}, since all s^μ are Nash equilibria. Note, however, they may not have come from the same equilibrium. By Proposition 2.1.13, (s_i, s_j) is a γ-Nash of G_{ij}. Thus,

$$\sum_{(i,j)\in E_i} \max_{s_i^*} u_{ij}(s_i^*, s_j) - u_{ij}(s_i, s_j) \leqslant \sum_{(i,j)\in E_i} \gamma = |E_i|\gamma.$$

□

Theorem 3.3.3 tells us that using self-play to compute a marginal strategy s^μ on constant-sum polymatrix games will have low vulnerability against worst-case opponents if γ is low. Thus, these are a set of multi-player games where self-play is an effective training procedure. Moreover, if a player were to play against agents who computed their strategies in a separate self-play training instance, they are playing an approximate best response.

In 2 player constant-sum games, all Nash equilibria give a player the same about of utility, that players *value*. An approximate version of this result holds for subgame stable games, where the utility of all equilibria lie in a neighbourhood defined by γ.

Proposition 3.3.4. *Let G be a $(0, \gamma)$-subgame stable CSP game. Let $v_i \doteq \sum_{(i,j)\in E_i} v_{ij}$ where v_{ij} is the value of G_{ij} for player i. Then for any Nash equilibrium s, $|v_i - u_i(s)| \leqslant |E_i|\gamma$.*

Proof.

$$|v_i - u_i(s)| = \left| \sum_{(i,j)\in E_i} v_{ij} - \sum_{(i,j)\in E_i} u_{ij}(s_i, s_j) \right| \tag{3.6}$$

$$= \left| \sum_{(i,j)\in E_i} v_{ij} - u_{ij}(s_i, s_j) \right| \tag{3.7}$$

$$\leqslant \sum_{(i,j)\in E_i} |v_{ij} - u_{ij}(s_i, s_j)| \tag{3.8}$$

By subgame stability, we have s_i, s_j is a γ-Nash in G_{ij}. In each G_{ij}, if $v_{ij} \geqslant u_{ij}(s_i, s_j)$ then $|v_{ij} - u_{ij}(s_i, s_j)| = v_{ij} - u_{ij}(s_i, s_j) \leqslant \gamma$ by Proposition 2.1.11.

Otherwise, let c_{ij} be the constant for G_{ij}. If $v_{ij} < u_{ij}(s_i, s_j)$, we have

$$\begin{aligned} 0 &= c_{ij} - c_{ij} \\ &= v_{ij} + v_{ji} - (u_{ij}(s_i, s_j) + u_{ji}(s_i, s_j)) \\ &= \underbrace{(v_{ij} - u_{ij}(s_i, s_j))}_{(a)} + \underbrace{(v_{ji} - u_{ji}(s_i, s_j))}_{(b)}. \end{aligned}$$

But $(a) < 0$, so $(b) > 0$. (b) is upper bounded by γ by Proposition 2.1.11. This implies $|v_{ij} - u_{ij}(s_i, s_j)| \leqslant \gamma$. This means 3.8 is upper bounded by

$$\leqslant \sum_{(i,j) \in E_i} \gamma = |E_i|\gamma.$$

□

3.3.1 Computing Subgame Stability

Let γ be the minimum γ such that G is $(0, \gamma)$-subgame stable. How do we compute γ? Does it involve computing all equilibria of G and checking their subgame stability? The answer is no, it can be done in polynomial time in the number of pure strategies. We next provide an algorithm for computing γ. The algorithm involves solving a linear program for each edge in the graph and each pure strategy of those players. This linear program takes a pure strategy ρ_i', and finds a Nash equilibrium of G that maximizes i's incentive to deviate to ρ_i' when *only* considering their utility in G_{ij}; call this quantity $\gamma_{ij}^{\rho_i'}$. If there are no such Nash equilibria the solver returns "infeasible". If the solver does not return "infeasible", we update $\gamma = \max_{(i,j) \in E_i} \max_{\rho_i' \in \mathrm{P}_i} \gamma_{ij}^{\rho_i'}$.

Let $a_i(\rho_i', \mu)$ be the advantage of deviating to ρ_i' from a joint distribution over pure strategies:

$$a_i(\rho_i', \mu) \doteq \sum_{(i,j) \in E_i} \underbrace{u_{ij}(\rho_i', s_j^\mu)}_{(a)} - \underbrace{\mathbb{E}_{\rho \sim \mu} \left[\sum_{(i,j) \in E_i} u_{ij}(\rho_i, \rho_j) \right]}_{(b)}$$

Note that (a) is a linear function of μ, since s_j^μ is a marginal strategy. (b) is also a linear function of μ, and so $a_i(\rho_i', \mu)$ is a linear function of μ. Likewise,

Algorithm 1 Compute γ

Input: $G = (N, E, \mathrm{P}, u)$, a polymatrix game
$\gamma \leftarrow -\infty$
for $(i, j) \in E$ **do**
  for $\rho_i' \in \mathrm{P}_i$ **do**
    if $\mathrm{LP1}(i, j, \rho_i')$ not infeasible **then**
      $\gamma_{ij}^{\rho_i'} \leftarrow \mathrm{LP1}(i, j, \rho_i')$
      $\gamma \leftarrow \max(\gamma, \gamma_{ij}^{\rho_i'})$
    end if
  end for
  for $\rho_j' \in \mathrm{P}_j$ **do**
    if $\mathrm{LP1}(j, i, \rho_j')$ not infeasible **then**
      $\gamma_{ji}^{\rho_j'} \leftarrow \mathrm{LP1}(j, i, \rho_j')$
      $\gamma \leftarrow \max(\gamma, \gamma_{ji}^{\rho_j'})$
    end if
  end for
end for
return γ

let

$$a_{ij}(\rho_i', \mu) \doteq u_{ij}(\rho_i', s_j^{\mu}) - \mathbb{E}_{\rho \sim \mu}\left[u_{ij}(\rho_i, \rho_j)\right].$$

be the advantage of ρ_i' in the subgame between i and j. $\mathrm{LP1}(i, j, \rho_i')$ is given below. The decision variables are the weights of μ for each $\rho \in \mathrm{P}$ and $\gamma_{ij}^{\rho_i'}$.

LP 1

$$\begin{aligned} \max \quad & \gamma_{ij}^{\rho_i'} \\ \text{s.t.} \quad & a_i(\rho_i, \mu) \leqslant 0 \quad \forall i \in N, \rho_i \in \mathrm{P}_i \\ & a_{ij}(\rho_i', \mu) \geqslant \gamma_{ij}^{\rho_i'} \\ & \sum_{\rho \in \mathrm{P}} \mu(\rho) = 1 \\ & \mu(\rho) \in [0, 1] \quad \forall \rho \in \mathrm{P} \end{aligned}$$

We can get away with computing a CCE rather than targeting Nash equilibria because the marginals of any CCE are Nash equilibria in CSP games [11].

The whole procedure runs in polynomial time in the size of the game.

We need to solve an LP, which takes polynomial time, at most $n^2 \max_{i \in N} |\mathrm{P}_i|$ times.

3.4 Approximate Polymatrix Games

Most games are *not* factorizable into polymatrix games. Fortunately, we can take any game G and project it into the space of CSP games. Games that are near to the space of CSP games have relaxed versions of the desirable properties of CSP games, where the degree of relaxation depends on this distance.

Definition 3.4.1 (δ-constant sum polymatrix)**.** A game G is *δ-constant sum polymatrix* (δ-CSP) if there exists a CSP game $\check{G}$ with global utility function $\check{u}$ such that $\forall i \in N, \rho \in \mathrm{P}$, $|u_i(\rho) - \check{u}_i(\rho)| \leqslant \delta$. We denote the set of such CSP games as $\mathrm{CSP}_\delta(G)$.

Throughout this thesis, we use the symbol $\check{}$ ("check") for CSP games that are the approximate decomposition of some game; i.e. $\check{G}$ is an approximate CSP decomposition of G.

Proposition 3.4.2. *In a δ-CSP game G the following hold.*

1. *Any CCE of G is a 2δ-CCE of any $\check{G} \in \mathrm{CSP}_\delta(G)$.*
2. *The marginal strategy profile of any CCE of G is a $2n\delta$-Nash equilibrium of any $\check{G} \in \mathrm{CSP}_\delta(G)$.*
3. *The marginal strategy profile of any CCE of G is a $2(n+1)\delta$-Nash equilibrium of G.*

From (3) we have that the removal of the mediator impacts players utilities by a bounded amount in δ-CSP games. The proof is given in Section 3.6.1.

Projecting a game into the space of CSP games with minimum δ can be done with a linear program. Let $\underline{\delta}$ be the minimum δ such that G is δ-CSP. We give a linear program that finds $\underline{\delta}$ and returns a CSP game $\check{G} \in \mathrm{CSP}_{\underline{\delta}}(G)$. The decision variables are the values of $\check{u}_{ij}(\rho)$ for all $i \neq j \in N, \rho \in \mathrm{P}$, $\underline{\delta}$ and constants for each subgame c_{ij}, for all $i \neq j$.

LP 2

$$\begin{aligned}
\min \quad & \underline{\delta} \\
\text{s.t.} \quad & u_i(\rho) - \sum_{j \in -i} \check{u}_{ij}(\rho_i, \rho_j) \leqslant \underline{\delta} \quad \forall i \in N, \rho \in \mathrm{P} \\
& u_i(\rho) - \sum_{j \in -i} \check{u}_{ij}(\rho_i, \rho_j) \geqslant -\underline{\delta} \quad \forall i \in N, \rho \in \mathrm{P} \\
& \check{u}_{ij}(\rho_i, \rho_j) + \check{u}_{ji}(\rho_i, \rho_j) = c_{ij} \quad \forall i \neq j \in N, (\rho_i, \rho_j) \in \mathrm{P}_{ij},
\end{aligned}$$

Combining δ-CSP with (ϵ, γ)-subgame stability lets us bound vulnerability and exchangeability in *any* game.

Theorem 3.4.3. *If G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable and μ is a CCE of G, then*

$$\mathrm{Vul}_i\left(s^\mu, S_{-i}\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta,$$

and

$$\mathrm{Ex}(S^\mu) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

We give the proof in Section 3.6.

Theorem 3.4.3 shows that games that are close to the space of subgame stable CSP (SS-CSP) games are cases where the marginal strategies learned through self-play have bounded worst-case performance. This makes these games suitable for any no-external-regret learning algorithm. The exchangeability of S^μ is also bounded, which means i will boundedly regret their marginal strategy s_i^μ when faced with agents who also learned in self-play.

We can also show that all utility of any marginal strategy profile falls within a bounded range. In contrast to Proposition 3.3.4, we have $\check{G}$ is $(2n\delta, \gamma)$-subgame stable.

Proposition 3.4.4. *If G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable, let $\check{v}_i \doteq \sum_{(i,j) \in E_i} \check{v}_{ij}$ where $\check{v}_{ij}$ is the value of $\check{G}_{ij}$ for player i. Then for any marginal strategy of a CCE of G, s^μ, we have $|\check{v}_i - u_i(s^\mu)| \leqslant |E_i|\gamma + \delta$.*

The proof is given in Section 3.6.3.

3.5 Vulnerability Against Other Self-Taught Agents

Theorem 3.4.3 bounds vulnerability in the worst-case scenarios, where $-i$ play any strategy profile to minimize i's utility. In reality, however, each player $j \in -i$ has their own interests and would only play a strategy that is reasonable under these own interests. In particular, what if each agent were also determining their own strategy via self-play in a separate training instance. How much utility can i guarantee themselves in this setup?

While no-external-regret learning algorithms converge to the set of CCE, other assumptions can be made with additional information about the type of regret being minimized. For example, no-external-regret learning algorithms will play strictly dominated strategies with vanishing probability and CFR will play dominated actions with vanishing probability [22]. These refinements can tighten our bounds, since the part of the game that no-regret learning algorithms converge to might be closer to a CSP game than the game overall.

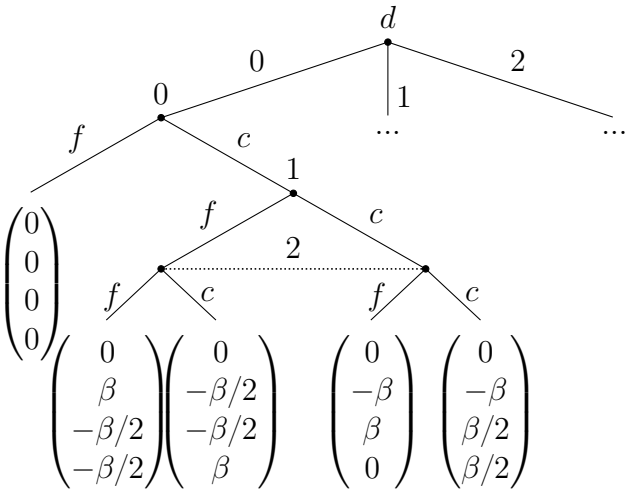


Figure 3.2: Bad Card: a game that is not overall polymatrix, but the subset of strategies learnable by self-play are. At the terminals, we show the dealers utility first, followed by players $0, 1$ and 2, respectively.

Consider the game shown in Figure 3.2, called "Bad Card". The game starts with each player except the dealer putting $\beta/2$ into the pot. A dealer player d—who receives utility 0 regardless of the strategies of the other players—

then selects a player from $\{0, 1, 2\}$ to receive a "bad card", while the other two players receive a "good card". The player who receives the bad card has an option to fold, after which the game ends and all players receive their ante back. Otherwise if this player calls, the other two players can either fold or call. The pot of β is divided among the players with good cards who call. If one player with a good card calls, they win the pot of β. If both good card players call then they split the pot. If both players with good cards fold, then the player with the bad card wins the pot.

As we shall soon show, Bad Card does not have a constant-sum polymatrix decomposition—in fact it does not have *any* polymatrix decomposition. Since Bad Card is an extensive-form game without chance, each pure strategy profile leads to a single terminal history. Let $\mathrm{P}(z)$ be the set of pure strategy profiles that play to a terminal z. In order for Bad Card to be polymatrix, we would need to find subgame utility functions such that $\forall \rho \in \mathrm{P}$, $u_0(\rho) = u_{0,d}(\rho_0, \rho_d) + u_{0,1}(\rho_0, \rho_1) + u_{0,2}(\rho_0, \rho_2)$. Equivalently, we could write $\forall z \in Z, \rho \in \mathrm{P}(z), u_0(z) = u_{0,d}(\rho_0, \rho_d) + u_{0,1}(\rho_0, \rho_1) + u_{0,2}(\rho_0, \rho_2)$ where Z is the set of terminals. A subset of these constraints results in an infeasible system of equations.

Consider the terminals in the subtree shown in Figure 3.2: $z^1 = (0, c, c, c)$, $z^2 = (0, c, c, f)$, $z^3 = (0, c, f, c)$ and $z^4 = (0, c, f, f)$. Let ρ_i^c be any pure strategy that plays c in this subtree and ρ_i^f be any strategy that plays f in this subtree for player i. In order for Bad Card to decompose into a polymatrix game we would need to solve the following infeasible system of linear equations:

$$
\begin{aligned}
u_0(z^1) &= u_{0,d}(\rho_0^c, 0) + u_{0,1}(\rho_0^c, \rho_1^c) + u_{0,2}(\rho_0^c, \rho_2^c) = -\beta \\
u_0(z^2) &= u_{0,d}(\rho_0^c, 0) + u_{0,1}(\rho_0^c, \rho_1^c) + u_{0,2}(\rho_0^c, \rho_1^f) = -\beta \\
u_0(z^3) &= u_{0,d}(\rho_0^c, 0) + u_{0,1}(\rho_0^c, \rho_1^f) + u_{0,2}(\rho_0^c, \rho_2^c) = -\beta \\
u_0(z^4) &= u_{0,d}(\rho_0^c, 0) + u_{0,1}(\rho_0^c, \rho_1^f) + u_{0,2}(\rho_0^c, \rho_2^f) = \beta
\end{aligned}
$$

Thus, Bad Card is not a constant-sum polymatrix game, although it is a β-CSP game. However, if we prune out dominated actions (i.e., those in which a player folds after receiving a good card), the resulting game is indeed a 0-CSP game.

Let $\mathcal{M}(\mathcal{A})$ be the set of mediated equilibria than an algorithm $\mathcal{A}$ converges to in self-play. For example, if $\mathcal{A}$ is a no-external-regret algorithm, $\mathcal{M}(\mathcal{A})$ is the set of CCE without strictly dominated strategies in their support. Let $S(\mathcal{A}_i) = \{s^\mu \mid (\mu, (\Phi_i)_{i\in N}) \in \mathcal{M}(\mathcal{A}_i)\}$ be the set of marginal strategy profiles of $\mathcal{M}(\mathcal{A}_i)$, and let $S_i(\mathcal{A}_i) = \{s_i \mid s \in S(\mathcal{A}_i)\}$ be the set of i's marginal strategies from $S(\mathcal{A}_i)$.

Now, consider if each player i learns with their own self-play algorithm $\mathcal{A}_i$. Let $\mathcal{A}_N = (\mathcal{A}_1, ...\mathcal{A}_n)$ be the profile of learning algorithms, then $S^\times(\mathcal{A}_N) = \times_{i\in N} S_i(\mathcal{A}_i)$. Summarizing, if each player learns with a no-Φ_i-regret learning algorithm $\mathcal{A}_i$, they will converge to the set of $\mathcal{M}(\mathcal{A}_i)$ equilibria. The set of marginal strategies from this set of equilibria is $S_i(\mathcal{A}_i)$ and the set of marginal strategy profiles is $S(\mathcal{A}_i)$. If each player plays a (potentially) different learning algorithm, $S^\times(\mathcal{A}_N)$ is the set of possible joint match-ups if each player plays a marginal strategy from their own algorithm's set of equilibria.

Definition 3.5.1. We say a game G is *δ-CSP in the neighbourhood of $S' \subseteq S$* if there exists a CSP game $\check{G}$ such that $\forall s \in S'$ we have $|u_i(s) - \check{u}_i(s)| \leqslant \delta$. We denote the set of such CSP games as $\mathrm{CSP}_\delta(G, S')$.

Definition 3.5.2. We say a CSP game G is *γ-subgame stable in the neighbourhood of S'* if $\forall s \in S', \forall (i,j) \in E$ we have that (s_i, s_j) is a γ-Nash of G_{ij}.

These definitions allow us to prove the following generalization of Theorem 3.4.3. The proof is given in Section 3.6.4.

Theorem 3.5.3. *For any $i \in N$, if G is δ-CSP in the neighbourhood of $S^\times(\mathcal{A}_N)$ and $\exists \check{G} \in \mathrm{CSP}_\delta(G, S^\times(\mathcal{A}_N))$ that is γ-subgame stable in the the neighbourhood of $S(\mathcal{A}_i)$, then for any $s \in S(\mathcal{A}_i)$*

$$\mathrm{Vul}_i\left(s, S^\times_{-i}(\mathcal{A}_N)\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

An implication of Theorem 3.5.3 is that if agents use self-play to compute a marginal strategy from some mediated equilibrium and there is a subgame stable CSP game that is close to the original game for these strategies, then this is sufficient to bound vulnerability against strategies learned in self-play.

Given a finite set $S(\mathcal{A}_i)\forall i \in N$, we can compute a CSP game $\check{G}$ that is δ-CSP in the neighbourhood of $S^{\times}(\mathcal{A}_N)$ and γ-subgame stable in the the neighbourhood of $S(\mathcal{A}_i)\ \forall i \in N$ so that our bounds are minimized using the following LP.

LP 3 The decision variables are the values of $\check{u}_{ij}(\rho)$ for all $i \neq j \in N, \rho \in \mathrm{P}$, δ, γ, and constants for each subgame c_{ij} for all $i \neq j$:

$$
\begin{aligned}
\min \quad & (n-1)\gamma + 2\delta \\
\text{s.t.} \quad & \check{u}_{ij}(\rho_i, s_j) - u_{ij}(s_i, s_j) \leqslant \gamma \quad \forall i \in N, \rho_i \in \mathrm{P}_i, s \in S(\mathcal{A}_i) \\
& u_i(s) - \sum_{j \in -i} \check{u}_{ij}(s_i, s_j) \leqslant \delta \quad \forall i \in N, s \in S^{\times}(\mathcal{A}_N) \\
& u_i(s) - \sum_{j \in -i} \check{u}_{ij}(s_i, s_j) \geqslant -\delta \quad \forall i \in N, s \in S^{\times}(\mathcal{A}_N) \\
& \check{u}_{ij}(\rho_i, \rho_j) + \check{u}_{ji}(\rho_i, \rho_j) = c_{ij} \quad \forall i \neq j \in N, (\rho_i, \rho_j) \in \mathrm{P}_{ij} \\
& \delta \geqslant 0 \\
& \gamma \geqslant 0
\end{aligned}
$$

3.6 Omitted Proofs

Here we give omitted proofs for this chapter.

3.6.1 Proof of Proposition 3.4.2

Proposition 3.4.2. *In a δ-CSP game G the following hold*

1. *Any CCE of G is a 2δ-CCE of any $\check{G} \in \mathrm{CSP}_{\delta}(G)$.*
2. *The marginalized strategy profile of any CCE of G is a $2n\delta$-Nash equilibrium of any $\check{G} \in \mathrm{CSP}_{\delta}(G)$.*
3. *The marginalized strategy profile of any CCE is a $2(n+1)\delta$-Nash equilibrium of G*

Proof. First we prove claim 1. Let $\check{u}_i$ denote the utility function of i in $\check{G}$. Note that $\forall \rho \in \mathrm{P}$ we have $|\check{u}_i(\rho) - u_i(\rho)| \leqslant \delta\ \forall i \in N$. Let μ be any CCE of G.

The definition of CCE states

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i}) - u_i(\rho)\right] \leqslant 0 \quad \forall i \in N, \rho_i' \in \mathrm{P}_i.$$

It is sufficient to consider only player i. We can preserve the inequality by substituting $\check{u}_i(\rho_i', \rho_{-i}) - \delta$ in place of $u_i(\rho_i', \rho_{-i})$ and $\check{u}_i(\rho) + \delta$ in place of $u_i(\rho)$. This gives us

$$\begin{aligned} &\mathbb{E}_{\rho\sim\mu}\left[\check{u}_i(\rho_i', \rho_{-i}) - \delta - (\check{u}_i(\rho) + \delta)\right] \leqslant 0 \quad \forall \rho_i' \in \mathrm{P}_i \\ \Longrightarrow\ &\mathbb{E}_{\rho\sim\mu}\left[\check{u}_i(\rho_i', \rho_{-i}) - \check{u}_i(\rho)\right] \leqslant 2\delta \quad \forall \rho_i' \in \mathrm{P}_i. \end{aligned}$$

Thus claim 1 is shown. Claim 2 is an immediate corollary of claim 1 and Proposition 3.2.4. Lastly, we show claim 3. By claim 2, we have the marginalized strategy of μ, s^μ, is a $2n\delta$-Nash equilibrium of $\check{G} \in \mathrm{CSP}_\delta(G)$. That is for any $i \in N$,

$$\check{u}_i(\rho_i', s_{-i}^\mu) - \check{u}_i(s_i^\mu, s_{-i}^\mu) \leqslant 2n\delta \quad \forall \rho_i' \in \mathrm{P}_i.$$

However, since G is δ-CSP, we may substitute $u_i(\rho_i', s_{-i}^\mu) - \delta$ in place of $\check{u}_i(\rho_i', s_{-i}^\mu)$ and $u_i(s_i^\mu, s_{-i}^\mu) + \delta$ in place of $\check{u}_i(s_i^\mu, s_{-i}^\mu)$ as preserve the inequality.

$$\left(u_i(\rho_i', s_{-i}^\mu) - \delta\right) - \left(u_i(s_i^\mu, s_{-i}^\mu) + \delta)\right) \leqslant 2n\delta \quad \forall \rho_i' \in \mathrm{P}_i.$$

Rearranging, this gives us

$$u_i(\rho_i', s_{-i}^\mu) - u_i(s_i^\mu, s_{-i}^\mu) \leqslant 2n\delta + 2\delta = 2(n+1)\delta \quad \forall \rho_i' \in \mathrm{P}_i.$$

□

3.6.2 Proof of Theorem 3.4.3

Theorem 3.4.3. *If G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable and μ is a CCE of G, then*

$$\mathrm{Vul}_i\left(s^\mu, S_{-i}\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta,$$

and

$$\mathrm{Ex}(S^\mu) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

The proof is largely the same as Theorem 3.3.3, with added approximation since G is no longer CSP.

Proof. Let $\check{G}$ be a polymatrix game that is $(2n\delta, \gamma)$-subgame stable such that $\check{G} \in \mathrm{CSP}_\delta(G)$. Let $\check{u}_i$ denote the utility function of i in $\check{G}$. By Proposition 3.4.2, μ is a $2n\delta$-Nash equilibrium of $\check{G}$. Then,

$$\begin{aligned}\mathrm{Vul}_i\left(s^\mu, S_{-i}\right) &\doteq u_i(s^\mu) - \min_{s'_{-i} \in S_{-i}} u_i(s_i^\mu, s'_{-i}) \\ &\leqslant \check{u}_i(s^\mu) - \min_{s'_{-i} \in S_{-i}} \check{u}_i(s_i^\mu, s'_{-i}) + 2\delta,\end{aligned}$$

since G is δ-CSP. Then expanding $\check{u}_i$ across i's subgames we have

$$\begin{aligned}&\sum_{(i,j)\in E_i} \check{u}_{ij}(s_i^\mu, s_j^\mu) - \min_{s'_{-i} \in S_{-i}} \sum_{(i,j)\in E_i} \check{u}_i(s_i^\mu, s_j) + 2\delta \\ =& \sum_{(i,j)\in E_i} \check{u}_{ij}(s_i^\mu, s_j^\mu) - \sum_{(i,j)\in E_i} \min_{s'_j \in S_j} \check{u}_i(s_i^\mu, s'_j) + 2\delta.\end{aligned}$$

Where, as in Theorem 3.3.3, the last line uses the fact that $\check{G}$ is polymatrix, G_{ij} is constant-sum and $-i$ minimize $i's$ utility and can do so by without coordinating. Continuing, we have

$$\begin{aligned}&\sum_{(i,j)\in E_i} \check{u}_{ij}(s_i^\mu, s_j^\mu) - \sum_{(i,j)\in E_i} \min_{s'_j \in S_j} \check{u}_i(s_i^\mu, s'_j) + 2\delta \\ &= \sum_{(i,j)\in E_i} \left(\check{u}_{ij}(s_i^\mu, s_j^\mu) - \min_{s'_j \in S_j} \check{u}_i(s_i^\mu, s'_j) \right) + 2\delta \\ &\leqslant \sum_{(i,j)\in E_i} \gamma + 2\delta \\ &= |E_i|\gamma + 2\delta \\ &\leqslant (n-1)\gamma + 2\delta.\end{aligned}$$

Where by $(2n\delta, \gamma)$-subgame stability of each G_{ij}, (s_i^μ, s_i^μ) is a γ-Nash of G_{ij}. By Proposition 2.1.10, s_i^μ can lose at most γ to a worst case opponent s'_j in each subgame, since $\check{G}_{ij}$ is two-player constant-sum.

Next we show

$$\mathrm{Ex}(S^\mu) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

Let s be a strategy profile such that $\forall i \in N$, s_i is the marginal strategy from some CCE μ^i. We bound the incentive that any player has to deviate from s in G, given by

$$u_i(\rho_i', s_{-i}) - u_i(s_i, s_{-i}) \quad \forall \rho_i' \in \mathrm{P}_i.$$

Then we have

$$\begin{aligned} u_i(\rho_i', s_{-i}) - u_i(s_i, s_{-i}) &\leqslant \check{u}_i(\rho_i', s_{-i}) - \check{u}_i(s_i, s_{-i}) + 2\delta \quad \forall \rho_i' \in \mathrm{P}_i && (3.9) \\ &\leqslant \sum_{(i,j)\in E_i} (\check{u}_{ij}(\rho_i', s_j) - \check{u}_{ij}(s_i, s_j)) + 2\delta \quad \forall \rho_i' \in \mathrm{P}_i. && (3.10) \end{aligned}$$

For any ρ_i', (3.10) is less than

$$\leqslant \sum_{(i,j)\in E_i} \left(\max_{\rho_i^* \in \mathrm{P}_i} \check{u}_{ij}(\rho_i^*, s_j) - \check{u}_{ij}(s_i, s_j) \right) + 2\delta. \tag{3.11}$$

By $(2n\delta, \gamma)$-subgame stability, s_i and s_j are γ-Nash-equilibrium strategies of $\check{G}_{ij}$. By Proposition 2.1.13 (s_i, s_j) is also a γ-Nash of $\check{G}_{ij}$. Thus,

$$(3.11) \leqslant \sum_{(i,j)\in E_i} \gamma + 2\delta = |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

□

3.6.3 Proof of Proposition 3.4.4

Proposition 3.4.4. *If G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable, let $\check{v}_i \doteq \sum_{(i,j)\in E_i} \check{v}_{ij}$ where $\check{v}_{ij}$ is the value of $\check{G}_{ij}$ for player i. Then for any marginal strategy of a CCE of G, s^μ, we have $|\check{v}_i - u_i(s^\mu)| \leqslant |E_i|\gamma + \delta$.*

Proof. The proof is largely the same as Proposition 3.3.4. By δ-CSP we have

$$|\check{v}_i - u_i(s^\mu)| \leqslant \underbrace{\left| \sum_{(i,j)\in E_i} \check{v}_{ij} - \check{u}_{ij}(s_i^\mu, s_j^\mu) \right|}_{(a)} + \delta.$$

By $(2n\delta, \gamma)$-subgame stability, each (s_i^μ, s_j^μ) is a γ-Nash equilibrium of $\check{G}_{ij}$. From here, we can upper bound (a) by following the steps of Proposition 3.3.4. This gives us

$$|\check{v}_i - u_i(s^\mu)| \leqslant |E_i|\gamma + \delta.$$

□

3.6.4 Proof of Theorem 3.5.3

Theorem 3.5.3. *If G is δ-CSP in the neighbourhood of $S^{\times}(\mathcal{A}_N)$ and $\exists \check{G} \in \mathrm{CSP}_{\delta}(G, S^{\times}(\mathcal{A}_N))$ that is γ-subgame stable in the the neighbourhood of $S(\mathcal{A}_i)$, then for any $s \in S(\mathcal{A}_i)$*

$$\mathrm{Vul}_i\left(s, S^{\times}_{-i}(\mathcal{A}_N)\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

Proof. The proof is very similar to Theorem 3.4.3. Writing the definition of vulnerability we have

$$\mathrm{Vul}_i\left(s, S(\mathcal{A})\right) \doteq u_i(s) - \min_{s'_{-i} \in S^{\times}_{-i}(\mathcal{A}_N)} u_i(s, s'_{-i}), \tag{3.12}$$

since G is δ-CSP in the neighbourhood of $S^{\times}(\mathcal{A}_N)$. Swapping out the utility of u_i for $\check{u}$, we have

$$(3.12) \leqslant \check{u}_i(s) - \min_{s'_{-i} \in S^{\times}_{-i}(\mathcal{A}_N)} \check{u}_i(s_i, s'_{-i}) + 2\delta$$

Since $\check{G}$ is a polymatrix game,

$$\check{u}_i(s) - \min_{s'_{-i} \in S^{\times}_{-i}(\mathcal{A}_N)} \check{u}_i(s_i, s'_{-i}) + 2\delta \tag{3.13}$$

$$= \sum_{(i,j) \in E_i} \check{u}_{ij}(s_i, s_j) - \min_{s'_{-i} \in S^{\times}_{-i}(\mathcal{A}_N)} \sum_{(i,j) \in E_i} \check{u}_{ij}(s_i, s_j) + 2\delta \tag{3.14}$$

$$= \left(\sum_{(i,j) \in E_i} \check{u}_{ij}(s_i, s_j) - \min_{s'_j \in S_j(\mathcal{A}_j)} \check{u}_{ij}(s_i, s'_j) \right) + 2\delta. \tag{3.15}$$

Where, as in Theorem 3.3.3 and Theorem 3.4.3, the last line uses the fact that $\check{G}$ is polymatrix, G_{ij} is constant-sum and $-i$ minimize $i's$ utility and can do so by without coordinating.

Since $\check{G}$ is γ-subgame stable in the neighbourhood of $S(\mathcal{A}_i)$ and $s \in S(\mathcal{A}_i)$, then means (s_i, s_j) is a γ-Nash for each subgame $\check{G}_{ij}$, so has bounded vulnerability within that subgame.

$$\begin{aligned}(3.15) \leqslant& \left(\sum_{(i,j) \in E_i} \gamma \right) + 2\delta \\ \leqslant& |E_i|\gamma + 2\delta \\ \leqslant& (n-1)\gamma + 2\delta\end{aligned}$$

□

Chapter 4

Guarantees for Self-Play in Extensive-Form Games

The previous chapter developed notions of approximately CSP and subgame stable in the context of normal-form games. In this chapter, we apply these concepts to extensive-form games. While any extensive-form game has an equivalent induced normal-form game, analysing properties of an EFG through its induced normal is intractable for moderately-sized EFGs, since the size the normal-form representation is exponentially larger.

We begin by introducing a novel "extensive-form version" of normal-form polymatrix games, which we call *poly-EFGs*. The major benefit of poly-EFGs over normal-form polymatrix games is their efficiently: poly-EFGs are exponentially more compact than an equivalent normal-form polymatrix game. The results of this chapter extend the theory of the previous chapter using this more efficient representation.

We give a proof-of-concept showing that poly-EFGs can be used to efficiently decompose extensive-form games by giving an algorithm for decomposing a perfect information EFG into a poly-EFG. Finally, we give an algorithm for finding a subgame stable poly-EFG decomposition in a neighbourhood that uses stochastic gradient descent. We will use this algorithm in the next chapter to find poly-EFG decompositions of Kuhn and Leduc Poker.

4.1 Poly-EFGs

What is the appropriate extension of polymatrix games to EFGs? Given some n-player EFG G, what should the subgame between i and j be in the graph? Unlike in normal form games, players act sequentially, and i and j may either observe actions or have their utility impacted by other players. There is additional structure present in EFGs beyond the players' pure strategies.

The approach we take is to have an EFG for each pair of players i and j. For simplicity, we assume that each subgame G'_{ij} shares the same structure as some n-player game G, but information sets where $P(I) \notin \{i, j\}$ now belong to the chance player c.

Definition 4.1.1 (Subgame)**.** Let $G = (N, \mathcal{A}, H, Z, A, P, u, \mathcal{I}, c, \pi_c)$ be some EFG. We define a *subgame* $G'_{ij} = (\{i, j\}, \mathcal{A}, H, Z, A, P', (u'_{ij}, u'_{ji}), \mathcal{I}, c, \pi'_c)$ as a structurally identical game to G between i and j with player function P' and utility functions (u'_{ij}, u'_{ji}), then

$$P'(h) \doteq \begin{cases} P(h) & \text{if } P(h) \in \{i, j\} \\ c & \text{o.w.} \end{cases}$$

and let π'_c be the strategy of the chance player in G'_{ij}. We put no restrictions on π'_c.

Note that u'_{ij}, u'_{ji} are not necessarily defined in the above definition. They may take any values. We merely want the subgame G'_{ij} to share the structure of G. In G'_{ij}, i and j's utility only depends on their strategies π_i, π_j and chance's actions π'_c:

$$\begin{aligned} u'_{ij}(\pi_i, \pi_j) &\doteq \mathbb{E}_{z \sim (\pi_i, \pi_j, \pi_j)} \left[u'_{ij}(z) \right] \\ &= \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi'_c) u'_{ij}(z). \end{aligned}$$

What should π'_c be defined as? This turns out to not matter very much. Given any subgame G'_{ij} with chance strategy π'_c and utility functions u'_{ij}, u'_{ji}, for any G''_{ij} with chance strategy π''_c, we can find u''_{ij}, u''_{ji} so that the utility of players between the two games will always be equal for any strategy profile (π_i, π_j).

Definition 4.1.2. We say π_c is *fully mixed* if $\pi'_c(a,h) > 0 \ \forall h \in \{h \in H \mid P(h) = c\}, a \in A(h)$

Proposition 4.1.3. *Let G be an EFG and G'_{ij} a subgame between i and j with utility functions u'_{ij}, u'_{ji} and fully mixed chance strategy π'_c. Given G''_{ij}, a subgame between i and j with fully mixed chance strategy π''_c, we may find u''_{ij}, u''_{ji} such that $\forall \pi_i, \pi_j$ we have $u'_{ij}(\pi_i, \pi_j) = u''_{ij}(\pi_i, \pi_j)$ and $u'_{ji}(\pi_i, \pi_j) = u''_{ji}(\pi_i, \pi_j)$.*

Proof. Note that $p_c(z, \pi'_c), p_c(z, \pi''_c) \neq 0$. Then $\forall z \in Z$, define $u''_{ij}(z) = \frac{p_c(z,\pi'_c)}{p_c(z,\pi''_c)} u'_{ij}(z)$ and $u''_{ji}(z) = \frac{p_c(z,\pi'_c)}{p_c(z,\pi''_c)} u'_{ji}(z)$. Then

$$\begin{aligned}
u''_{ij}(\pi_i, \pi_j) &= \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi''_c) u''_{ij}(z) \\
&= \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi''_c) \frac{p_c(z, \pi'_c)}{p_c(z, \pi''_c)} u'_{ij}(z) \\
&= \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi'_c) u'_{ij}(z) \\
&= u'_{ij}(\pi_i, \pi_j).
\end{aligned}$$

□

For the remainder of this thesis, when defining a subgame between i and j given an EFG G, we define the subgame chance player's strategy to be equal to π_c in G at information sets I where $P(I) = c$ in G and uniform randomly otherwise.

Having defined subgames, we may now define our representation of extensive-form polymatrix games.

Definition 4.1.4 (Poly-EFG)**.** A poly-EFG $(N, E, \mathcal{G})$ is defined by a graph with nodes N, one for each player, a set of edges E and a set of games $\mathcal{G} = \{G_{ij} \mid \forall (i,j) \in E\}$ where $G_{ij} \in \mathcal{G}$ is a two player EFG between i and j and all $G_{ij} \in \mathcal{G}$ are a subgame between i and j and all subgames are defined with respect to some EFG G.

Let $G_{ij} \in \mathcal{G}$ denote the subgame between i and j. We use subscript ij (e.g. Z_{ij}) to refer to parts of G_{ij}.

Since each subgame of the poly-EFG is the subgame of the same G, the space of pure, mixed and behaviour strategies is the same for each subgame for any player. A player chooses a strategy (whether pure, mixed or behaviour) and plays this strategy in each subgame. A player's global utility is the sum of their utility in each of their subgames. We have

$$u_i(\pi) = \sum_{(i,j)\in E_i} u_{ij}(\pi_i, \pi_j),$$

where $E_i \subseteq E$ is the set of edges connected to i in the graph and u_{ij} is the utility function of i in subgame G_{ij}.

4.1.1 Constant-Sum and Subgame Stable Poly-EFGs

Here, we give definitions of constant-sum and subgame stability for poly-EFGs. These definitions are largely identical to their normal-form counterparts, we merely provide them here for completeness.

Poly-EFGs are constant-sum if, for each subgame, the utilities at the terminals add up to a constant.

Definition 4.1.5 (Constant-sum)**.** We say a poly-EFG G is constant-sum if $\forall G_{ij} \in \mathcal{G}, z \in Z_{ij}$, $u_{ij}(z) + u_{ji}(z) = c_{ij}$ where Z_{ij} is the set of terminal histories of $\forall G_{ij}$ and c_{ij} is a constant.

We may also define approximate poly-EFGs in the same way as in normal-form games.

Definition 4.1.6 (δ-constant sum poly-EFG)**.** An EFG G is δ-constant sum poly-EFG (δ-CSP) if there exists a constant-sum poly-EFG $\check{G}$ with global utility function $\check{u}$ such that $\forall i \in N, \pi \in \Pi$, $|u_i(\pi) - \check{u}_i(\pi)| \leqslant \delta$. We denote the set of such CSP games as $\text{CSP}_\delta(G)$.

Finally, we define subgame stability for poly-EFGs. Our definitions are near-identical from the normal-form definitions.

Definition 4.1.7 (Subgame stable profile)**.** Let G be a poly-EFG. We say a strategy profile π is γ-subgame stable if $\forall (i,j) \in E$, we have (π_i, π_j) is a γ-Nash of G_{ij}.

Definition 4.1.8 (Subgame stable game)**.** Let G be a poly-EFG. We say G is (ϵ, γ)-subgame stable if for *any* ϵ-Nash equilibrium π of G, π is γ-subgame stable.

4.2 Theoretical Results For Poly-EFGs

Our theoretical results from Chapter 3 continue to hold in poly-EFGs. The idea is to use the induced normal-form of a poly-EFG and then apply the theory of the previous chapter. Only the main results are given here, full details are given in Appendix B. We assume perfect recall for the remainder of this thesis.

First, we will characterize what self-play will produce in EFGs. These are called *marginal behaviour strategies*.

Definition 4.2.1 (Marginal behaviour strategy)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let π_i^μ be the *marginal behaviour strategy* for i where $\pi_i^\mu(a, I)$ is defined arbitrarily if $\sum_{\rho_i' \in \mathrm{P}_i(I)} s_i^\mu(\rho_i') = 0$ and otherwise

$$\pi_i^\mu(a, I) \doteq \frac{\sum_{\rho_i \in \mathrm{P}_i(a,I)} s_i^\mu(\rho_i)}{\sum_{\rho_i' \in \mathrm{P}_i(I)} s_i^\mu(\rho_i')} \quad \forall I \in \mathcal{I}_i, a \in A(I),$$

where $s_i^\mu(\rho_i) \doteq \sum_{\rho_{-i} \in \mathrm{P}_{-i}} \mu(\rho_i, \rho_{-i})$.

Definition 4.2.2 (Marginal behaviour strategy profile)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let π^μ be a *marginal behaviour strategy profile*, where π_i^μ is a marginal behaviour strategy $\forall i \in N$.

An extension of Theorem 3.4.3 holds for poly-EFGs. Let Π^μ be the set of marginal behaviour strategy profiles for any CCE of G and Π_i^μ be the set of marginal behaviour strategies for i.

Proposition 4.2.3. *If an EFG G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable and μ is a CCE* [1] *of G, then*

$$\mathrm{Vul}_i(\pi^\mu, \Pi_{-i}) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta,$$

[1] Note that "CCE" refers to a *normal-form* CCE (NFCCE) in the language of [17].

and

$$\mathrm{Ex}(\Pi^{\mu}) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

4.2.1 Vulnerability Against Self-Taught Agents in EFGs

Next we show an analogue of Theorem 3.5.3 for extensive-form games. In Section 3.5 we defined $S(\mathcal{A}_i)$ as the set of marginal strategy profiles for a no-regret learning algorithm $\mathcal{A}_i$. Many algorithms for EFGs will compute behaviour strategies, so we use $\Pi(\mathcal{A}_i) = \{\pi^{\mu} \mid (\mu, (\Phi_i)_{i\in N}) \in \mathcal{M}(\mathcal{A}_i)\}$ as the set of marginal behaviour strategy profiles of $\mathcal{M}(\mathcal{A}_i)$ (recall that $\mathcal{M}(\mathcal{A}_i)$ is the set of mediated equilibria reachable by learning algorithm $\mathcal{A}_i$. Then let $\Pi_i(\mathcal{A}_i) \doteq \{\pi_i \mid \pi \in \Pi(\mathcal{A}_i)\}$ be the set of i's marginal strategies from $\Pi(\mathcal{A}_i)$. For example, if $\mathcal{A}$ is CFR, $\mathcal{M}(\mathcal{A})$ is the set of observably sequentially rational CFCCE [45] and $\Pi(\mathcal{A})$ is the set of behaviour strategy profiles computable by CFR. Next, suppose each player i learns with their own self-play algorithm $\mathcal{A}_i$. Let $\mathcal{A}_N \doteq (\mathcal{A}_1, ...\mathcal{A}_n)$ be the profile of learning algorithms and $\Pi^{\times}(\mathcal{A}_N) \doteq \times_{i\in N}\Pi_i(\mathcal{A}_i)$ be the set of all possible match-ups between strategies learned in self-play by those learning algorithms.

Definition 4.2.4. We say a game G is *δ-CSP in the neighbourhood of* $\Pi' \subseteq \Pi$ if there exists a constant sum poly-EFG $\check{G}$ such that $\forall \pi \in \Pi'$ we have $|u_i(\pi) - \check{u}_i(\pi)| \leqslant \delta$. We denote the set of such CSP games as $\mathrm{CSP}_{\delta}(G, \Pi')$.

Definition 4.2.5. We say a CSP game G is *γ-subgame stable in the neighbourhood of* Π' if $\forall \pi \in \Pi', \forall (i,j) \in E$ we have that (π_i, π_j) is a γ-Nash of G_{ij}.

The following proposition bounds the vulnerability if each agent i learns via a no-regret learning algorithm $\mathcal{A}_i$.

Proposition 4.2.6. *If G is δ-CSP in the neighbourhood of $\Pi^{\times}(\mathcal{A}_N)$ and $\exists \check{G} \in \mathrm{CSP}_{\delta}(G, \Pi^{\times}(\mathcal{A}_N))$ that is γ-subgame stable in the the neighbourhood of $\Pi(\mathcal{A}_i)$, then for any $\Pi \in \Pi(\mathcal{A}_i)$*

$$\mathrm{Vul}_i\left(\pi, \Pi^{\times}_{-i}(\mathcal{A}_N)\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

4.3 Leveraging the Poly-EFG Representation for Computing CSP Decompositions

The poly-EFG representation gives rise to more efficient algorithms for computing a poly-EFG decomposition. As a proof of concept, we show that in perfect information EFGs, we can write a linear program to compute the optimal polymatrix decomposition for an EFG that is exponentially smaller than LP 2 from Section 3.4. Recall that $\underline{\delta}$ is the *minimum* value of δ such that a game is δ-CSP. In an perfect information EFG, we can compute $\underline{\delta}$ with the following LP. The decision variables are $\underline{\delta}$, the values of $\check{u}_{ij}(z)\ \forall z \in Z, (i,j) \in E$ and $c_{ij}\ \forall (i,j) \in E$.

LP 4

$$
\begin{aligned}
\min \quad & \underline{\delta} \\
\text{s.t.} \quad & u_i(z) - \sum_{j \in -i} \check{u}_{ij}(z) \leqslant \underline{\delta} \quad \forall i \in N, z \in Z \\
& u_i(z) - \sum_{j \in -i} \check{u}_{ij}(z) \geqslant -\underline{\delta} \quad \forall i \in N, z \in Z \\
& \check{u}_{ij}(z) + \check{u}_{ji}(z) = c_{ij} \quad \forall i \neq j \in N, z \in Z
\end{aligned}
$$

The trick is that in perfect information EFGs, each pure strategy profile leads to a single terminal. Hence, rather than having a constraint for each pure strategy profile, a constraint for each terminal will suffice. This leads to an exponential reduction in the number of constraints over LP 2.

4.4 Computing a SS-CSP Decomposition in a Neighbourhood

Next, we give an algorithm for computing a polymatrix decomposition that is approximately subgame stable in the neighbourhood of some set of strategy profiles Π' and approximately CSP in the neighbourhood of $\Pi^{\times} = \times_{i \in N} \Pi'_i$. Given a game G as input, it uses stochastic gradient descent (SGD) to compute a CSP game $\check{G}$ that minimizes a loss function with two components: how close $\check{G}$ is to G in the neighbourhood of $\Pi^{\times}$ and how subgame stable $\check{G}$ is for the

neighbourhood of Π'. In principle, this procedure can be used with either normal-form polymatrix games or poly-EFGs, but we show it here with poly-EFGs since it is primarily intended for use with games that are too large to use linear programming.

For each subgame $\check{G}_{ij}$ we store a single vector $\check{u}_{ij}$ where the entry $\check{u}_{ij}(z)$ gives the value of the utility for corresponding terminal z. We additionally store a constant $\check{c}_{ij}$ for each subgame. Player i will use $\check{u}_{ij}$ when computing $\check{u}_{ij}(\pi_i, \pi_j)$:

$$\check{u}_{ij}(\pi_i, \pi_j) = \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi_c) \check{u}_{ij}(z).$$

Whereas we compute $\check{u}_{ji}(\pi_i, \pi_j)$ as follows.

$$\check{u}_{ji}(\pi_i, \pi_j) = \sum_{z \in Z} p_i(z, \pi_i) p_j(z, \pi_j) p_c(z, \pi_c) \left(\check{c}_{ij} - \check{u}_{ij}(z)\right).$$

For simplicity, let $\check{u}$ denote the stacked vector of all $\check{u}_{ij}$ and $\check{c}_{ij}$. We additionally initialize $\check{G}$ as a fully connected graph.

The overall loss function which we minimize has two components: first, $\mathcal{L}^\delta$ is the error between the utility functions of G and $\check{G}$; it is a proxy for δ in δ-CSP.

$$\begin{aligned}
\mathcal{L}^\delta(\pi; \check{u}, u) &\doteq \sum_{i \in N} \left| \check{u}_i(\pi) - u_i(\pi) \right| \\
&= \sum_{i \in N} \left| \left(\sum_{(i,j) \in E_i} \check{u}_{ij}(\pi_i, \pi_j) \right) - u_i(\pi) \right|.
\end{aligned}$$

The other component of the overall loss function, $\mathcal{L}^\gamma$, measures the subgame-stability. First, we define $\mathcal{L}^\gamma_{ij}$, which only applies to a single subgame.

$$\begin{aligned}
\mathcal{L}^\gamma_{ij}(\pi_{ij}, \pi^*_{ij}; \check{u}) \doteq &\max\left(\check{u}_{ij}(\pi^*_i, \pi_j) - \check{u}_{ij}(\pi_{ij}), 0\right) \\
&+ \max\left(\check{u}_{ji}(\pi_i, \pi^*_j) - \check{u}_{ji}(\pi_{ij}), 0\right).
\end{aligned}$$

Where $\pi_{ij} = (\pi_i, \pi_j)$ is a profile and $\pi^*_{ij} = (\pi^*_i, \pi^*_j)$ is a profile of deviations. Then

$$\mathcal{L}^\gamma(\pi, \pi^*; \check{u}) \doteq \sum_{(i,j) \in E} \mathcal{L}^\gamma_{ij}(\pi_{ij}, \pi^*_{ij}; \check{u}).$$

Algorithm 2 shows how to compute a subgame-stable constant-sum polymatrix decomposition via SGD. As input, the algorithm receives a game G, a finite set of strategy profiles Π', learning rate η, number of training epochs T, hyperparameter $\lambda \in [0, 1]$ and batch size B. First, we initialize $\Pi^{\times}$ as the set of all match-ups amongst strategies in Π'.

We then repeat the following steps for each epoch. First, we compute a best-response (for example, via sequence-form linear programming) to each strategy π'_i in Π' in each subgame; the full process is shown in Algorithm 3. After computing these best-responses for the current utility function of $\check{G}$, we try to fit $\check{u}$ to be nearly CSP in the neighbourhood of $\Pi^{\times}$ and subgame stable in the neighbourhood of Π'. Since $\Pi^{\times}$ is exponentially larger than Π', we partition it into batches, then use batch gradient descent[2]. We use the following batch loss function, which computes the average values of $\mathcal{L}^{\delta}$ and $\mathcal{L}^{\gamma}$ over the batches, then weights the losses with λ. Let Π^b denote a batch of strategy profiles from $\Pi^{\times}$ with size B, the overall loss function is

$$\mathcal{L}(\Pi^b, \Pi', \Pi^*; \check{u}, u) \doteq \frac{\lambda}{B} \sum_{\pi \in \Pi^b} \mathcal{L}^{\delta}(\pi; \check{u}, u) + \frac{(1-\lambda)}{|\Pi'|} \sum_{\pi \in \Pi'} \sum_{\pi^* \in \Pi^*} \mathcal{L}^{\gamma}(\pi, \pi^*; \check{u}).$$

We take this loss and find its gradient with respect to $\check{u}$, then update $\check{u}$:

$$\check{u} \leftarrow \check{u} - \eta \cdot \nabla_{\check{u}} \mathcal{L}(\Pi^b, \Pi', \Pi^*; \check{u}, u).$$

We found that in practise the gradient can be quite large relative to $\check{u}$, which has the potential to destabilize optimization. This is alleviated by normalizing the gradient by its l^2 norm.

$$g \leftarrow \nabla_{\check{u}} \mathcal{L}(\Pi^b, \Pi', \Pi^*; \check{u}, u)$$
$$\check{u} \leftarrow \check{u} - \eta \cdot \frac{g}{\|g\|_2}$$

[2]One could also partition Π' into batches if it were too large. However, in this thesis, Π' will always be relatively small.

Algorithm 2 Compute $\check{G}$

Input: G, Π', η, T, λ, B
Initialize $\check{u}$ to all 0
$\Pi^{\times} \leftarrow \times_{i \in N} \hat{\Pi}_i$
for $t \in 1...T$ **do**
  $\Pi^* \leftarrow \text{getBRs}(\check{G}, \Pi')$
  $\mathcal{B} \leftarrow$ partition of $\Pi^{\times}$ into batches of size B
  for $\Pi^b \in \mathcal{B}$ **do**
    $g \leftarrow \nabla_{\check{u}} \mathcal{L}(\Pi^b, \Pi', \Pi^*; \check{u}, u)$
    $\check{u} \leftarrow \check{u} - \eta \cdot \frac{g}{\|g\|_2}$
  end for
end for
{Lastly, output δ and γ}
$\delta \leftarrow \max_{\pi \in \Pi^{\times}} |u_i(\pi) - \check{u}_i(\pi)|$
$\gamma \leftarrow \max_{\pi \in \Pi'} \max_{i \neq j \in N \times N} (\check{u}_{ij}(BR_{ij}(\pi_j), \pi_j) - \check{u}_{ij}(\pi_i, \pi_j))$
return $\check{u}, \gamma, \delta$

Algorithm 3 getBRs

Input: $\check{G}$, Π'
$\Pi_i^* \leftarrow \varnothing \ \forall i \in N$
for $i \neq j \in N \times N$ **do**
  for $\pi_j \in \Pi'_j$ **do**
    compute $\pi_{ij}^* \in \arg\max_{\pi'_i \in \Pi_i} \check{u}_{ij}(\pi'_i, \pi_j)$
    $\Pi_i^* \leftarrow \Pi_i^* \cup \{\pi_{ij}^*\}$
  end for
end for
return $\times_{i \in N} \Pi_i^*$

Chapter 5

Experiments: Is Poker Approximately Subgame Stable Constant-Sum Polymatrix?

Self-play with regret minimization has been very successful in producing strong strategies in multi-player Texas hold 'em. Could this be because these strategies come from a subgame stable CSP part of Texas hold 'em?

Conjecture 5.0.1. Self-play with regret minimization performs well in multi-player Texas hold 'em because "good" players (whether professional players or strategies learned by self-play) play in a part of the game's strategy space that is close to a subgame stable CSP game for some low values of γ, δ.

Unfortunately, multi-player Texas hold'em is too large to analyse using the algorithms outlined in this thesis. However, we instead analyse two smaller poker games, called Kuhn Poker and Leduc Poker.

5.1 Kuhn Poker

Kuhn poker [32] was originally developed for two players but was extended to a 3-player variant by Abou Risk and Szafron [1]. The game has 4 cards, with each player receiving one privately (the remaining card is hidden, i.e. there are no public cards), followed by one round of betting with a fixed bet size. Players may pass and stay in the game if there is no outstanding bet, otherwise they can only call or fold. Figure 5.1 shows one subtree of Kuhn poker. CFR

was previously shown to generate approximate Nash equilibria in Kuhn poker [1].

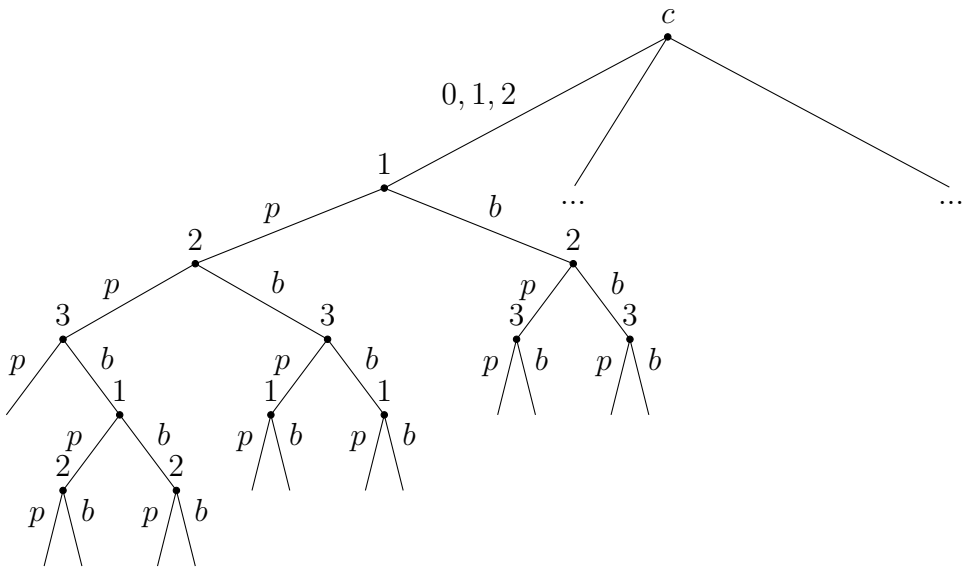


Figure 5.1: A partial subtree of Kuhn Poker. Chance (c) has dealt cards $0, 1, 2$ to players 1, 2 and 3, respectively. Player 1 may either pass (p) or bet (b) before play goes to player 2. We omit payoffs at terminals for clarity.

5.1.1 Is Kuhn Poker Approximately CSP?

While Kuhn Poker is relatively small for an extensive-form game (each player has 16 information sets in 3 player Kuhn Poker), this is still far too large for a naive decomposition using the induced normal-form of Kuhn Poker, since each player has 2^{16} pure strategies and using methods such as LP would require solving a linear program with $\approx 3 * (2^{16})^3$ constraints. Instead, we analytically find a lower bound for $\underline{\delta}$.

Proposition 5.1.1. *3 player Kuhn Poker is at least* 0.125*-CSP.*

Proof. Consider the following pure strategies:

1. ρ_1^b: player 1 always bets.
2. ρ_2^b: player 2 always bets.
3. ρ_2^p: player 2 always passes.

4. $\rho_3^{2\to b}$: player 3 bets if player 2 does, otherwise they pass.

5. ρ_3^{p}: player 3 always passes.

Consider the following pure strategy profiles:

$$\rho^1 = (\rho_1^b, \rho_2^p, \rho_3^p)$$
$$\rho^2 = (\rho_1^b, \rho_2^b, \rho_3^p)$$
$$\rho^3 = (\rho_1^b, \rho_2^p, \rho_3^{2\to b})$$
$$\rho^4 = (\rho_1^b, \rho_2^b, \rho_3^{2\to b})$$

Then we have $u_1(\rho^1) = 1$ since 1 wins the ante of every hand, giving them a net winning of 1. Likewise $u_1(\rho^3) = 1$ since both 2 and 3 will continue to always pass.

$u_1(\rho^2) = 0.5$. Since 3 always passes, and 1 and 2 always bet, the pot will be 5, 3 from the ante and 2 from the bets of 1 and 2; thus the net winnings for any player will be 3. 1 can win with one of cards $1, 2, 3$. If they are dealt a 3, they always win. If they are dealt a 2, they win if 2 does not have a 3 which has probability $1 - P(2 \text{ has a } 3 \mid 1 \text{ has a } 2) = 1 - \frac{1}{3} = \frac{2}{3}$. If they are dealt a 1, they win if 2 does not have 2 or 3 which has probability $1 - P(2 \text{ has a } 2 \text{ or } 3 \mid 1 \text{ has a } 1) = 1 - \frac{2}{3} = \frac{1}{3}$. Altogether, 1 has a probability of winning equal to $\frac{1}{4} + \frac{1}{4} \cdot \frac{2}{3} + \frac{1}{4} \cdot \frac{1}{3} = \frac{1}{2}$. Thus $u_1(\rho^2) = \frac{1}{2} \cdot 3 + \frac{1}{2} \cdot -2 = 0.5$

Lastly, $u_1(\rho^4) = 0$. Since all players always bet (recall that 3 bets if 2 does), the pot will be 6 and net winnings are 4. 1 can win by either holding a 3 or 2. They hold a 3 with probability $\frac{1}{4}$. 1 wins if they hold a 2 if neither 2 nor 3 hold a 3, which happens with probability $\frac{2}{3} \cdot \frac{1}{2}$. Thus 1's total probability of winning is $\frac{1}{4} + \frac{1}{4} \cdot \frac{2}{3} \cdot \frac{1}{2} = \frac{1}{3}$ so $u_1(\rho^4) = \frac{1}{3} \cdot 4 + \frac{2}{3} \cdot -2 = 0$.

We then solve the following LP to obtain the lower bound on $\underline{\delta}$.

$$\min_{\check{u}_{ij}(\cdot), \delta, c_{ij}} \quad \delta$$
$$u_i(\rho) - \sum_{j \in -i} \check{u}_{ij}(\rho_i, \rho_j) \leqslant \delta \quad \forall i \in N, \rho \in \{\rho^1, \rho^2, \rho^3, \rho^4\}$$
$$u_i(\rho) - \sum_{j \in -i} \check{u}_{ij}(\rho_i, \rho_j) \geqslant -\delta \quad \forall i \in N, \rho \in \{\rho^1, \rho^2, \rho^3, \rho^4\}$$
$$\check{u}_{ij}(\rho_i, \rho_j) + \check{u}_{ji}(\rho_i, \rho_j) = c_{ij} \quad \forall i \neq j \in N, \rho \in \{\rho^1, \rho^2, \rho^3, \rho^4\}$$

We obtain a solution of $\delta = 0.125$. □

5.1.2 CFR Converges to a Nearly SS-CSP Neighbourhood in Kuhn Poker

Perhaps the part of the game that "good" agents (i.e. skilled human players or no-regret learning algorithms) play in is closer to an SS-CSP part of the game than Kuhn poker is overall. While we cannot exhaustively check whether skilled human players or all no-regret learning algorithms play in a SS-CSP neighbourhood of the strategy space, we can generate a set of strategies using self-play as a proxy and test this claim empirically.

We use CFR to compute a set of approximate marginal strategies.[1] CFR is a deterministic algorithm, so we use different random initializations of CFR's initial strategy in order to generate a set of marginal strategies. We train random initializations of CFR for 100,000 iterations.

Do the CFR strategies lie in a near subgame-stable CSP neighbourhood? We ran Algorithm 2 30 times to answer this question. Each run used its own set of 30 CFR-generated strategies (i.e. we trained CFR 900 times in total). Let $\Pi(\text{CFR})_j$ denote the CFR-generated strategy profiles for the jth run of Algorithm 2 and

$$\Pi^{\times}(\text{CFR})_j = \bigtimes_{i \in N} \Pi_i(\text{CFR})_j$$

to denote the set of all match-ups between these 30 strategy profiles. Given that the size of each $\Pi(\text{CFR})_j$ is 30, there are $30^3 = 27000$ strategy profiles in each $\Pi^{\times}(\text{CFR})_j$.

First, we measure the diversity of strategy profiles used in each run of Algorithm 2. We do this for each $\Pi(\text{CFR})_j$ by taking each pair of strategy profiles $\pi, \pi' \in \Pi(\text{CFR})_j$ and computing the total variation between these two probability distributions induced over the terminal histories of Kuhn poker. We denote the maximum total variation for run j as TV_j, where

$$TV_j \doteq \max_{\pi, \pi' \in \Pi(\text{CFR})_j} \frac{1}{2} \sum_{z \in Z} |p(z, \pi) - p(z, \pi')|.$$

[1] We use the OpenSpiel implementation [34].

TV_j is constrained to be between 0 and 1, where 0 means the two distributions are the same and 1 means they are maximally different. Figure 5.2 shows the maximum total variation between any two of the strategy profiles used in each run.

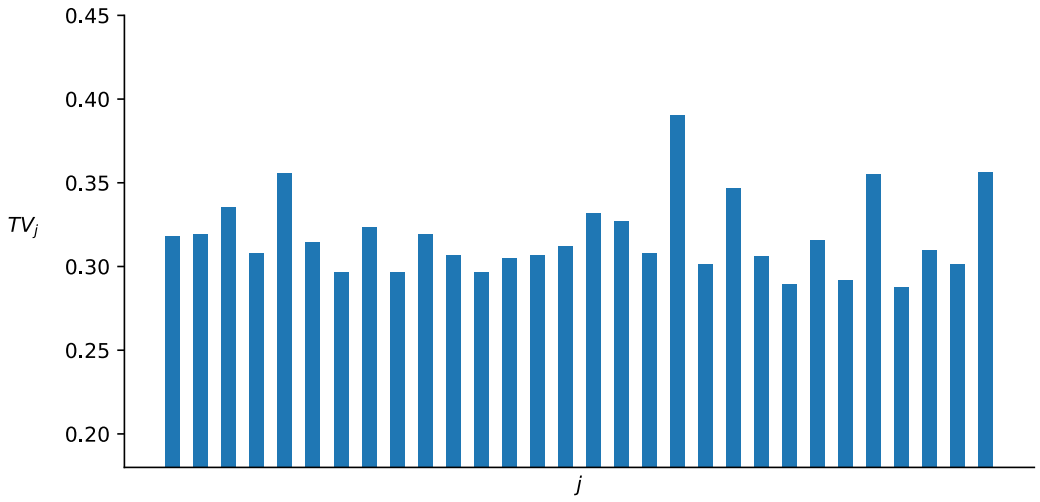
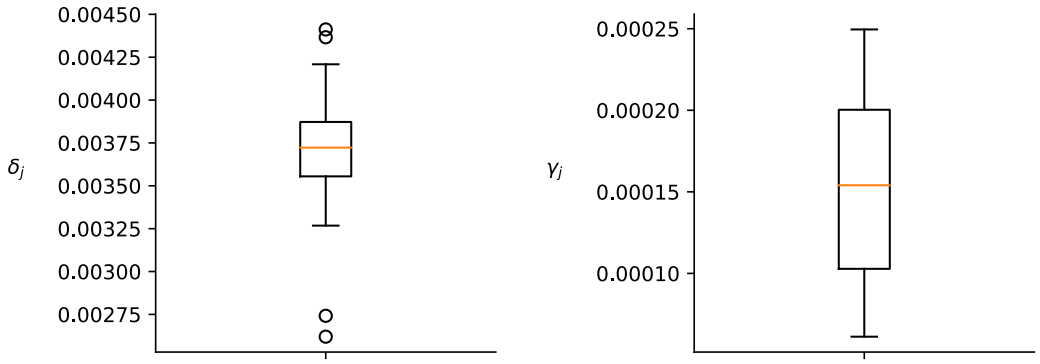


Figure 5.2: The maximum total variation for each $\Pi(\text{CFR})_j$ used in each run of Algorithm 2 in **Kuhn Poker**. Different runs are shown on the x-axis, and the corresponding TV_j for run j is shown with the bars A value of 0 indicates minimal diversity and 1 means maximal diversity. The minimum, mean, maximum and standard error across runs are 0.29, 0.32, 0.39 and 0.0043, respectively.

We used Algorithm 2 with $\lambda = 0.5$, $B = 30$, $T = 1000$ and a learning rate schedule: η is initialized to 2^{-6}, then divided by half every 5 epochs until it reaches 2^{-17}. The values of η were chosen manually to encourage convergence and λ was chosen with a coarse hyperparameter sweep.

We ran Algorithm 2 a total of 30 times. The jth run will compute δ_j and γ_j such that Kuhn Poker is δ_j-CSP in the neighbourhood of $\Pi^{\times}(\text{CFR})_j$ and γ_j-subgame stable in the neighbourhood of $\Pi(\text{CFR})_j$. Figure 5.3 shows our results: we found that across the 30 runs, Algorithm 2 shows that Kuhn Poker was at most 0.0044-CSP and 0.00025-subgame stable across the runs.

How well do these values bound vulnerability with respect to other CFR-learned strategies? For each of the runs, we computed the vulnerability with respect to the strategies of that run, $\Pi(\text{CFR})_j$, by evaluating each strategy



(a) (b)

Figure 5.3: Boxplots showing the values of δ_j and γ_j for each of the 30 runs of Algorithm 2 in **Kuhn Poker**. Figure 5.3a shows the values of δ_j, with the minimum, mean, maximum and standard error being 0.0026, 0.0037, 0.0044 , and $7.03e-5$, respectively. Figure 5.3b shows the values of γ_j, with the minimum, mean, maximum and standard error being $6.12e-5$, 0.00015, 0.00025 and $1.06e-05$, respectively.

against each other and taking the maximum vulnerability:

$$\mathrm{Vul}_j \doteq \max_{i \in N} \max_{\pi \in \Pi(\mathrm{CFR})_j} \mathrm{Vul}_i \left(\pi, \Pi^{\times}_{-i}(\mathrm{CFR})_j\right). \tag{5.1}$$

Figure 5.4 shows, for each run, the bounds implied by Proposition 4.2.6 and the values of δ_j and γ_j and the true vulnerability. We see that the bounds do a good job of upper bounding the vulnerability. Across the runs, the bounds are at minimum 1.06 times the vulnerability, at maximum 1.92 times the vulnerability and on average 1.40 times as large, with a standard error of 0.041.

5.1.3 Leduc Poker

Leduc poker was also extended to a 3-player variant by Abou Risk and Szafron [1]. The game has 8 cards with 4 ranks and 2 suits. At the start, each player antes 1 and receives one card privately before an initial round of betting commences. A public card is then revealed before the second round of betting. The bet values in the first and second round are 2 and 4, respectively. Leduc poker is substantially larger than Kuhn poker; Kuhn poker has a total of 48 information sets while Leduc poker has 25800.

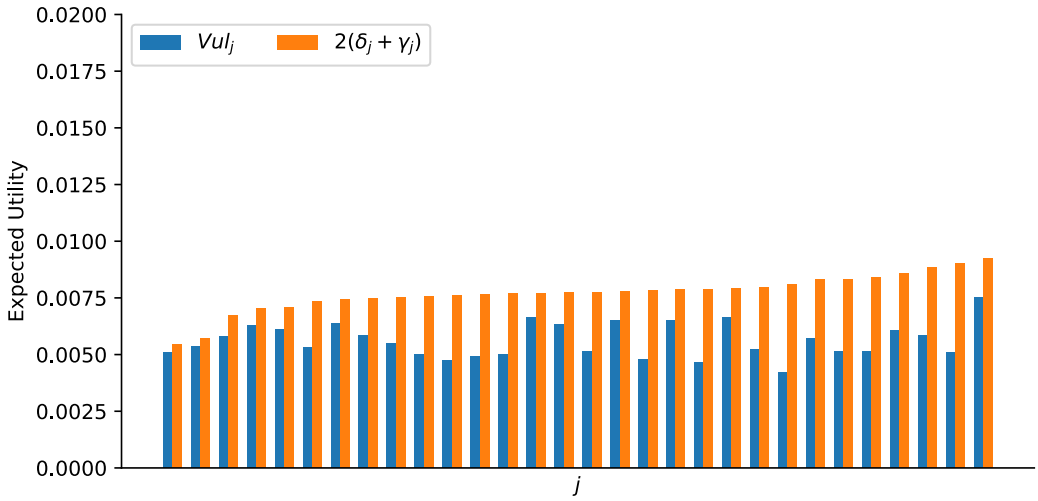


Figure 5.4: Bounds on vulnerability compared to true vulnerability in **Kuhn Poker** for each run. Each of the 30 runs are shown on the x-axis. For each run j, we compute the bounds determined by Proposition 4.2.6, which are $(n-1)\gamma_j + 2\delta_j = 2(\gamma_j + \delta_j)$. These value are shown in orange. The blue bars are the maximum vulnerability in each run, computed using (5.1). The ordering of bars in this plot matches the ordering of bars in Figure 5.2.

5.1.4 CFR+ Converges to a Nearly SS-CSP Neighbourhood in Leduc Poker

We repeated the above experiments with Leduc poker using CFR+ as the self-play algorithm rather than CFR, since Leduc poker is substantially larger than Kuhn poker and CFR+ converges faster than CFR. Leduc poker is again too large to fully project into the space of SS-CSP games for a reasonable value of δ, but we can still see if CFR+ plays in an SS-CSP neighbourhood of Leduc Poker.

We again ran Algorithm 2 30 times, each time with its own set of 30 strategy profiles. These 900 strategy profiles are generated with CFR+ in self-play for 1000 iterations with random initializations. Interestingly, we found that CFR+ converges to approximate Nash equilibria in Leduc poker, with a maximum value of ϵ equal to 0.013 after 1000 iterations. As we will show in Section 5.1.5, CFR also empirically produces approximate Nash equilibria in Leduc poker.

Let $\Pi(\text{CFR+})_j$ denote the set of CFR+-learned strategy profiles for run

j; and $\Pi^{\times}(\text{CFR+})_j$ denote the set of all match-ups between these 30 strategy profiles. Figure 5.5 shows diversity of $\Pi(\text{CFR+})_j$ for each of the 30 runs, measured with maximum total variation.

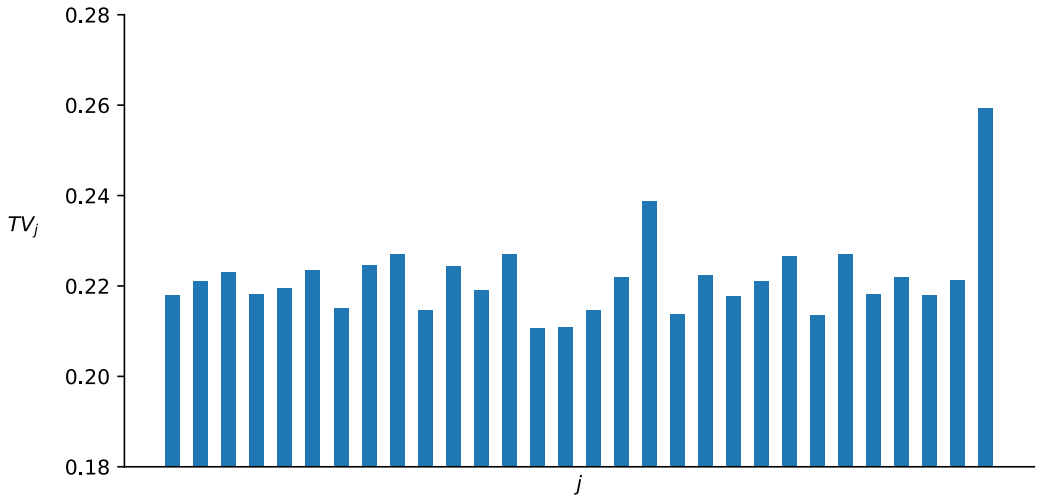
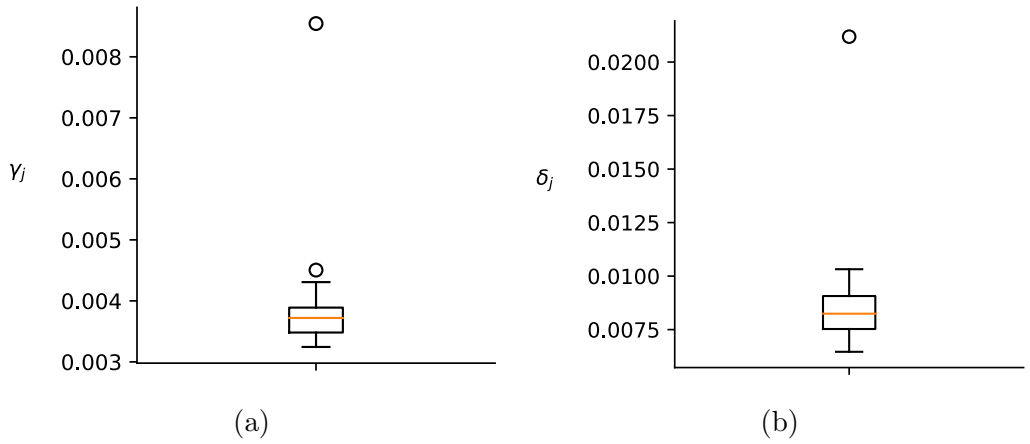


Figure 5.5: The maximum total variation for each $\Pi(\text{CFR})_j$ used in different runs of Algorithm 2 in **Leduc Poker**. Different runs are shown on the x-axis, and the corresponding TV_j for run j is shown with the bars. A value of 0 indicates minimal diversity and 1 means maximal diversity. The minimum, mean, maximum and standard error across runs are 0.21, 0.22, 0.26 and 0.0016, respectively.

We used the same parameters for each run of Algorithm 2: $\lambda = 0.5$, $B = 30$, $T = 200$. We used a learning rate schedule where the learning rate η begins at 2^{-6}, then halves every 5 epochs until reaching 2^{-17}. Our results are shown in Figure 5.6. We see that across the 30 runs of Algorithm 2, Leduc poker is at most 0.021-CSP in the neighbourhood of $\Pi^{\times}(\text{CFR+})_j$ and 0.009-subgame stable in the neighbourhood of $\Pi(\text{CFR+})_j$. Figure 5.7 shows the Vul_j for each of the runs compared to the bounds on vulnerability given by Proposition 4.2.6. We see that the bounds are between 1.89 and 3.05 times the actual vulnerability, and are on average 2.51 times larger with a standard error of 0.049.



(a) (b)

Figure 5.6: Boxplots showing the values of δ_j and γ_j for each of the 30 runs of Algorithm 2 in **Leduc Poker**. Figure 5.3a shows the values of δ_j, with the minimum, mean, maximum and standard error being 0.006, 0.009, 0.021 and 0.00046, respectively. Figure 5.3b shows the values of γ_j, with the minimum, mean, maximum and standard error being 0.003, 0.004, 0.009 and 0.00016, respectively.

5.1.5 CFR Finds Approximate Nash in Leduc Poker

It was previously believed that CFR does not compute an ϵ-Nash equilibrium on 3-player Leduc for any reasonable value of ϵ. Previous work found that CFR computed a 0.130-Nash equilibrium after 10^8 iterations [1]. We saw in the previous section that CFR+ computes approximate Nash equilibria in Leduc poker—does this hold for CFR as well?

We ran 30 runs of CFR in self-play for 10,000 iterations and found that *all* of our strategies converged to an approximate Nash equilibrium with the maximum $\epsilon = 0.017$ after only 10^4 iterations. Figure 5.8b shows the shows the maximum deviation incentive

$$\epsilon = \max_{\pi_i'} u_i(\pi_i', \pi_{-i}) - u_i(\pi)$$

for each of the CFR strategies π computed by CFR in Leduc Poker. Each column is for one of the players and each point is one of the random seeds. We see the maximum value of ϵ after 10,000 iterations is 0.017. Figure 5.8a shows the maximum deviation incentive ϵ over 10,000 iterations. We average learning curves over 30 random seeds.

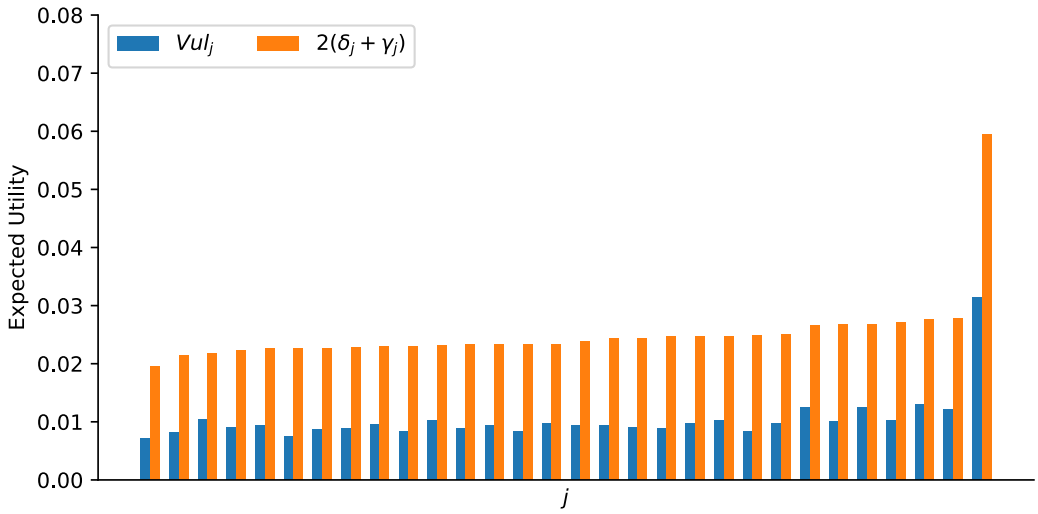


Figure 5.7: Bounds on vulnerability compared to true vulnerability in **Leduc Poker** for each run. Each of the 30 runs are shown on the x-axis. For each run j, we compute the bounds determined by Proposition 4.2.6, which are $(n-1)\gamma_j + 2\delta_j = 2(\gamma_j + \delta_j)$. These value are shown in orange. The blue bars are the maximum vulnerability in each run, computed using (5.1). The ordering of bars in this plot matches the ordering of bars in Figure 5.5. The rightmost run had both the highest vulnerability and highest diversity.

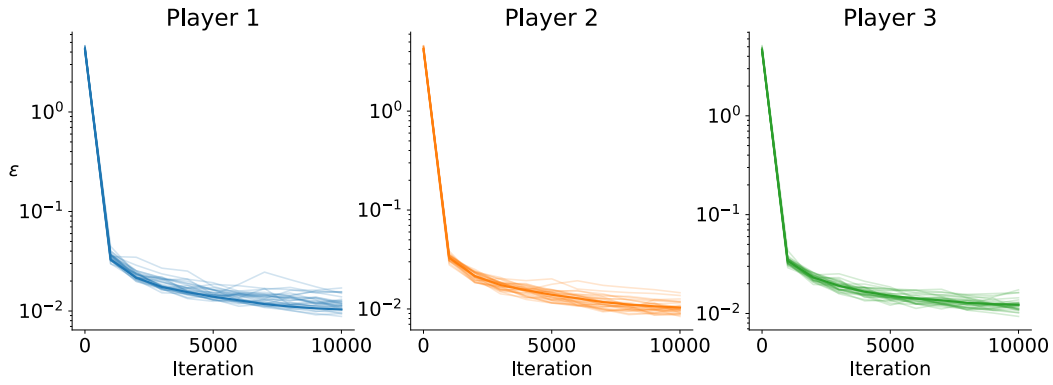
5.2 Vulnerability in a Cooperative Game

In games with low values of δ and γ, self-play will perform well against new opponents; however is the converse also true? Do games where self-play performs poorly against new opponents have high values of δ and γ?

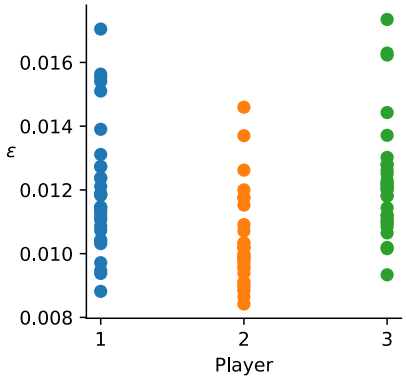
Self-play has been known to perform poorly against new agents in games with highly specialized conventions [27]. In training, a self-play algorithm may learn conventions that are incompatible with the conventions an independent instance of self-play may have learned. Much of this has to do with symmetries in the game [27].

Hanabi is one such game [5]. Hanabi is a cooperative game where players cannot see their own hands, but can see all other players hands; therefore players must give each other hints on how to act.

We show that a small version of the game of Hanabi—called Tiny Hanabi in the Openspiel framework [34]—is not close to the space of CSP games and self-play is quite vulnerable. Figure 5.9 shows this game. Chance deals one



(a)



(b)

Figure 5.8: CFR empirically computes Nash Equilibria in Leduc Poker. (a) shows learning curves over iterations for each of the players. We measure ϵ by finding a best-response with sequence-form linear programming every 1000 iterations. We show each of the individual instances of CFR with different random initializations in light-coloured lines and the average across seeds in bold. (b) shows the distribution of ϵ at iteration 10,000.

of two hands, A or B with equal probability. Only player 1 may observe this hand and must signal to other players through their actions, σ_1 and σ_2, which hand chance has dealt. If both players 2 and 3 then correctly choose their actions to match chance's deal ((a, a) for A or (b, b) for B) then all players get payoff equal to 1, otherwise all players get 0.

σ_1 and σ_2 can come to mean different things, σ_1 could signal to 2 and 3 to play a, or b. Self-play may learn either of these conventions. However, if a player trained in self-play encounters a set of players trained in an independent instance of self-play, they may not have compatible conventions.

This is indeed what happens when we train CFR on Tiny Hanabi in Fig-

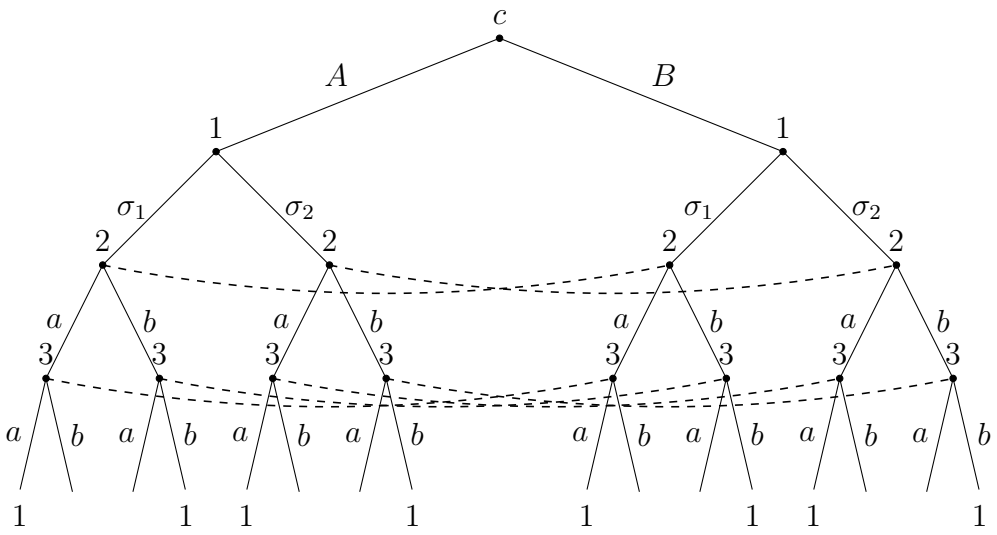


Figure 5.9: Tiny Hanabi. We omit payoffs of 0 at terminals.

ure 5.9. We trained 30 runs in self-play with different random initializations for 10,000 iterations. Some of these runs converged to each convention and when played against each other miscoordinated. We found

$$\max_{i \in N} \max_{\pi \in \Pi(\text{CFR})} \text{Vul}_i\left(\pi, \Pi^{\times}_{-i}(\text{CFR})\right) \approx 1,$$

as expected.

Tiny Hanabi is small enough that we can use LP3 (from Section 3.5) to decompose it into an approximate SS-CSP game. We found $\delta = 0.5$ and $\gamma \approx 0$, meaning the true vulnerability matched what our bounds predicted since $(n-1)\gamma + 2\delta \approx 1$. Why is $\gamma \approx 0$? We found that this was because our algorithm was setting the payoffs to equal to 0.50 for all terminal histories, which is trivially polymatrix.

Chapter 6

Strategic Equivalence to Polymatrix Games

In their seminal paper, Moulin and Vial [46] show that games which are *strategically equivalent* to a two-player zero-sum game share their equilibria, thus implying exchangeability. However, strategic equivalence to an n-player ($n > 3$) zero-sum game fails to guarantee exchangeability. Moulin and Vial leave unanswered what the appropriate generalization to n-player games should be.

In this chapter, we show that games which are strategically equivalent to CSP and SS-CSP games are a generalization to n-player games that preserve the nice properties of strategically two-player zero-sum games. First, we generalize the algebraic characterization of strategic equivalence from Moulin and Vial [46] to n-player games. Leveraging this result, we show three additional results. First, that the set of CCE are preserved between strategically equivalent games; second, that CCE of strategically approximate CSP games are approximately Nash equilibria; third, that we can bound the exchangeability of the set of marginal CCE strategies if a game is strategically equivalent to a SS-CSP game.

6.1 Strategic Equivalence

We begin with the definition of strategic equivalence from Moulin and Vial [46]. Their original definition was intended for 2 player games, but it easily generalizes to n-players.

Definition 6.1.1 (Strategic Equivalence)**.** G is strategically equivalent (SE) to G' for player i if

$$u_i(s'_i, s_{-i}) \geqslant u_i(s_i, s_{-i}) \iff u'_i(s'_i, s_{-i}) \geqslant u'_i(s_i, s_{-i}) \quad \forall s_i, s'_i \in S_i, s_{-i} \in S_{-i}.$$

Two strategically equivalent games share the same set of Nash equilibria—this is an immediate corollary of the definition. Moreover, if a game is strategically equivalent to a two-player zero-sum game, then its equilibria are exchangeable since the equilibria of the two-player zero-sum game are.

We will give an algebraic characterization of strategic equivalence later on. This is accomplished by showing that a stronger notion of strategic equivalence, called correlated strategic equivalence (CSE), holds if and only if strategic equivalence holds. In CSE, $-i$ may correlate their strategies; this correlation will become useful in showing the algebraic characterization.

Definition 6.1.2 (Correlated strategic equivalence)**.** G is correlated strategically equivalent (CSE) to G' for player i if $\forall \mu_{-i} \in \Delta(\mathrm{P}_{-i}), s_i, s'_i \in S_i$ we have

$$\mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i(s'_i, \rho_{-i})] \geqslant \mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i(s_i, \rho_{-i})]$$

$$\iff$$

$$\mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u'_i(s'_i, \rho_{-i})] \geqslant \mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u'_i(s_i, \rho_{-i})].$$

If a game is SE (or CSE) for *all* players to another game, we say these two games are simply SE (or CSE).

Before moving forward, it is useful to express a game in matrix form. Let U_i be a $|\mathrm{P}_i| \times |\mathrm{P}_{-i}|$ payoff matrix of player i. A strategy for i is any vector $x \in \mathbb{R}^{|\mathrm{P}_i|}_{\geqslant 0}$ such that $\sum_j (x)_j = 1$. A (potentially) correlated strategy profile for $-i$ is any vector $y \in \mathbb{R}^{|\mathrm{P}_{-i}|}_{\geqslant 0}$ such that $\sum_j (y)_j = 1$. Given x and y, the payoff to x is then xU_iy.

The following lemma proves the equivalence between SE and CSE.

Lemma 6.1.3. *G and G' are SE for player i if and only they are CSE for player i.*

Proof. If G and G' are CSE for i, then they are clearly SE for i, since any $s_{-i} \in \Delta(\mathrm{P}_{-i})$. Next suppose that G and G' are SE for i, we show this implies G and G' are CSE for i. Assume for the sake of contradiction that G and G' are SE for i and G and G' are not CSE for i. Then, $\exists, \mu_{-i} \in \Delta(\mathrm{P}_{-i}), s_i, s_i' \in S_i$ such that

$$\mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i(s_i', \rho_{-i})] \geqslant \mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i(s_i, \rho_{-i})]$$

and

$$\mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i'(s_i', \rho_{-i})] < \mathbb{E}_{\rho_{-i} \sim \mu_{-i}}[u_i'(s_i, \rho_{-i})].$$

We may express these two equations in matrix-form as follows:

$$x^2 U_i y \geqslant x^1 U_i y \tag{6.1}$$

$$x^2 U_i' y < x^1 U_i' y, \tag{6.2}$$

where x^1 and x^2 are the vectors corresponding to s and s', respectively and y is vector corresponding to μ_{-i}. Rearranging, we have:

$$x^2 U_i y - x^1 U_i y \geqslant 0 \tag{6.3}$$

$$x^2 U_i' y - x^1 U_i' y < 0. \tag{6.4}$$

Then,

$$(x^2 - x^1) U_i y \geqslant 0 \tag{6.5}$$

$$(x^2 - x^1) U_i' y < 0. \tag{6.6}$$

Let

$$a \doteq (x^2 - x^1) U_i \in \mathbb{R}^{|\mathrm{P}_{-i}|}$$

$$b \doteq (x^2 - x^1) U_i' \in \mathbb{R}^{|\mathrm{P}_{-i}|},$$

Then (6.5) and (6.6) may be written as:

$$a^\top y \geqslant 0 \tag{6.7}$$

$$b^\top y < 0. \tag{6.8}$$

Next, consider the following two statements:

$$a^\top y \geqslant 0 \text{ s.t. } [b^\top y < 0] \wedge [y \geqslant 0] \tag{6.9}$$

$$\exists c \geqslant 0 \text{ s.t. } ac \leqslant b. \tag{6.10}$$

(6.7) and (6.8) imply (6.9) holds true. By Farkas' lemma (6.10) cannot hold, which means $\forall c \geqslant 0$, we have $ac > b$. Consider each element $(a)_j$, there are 3 cases.

1. $(a)_j > 0$
2. $(a)_j < 0$
3. $(a)_j = 0$

1. if $(a)_j > 0$, then $(b)_j < 0$, otherwise we could find a $c \geqslant 0$ such that $c(a)_j \leqslant (b)_j$.

2. Suppose first that $(a)_j < 0$ and $(b)_j \geqslant 0$, then taking $c = 0$ would give $c(a)_j \leqslant (b)_j$, a contradiction. Likewise if $(a)_j < 0$ and $(b)_j < 0$, then we could find a c to make $c(a)_j \leqslant (b)_j$ hold true.

3. if $(a)_j = 0$, then we must have $(b)_j < 0$.

Thus, we have $a \geqslant 0$ and $b < 0$. Recalling the definition of a and b, we have:

$$(x^2 - x^1)U_i \geqslant 0$$
$$(x^2 - x^1)U_i' < 0$$

Take any standard basis vector e_j of $\mathbb{R}^{|\mathrm{P}_{-i}|}$. e_j corresponds to $-i$ playing a pure strategy profile ρ^j_{-i}. Then,

$$(x^2 - x^1)U_i e_j \geqslant 0$$
$$(x^2 - x^1)U_i' e_j < 0.$$

Rearranging,

$$x^2 U_i e_j \geqslant x^1 U_i e_j$$
$$x^2 U_i' e_j < x^1 U_i' e_j$$

Which contradicts the assumption that G and G' are SE, since we have found a strategy of $-i$, ρ^j_{-i} where s'_i weakly preferred in G, yet s_i is strictly preferred in G'. □

Thus, CSE implies SE and SE implies CSE. This means that if two n-player games are SE for i, they are also CSE for i. This lets us generalize Theorem 4 of Moulin and Vial [46] to derive an algebraic characterization of strategic equivalence for n-player games. The algebraic characterization falls into two categories: equivalently trivial and algebraic equivalence. We give these two definitions next.

Let $\hat{\mathcal{U}}_i$ be the set of matrices of size $|\mathrm{P}_i| \times |\mathrm{P}_{-i}|$ such that each element $\hat{U}_i \in \hat{\mathcal{U}}_i$ is a matrix with identical entries in each column.

Definition 6.1.4 (Equivalently trivial)**.** G and G' are equivalently trivial for player i if there exists $\alpha \in \mathbb{R}^{|\mathrm{P}_i|}$ and $\beta_1, \beta_2 \in \mathbb{R}^{|\mathrm{P}_{-i}|}_{\geqslant 0}$ such that $U_i = \alpha\beta_1^\top + \hat{U}_1$ and $U'_i = \alpha\beta_2^\top + \hat{U}_2$ where $\hat{U}_1, \hat{U}_2 \in \hat{\mathcal{U}}_i$ and $(\beta_1)_j = 0 \iff (\beta_2)_j = 0$

Definition 6.1.5 (Algebraically equivalent.)**.** We say G and G' are algebraically equivalent for player i if $\exists \lambda_i > 0$ and $\hat{U}_i \in \hat{\mathcal{U}}_i$ such that

$$U_i = \lambda_i U'_i + \hat{U}_i, \tag{6.11}$$

In strategic form, (6.11) is given as

$$u_i(\rho) = \lambda_i u'_i(\rho) + \hat{u}_i(\rho_{-i}) \quad \forall \rho \in \mathrm{P}, \tag{6.12}$$

The following theorem, based on Theorem 4 of Moulin and Vial [46], uses equivalent triviality and algebraic equivalence to characterize strategic equivalence. The idea is that SE implies CSE, which means we can take two n-player games which are SE and produce two 2 player games which are CSE, and then apply the original two-player theorem of Moulin and Vial [46]. The full proof is given in Appendix C.

Theorem 6.1.6. *G and G' are SE for player i if and only if G and G' are equivalently trivial for player i, or G and G' are algebraically equivalent for player i.*

6.2 CCE Are Preserved by Strategic Equivalence

We know from the definition of SE that if two games are SE, they share a set of Nash equilibria. Does the same hold for CCE? The answer is yes—our next result shows that any CCE of a game G is also a CCE of G' if G and G' are SE.

Theorem 6.2.1. *Let G and G' be SE, then any CCE of G is a CCE of G'.*

Proof. Consider player $i \in N$.

Case 1. Consider first if G and G' are algebraically trivial. Writing out the definition of CCE, we have:

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant 0 \quad \forall \rho_i' \in \mathrm{P}_i. \tag{6.13}$$

By Theorem C.0.6 we have a scalar $\lambda_i > 0$ s.t.

$$\mathbb{E}_{\rho\sim\mu}\left[\lambda_i u_i'(\rho_i', \rho_{-i}) + \hat{u}_i(\rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[\lambda_i u_i'(\rho) + \hat{u}_i(\rho_{-i})\right] \leqslant 0 \quad \forall \rho_i' \in \mathrm{P}_i.$$

By linearity of expectation, we have

$$\begin{aligned}&\lambda_i \mathbb{E}_{\rho\sim\mu}\left[u_i'(\rho_i', \rho_{-i})\right] + \mathbb{E}_{\rho\sim\mu}\left[\hat{u}_i(\rho_{-i})\right] \\ &- \lambda_i \mathbb{E}_{\rho\sim\mu}\left[u_i'(\rho)\right] - \mathbb{E}_{\rho\sim\mu}\left[\hat{u}_i(\rho_{-i})\right] \leqslant 0 \quad \forall \rho_i' \in \mathrm{P}_i.\end{aligned}$$

Then, cancelling out like terms:

$$\lambda_i \mathbb{E}_{\rho\sim\mu}\left[u_i'(s_i', \rho_{-i})\right] - \lambda_i \mathbb{E}_{\rho\sim\mu}\left[u_i'(\rho)\right] \leqslant 0 \quad \forall s_i \in S_i.$$

Which implies

$$\mathbb{E}_{\rho\sim\mu}\left[u_i'(s_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i'(\rho)\right] \leqslant \frac{0}{\lambda_i} = 0 \quad \forall s_i \in S_i.$$

Case 2. Next, consider the case where G and G' are equivalently trivial. Let P_i^* be the set of dominant pure strategies in both G and G'. The fact that P_i^* is non-empty and the set of dominant pure strategies is the same in both G and G' follows from the fact that all columns in U_i and U_i' are equal to α scaled by

some non-negative number. Additionally, note that $u_i(\rho_i^*, \rho_{-i}) = u_i(\rho_i^{**}, \rho_{-i})$ and $u_i'(\rho_i^*, \rho_{-i}) = u_i'(\rho_i^{**}, \rho_{-i})$ for any two $\rho_i^{**}, \rho_i^* \in \mathrm{P}_i^*$.

For any distribution over pure strategies μ', we have

$$\mathbb{E}_{\rho\sim\mu'}\left[u_i(\rho_i^*, \rho_{-i})\right] \geqslant \mathbb{E}_{\rho\sim\mu'}\left[u_i(\rho)\right] \quad \forall \rho_i^* \in \mathrm{P}_i^* \tag{6.14}$$

Since

$$\begin{aligned}\mathbb{E}_{\rho\sim\mu'}\left[u_i(\rho)\right] &= \sum_{\rho\in\mathrm{P}} \mu(\rho) u_i(\rho) \\ &\leqslant \sum_{\rho\in\mathrm{P}} \mu(\rho) u_i(\rho_i^*, \rho_{-i}) \\ &= \mathbb{E}_{\rho\sim\mu'}\left[u_i(\rho_i^*, \rho_{-i})\right]\end{aligned}$$

Combining (6.14) and (6.13), we have

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i^*, \rho_{-i})\right] = \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \quad \forall \rho_i^* \in \mathrm{P}_i^* \tag{6.15}$$

Note that (6.15) holds only if μ only puts non-zero probability on strategies profiles where $\rho_i \in \mathrm{P}_i^*$. Let P^* be the set of pure strategy profiles where $\rho_i \in \mathrm{P}_i^*$. Assume for the sake of contradiction that $\exists \rho \notin \mathrm{P}^*$ such that $\mu(\rho) > 0$. Then

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] = \sum_{\rho^*\in\mathrm{P}^*} \mu(\rho) u_i(\rho_i^*, \rho_{-i}) + \sum_{\rho\in\mathrm{P}\backslash\mathrm{P}^*} \mu(\rho) u_i(\rho_i, \rho_{-i}) \tag{6.16}$$

$$< \sum_{\rho^*\in\mathrm{P}^*} \mu(\rho) u_i(\rho_i^*, \rho_{-i}) + \sum_{\rho\in\mathrm{P}\backslash\mathrm{P}^*} \mu(\rho) u_i(\rho_i^*, \rho_{-i}) \tag{6.17}$$

since $\exists \rho \in \mathrm{P}\backslash\mathrm{P}^* : \mu(\rho) > 0$. Continuing from (6.17),

$$\begin{aligned}&= \sum_{\rho\in\mathrm{P}} \mu(\rho) u_i(\rho_i^*, \rho_{-i}) \\ &= \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i^*, \rho_{-i})\right]\end{aligned}$$

This gives us the following inequality:

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] < \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i^*, \rho_{-i})\right]$$

which contradicts (6.15). We conclude μ only puts positive probability on elements of P^*. Then in G' we have that for any deviation ρ_i'

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i^*, \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \quad \forall \rho_i^* \in \mathrm{P}_i^*$$

Since $\rho_i^* \in \mathrm{P}_i^*$ are dominant pure strategies in G'. Then, for any $\rho_i^* \in \mathrm{P}_i^*$,

$$\begin{aligned}
&\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i^*, \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \\
=&\mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i^*)\beta'(\rho_{-i}) + \hat{u}(\rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i)\beta'(\rho_{-i}) + \hat{u}(\rho_{-i})\right] \\
=&\mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i^*)\beta'(\rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i)\beta'(\rho_{-i})\right]
\end{aligned}$$

But μ only puts positive probability on elements of P^*, which all have the same corresponding value of α. This means for any deviation ρ_i',

$$\begin{aligned}
\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant&\mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i^*)\beta'(\rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[\alpha(\rho_i^*)\beta'(\rho_{-i})\right] \\
=&0
\end{aligned}$$

Thus μ is also a CCE of G'. □

Thus, if two games are strategically equivalent, then they also share the same set of CCE.

6.3 Strategic Equivalence to CSP Games

Not all games have an exact CSP decomposition (i.e. are 0-CSP), yet some of these games may nonetheless behave much like CSP games. We next consider this class of games—games that are SE to CSP and SS-CSP games. Unlike Cai and Daskalakis [12], who study polymatrix games where each subgame is strictly competitive[1], we consider games which are *strategically equivalent* to CSP games. This distinction is important: computing a Nash equilibrium of Cai and Daskalakis' category of games is PPAD complete, whereas in games that are SE to CSP games, they may be computed in polynomial time. This is because games that are SE to CSP games share the CSP games' equilibria, which may be computed in polynomial time [11].

We first show that the set of games which are SE to a CSP game yet are not CSP themselves is non-empty, then give guarantees for these games.

[1]Strictly competitive games [3] are games whose payoff matrices are affine transformations of two-player zero-sum games [2]. Strictly competitive games are a subset of games that are strategically equivalent to two-player zero-sum games [46]

Proposition 6.3.1. *There are games which are SE to CSP games, that are not CSP themselves.*

Proof. Consider the modified version of Matching Pennies, shown in Figure 6.1. A matrix player M chooses a matrix for row and column to play. The matrix player receives payoff 0 in all outcomes, we omit them from Figure 6.1. This game is not a polymatrix game, since row's utility depend on a non-additive function of M and column's strategies. However, this game is strategically equivalent to Matching Pennies with a dummy player, which is a CSP game. □

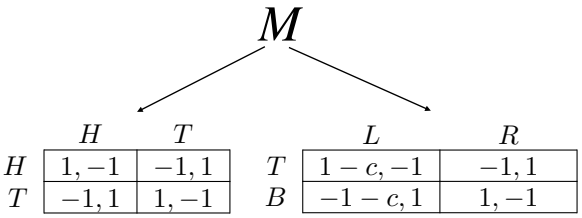


Figure 6.1: Modified Matching Pennies, a game that is strategically equivalent to a constant-sum polymatrix game, yet is itself not a constant-sum polymatrix game.

In games that are SE to approximate CSP games, we can establish guarantees on self-play by showing properties of the marginal strategies of CCE. First, we show that in games that are SE to a δ-CSP game, the marginals of CCE are approximate Nash equilibria, with the level of approximation depending on δ. This result generalizes Proposition 3.4.2 to games that are merely SE to a δ-CSP game, rather than being δ-CSP themselves. We introduce δ once more to allow these results to apply to any game. Note that a game G is SE to a CSP game if $\delta = 0$.

Theorem 6.3.2. *If G and G' are SE for player i and G' is δ-CSP, then for any marginal strategy profile s^μ of a CCE of G, i can gain at most $2(n+1)\delta\psi$*

utility in G by deviating, where ψ is a constant defined by

$$\psi = \begin{cases} \lambda_i & \text{if } G \text{ and } G' \text{ are algebraically equivalent} \\ \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j} & \text{if } G \text{ and } G' \text{ are equivalently trivial and } \beta \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

Proof. By Theorem 6.2.1, any CCE of G is a CCE of G'.

Case 1. Consider first the case where G and G' are algebraically equivalent. Proposition 3.4.2 implies the marginal strategy profile s^μ is a $2(n+1)\delta$-Nash of G' since G' is a δ-CSP game:

$$u_i'(s_i', s_{-i}^\mu) - u_i'(s^\mu) \leqslant 2(n+1)\delta \quad \forall s_i' \in S_i. \tag{6.18}$$

Since G' is NT-SE to G, we have

$$\left(\frac{1}{\lambda_i} u_i(s_i', s_{-i}^\mu) - \hat{u}_i(s_{-i}^\mu)\right) - \left(\frac{1}{\lambda_i} u_i(s^\mu) - \hat{u}_i(s_{-i}^\mu)\right) \leqslant 2(n+1)\delta \quad \forall s_i' \in S_i.$$

Since $u_i(s) = \lambda_i u_i'(s) + \hat{u}_i(s) \implies u_i'(s) = \frac{1}{\lambda_i} u_i(s) - \hat{u}_i(s)$. By cancelling out like terms, this gives us

$$\frac{1}{\lambda_i} u_i(s_i', s_{-i}^\mu) - \frac{1}{\lambda_i} u_i(s^\mu) \leqslant 2(n+1)\delta \quad \forall s_i' \in S_i,$$

and finally,

$$u_i(s_i', s_{-i}^\mu) - u_i(s^\mu) \leqslant 2(n+1)\delta\lambda_i \quad \forall s_i' \in S_i.$$

Case 2. Now consider if G and G' are equivalently trivial. We proceed with the matrix representation Let x^μ be the vector form of i's marginal strategy and y^μ be $-i$'s marginal strategy profile. Then we may express (6.18) as

$$x^\top U_i' y^\mu - x^{\mu\top} U_i' y^\mu \leqslant 2(n+1)\delta \quad \forall x \in X_i \tag{6.19}$$

$$\implies (x^\top \alpha)(\beta'^\top y^\mu) - (x^{\mu\top}\alpha)(\beta'^\top y^\mu) \leqslant 2(n+1)\delta \quad \forall x \in X_i \tag{6.20}$$

Recall that $(\beta)_j = 0 \iff (\beta')_j = 0$, which means if $\beta = 0$ then $\beta' = 0$. We consider 3 cases: (1) $\beta = \beta' = 0$, (2) $\beta, \beta' \neq 0$ and $\beta^\top y^\mu = 0$ and (3) $\beta, \beta' \neq 0$ and $\beta^\top y^\mu > 0$.

In case (1), clearly i has no incentive to deviate since

$$(x^\top \alpha)(\beta^\top y^\mu) - (x^{\mu\top}\alpha)(\beta^\top y^\mu) = (x^\top \alpha)0 - (x^{\mu\top}\alpha)0 = 0 \quad \forall x \in X_i$$

In case (2), y^μ only has support on 0 entries in β, β' since $\beta^\top y^\mu \geqslant 0$. Let $\psi \doteq \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j}$ be the maximum ratio between the non-zero elements of β and β'. We know this ratio is well-defined since we are not in case (1). Then,

$$(x^\top \alpha)(\beta^\top y^\mu) - (x^{\mu\top}\alpha)(\beta^\top y^\mu) = (x^\top \alpha)0 - (x^{\mu\top}\alpha)0 = 0 \leqslant 2(n+1)\delta\psi \quad \forall x \in X_i$$

For case (3) we may multiply the left side of (6.20) by $\frac{(\beta^\top y^\mu)}{(\beta^\top y^\mu)}$, giving us

$$\frac{(\beta^\top y^\mu)}{(\beta^\top y^\mu)} \left((x^\top \alpha)(\beta'^\top y^\mu) - (x^{\mu\top}\alpha)(\beta'^\top y^\mu)\right) \leqslant 2(n+1)\delta \quad \forall x \in X_i$$

Rearranging, we get

$$\begin{aligned}(x^\top \alpha)(\beta^\top y^\mu) - (x^{\mu\top}\alpha)(\beta^\top y^\mu) &\leqslant \frac{2(n+1)\delta(\beta^\top y^\mu)}{\beta'^\top y^\mu} \quad \forall x \in X_i \\ &\leqslant 2(n+1)\delta\psi \quad \forall x \in X_i\end{aligned}$$

where again $\psi \doteq \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j}$ is the maximum ratio between the non-zero elements of β and β'.

Summarizing, if $\beta = \beta' = 0$, set $\psi \doteq 0$ otherwise let $\psi \doteq \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j}$. Then i has at most $2(n+1)\delta\psi$ incentive to deviate from s^μ. Clearly in case (2) our bounds are not tight. □

Thus, in games are the SE to δ-CSP games, the marginals of CCE are approximate Nash equilibria, with the level of approximation depending on δ. This means self-play will produce approximate Nash equilibria in these games.

6.3.1 Strategic Equivalence to SS-CSP Games

In Chapter 3 we saw that SS-CSP games have both exchangeability and vulnerability guarantees. What can be said for games that are SE to a SS-CSP game? It turns out we can give exchangeability guarantees, but not vulnerability guarantees. In fact, we do not even have vulnerability guarantees for games that are SE to two-player zero-sum games (a subset of strategically SS-CSP games). Consider the two-player zero-sum game in Figure 6.2a and a

game that is strategically equivalent to it in Figure 6.2b. Since they are SE, they have the same set of Nash equilibria; thus (T, R) is a Nash equilibrium of both games. However, in Figure 6.2b row can lose c utility if the column player deviates to L.

	L	R
T	$0, 0$	$0, 0$
B	$1, -1$	$-1, 1$

(a)

	L	R
T	$-c, 0$	$0, 0$
B	$1 - c, -1$	$-1, 1$

(b)

Figure 6.2: Two games that are strategically equivalent: (a) is a two-player zero-sum game and (b) is a strategically equilvant game. (T, R) is a Nash equilibrium of both games, yet in (b) it has vulnerability c for the row player.

Additionally, not all equilibria of strategically two-player zero-sum games have the same value. Consider the game "Matching Pennies with Opt-Out", shown in Figure 6.3a. This game is identical to Matching Pennies except each player can now choose an opt-out action O, where both players get payoff of 0. Note that if both players uniformly randomize between H and T, this is a Nash equilibrium, as is (O, O). Matching Pennies with Opt-Out is strategically equivalent to the game show in 6.3b, yet the two aforementioned equilibria have different values for row in the second game.

	H	T	O
H	$1, -1$	$-1, 1$	$0, 0$
T	$-1, 1$	$1, -1$	$0, 0$
O	$0, 0$	$0, 0$	$0, 0$

(a)

	H	T	O
H	$1, -1$	$-1, 1$	$c, 0$
T	$-1, 1$	$1, -1$	$c, 0$
O	$0, 0$	$0, 0$	$c, 0$

(b)

Figure 6.3: Not all Nash equilibria of strategically two-player zero-sum games have the same value. The game in (b) is strategically equivalent to the game in (a), a two-player zero-sum game, but has two Nash equilibria with different values for row.

Fortunately, we can still establish guarantees on exchangeability in games that are SE to a SS-CSP game.

Proposition 6.3.3. *If G is SE for player i to a $(0, \gamma)$-SS-CSP game $\check{G}$, then for any combination of marginal strategies from CCE of G, i can gain at most*

$|E_i|\gamma\psi$ *utility in* G *by deviating, where* ψ *is a constant defined by*

$$\psi = \begin{cases} \lambda_i & \text{if } G \text{ and } G' \text{ are algebraically equivalent} \\ \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j} & \text{if } G \text{ and } G' \text{ are equivalently trivial and } \beta \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

Proof. The proof is similar to Theorem 6.3.2. First we consider the case where G and G' are algebraically equivalent.

Case 1. Let s be a strategy profile such that $\forall i \in N$, s_i is the marginal strategy from some CCE μ^i of G. Note that each marginal strategy profile s^{μ^i} is a Nash equilibrium of both G and G' since G is SE to a G', a 0-CSP game, so we may apply Theorem 6.3.2.

By $(0, \gamma)$-subgame stability, each (s_i, s_j) is a γ-Nash of $\check{G}_{ij}$, a subgame of $\check{G}$, since they come from (potentially different) marginal strategy profiles that are both Nash equilibria. This gives us

$$\begin{aligned} & \check{u}_i(s_i', s_{-i}) - \check{u}_i(s_i, s_{-i}) \leqslant |E_i|\gamma \quad \forall s_i' \in S_i \\ \implies & \left(\frac{1}{\lambda_i} u_i(s_i', s_{-i}) - \hat{u}_i(s_{-i})\right) - \left(\frac{1}{\lambda_i} u_i(s_i, s_{-i}) - \hat{u}_i(s_{-i})\right) \leqslant |E_i|\gamma \quad \forall s_i' \in S_i \\ \implies & \left(\frac{1}{\lambda_i} u_i(s_i', s_{-i})\right) - \left(\frac{1}{\lambda_i} u_i(s_i, s_{-i})\right) \leqslant |E_i|\gamma \quad \forall s_i' \in S_i \\ \\ \implies & u_i(s_i', s_{-i}) - u_i(s_i, s_{-i}) \leqslant |E_i|\gamma\lambda_i \quad \forall s_i' \in S_i. \end{aligned}$$

Case 2. Next, we consider the case where G and G' are equivalently trivial. We again use the matrix form, let x, y be the vector-form strategy and strategy profile for s_i and s_{-i}, respectively. Since G' is subgame stable, we have

$$x'^{\top} U_i' y - x U_i' y \leqslant |E_i|\gamma \quad \forall x' \in X_i \tag{6.21}$$

$$(x'^{\top}\alpha)(\beta'^{\top} y) - (x^{\top}\alpha)(\beta'^{\top} y) \leqslant |E_i|\gamma \quad \forall x' \in X_i \tag{6.22}$$

We again consider 3 cases: (1) $\beta = \beta' = 0$, (2) $\beta, \beta' \neq 0$ and $\beta^{\top} y^{\mu} = 0$ and (3) $\beta, \beta' \neq 0$ and $\beta^{\top} y^{\mu} > 0$. In (1), i cannot gain by deviating since

$$(x'^{\top}\alpha)(\beta^{\top} y) - (x^{\top}\alpha)(\beta^{\top} y) = (x'^{\top}\alpha)0 - (x^{\top}\alpha)0 = 0 \quad \forall x' \in X_i$$

In case (2), we also have that i has no incentive to deviate. Let $\psi \doteq \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j}$ be the maximum ratio between the non-zero elements of β and β'. Then,

$$(x'^{\top}\alpha)(\beta^{\top}y) - (x^{\top}\alpha)(\beta^{\top}y) \leqslant |E_i|\gamma\psi \quad \forall x' \in X_i$$

For case (3) we multiply the left side of (6.22) by $\frac{(\beta^{\top}y)}{(\beta^{\top}y)}$.

$$\frac{(\beta^{\top}y)}{(\beta^{\top}y)}(x'^{\top}\alpha)(\beta'^{\top}y) - (x^{\top}\alpha)(\beta'^{\top}y) \leqslant |E_i|\gamma \quad \forall x \in X_i \tag{6.23}$$

Rearranging, we get

$$(x'^{\top}\alpha)(\beta^{\top}y - (x^{\top}\alpha)(\beta^{\top}y) \leqslant |E_i|\gamma\frac{(\beta^{\top}y)}{(\beta'^{\top}y)} \leqslant |E_i|\gamma\psi \quad \forall x \in X_i$$

where again $\psi \doteq \max_{j:(\beta)_j \neq 0} \frac{(\beta)_j}{(\beta')_j}$ is the maximum ratio between the non-zero elements of β and β. □

Proposition 6.3.3 tells us if an agent trained in self-play plays against agents from other training instances, they will boundedly regret their strategies.

Chapter 7

Additional Results

7.1 Aligned Games

Where does subgame stability come from? Are there other properties of polymatrix games that imply subgame stability? We identify one such example, which we call *aligned games*. Intuitively, these are games where global behaviour is "aligned" with local behaviour for all strategies. While subgame-stability is a property that guarantees global equilibria continue to be equilibria in each subgame, alignment means that the regret of an agent is aligned globally and at all subgames for *any* opponent's strategy; it is a stronger property than subgame stability.

Let $R_{ij}(\rho_i', (s_i, s_j))$ be the regret for player i in their subgame against player j: $R_{ij}(\rho_i', (s_i, s_j)) \doteq u_{ij}(\rho_i', s_j) - u_i(s_i, s_j)$ For ease of notation, let $R_{ij}(\rho_i', s) \doteq R_{ij}(\rho_i', (s_i, s_j))$. We define β-aligned with a definition that uses pure strategies, then show that this implies alignment holds for mixed strategies as well.

Definition 7.1.1 (Aligned game)**.** We say a polymatrix game is β-aligned for player i if $\forall \rho \in \mathrm{P}, \rho_i' \in \mathrm{P}_i, j \neq k \in -i$ we have $|R_{ij}(\rho_i', \rho) - \alpha_{ik} R_{ik}(\rho_i', \rho)| \leqslant \beta$ for some $\alpha_{ik} > 0$, $\beta \geqslant 0$.

Note that the regret of an agent i w.r.t. a mixed strategy s is a convex

combination of the regret w.r.t. each pure strategy:

$$\begin{aligned} R_i(\rho_i', s) &= u_i(\rho_i', s_{-i}) - u_i(s) \\ &= \sum_{\rho \in \mathrm{P}} s(\rho) u_i(\rho_i', \rho_{-i}) - \sum_{\rho \in \mathrm{P}} s(\rho) u_i(\rho) \\ &= \sum_{\rho \in \mathrm{P}} s(\rho) R_i(\rho_i', \rho), \end{aligned}$$

where $s(\rho) = \prod_{i \in N} s_i(\rho_i)$. Additionally note that for subgame G_{ij} we have

$$R_{ij}(\rho_i', s) = \sum_{\rho \in \mathrm{P}} s(\rho) R_{ij}(\rho_i', \rho),$$

since

$$\begin{aligned} R_{ij}(\rho_i', s) &= \sum_{\rho_i \in \mathrm{P}_i} \sum_{\rho_j \in \mathrm{P}_j} s_i(\rho_i) s_j(\rho_j) R_{ij}(\rho_i', (\rho_i, \rho_j)) \\ &= \sum_{\rho_{-ij} \in \mathrm{P}_{-ij}} s_{-ij}(\rho_{-ij}) \sum_{\rho_i \in \mathrm{P}_i} \sum_{\rho_j \in \mathrm{P}_j} s_i(\rho_i) s_j(\rho_j) R_{ij}(\rho_i', (\rho_i, \rho_j)) \\ &= \sum_{\rho_{-ij} \in \mathrm{P}_{-ij}} \sum_{\rho_i \in \mathrm{P}_i} \sum_{\rho_j \in \mathrm{P}_j} s_{-ij}(\rho_{-ij}) s_i(\rho_i) s_j(\rho_j) R_{ij}(\rho_i', (\rho_i, \rho_j)) \\ &= \sum_{\rho \in \mathrm{P}} s(\rho) R_{ij}(\rho_i', \rho). \end{aligned}$$

Corollary 7.1.2. *If a game is β-aligned, then $\forall s \in S, \rho_i' \in \mathrm{P}_i, j \neq k \in -i$ we have $|R_{ij}(\rho_i', s) - \alpha_{ik} R_{ik}(\rho_i', s)| \leqslant \beta$.*

Proof.

$$\begin{aligned} & |R_{ij}(\rho_i', s) - \alpha_{ik}\ R_{ik}(\rho_i', s)| && (7.1) \\ =& \left| \sum_{\rho \in \mathrm{P}} s(\rho) R_{ij}(\rho_i', \rho) - \alpha_{ik} \sum_{\rho \in \mathrm{P}} s(\rho) R_{ik}(\rho_i', \rho) \right| && (7.2) \\ =& \left| \sum_{\rho \in \mathrm{P}} s(\rho) \left(R_{ij}(\rho_i', \rho) - \alpha_{ik} R_{ik}(\rho_i', \rho) \right) \right| && (7.3) \\ \leqslant& \max_{\rho \in \mathrm{P}} |R_{ij}(\rho_i', \rho) - \alpha_{ik} R_{ik}(\rho_i', \rho)| && (7.4) \\ \leqslant& \beta && (7.5) \end{aligned}$$

Where (7.4) is because $s(\rho) \in [0, 1]$ and $\sum_{\rho \in \mathrm{P}} s(\rho) = 1$. □

If a game is β-aligned, then this implies approximate subgame stability.

Proposition 7.1.3. *If G is β-aligned, any ϵ-Nash equilibrium s is a $\frac{\epsilon+|E_i|\beta}{\alpha_i}$-Nash of each subgame G_{ik} where $\alpha_i = \sum_{(i,j)\in E_i} \alpha_{ij}$.*

Proof. Since s is an ϵ-Nash, we have $R_i(\rho_i', s) \leqslant \epsilon \ \forall \rho_i' \in \mathrm{P}_i$. G is polymatrix, so $R_i(\rho_i', s) = \sum_{(i,j)\in E_i} R_{ij}(\rho_i', s)$. Consider subgame G_{ik}—we have for each other subgame for i G_{ij}, where $j \neq k$ that $|R_{ik}(\rho_i', s) - \alpha_{ij} R_{ij}(\rho_i', s)| \leqslant \beta \ \forall \rho_i' \in \mathrm{P}_i$, which means we can write $R_i(\rho_i', s)$ as a weighted sum of $R_{ik}(\rho_i', s)$ plus some error $\hat{\beta}_{ij}$:

$$R_i(\rho_i', s) = \sum_{(i,j)\in E_i} \alpha_{ij} \left(R_{ik}(\rho_i', s) + \hat{\beta}_{ij} \right) \quad \forall \rho_i' \in \mathrm{P}_i.$$

Since s is a Nash equilibrium,

$$\begin{aligned}
&\sum_{(i,j)\in E_i} \alpha_{ij} R_{ik}(\rho_i', s) + \hat{\beta}_{ij} \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i \\
\implies &\left(R_{ik}(\rho_i', s) \sum_{(i,j)\in E_i} \alpha_{ij} + \sum_{(i,j)\in E_i} \hat{\beta}_{ij} \right) \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i \\
\implies &R_{ik}(\rho_i', s)\alpha_i - |E_i|\beta \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i \\
\implies &R_{ik}(\rho_i', s)\alpha_i \leqslant \epsilon + |E_i|\beta \quad \forall \rho_i' \in \mathrm{P}_i \\
\implies &R_{ik}(\rho_i', s) \leqslant \frac{\epsilon + |E_i|\beta}{\alpha_i} \quad \forall \rho_i' \in \mathrm{P}_i.
\end{aligned}$$

Where we use the fact that all $\alpha_{ij} > 0$, so $\alpha_i > 0$. The final equation implies $u_i(\rho_i', s_{-i}) - u_i(s) \leqslant \frac{\epsilon+|E_i|\beta}{\alpha_i}$ for any $\rho_i' \in \mathrm{P}_i$.

□

In summary, one possible cause of subgame stability is if a CSP game is aligned. This property might be easier to show for any given game than subgame stability since it holds everywhere, not just at equilibria.

7.2 Multi-player Minimax Games

We have seen that SS-CSP games generalize many of the properties of two-player zero-sum games to multi-player settings. We conclude this by showing that SS-CSP games are a particular instance of a broader class of games, which we call *multi-player minimax games*. We define this class by generalizing the

spirit of Aumann's notion of "strict competition" to the n-player setting. Two player strictly competitive games are games where "for each player, helping himself and hurting his opponent are equivalent" [3].

In the multi-player setting, we additionally require that players helping themselves jointly minimizes the utility of opponents. If this property holds at a strategy profile s, we say the game is *locally multi-player minimax* at s.

Definition 7.2.1 (Locally multi-player minimax)**.** Let $s \in S$ be a strategy profile. For each $j \in N$, let $s_j^* \in \arg\max_{s'_j \in S_j} u_j(s'_j, s_{-j})$. We say the game is *locally multi-player minimax* (LMM) at s if $\forall i \in N$

$$u_i(s_i, s^*_{-i}) = \min_{s'_{-i} \in S_{-i}} u_i(s_i, s'_{-i}). \tag{7.6}$$

If a game is LMM at the set of Nash equilibria, a generalized version of the minimax theorem holds.

Proposition 7.2.2. *If a game is LMM at the set of Nash equilibria S^* , then in any Nash equilibrium each player receives a payoff that is equal to both their maxmin and minmax value.*

Proof. Let $\bar{v}_i, \underline{v}_i$ be i's maxmin and minmax values, respectively. Let $s \in S^*$ be a Nash equilibrium and $v_i = u_i(s)$. Note that we cannot have $\bar{v}_i > v_i$, otherwise i would want to deviate to their maxmin strategy. Since s is a Nash equilibrium, each player plays $s_j^* \in \arg\max_{s'_j \in S_j} u_j(s'_j, s_{-j})$ and by LMM at s, we have

$$v_i = \min_{s'_{-i} \in S_{-i}} u_i(s_i, s'_{-i}) \leqslant \max_{s'_i \in S_i} \min_{s'_{-i} \in S_{-i}} u_i(s'_i, s'_{-i}) = \bar{v}_i.$$

Thus $v_i = \bar{v}_i$ since we cannot have $\bar{v}_i > v_i$. Next we show that $v_i = \underline{v}_i$. Note that we cannot have $\underline{v}_i < v_i$, otherwise $-i$ would not be jointly minimizing i's utility, which contradicts LMM. Then,

$$v_i = \max_{s'_i \in S_i} u_i(s'_i, s_{-i}) \geqslant \min_{s'_{-i} \in S_{-i}} \max_{s'_i \in S_i} u_i(s'_i, s'_{-i}) = \underline{v}_i,$$

which implies $v_i = \underline{v}_i$ since we cannot have $\underline{v}_i < v_i$.

□

Subgame stable CSP games are LMM $\forall s^{\mu} \in S^{\mu}$, since each agent in $-i$ is playing a best response in their subgame against i, which minimizes i's utility in that subgame. Moreover, $-i$ do not need to coordinate to minimize i's utility since i's utility decomposes amongst their subgames.

Chapter 8

Conclusion

Self-play has been incredibly successful in producing strategies that perform well against new opponents in two-player constant-sum games. Despite a lack of theoretical guarantees, self-play seems to also produce good strategies in some multi-player games. In this thesis, we have identified structural properties of multi-player, general-sum games that allow us to establish guarantees on the performance of strategies learned via self-play against new opponents. We show that any game can be projected into the space of constant-sum polymatrix (CSP) games, and if there exists a game within this set with high subgame stability (low γ), strategies learned through self-play have bounded exchangeability, bounded values and bounded loss of performance against new opponents. In normal-form games, CSP decompositions and checks for subgame stability can be done efficiently in polynomial time using linear programming. Subgame stable CSP games are a subset of locally multi-player minimax games. Our novel poly-EFG representation gives rise to an efficient algorithm for producing approximate subgame-stable CSP decompositions.

We conjecture that Texas hold 'em is one such game. We investigate this claim on Kuhn and Leduc poker, and find that CFR plays strategies from a nearly subgame stable CSP part of the strategy space within these games. Conversely, in a toy Hanabi game where the strategies learned in self-play does not perform well, the game is not approximately CSP and subgame stable.

We extend out results to cases where a game may not decompose into a CSP game, but is instead strategically equivalent to one. We give an algebraic

characterization for n-player games. Using this characterization, we find that the set of CCE are the same between strategically equivalent games. We also show that the marginal strategies of CCE in games that are strategically equivalent to approximate CSP games are approximate Nash equilibria. Lastly, in games that are strategically equivalent to approximate subgame stable CSP games, the marginal strategies of CCE are all approximately exchangeable.

Machine learning in general-sum, multi-player games is a challenging problem domain with great potential. Nearly all real-world multi-agent systems (such as driving coordination or stock trading) have elements of both cooperation and competition. Few involve only two agents. This thesis lays the groundwork for guarantees for self-play in general-sum multi-player games by studying multi-player games that behave in similar ways to two player zero-sum games. Our main results may be applied to any game, however there are surely ways to refine our work in particular instances. We hope that future work will deepen our understanding of self-play and when it is a desirable training procedure. Approximate SS-CSP games may not be the only class of games where self-play is guaranteed to perform well.

Self-play has been a workhorse for training machine learning systems in multi-agent settings. While we have shown that there are many cases where self-play is a desirable learning procedure outside of two-player constant-sum games, many settings will require fast and continual adaptation to new agents in the environment, rather than a fixed strategy being generated through self-play. Continual learning is not incompatible with self-play—indeed pre-training through self-play to produce a "blueprint strategy" has been used in combination with real-time search to produce super-human multi-player poker AI [10]. An exciting future direction is what blueprint strategies are good starting places for continual learning.

References

[1] N. Abou Risk and D. Szafron, "Using counterfactual regret minimization to create competitive multiplayer poker agents.," *International Conference on Autonomous Agents and Multiagent Systems*, 2010.

[2] I. Adler, C. Daskalakis, and C. H. Papadimitriou, "A note on strictly competitive games," in *Internet and Network Economics: 5th International Workshop, WINE 2009, Rome, Italy, December 14-18, 2009. Proceedings 5*, Springer, 2009, pp. 471–474.

[3] R. J. Aumann, "Almost strictly competitive games," *Journal of the Society for Industrial and Applied Mathematics*, vol. 9, no. 4, pp. 544–550, 1961.

[4] R. J. Aumann, "Subjectivity and correlation in randomized strategies," *Journal of Mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[5] N. Bard, J. N. Foerster, S. Chandar, *et al.*, "The hanabi challenge: A new frontier for ai research," *Artificial Intelligence*, 2020.

[6] L. Bergman and I. Fokin, "On separable non-cooperative zero-sum games," *Optimization*, vol. 44, no. 1, pp. 69–84, 1998.

[7] M. Bowling, N. Burch, M. Johanson, and O. Tammelin, "Heads-up limit hold'em poker is solved," *Science*, vol. 347, no. 6218, pp. 145–149, 2015.

[8] N. Brown, A. Lerer, S. Gross, and T. Sandholm, "Deep counterfactual regret minimization," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, 2019.

[9] N. Brown and T. Sandholm, "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals," *Science*, vol. 359, no. 6374, pp. 418–424, 2018.

[10] N. Brown and T. Sandholm, "Superhuman AI for multiplayer poker," *Science*, vol. 365, no. 6456, pp. 885–890, 2019.

[11] Y. Cai, O. Candogan, C. Daskalakis, and C. Papadimitriou, "Zero-sum polymatrix games: A generalization of minmax," *Mathematics of Operations Research*, vol. 41, no. 2, pp. 648–655, 2016.

[12] Y. Cai and C. Daskalakis, "On minmax theorems for multiplayer games," *ACM-SIAM Symposium on Discrete algorithms*, 2011.

[13] A. Celli, A. Marchesi, T. Bianchi, and N. Gatti, "Learning to correlate in multi-player general-sum sequential games," *Neural Information Processing Systems*, 2019.

[14] A. Celli, A. Marchesi, G. Farina, and N. Gatti, "No-regret learning dynamics for extensive-form correlated equilibrium," *Neural Information Processing Systems*, 2020.

[15] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou, "The complexity of computing a nash equilibrium," *SIAM Journal on Computing*, vol. 39, no. 1, pp. 195–259, 2009.

[16] C. Daskalakis and C. H. Papadimitriou, "Three-player games are hard," *Electron. Colloquium Comput. Complex.*, 2005.

[17] G. Farina, T. Bianchi, and T. Sandholm, "Coarse correlation in extensive-form games," *AAAI conference on Artificial Intelligence*, 2020.

[18] G. Farina, C. K. Ling, F. Fang, and T. Sandholm, "Correlation in extensive-form games: Saddle-point formulation and benchmarks," *Neural Information Processing Systems*, 2019.

[19] D. P. Foster and R. Vohra, "Regret in the on-line decision problem," *Games and Economic Behavior*, vol. 29, no. 1, pp. 7–35, 1999.

[20] D. Fudenberg and J. Tirole, *Game Theory* (MIT Press Books). The MIT Press, 1991.

[21] R. Gibson, M. Lanctot, N. Burch, D. Szafron, and M. Bowling, "Generalized sampling and variance in counterfactual regret minimization," 2012.

[22] R. G. Gibson, "Regret minimization in games and the development of champion multiplayer computer poker-playing agents," *Ph.D. Thesis*, 2014.

[23] A. Greenwald and A. Jafari, "A general class of no-regret learning algorithms and game-theoretic equilibria," in *COLT*, vol. 3, 2003.

[24] A. Greenwald, A. Jafari, and C. Marks, "No-Φ-regret: A connection between computational learning theory and game theory," *Games, Norms and Reasons: Logic at the Crossroads*, 2011.

[25] J. C. Harsanyi, R. Selten, *et al.*, "A general theory of equilibrium selection in games," *MIT Press Books*, 1988.

[26] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[27] H. Hu, A. Lerer, A. Peysakhovich, and J. Foerster, ""other-play" for zero-shot coordination," *International Conference on Machine Learning*, 2020.

[28] S.-H. Hwang and L. Rey-Bellet, "Strategic decompositions of normal form games: Zero-sum games and potential games," *Games and Economic Behavior*, vol. 122, pp. 370–390, 2020.

[29] C. Jin, Q. Liu, Y. Wang, and T. Yu, "V-learning–a simple, efficient, decentralized algorithm for multiagent rl," *arXiv preprint arXiv:2110.14555*, 2021.

[30] M. Johanson, N. Bard, M. Lanctot, R. Gibson, and M. Bowling, "Efficient nash equilibrium approximation through monte carlo counterfactual regret minimization," *International Conference on Autonomous Agents and Multiagent Systems*, 2012.

[31] M. Kearns, M. L. Littman, and S. Singh, *Graphical models for game theory*, 2013.

[32] H. W. Kuhn, "A simplified two-person poker," *Contributions to the Theory of Games*, vol. 1, pp. 97–103, 1950.

[33] H. W. Kuhn, "Extensive games and the problem of information," *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 193–216, 1953.

[34] M. Lanctot, E. Lockhart, J.-B. Lespiau, *et al.*, "Openspiel: A framework for reinforcement learning in games," *arXiv preprint arXiv:1908.09453*, 2019.

[35] M. Lanctot, K. Waugh, M. Zinkevich, and M. Bowling, "Monte carlo sampling for regret minimization in extensive games," *Neural Information Processing Systems*, 2009.

[36] M. Lanctot, V. Zambaldi, A. Gruslys, *et al.*, "A unified game-theoretic approach to multiagent reinforcement learning," *Neural Information Processing Systems*, 2017.

[37] Q. Liu, T. Yu, Y. Bai, and C. Jin, "A sharp analysis of model-based reinforcement learning with self-play," *International Conference on Machine Learning*, 2021.

[38] C. Luckhart and K. B. Irani, "An algorithmic solution of n-person games.," *AAAI Conference on Artificial Intelligence*, 1986.

[39] R. MacQueen and J. R. Wright, "Guarantees for self-play via polymatrix decomposability," *Neural Information Processing Systems*, 2023.

[40] L. Marris, P. Muller, M. Lanctot, K. Tuyls, and T. Graepel, "Multi-agent training beyond zero-sum with correlated equilibrium meta-solvers," *International Conference on Machine Learning*, 2021.

[41] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Independent reinforcement learners in cooperative markov games: A survey regarding coordination problems," *The Knowledge Engineering Review*, vol. 27, no. 1, pp. 1–31, 2012.

[42] M. Moravcik, M. Schmid, N. Burch, *et al.*, "Deepstack: Expert-level artificial intelligence in heads-up no-limit poker," *Science*, vol. 356, no. 6337, pp. 508–513, 2017.

[43] D. Morrill, "Hindsight rational learning for sequential decision-making: Foundations and experimental applications," *Ph.D. Thesis*, 2022.

[44] D. Morrill, R. D'Orazio, M. Lanctot, J. R. Wright, M. Bowling, and A. R. Greenwald, "Efficient deviation types and learning for hindsight rationality in extensive-form games," *International Conference on Machine Learning*, 2021.

[45] D. Morrill, R. D'Orazio, R. Sarfati, *et al.*, "Hindsight and sequential rationality of correlated play," *AAAI Conference on Artificial Intelligence*, 2021.

[46] H. Moulin and J. Vial, "Strategically zero-sum games: The class of games whose completely mixed equilibria cannot be improved upon," *International Journal of Game Theory*, vol. 7, no. 3, pp. 201–221, 1978.

[47] J. Nash, "Equilibrium points in n-person games," *Proceedings of the National Academy of Sciences*, 1950.

[48] P. Paquette, Y. Lu, S. S. Bocco, *et al.*, "No-press diplomacy: Modeling multi-agent gameplay," *Neural Information Processing Systems*, 2019.

[49] J. Perolat, B. D. Vylder, D. Hennes, *et al.*, "Mastering the game of stratego with model-free multiagent reinforcement learning," *Science*, vol. 378, no. 6623, pp. 990–996, 2022.

[50] M. Riazi-Kermani, *Orthogonal complement of vector of all 1's*, Math Stack Exchange, 2019.

[51] L. Samuelson, *Evolutionary games and equilibrium selection*. MIT press, 1997, vol. 1.

[52] M. Schmid, N. Burch, M. Lanctot, M. Moravcik, R. Kadlec, and M. Bowling, "Variance reduction in monte carlo counterfactual regret minimization (vr-mccfr) for extensive form games using baselines," *AAAI Conference on Artificial Intelligence*, 2019.

[53] R. Selten, "Reexamination of the perfectness concept for equilibrium points in extensive games," in *Models of Strategic Rationality*, Springer, 1988, pp. 1–31.

[54] Y. Shoham and K. Leyton-Brown, *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[55] Y. Shoham, R. Powers, and T. Grenager, "If multi-agent learning is the answer, what is the question?" *Artificial intelligence*, vol. 171, no. 7, pp. 365–377, 2007.

[56] D. Silver, A. Huang, C. J. Maddison, *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[57] D. Silver, T. Hubert, J. Schrittwieser, *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[58] E. Steinberger, A. Lerer, and N. Brown, "Dream: Deep regret minimization with advantage baselines and model-free learning," *arXiv preprint arXiv:2006.10410*, 2020.

[59] N. Sturtevant, "Current challenges in multi-player game search," *International Conference on Computers and Games*, 2004.

[60] N. Sturtevant and M. Bowling, "Robust game play against unknown opponents," 2006.

[61] N. Sturtevant, M. Zinkevich, and M. Bowling, "Prob-maxˆ n: Playing n-player games with opponent models," *AAAI Conference on Artificial Intelligence*, 2006.

[62] N. R. Sturtevant and R. E. Korf, "On pruning techniques for multi-player games," *AAAI Conference on Artificial Intelligence*, 2000.

[63] O. Tammelin, "Solving large imperfect information games using cfr+," *arXiv preprint arXiv:1407.5042*, 2014.

[64] O. Tammelin, N. Burch, M. Johanson, and M. Bowling, "Solving heads-up limit texas hold'em," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[65] O. Vinyals, I. Babuschkin, W. M. Czarnecki, *et al.*, "Grandmaster level in Starcraft II using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[66] J. von Neumann, "Zur theorie der gesellschaftsspiele," *Mathematische annalen*, vol. 100, no. 1, pp. 295–320, 1928.

[67] B. Von Stengel and F. Forges, "Extensive-form correlated equilibrium: Definition and computational complexity," *Mathematics of Operations Research*, vol. 33, no. 4, pp. 1002–1022, 2008.

[68] K. Waugh, D. Morrill, J. Bagnell, and M. Bowling, "Solving games with functional regret estimation," *AAAI Conference on Artificial Intelligence*, 2015.

[69] M. Zinkevich, M. Johanson, M. Bowling, and C. Piccione, "Regret minimization in games with incomplete information," *Neural Information Processing Systems*, 2008.

Appendix A

Proofs of Well-Known Results

This appendix gives proofs and examples for a few well-known results. We show them here since we could not find work in the literature showing them.

A.1 Hindsight Rationality With Respect to Action Deviations Does Not Imply Nash

Here we show that hindsight rationality with respect to action deviations does not imply Nash equilibrium in 2 player constant-sum games. We show this with a 1 player game. Consider the agents strategy, shown in blue, which receives utility of 1. Deviating to $[I_1 : b, I_2 : a]$ will increase the player's utility to 2, so the blue strategy is not a Nash equilibrium. However, this would require two simultaneous action deviations, one at I_1 to b and one at I_2 to a. Neither of these deviations increases the player's utility on their own, so the player is hindsight rational w.r.t. action deviations.

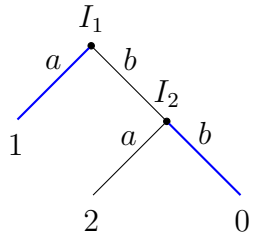


Figure A.1: Action deviations in a simple game.

A.2 CCE Imply Nash in Two Player Zero-Sum Games

Here we show a proof that CCE imply Nash in two-player zero-sum games. We begin showing that if a player i deviates from their CCE recommendations, if $-i$ continues to play their CCE recommendations, this is equivalent to $-i$ playing their marginal strategy for the CCE.

Lemma A.2.1. *If $\mu \in \Delta(\mathrm{P})$, then in a two-player game, for any deviation s_i, we have $\mathbb{E}_{\rho\sim\mu}[u_i(s_i, \rho_{-i})] = u_i(s_i, s^{\mu}_{-i})$.*

Proof. We have

$$\mathbb{E}_{\rho\sim\mu}[u_i(s_i, \rho_{-i})] = \sum_{\rho_i\in\mathrm{P}_i} \sum_{\rho_{-i}\in\mathrm{P}_{-i}} s_i(\rho_i)\mu_{-i}(\rho_{-i})u_i(\rho_i, \rho_{-i}),$$

where $\mu_{-i}(\rho_{-i}) \doteq \sum_{\rho_{-i}\in\mathrm{P}_{-i}} \mu(\rho_i, \rho_{-i})$. Note, however, that $\mu_{-i}(\rho_{-i}) = s^{\mu}_{-i}(\rho_{-i})$ so

$$\begin{aligned}\mathbb{E}_{\rho\sim\mu}[u_i(s_i, \rho_{-i})] &= \sum_{\rho_i\in\mathrm{P}_i} \sum_{\rho_{-i}\in\mathrm{P}_{-i}} s_i(\rho_i)s^{\mu}_{-i}(\rho_{-i})u_i(\rho_i, \rho_{-i}) \\ &= u_i(s_i, s^{\mu}_{-i}).\end{aligned}$$

□

Proposition A.2.2. *If μ is an ϵ-CCE of a two-player constant-sum game, then $|\mathbb{E}_{\rho\sim\mu}[u_i(\rho)] - u_i(s^{\mu})| \leqslant \epsilon$*

Proof. Suppose not, then $|\mathbb{E}_{\rho\sim\mu}[u_i(\rho)] - u_i(s^{\mu})| > \epsilon$ which means either

$$u_i(s^{\mu}) - \mathbb{E}_{\rho\sim\mu}[u_i(\rho)] > \epsilon \tag{A.1}$$

or

$$\mathbb{E}_{\rho\sim\mu}[u_i(\rho)] - u_i(s^{\mu}) > \epsilon. \tag{A.2}$$

Consider (A.1). By Lemma A.2.1 we have $u_i(s^{\mu}) = \mathbb{E}_{\rho\sim\mu}[u_i(s^{\mu}_i, \rho_{-i})]$, which means

$$\mathbb{E}_{\rho\sim\mu}[u_i(s^{\mu}_i, \rho_{-i})] - \mathbb{E}_{\rho\sim\mu}[u_i(\rho)] > \epsilon,$$

which contradicts the fact that μ is an ϵ-CCE, since s_i^μ is a deviation that is more than ϵ-profitable for player i. Next, consider (A.2); since the game is two-player zero-sum, we have:

$$\begin{aligned} &\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] - u_i(s^\mu) > \epsilon \\ \implies &-\mathbb{E}_{\rho\sim\mu}\left[u_{-i}(\rho)\right] - (-u_{-i}(s^\mu)) > \epsilon \\ \implies &u_{-i}(s^\mu) - \mathbb{E}_{\rho\sim\mu}\left[u_{-i}(\rho)\right] > \epsilon. \end{aligned}$$

At this point we may repeat the steps above to show that μ is not an ϵ-CCE, since s_{-i}^μ is a deviation that is more than ϵ-profitable for player $-i$. □

Proposition A.2.3. *If μ is an ϵ-CCE of a two-player constant-sum game G, then s^μ is a 2ϵ-Nash equilibrium.*

Proof. Choose player i arbitrarily. Either $u_i(s^\mu) \geqslant \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right]$ or $u_i(s^\mu) < \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right]$. Consider the first case. Starting from the definition of ϵ-CCE:

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i.$$

But since $u_i(s^\mu) \geqslant \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right]$ we have

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - u_i(s^\mu) \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i.$$

Which by Lemma A.2.1 means

$$u_i(\rho_i', s_{-i}^\mu) - u_i(s^\mu) \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i.$$

Thus s^μ is an ϵ-Nash. Next, suppose $u_i(s^\mu) < \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right]$. By Proposition A.2.2 we have,

$$\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] - u_i(s^\mu) \leqslant \epsilon \tag{A.3}$$

$$\implies \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant u_i(s^\mu) + \epsilon. \tag{A.4}$$

Then, starting from the definition of ϵ-CCE and applying (A.4),

$$\begin{aligned} &\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - \mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i \\ \implies &\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - (u_i(s^\mu) + \epsilon) \leqslant \epsilon \quad \forall \rho_i' \in \mathrm{P}_i \\ \implies &\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho_i', \rho_{-i})\right] - u_i(s^\mu) \leqslant 2\epsilon \quad \forall \rho_i' \in \mathrm{P}_i. \end{aligned}$$

By Lemma A.2.1 we have

$$u_i(\rho_i', s_{-i}^{\mu}) - u_i(s^{\mu}) \leqslant 2\epsilon \quad \forall \rho_i' \in \mathrm{P}_i.$$

□

A.3 Marginals of a CCE May Not Be a CCE

	a	b
a	$1, 0$	$-1, -1$
a	$-1, -1$	$0, 1$

Figure A.2: The marginal strategies of a CCE do not generally form a CCE themselves.

Here we give an example showing the marginal strategies of a CCE may not form a CCE. Consider μ s.t. $\mu(a, a) = 0.5$ and $\mu(b, b) = 0.5$. μ is a CCE. $\mathbb{E}_{\rho\sim\mu}\left[u_i(\rho)\right] = 0.5$ for each player. If row (r) were to player a and column continues to play according to μ, row's utility is 0; if r plays b instead, their utility is now -0.5. Thus r has no profitable deviations from the CCE recommendations. Column does not either, this can be shown with a symmetric argument.

Row's marginal strategy s_r^{μ} plays a with probability 0.5 and b with probability 0.5, s_c^{μ} does likewise. $u_r(s_r^{\mu}, s_c^{\mu}) = u_c(s_r^{\mu}, s_c^{\mu}) = -0.25$. However, a is a profitable deviation for r now since $0 > -0.25$, thus the decorrelated strategies from the same CCE are also not a CCE.

Appendix B

Poly-EFG Details

Here we include the full details of Section 4.2.

Definition B.0.1 (Induced normal-form polymatrix game)**.** Given a poly-EFG $G = (N, E, \mathcal{G})$, the *induced normal-form polymatrix game* is a polymatrix game $G' = (N, E, \mathrm{P}, u')$ such that P_i is equal to i's set of pure strategies in each G_{ij} and $u'_{ij}(\rho_i, \rho_j) = u_{ij}(\rho_i, \rho_j) = \sum_{z \in Z} p_i(z, \rho_i) p_j(z, \rho_j) p_c(z, \pi'_c) u_{ij}(z)$ where u_{ij} is the utility function of i in G_{ij}.

In games of perfect recall, every behaviour strategy π_i has an equivalent mixed strategy $s_i^{\pi_i}$ (by Theorem 2.3.2), which means for any perfect recall EFG G, we can use the poly-EFG representation instead of turning G into a normal-form game then using a normal-form polymatrix game to get the same vulnerability bounds on G. Given π, let s^π be a profile of equivalent mixed strategies.

From Theorem 2.3.2 and an assumption that each $G_{ij} \in \mathcal{G}$ has perfect recall, we derive two immediate corollaries.

Corollary B.0.2. *If π is γ subgame stable for a poly-EFG $\check{G}$ where each subgame has perfect recall , then s^π is γ subgame stable in the induced normal-form polymatrix game of $\check{G}$.*

Corollary B.0.3. *For EFG of perfect recall G, if G is δ-CSP-EFG then the induced normal form of G is δ-CSP.*

Recall the definition of marginal behaviour strategies.

Definition B.0.4 (Marginal behaviour strategy)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let π_i^μ be the *marginal behaviour strategy* for i where $\pi_i^\mu(a, I)$ is defined arbitrarily if $\sum_{\rho_i' \in \mathrm{P}_i(I)} s_i^\mu(\rho_i') = 0$ and otherwise

$$\pi_i^\mu(a, I) \doteq \frac{\sum_{\rho_i \in \mathrm{P}_i(a,I)} s_i^\mu(\rho_i)}{\sum_{\rho_i' \in \mathrm{P}_i(I)} s_i^\mu(\rho_i')} \quad \forall I \in \mathcal{I}_i, a \in A(I),$$

where $s_i^\mu(\rho_i) \doteq \sum_{\rho_{-i} \in \mathrm{P}_{-i}} \mu(\rho_i, \rho_{-i})$.

Definition B.0.5 (Marginal behaviour strategy profile)**.** Given some mediated equilibrium $(\mu, (\Phi_i)_{i=1}^N)$, let π^μ be a *marginal behaviour strategy profile*, where π_i^μ is a marginal behaviour strategy $\forall i \in N$.

Recall that Π^μ be the set of marginal behaviour strategy profiles for any CCE of G and Π_i^μ is the set of marginal behaviour strategies for i. An extension of Theorem 3.4.3 for poly-EFGs is given next.

Proposition 4.2.3. *If an EFG G is δ-CSP and $\exists \check{G} \in \mathrm{CSP}_\delta(G)$ that is $(2n\delta, \gamma)$-subgame stable and μ is a CCE of G, then*

$$\mathrm{Vul}_i\left(\pi^\mu, \Pi_{-i}\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta,$$

and

$$\mathrm{Ex}(\Pi^\mu) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

Proof. Transform $\check{G}$ into its induced normal-form polymatrix game $\check{G}'$. By Corollaries B.0.2 and B.0.3 the induced normal form of G is δ-CSP and $(2n\delta, \gamma)$-subgame stable. By perfect recall, we can convert π^μ to an equivalent mixed strategy profile s^μ and do likewise with any $\pi_{-i} \in \Pi_{-i}$. Then apply Theorem 3.4.3 using s^μ, S_{-i} and the induced normal-form polymatrix game of $\check{G}$ to bound vulnerability on G's induced normal form, and hence G. For the exchangeability bound, we can also convert any $\pi \in \bigtimes_{i \in N} \Pi_i^\mu$ into an equivalent mixed strategy profile, then apply Theorem 3.4.3. □

B.0.1 Vulnerability Against Self-Taught Agents in EFGs

Next we show an analogue of Theorem 3.5.3 for extensive-form games. Recall from Section 4.2.1 that $\Pi(\mathcal{A}_i)$ is the set of marginal behaviour strategy profiles

of the mediated equilibria of algorithm $\mathcal{A}_i$; $\Pi_i(\mathcal{A}_i)$ is the set of i's marginal strategies from this set of strategy profiles; and if $\mathcal{A}_N \doteq (\mathcal{A}_1, ...\mathcal{A}_n)$ is the profile of learning algorithms, then $\Pi^\times(\mathcal{A}_N)$ is the set of all possible match-ups between strategies learned in self-play by those learning algorithms.

Definition B.0.7. We say a game G is *δ-CSP in the neighbourhood of* $\Pi' \subseteq \Pi$ if there exists a constant sum poly-EFG $\check{G}$ such that $\forall \pi \in \Pi'$ we have $|u_i(\pi) - \check{u}_i(\pi)| \leqslant \delta$. We denote the set of such CSP games as $\mathrm{CSP}_\delta(G, \Pi')$.

Definition B.0.8. We say a CSP game G is *γ-subgame stable in the neighbourhood of* Π' if $\forall \pi \in \Pi', \forall (i,j) \in E$ we have that (π_i, π_j) is a γ-Nash of G_{ij}.

Proposition 4.2.6. *If G is δ-CSP in the neighbourhood of $\Pi^\times(\mathcal{A}_N)$ and $\exists \check{G} \in \mathrm{CSP}_\delta(G, \Pi^\times(\mathcal{A}_N))$ that is γ-subgame stable in the the neighbourhood of $\Pi(\mathcal{A}_i)$, then for any $\Pi \in \Pi(\mathcal{A}_i)$*

$$\mathrm{Vul}_i\left(\pi, \Pi^\times_{-i}(\mathcal{A}_N)\right) \leqslant |E_i|\gamma + 2\delta \leqslant (n-1)\gamma + 2\delta.$$

The proof goes the same way as in Corollary 4.2.3. Use the induced normal-form polymatrix game of $\check{G}$ and Theorem 3.5.3 to derive bounds for the induced normal form of G, which then apply to G.

Appendix C

Proof of Algebraic Characterization of Strategic Equivalence for n-Player Games

Next we show the proof of Theorem C.0.6. The proof is largely the same as the original proof in Moulin and Vial [46], we rewrite it here for clarity. We break the larger proof into a number of smaller lemmas for readability.

Lemma C.0.1. *If G and G' are algebraically equivalent for player i, then they are strategically equivalent for player i.*

Proof. In strategic-form, (6.11) is given as

$$u_i(\rho) = \lambda_i u_i'(\rho) + \hat{u}_i(\rho_{-i}) \quad \forall \rho \in \mathrm{P}. \tag{C.1}$$

Suppose that (C.1) holds. Then if

$$u_i(s_i', s_{-i}) \geqslant u_i(s_i, s_{-i})$$

holds, we have

$$\begin{aligned} \lambda_i u_i'(s_i', s_{-i}) + \hat{u}_i(s_{-i}) &\geqslant \lambda_i u_i'(s_i, s_{-i}) + \hat{u}_i(s_{-i}) \\ \implies \lambda_i u_i'(s_i', s_{-i}) &\geqslant \lambda_i u_i'(s_i, s_{-i}) \\ \implies u_i'(s_i', s_{-i}) &\geqslant u_i'(s_i, s_{-i}), \end{aligned}$$

since $\lambda_i > 0$. Thus G and G' are SE. □

Lemma C.0.2. *If G and G' are equivalently trivial for i, then they are strategically equivalent for i.*

Proof. Let x_1, x_2 be two strategies of i and y be any *non-correlated* strategy profile of $-i$. Then

$$x_1^\top U_i y \geqslant x_2^\top U_i y \tag{C.2}$$
$$\iff x_1^\top (\alpha\beta^\top) y \geqslant x_2^\top (\alpha\beta^\top) y \tag{C.3}$$
$$\iff (x_1^\top \alpha)(\beta^\top y) \geqslant (x_2^\top \alpha)(\beta^\top y) \tag{C.4}$$

and

$$x_1^\top U_i' y \geqslant x_2^\top U_i' y \tag{C.5}$$
$$\iff x_1^\top (\alpha\beta'^\top) y \geqslant x_2^\top (\alpha\beta'^\top) y \tag{C.6}$$
$$\iff (x_1^\top \alpha)(\beta'^\top y) \geqslant (x_2^\top \alpha)(\beta'^\top y) \tag{C.7}$$

But $(\beta^\top y) = (\beta'^\top y) = 0$ or $(\beta^\top y) > 0$ and $(\beta'^\top y) > 0$. This means that

$$(x_1^\top \alpha)(\beta^\top y) \geqslant (x_2^\top \alpha)(\beta^\top y)$$

holds if and only if

$$(x_1^\top \alpha)(\beta'^\top y) \geqslant (x_2^\top \alpha)(\beta'^\top y),$$

thus G and G' are SE. □

Let $e \in \mathbb{R}^{|\mathrm{P}_i|}$ be a vector of all ones. Let $p_e(x) = \frac{x^\top e}{|\mathrm{P}_i|} e$, be the vector projection of x onto e. If we apply p_e, to each of the columns of U_i, we get a payoff matrix $\hat{U}_i = p_e \circ U_i$ where each column is equal to ce for some scalar c. For any strategy profile of $-i$, i will be indifferent between all of their strategies with $\hat{U}_i$. What is left over when we subtract $\hat{U}_i$ from U_i is the strategic essence of the game, what we call the *strategic kernel*.

Definition C.0.3 (Strategic kernel)**.** Given a utility matrix U_i, the strategic kernel of U_i is a utility matrix $\tilde{U}_i = U_i - p_e \circ U_i$ where p_e is applied to each column of U_i.

Lemma C.0.4. *If G and G' are SE for i, then G and G' are equivalently trivial for i or G and G' are algebraically equivalent for i.*

Proof. Suppose that G and G' are strategically equivalent for i. By Lemma 6.1.3, we have that G and G' are correlated strategically equivalent for i. In matrix-form, this expressed as

$$x_1 \top U_i y \geqslant x_2^\top U_i y \iff x_1^\top U_i' y \geqslant x_2^\top U_i' y \quad \forall x_1, x_2 \in X_i, y \in Y. \tag{C.8}$$

Where $X_i = \{x \in [0,1]^{|\mathrm{P}_i|} \mid \sum_{j=1}^{|\mathrm{P}_i|}(x)_j = 1\}$ is the set of strategies for player i and $Y = \{y \in [0,1]^{|\mathrm{P}_{-i}|} \mid \sum_{j=1}^{|\mathrm{P}_{-i}|}(y)_j = 1\}$ is the set of correlated strategy profiles for $-i$. Rearranging (C.8), we have

$$(x_2^\top - x_1^\top)U_i y \leqslant 0 \iff (x_2^\top - x_1^\top)U_i' y \leqslant 0 \quad \forall x_1, x_2 \in X_i, y \in Y. \tag{C.9}$$

Let $e \in \mathbb{R}^{|\mathrm{P}_i|}$ we a vector such that all entries are 1. Let $Z = \{x_2 - x_1 \mid x_1, x_2 \in X_i\}$. Then (C.9) is equivalent to

$$z^\top U_i y \leqslant 0 \iff z^\top U_i' y \leqslant 0 \quad \forall z \in Z, y \in Y. \tag{C.10}$$

Let $\mathcal{H} = \{z \in \mathbb{R}^{|\mathrm{P}_i|} \mid z^\top e = 0\}$. $\mathcal{H}$ is a vector space spanned by the set of vectors with a single entry of 1 and -1, with the remainder of the entries being 0 [50]. Z is a subset of $\mathcal{H}$ characterized by elements having a sum of positive entries being at most 1: $Z = \{z \in H \mid \sum_{(z)_j>0}(z)_j \leqslant 1\}$. Any element of $\mathcal{H}$ can be written as a scaled element of Z by normalizing by its sum of positive entries. Then (C.10) holds if and only if

$$z^\top U_i y \leqslant 0 \iff z^\top U_i' y \leqslant 0 \quad \forall z \in \mathcal{H}, y \in Y.$$

Let $\tilde{U}_i = U_i - p_e \circ U_i$, $\tilde{U}_i' = U_i' - p_e \circ U_i'$ be the strategic kernels of U_i and U_i', respectively. Recall that $p : \mathbb{R}^{|\mathrm{P}_i|} \to \mathcal{H}$ where $p(u) = \frac{u^\top e}{|\mathrm{P}_i|}e$. Note that

$$z^\top U_i y = z^\top(\hat{U}_i + \tilde{U}_i)y = z^\top \hat{U}_i y + z^\top \tilde{U}_i y = z^\top \tilde{U}_i y,$$

since $z^\top \hat{U}_i = 0$ as z is orthogonal to all columns of $\hat{U}_i$. Likewise $z^\top U_i' y = z^\top \tilde{U}_i' y$ Thus,

$$z^\top \tilde{U}_i y \leqslant 0 \iff z^\top \tilde{U}_i' y \leqslant 0 \quad \forall z \in \mathcal{H}, y \in Y. \tag{C.11}$$

Next, note that $\tilde{U}_i' y, \tilde{U}_i y \in \mathcal{H}$ by the definition of p_e. Take $\tilde{U}_i' y$ and $\tilde{U}_i y$ to be two normal vectors dividing $\mathcal{H}$ into two half-spaces according to (C.11). (C.11)

states that all elements of $\mathcal{H}$ on one side of $\tilde{U}_i y$'s corresponding hyperplane are also on the same side of $\tilde{U}_i' y$'s hyperplane. The only way this is possible is that if $\tilde{U}_i y$ and $\tilde{U}_i' y$ are parallel. Then we have (C.11) is equivalent to

$$\forall y \in Y, \exists \lambda_y > 0 : \tilde{U}_i y = \lambda_y \tilde{U}_i' y. \tag{C.12}$$

At this point, consider two cases: 1. when the rank of $\tilde{U}_i$ is at least 2 and 2. when the rank is less than 2.

Case 1. Let E be the standard basis of $\mathbb{R}^{|\mathrm{P}_{-i}|}$. Note that all elements of E are elements of Y. Since the rank of $\tilde{U}$ is at least 2, we can find $e_j, e_k \in E$ such that $\tilde{U}_i e_j$ and $\tilde{U}_i e_k$ are linearly independent (if we could not, this would imply that $\tilde{U}_i$ has rank less than 2). Applying (C.12), we have

$$\begin{aligned}
\lambda_{e_j+e_k} \tilde{U}_i'(e_j + e_k) &= \tilde{U}_i(e_j + e_k) \\
&= \tilde{U}_i e_j + \tilde{U}_i e_k \\
&= \lambda_{e_j} \tilde{U}_i' e_j + \lambda_{e_k} \tilde{U}_i' e_k.
\end{aligned}$$

Which implies that $\lambda_{e_k} = \lambda_{e_j}$. Call this quantity λ_i.

Take any y that is not an element of $\ker(\tilde{U})$. Then at least one of the sets $\{\tilde{U}_i y, \tilde{U}_i e_j\}$, $\{\tilde{U}_i y, \tilde{U}_i e_k\}$ is linearly independent (if they were this would imply $\tilde{U}_i e_j$ and $\tilde{U}_i e_k$ are linearly dependent.)

We can then show that $\lambda_y = \lambda_i$ by the same argument as above. If y is instead an element of $\ker(\tilde{U})$ then $\tilde{U}_i' y = \lambda_y \tilde{U}_i y = 0$, so we may as well take $\lambda_y = \lambda_i$. This means that if the rank of $\tilde{U}_i$ is at least 2, then

$$\exists \lambda_i : \forall y \in Y, \tilde{U}_i y = \lambda_i \tilde{U}_i' y.$$

In particular,

$$\exists \lambda_i : \forall e_j \in E, \tilde{U}_i e_j = \lambda_i \tilde{U}_i' e_j, \tag{C.13}$$

which implies that

$$\tilde{U}_i = \lambda_i \tilde{U}_i, \tag{C.14}$$

since (C.13) means that each column of $\tilde{U}$ is equal to the corresponding column of $\tilde{U}_i'$ scaled by λ_i.

Case 2. The rank of $\tilde{U}$ is at most 1. By (C.12) $\tilde{U}'$ is also rank at most 1. This means that $\exists \alpha \mathbb{R}^{|\mathrm{P}_i|}$, $\beta, \beta' \in \mathbb{R}^{|\mathrm{P}_{-i}|}$ such that

$$\tilde{U}y = (\alpha\beta^\top)y = \alpha^\top(\beta^\top y) \quad \forall y \in Y$$
$$\tilde{U}'y = (\alpha\beta'^\top)y = \alpha^\top(\beta'^\top y) \quad \forall y \in Y$$

If $\alpha = 0$, then (C.13) holds. Suppose that $\alpha \neq 0$. Then by (C.12) we have

$$\alpha^\top(\beta^\top y) = \lambda_y \alpha^\top(\beta'^\top y) \quad \forall y \in Y \tag{C.15}$$

Then, either $(\beta^\top y) = (\beta'^\top y) = 0$, or $(\beta^\top y), (\beta'^\top y) \neq 0$ and have the same sign (since $\lambda_y > 0$).

Consider the following two statements:

$$\exists x \geqslant 0 \text{ s.t. } \beta x \leqslant \beta', \tag{C.16}$$
$$\exists y \geqslant 0 \text{ s.t. } (\beta^\top y) \geqslant 0 \text{ and } (\beta'^\top y) < 0. \tag{C.17}$$

By Farkas' lemma, only one of the two may hold. However, by (C.15) $(\beta^\top y)$ and $(\beta'^\top y)$ have the same sign or are both 0, since $\lambda_y > 0$. This means (C.17) cannot hold. This uses the fact that (C.17) holds if and only if

$$\exists y \in Y \geqslant 0 \text{ s.t. } (\beta^\top y) \geqslant 0 \text{ and } (\beta'^\top y) < 0.$$

Thus,

$$\beta' \geqslant x\beta. \tag{C.18}$$
$$\implies \beta' - \beta x \geqslant 0 \tag{C.19}$$

Applying the same steps we also find that

$$\beta \geqslant x'\beta' \tag{C.20}$$
$$\implies \beta - \beta' x' \geqslant 0 \tag{C.21}$$

Substituting (C.18) into (C.21), we get

$$\beta - \beta x x' \geqslant 0 \tag{C.22}$$
$$\implies (1 - xx')\beta \geqslant 0 \tag{C.23}$$
$$\implies c\beta \geqslant 0, \tag{C.24}$$

where $c \doteq (1 - xx')$. Substituting (C.20) into (C.19), and following similar steps, we get

$$(1 - xx')\beta' \geqslant 0 \tag{C.25}$$

$$\implies c\beta' \geqslant 0. \tag{C.26}$$

First, consider if $c = 0$, then $xx' = 1$. Without loss of generality, suppose that $x \geqslant 1$. Then, we have $x' = \frac{1}{x} \leqslant 1$. Thus,

$$\beta \geqslant x'\beta' \implies \beta \geqslant \frac{1}{x}\beta' \implies x\beta \geqslant \beta'.$$

And since $\beta' \geqslant x\beta$, we have $\beta' = x\beta$. Which is the same as (C.14).

Next, If $c > 0$, then (C.24) and (C.26) imply all elements of β and β' are positive or 0. If $c < 0$, then (C.24) and (C.26) imply all elements of β and β' are negative or 0. Thus, all elements of β and β' have the same sign. If $\beta', \beta \leqslant 0$, simply flip the sign of β, β' and α. Then $\beta \geqslant 0$ and $\beta' \geqslant 0$ and we have that G and G' are equivalently trivial.

Summarizing, we have shown that if G and G' are SE, then either they are equivalently trivial or $\exists \lambda_i \geqslant 0$ s.t. $\tilde{U}_i = \lambda_i \tilde{U}_i$. □

Theorem 6.1.3. *G and G' are SE for player i if and only if G and G' are equivalently trivial for player i, or G and G' are algebraically equivalent for player i.*

Theorem C.0.6. *G and G' are SE for player i if and only if G and G' are equivalently trivial for player i or G and G' are algebraically equivalent for player i.*

Proof. One direction is shown by be Lemma C.0.1 and Lemma C.0.2, the other is shown by Lemma C.0.4 □