Computing Game Symmetries and Equilibria That Respect Them

Emanuel Tewolde[1,2], Brian Hu Zhang[1], Caspar Oesterheld[1,2]
Tuomas Sandholm[1,3,4,5], Vincent Conitzer[1,2]
[1]Carnegie Mellon University, [2]Foundations of Cooperative AI Lab (FOCAL)
[3]Strategy Robot, Inc., [4]Strategic Machine, Inc., [5]Optimized Markets, Inc.
{etewolde, bhzhang, coesterh, sandholm, conitzer}@cs.cmu.edu

Abstract

Strategic interactions can be represented more concisely, and analyzed and solved more efficiently, if we are aware of the symmetries within the multiagent system. Symmetries also have conceptual implications, for example for equilibrium selection. We study the computational complexity of identifying and using symmetries. Using the classical framework of normal-form games, we consider game symmetries that can be across some or all players and/or actions. We find a strong connection between game symmetries and graph automorphisms, yielding graph automorphism and graph isomorphism completeness results for characterizing the symmetries present in a game. On the other hand, we also show that the problem becomes polynomial-time solvable when we restrict the consideration of actions in one of two ways.

Next, we investigate when exactly game symmetries can be successfully leveraged for Nash equilibrium computation. We show that finding a Nash equilibrium that respects a given set of symmetries is PPAD- and CLS-complete in general-sum and team games respectively—that is, exactly as hard as Brouwer fixed point and gradient descent problems. Finally, we present polynomial-time methods for the special cases where we are aware of a vast number of symmetries, or where the game is two-player zero-sum and we do not even know the symmetries.

1 Introduction

In AI and decision making, we appreciate the presence of symmetries, and they are of utmost importance in game theory and multiagent systems. For one, central concepts such as cooperation, conflict, and coordination are usually presented most simply on *totally symmetric* games[1], such as the Prisoner's Dilemma, Chicken, and Stag Hunt. The classic and performant Lemke-Howson algorithm for finding Nash equilibria is frequently (and without loss of generality) presented for totally symmetric games (Nisan et al. 2007)[Section 2.3] for the sake of clarity. Furthermore, we may find that interactions with symmetries can be described more concisely in comparison to enumerating the full outcome payoff functions: "Matching Pennies is a two-player game where each player has two actions $\{0, 1\}$. If both players play the same action, player 1 wins, otherwise, player 2 wins." This is oftentimes leveraged in games where we *design* the outcome and reward structures, such as in social choice (Brandt et al. 2015) and mechanism design (Moulin 2004) via anonymity, neutrality, and fairness axioms.

[1]To be defined later; informally, games in which players have the same strategy options and take on the same "role" in the game.

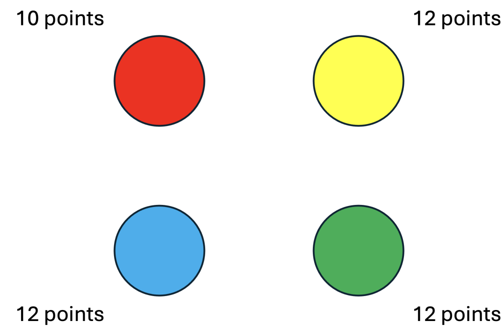

Figure 1: A two-player coordination game. If both players pick the same color, they each receive the associated utility points. If they miscoordinate, both receive 0 points. Without knowing who you are playing with, what color would you choose?

Indeed, notions of *fairness* have been connected to the premise that any participant of the game might be assigned to any player identity in the game, which creates a symmetry across participants (Gale, Kuhn, and Tucker 1952). For the sake of fairness, one would then like the player identities to be equally strong (*cf.* the Matching Pennies game, and the "veil of ignorance" philosophy (Rawls 1971)). Ham (2021), and references therein, give a formal treatment of this in terms of game symmetries.

This symmetry idea that any participant (AIs, humans, etc.) might take on any player identity in the game (*e.g.*, black versus white in chess) also reappears when reasoning about other agents of which we do not have a prior: since the beginnings of machine learning, it has been popular to learn good strategies in *self-play* (Samuel 1959), that is, to assume that other players would use the same strategy as oneself. Self-play continues to be a core contributor to AIs that can learn with no or limited access to human data, and reach super-human performance in domains such as Go (Silver et al. 2016, 2017), and two- and multi-player poker (Brown and Sandholm 2018, 2019). Beyond leverag-

	Game symmetries		Game isomorphisms		Computing a symmetric equilibrium
General games	GA-c; (Th. 1)	$\mathsf{XP}\left(\frac{\text{\#actions}}{\text{\#players}}\right)$ (Th. 5)	GI-c; (GGS11)	$\mathsf{XP}\left(\frac{\text{\#actions}}{\text{\#players}}\right)$ (Th. 5)	PPAD-c (Prop. 18); XP(#orbits) (Th. 9)
Team games					CLS-c (Th. 8)
Zero-sum games					FP (Th. 10)
1PL-actions symmetries	P (Th. 8)				PPAD-c (Prop. 18)
Player symmetries	P (Th. 6)				

Table 1: A high-level summary of some complexity results we obtain across various special cases of games and restrictions on the symmetry sets; though we refer to the associated results for exact statements. '-c' denotes completeness for the respective class. We obtain the hardness results for very narrow settings already, such as, for example, two-player games. $\mathsf{XP}(k)$ stands for runtimes in which the only exponent is k.

ing the player symmetry in chess and Go by always orienting the board from the moving player's perspective, Silver et al. also exploits the rotation and reflection symmetries in Go.

Related to self-play, we may also utilize game symmetries for the purposes of strategy pruning and equilibrium selection (Harsanyi and Selten 1988). Consider the coordination game in Figure 1. In an ideal scenario, the two players manage to coordinate on the same color between the three that yield the maximal reward of 12 points. However, if there is no further basis for distinguishing the high-reward colors (*cf.* focal points (Schelling 1960; Alós-Ferrer and Kuzmics 2013)), then the players run a significant risk of miscoordinating if they attempt to get the reward of 12. In fact, a natural strategy in this game is to instead go for red with the lower reward of 10. We can explain this formally by recognizing that there are symmetries permuting the colors {Y,B,G} for both players while keeping the players' preferences over colors unchanged. Hence, without access to some prior coordination device between the players, the players cannot properly differentiate between {Y, B, G}. Therefore, the players should assign the same likelihood of play across those colors, that is, select a strategy profile that respects the aforementioned symmetries. Under this constraint, both players picking red becomes the unique optimal (Nash equilibrium) profile. The equilibrium that uniformly randomizes over {Y,B,G} merely achieves an expected reward of 4.[2] In more recent work, Hu et al. (2020) and Treutlein et al. (2021) apply this argument to zero-shot coordination problems in order to tackle the shortcomings of standard self-play. We remark that respecting the color and player symmetries does not hurt the players if they play this game repeatedly instead. In that case, they can achieve a long-term average of 12 points by both playing the following symmetric strategy: In round 1, randomize uniformly over {Y, B, G}. In round $t \geq 2$, repeat last round's action if both of you coordinated successfully last round. Otherwise, repeat last round's action only with 50% chance, and the other player's action from last round with the other 50% chance.

Lastly, several methods for finding solutions to multiagent problems make great use of symmetries or awareness thereof. On the applied side of solvers, Marris et al. (2022) learn to compute Nash, correlated, and coarse correlated equilibria, and achieve sample efficiency by imposing game symmetry invariance onto their neural network architecture. Liu et al. (2024) extends this to transformer-based representation learning of normal-form games, with which they show state-of-the-art performance on various additional tasks such as predicting deviation incentives. Earlier work (Gilpin and Sandholm 2007) has developed an abstraction algorithm for solving large-scale extensive-form games that is based on detecting game symmetries (or a related notion thereof) and merging subgames accordingly. On the theory side, Fabrikant, Papadimitriou, and Talwar (2004) give a polytime algorithm for pure Nash equilibrium network congestion game whenever all players are symmetric, and Daskalakis and Papadimitriou (2007) develop a polytime approximation scheme for two-action *anonymous games*—a popular game class with particular kinds of symmetries.

A further discussion of related work can be found in the full version of this paper.

1.1 Structure and a First Overview

In Section 2, we start with background on game symmetries in normal-form games. Our general notion of symmetry encompasses any permutation of players *and* their action sets while keeping the utility payoffs unchanged. This is important: the players in Matching Pennies take on different roles in the game (matcher vs. mismatcher), and as such, can only be considered symmetric if we allow swapping the two actions of one player while simultaneously swapping the player identities. In another example, the symmetries discussed for the coordination game of Figure 1 keep player identities the same and only permute the action sets. In Section 3 we connect the presence of symmetries in a game to the presence of symmetries in a graph, and vice versa. The latter is a well-studied computational problem from which we obtain some of the complexity results summarized in Table 1. Not included in this table are Proposition 9 and Theorems 3 and 4. They focus on characterizing the *set* of game symmetries and relate it to the graph isomorphism problem. As a consequence, Theorem 4 resolves an open conjecture by Cao and Yang (2018) on deciding whether a game is *name-irrelevant symmetric*. Furthermore, our proof ideas can also be applied to the related *game isomorphism* problem, and so we simultaneously discuss those implications.

Section 4 introduces Nash equilibria that respects a given set of symmetries, then relates it to group-theoretic ideas involving orbits of actions, and further discusses computa-

[2] With a coordination device, the players are able to achieve a reward of 12 while respecting the symmetries, namely, by uniformly randomizing over profiles {(Y, Y), (B,B), (G,G)}. Correlated equilibria, however, will not be a focus in this paper.

tional preliminaries. In Section 5, we present a series of results on the complexities of computing Nash equilibria that respect a given set of symmetries or all symmetries. A summary can again be found in Table 1. We give a contextualized discussion of these results in the upcoming Section 1.2, and accompany it with additional insights.

Full proofs can be found in the full version of this paper.

1.2 Are Symmetries Actually Helpful for Solving Games?

As discussed in the introduction, symmetries have been successfully used for state-of-the-art equilibrium computation methods. Nonetheless, we should not be too quick to conclude that symmetries, if present, ought to be used.

Potential Harm We have already seen that in the coordination game of Figure 1, the players might actually strongly prefer to play Nash equilibria that fail to respect symmetries of the game. This effect is amplified in that game if we take away the color red from the alternatives, leaving us with a maximal symmetry-respecting payoff of 4. However, we also note here that a similar argument can be given for the opposite position. Take the totally symmetric two-player game of chicken, that is, the bimatrix game (A, A^T) where $A = \begin{pmatrix} 0 & -1 \\ 1 & -10 \end{pmatrix}$. In a Nash equilibrium that respects the symmetry that swaps the players, the players play their first strategy with probability 0.9, yielding each of them a payoff of 0.1. If they instead each search for an asymmetric Nash equilibrium, they may find distinct equilibria—for example, perhaps each player find the equilibrium that is best for that player. This results in the two players miscoordinating, resulting in the worst of all outcomes (each playing their second strategy).

Potential Slow-down Players also might want to ignore symmetries present in a game for the sake of faster computation. Take a totally symmetric bimatrix game $\Gamma = (A, A^T)$ with payoffs in $[0, 1]$. It has long been known (Gale, Kuhn, and Tucker 1952) that finding a symmetric Nash equilibrium of such a game cannot be easier than finding any Nash equilibrium of a general bimatrix game, which makes it PPAD-hard. Now consider the symmetric bimatrix game $(\tilde{A}, \tilde{A}^T)$ defined by $\tilde{A} = \begin{pmatrix} -10 & 2 \cdot \mathbb{1}^T \\ 2 \cdot \mathbb{1} & A \end{pmatrix}$, where $\mathbb{1}$ denotes the vector of appropriate dimension with all entries $= 1$. This game has its Pareto-optimal Nash equilibria located at strategy profiles that are obvious to find[3] for any participant: one player must play their first strategy and the other player any strategy but their first. Yet, if we restrict ourselves to respect the player symmetry, then both players playing the first strategy suddenly becomes unattractive. It leaving us with no choice but to find a Nash equilibrium of the original Γ, which is a PPAD-hard task. This phenomenon becomes even more omnipresent in team games—also known as identical-interest or common-payoff games—because such games are guaranteed to have Pareto-optimal Nash equilibria in a pure strategy profile. However, these profiles might not respect most or any nontrivial symmetries present in the game, as illustrated in the coordination game of Figure 1. Instead, the constraint of respecting symmetries leaves us with the harder computational problem of non-linear continuous optimization, as we will show in Theorem 8.

Results and General Conclusions This goes to show that for computational efficiency as well as for achieving high payoffs, one might want to be informed about the game before imposing the constraint of respecting symmetries. This stands in contrast to some self-play approaches—such as when using regret learning with full feedback—which implicitly respect symmetries, and other solving techniques mentioned in the introduction that have symmetry explicitly imposed into their architecture.

In Proposition 18, on the other hand, we show that the requirement of respecting a given set of symmetries does not make the search for a Nash equilibrium harder in the worst-case (PPAD-completeness), and in Theorem 8 we show that gradient descent methods are the best we can generally do in team games if we want a given set of symmetries to be respected (CLS-completeness). An additional special case that arises is with the class of two-player zero-sum games (Theorem 10): without having to compute any symmetry of the game (which we show to be graph automorphism hard), we can find a Nash equilibrium that respects *all* of the game's symmetries in polytime via a convex optimization approach.

A Positive Result When There Are Many Symmetries With many players in the game, the normal-form representation blows up exponentially, casting that representation impractical. That is why many-player games are usually represented more concisely, often making use of a vast number of symmetries present in the game. So what can we say then? If we are aware of enough symmetries between players and actions such that we are left with a constant number of *orbits* of actions, we can compute a Nash equilibrium that respects those symmetries in polytime (Theorem 9). This generalizes a result by Papadimitriou and Roughgarden (2008), and we can illustrate it on an N-player m-action variant of Rock-Paper-Scissors: each player i chooses an action j from $\{0, \ldots, m-1\}$, upon which they receive a payoff equal to "# wins - # losses" where # wins is the total number of players in the game choosing action $j - 1 \pmod m$ and # losses the number of players choosing $j + 1 \pmod m$. Figure 2 illustrates this game for $N = 2$ and $m = 5$, together with another variant that is prominently referenced in pop culture (RPS 2024). It is not hard to see that neither of these games change if we rotate the player identities by $1 \to \cdots \to i \to \cdots \to N \to 1$, or, instead, if we rotate the action labels by $1 \to \cdots \to j \to \cdots \to m \to 1$ for all players simultaneously. With those symmetries there is only a *single* orbit of actions. This renders the task of finding a symmetry-respecting Nash equilibrium not only polytime but trivial, because there is only one strategy profile left that respects those symmetries: each player uniformly randomizing over all of their action alternatives.

[3]Concretely, it requires parsing the full payoff matrix while recognizing that payoffs are in $[0, 1]$ in A. This takes linear time.

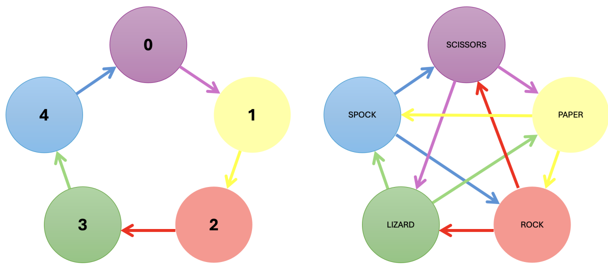

Figure 2: Two extensions to Rock-Paper-Scissors. In both, there is only one symmetries-respecting strategy profile. The right game is known as Rock-Paper-Scissors-Lizard-Spock.

A Remark On Approximate Symmetries The symmetry notion we study in this paper requires payoff profiles to match exactly. Earlier in this section, we argued that this is common in real-world scenarios; in particular, when they are human-designed. Furthermore, we believe that our results generalize meaningfully to settings in which it is unlikely to find exact payoff matches, *e.g.*, because utilities are drawn from a random distribution. To illustrate, let us revisit the color coordination game in Figure 1, except now, coordinating on {Y,B,G} yields 11.9, 12, and 12.1 points to both. van Damme (1997) then argues that the slightest uncertainties over payoffs—whether due to exogenous stochasticity or private information—may be reason enough for both players to pick red. We leave it to future work to give a general treatment of approximate notions of symmetries.

2 Preliminaries on Game Symmetries

Definition 1. *A* (normal-form) game Γ *consists of*

1. *A finite set of* players $\mathcal{N} := \{1, \dots, N\}$*, where* $N \geq 2$ *denotes the number of players,*
2. *A finite set of* actions $A^i := \{1, \dots, m^i\}$ *for each player* $i \in \mathcal{N}$*, where* m^i *denotes the number of actions, and*
3. *A* utility payoff *function* $u^i : A^1 \times \cdots \times A^N \to \mathbb{R}$ *for each player* $i \in \mathcal{N}$.

The players' goal is to maximize their own utility. An *action profile* $\mathbf{a}$ specifies what action each player takes and the set A denotes the set of all action profiles, that is, $\mathbf{a} = (\mathbf{a}_1, \dots, \mathbf{a}_N) \in A^1 \times \cdots \times A^N =: A$. We also denote the set of actions as $\mathbb{A} := \sqcup_{i \in \mathcal{N}} A^i$.

Remark 2. *For computational considerations,* u^i *is restricted to evaluate as rational values only. We assume a game* Γ *is given in explicit form, that is, it is stated as* $(\mathcal{N}, (A^i)_{i \in \mathcal{N}}, (T^i)_{i \in \mathcal{N}})$*, where* T^i *is a look-up table of length* $|A|$ *with all of player* i*'s payoffs under each* $\mathbf{a} \in A$.

T^i represents an N-dimensional payoff tensor; for example, for $N = 2$, that is a matrix. By abuse of notation, we also use cardinality $|\cdot|$ to denote the encoding size of an object that is not a set, *e.g.*, $|\Gamma|$ for a game Γ.

Definition 3 (Nash 1951)**.** *A* (game) symmetry *of a game* Γ *is a bijective map* $\phi : \mathbb{A} \to \mathbb{A}$ *that additionally satisfies*

1. *actions of the same player are mapped to the same player,* i.e.*, for each* $i \in \mathcal{N}$*, there is* $\pi(i) \in \mathcal{N}$ *satisfying* $a, a' \in A^i \implies \phi(a), \phi(a') \in A^{\pi(i)}$
2. *payoffs are symmetry-invariant, that is,* $u^i(\mathbf{a}) = u^{\pi(i)}(\phi(\mathbf{a}))$ *for all* $i \in \mathcal{N}$ *and* $\mathbf{a} \in A$.

To explain the notation $\phi(\mathbf{a})$, we first remark that the map ϕ induces a bijective player map $\pi : \mathcal{N} \to \mathcal{N}$ and bijective action set maps $\phi^i := \phi|_{A^i} : A^i \to A^{\pi(i)}$. Map π is henceforth referred to as a *player permutation*. By abuse of notation, ϕ can then be considered to map an action profile $\mathbf{a} \in A$ to action profile $\phi(\mathbf{a}) := \big(\phi(\mathbf{a}_{\pi^{-1}(j)})\big)_{j \in \mathcal{N}} \in A$.[4]

Let us revisit the bimatrix game (A, B) Matching Pennies, where $A = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$ and $B = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$. If PL1's and PL2's actions are {up,down} and {left,right} respectively, then this game has a symmetry "up $\to$ left $\to$ down $\to$ right $\to$ up". For instance, PL1 receives under profile (up,right) the same as PL2 under (up,left). Compare this with the another popular but more restrictive definition:

Definition 4 (von Neumann and Morgenstern 1944)**.** *A game* Γ *is called* totally symmetric *if each player has the same action set* A^**, and if for any player permutation* π *we have* $u^i(\mathbf{a}) = u^{\pi(i)}\Big((\mathbf{a}_{\pi^{-1}(j)})_{j \in \mathcal{N}}\Big)$ *for any player* i *and action profile* $\mathbf{a} \in \times_{i \in \mathcal{N}} A^*$.

In bimatrix games, this reduces to $B = A^T$. In particular, Matching Pennies has symmetries, but it is not totally symmetric.

Until Section 4, we assume that a game symmetry ϕ is represented in *explicit form*, which, again, means as a look-up table of evaluations of ϕ. Unlike with games, this explicit representation does not blow up exponentially with the number of players since ϕ only has $|\mathbb{A}|$ evaluations.

Note that the identity map $\mathrm{Id}_{\mathbb{A}}$ is always a game symmetry (henceforth the *trivial symmetry*), and that two symmetries ϕ and ϕ' compose to a third symmetry. Moreover, if ϕ is a symmetry, then ϕ^{-1} is one as well. Therefore:

Remark 5. *The set* $\mathrm{Sym}(\Gamma)$ *of symmetries of a game* Γ *forms a group together with map composition.*

Symmetry groups can be exponentially large; up to $N! \cdot \prod_{i=1}^N (m^i!)$ in our case. For purposes of algorithms, we will thus consider a group G as specified by a subset Z of generators[5]; writing $\langle Z \rangle = G$. Every finite group has a generator set of logarithmic size, which is $\log_2(|\mathbb{A}|!) = \tilde{\mathcal{O}}(|\mathbb{A}|)$ for us.

Last but not least, we define the graph problems that become relevant later on. A *simple graph* (hereafter just *graph*) $G = (V, E)$ consists of a set of vertices V and a set of edges $E \subseteq \binom{V}{2}$. The encoding size parameters are $|V|$ and $|E|$.

Definition 6 (GA)**.** *In the graph automorphism problem* GA*, we are given a graph* G*, and asked whether* G *admits a non-trivial* automorphism*, that is, a bijective map* $\phi : V \to V$ *that is not the identity function, and that satisfies* $(v, w) \in E \implies (\phi(v), \phi(w)) \in E$.

Definition 7 (GI)**.** *In the graph isomorphism problem* GI*, we are given two graphs* $G = (V, E)$ *and* $G' = (V', E')$*, and*

[4] We cannot simply define $\phi(\mathbf{a})$ as $(\phi(\mathbf{a}_i))_{i \in \mathcal{N}}$ because the j-th action in this vector is an action that belongs to player $\pi(j)$.

[5] Set $Z \subset G$ generates a finite group G if any $g \in G$ can be written as a composition of finitely many elements in Z.

asked whether there exists any isomorphism $G \to G'$, *that is, a bijective map* $\phi : V \to V'$ *that satisfies* $(v, w) \in E \iff (\phi(v), \phi(w)) \in E'$.

We call a decision problem GA- or GI-complete if it polytime reduces to GA (resp. GI) and vice versa. No polynomial-time algorithms for GA or GI are known, and GI is widely conjectured to be neither in P nor NP-hard.[6] GA many-one reduces to GI (Lozano and Torán 1992), but the reverse reduction is unknown. The best algorithm for GI is due to Babai (2016), and runs in time $\exp(\log^{\mathcal{O}(1)}(|V|))$.

3 Computation of Game Symmetries

In this section we study the complexity of characterizing the symmetries in a game. As a warm-up, we start with a simple-to-describe subgroup of symmetries that we have not discussed yet. We call ϕ a *1PL-actions* symmetry of Γ if it merely permutes the actions of a single player, *i.e.*, there is $i \in \mathcal{N}$ such that $\phi|_{\mathbb{A} \setminus A^i} = \mathrm{Id}_{\mathbb{A} \setminus A^i}$. Those relate to symmetries ϕ' that expose action duplicates, *i.e.*, that swap two actions $a, a' \in A^i$ for a player i and keep the rest fixed.

Proposition 8. *The 1PL-actions symmetries of a game* Γ *are generated by the symmetries that expose duplicate actions. The latter set can be computed in polytime* $\mathcal{O}(N \cdot \max_i(m^i)^2 \cdot |A|)$.

The factor $N \cdot \max_i(m^i)^2$ is the number of action pairs of a single player, and the factor $|A|$ comes from checking if swapping two actions keeps the utilities the same.

3.1 Complexity Results

We note that none of the symmetry examples from Section 1 are 1PL-actions symmetries. Instead, the {Y,B,G} symmetries described for the coordination game of Figure 1 is what we call *player-separable* because they keep player identities fixed, *i.e.*, a symmetry ϕ whose $\pi = \mathrm{Id}_{\mathcal{N}}$. We will show that such symmetries are already hard to characterize, let alone the whole set of symmetries Sym(Γ)—which, as we recall, allows an arbitrary permutation of players and their action sets simultaneously.

Theorem 1. *It is* GA*-complete to decide whether a game has a nontrivial symmetry. Hardness already holds for two-player {zero-sum / team} games that only possess game symmetries that are player-separable.*

The brackets indicate that the hardness works for the zero-sum restriction, but it also works for the team restriction.

Proof Idea. For membership, create an *edge-labeled* graph with node set $\mathcal{N} \cup \mathbb{A} \cup A$ and edges $\{(i, a) : i \in \mathcal{N}, a \in A^i\} \cup \{(a, \mathbf{a}) : a \in \mathbf{a} \in A\} \cup \{(i, \mathbf{a}) : i \in \mathcal{N}, \mathbf{a} \in A\}$. The first kind and second kind of edges shall receive two distinct labels, and edges $(i, \mathbf{a})$ are labeled with $u^i(\mathbf{a})$. Finally, we note that GA remains its complexity when the graph has edge labels. For hardness, create a two-player game with one action per vertex. Next, we give the players different payoffs depending on whether they play the same, neighboring, or non-neighboring vertices. To remove symmetries across players, we can give PL2 an additional dummy action. □

Our proof method carries over to a known GI-completeness result of the related *game isomorphism problem*. This problem is defined similarly to Definition 3, except now we are given two games Γ and Γ' and are asking for a player- and utility-preserving map $\phi : \mathbb{A} \to \mathbb{A}'$. In particular, a game symmetry is simply a game isomorphism from a game to itself.

Theorem 2 (Improved from Gabarró, García, and Serna 2011, Thm. 6)**.** *It is* GI*-complete to decide whether two games are isomorphic. Hardness already holds for two-player {zero-sum / team} games that only possess game symmetries that are player-separable.*

We think this result is worth noting because Gabarró, García, and Serna only establish hardness for mixed-motive 4-player games, and because they do not describe why their problem reduces to GI. Indeed, they partly accredit "personal communication" with another researcher as a reference.

Furthermore, the proof constructions in Theorem 1 additionally imply that the symmetries Sym(Γ) and automorphisms Aut(G) of associated game-graph pair (Γ, G) are isomorphic in a group-theoretic sense. Therefore, we can inherit further known results about graph automorphisms for our setting (Mathon 1979).

Proposition 9. *The following problems for a game* Γ *are polynomial-time Turing-equivalent to* GI*: (a) determining a generator set of* Sym(Γ)*, and (b) determining the cardinality of* Sym(Γ) *. Hardness already holds for two-player {zero-sum / team} with only player-separable symmetries.*

With an independent proof idea, we can additionally obtain hardness of deciding whether different players are symmetric to each other.

Theorem 3. *Deciding whether* Γ *has a symmetry* ϕ *that is not player-separable,* i.e.*, that maps at least one player to another player, is* GI*-complete. Hardness already holds for two-player zero-sum games.*

With this result, we can also prove an open conjecture by Cao and Yang (2018) in the affirmative.

Theorem 4. *It is* GI*-complete to decide whether a game* Γ *is* name-irrelevant symmetric*, that is, whether for all possible player permutations* $\pi : \mathcal{N} \to \mathcal{N}$ *there is symmetry* $\phi \in$ Sym(Γ) *of* Γ *that induces it. Hardness already holds for two-player zero-sum games.*

3.2 Efficient Computation

Next, we study efficient ways to compute the set of symmetries in a game (resp. isomorphisms between two games). For the results below, we require that each player i has $m^i \geq 2$ actions, that is, there is no player with no impact on the game. For the sake of space and presentation, we only present the statements in terms of game symmetries in this main body and defer to the full version of this paper for the treatment of game isomorphisms.

Theorem 5. *We can compute (a generator set of) the set* Sym(Γ) *of symmetries of a game* Γ *in time* $2^{\mathcal{O}(|\mathbb{A}|)}$.

[6]For example, GI being NP-complete would imply that the polynomial hierarchy collapses (Schöning 1988).

We prove Theorem 5 by reducing the problem to a *hypergraph* automorphism problem over a hypergraph with $|V| = \mathcal{O}(|\mathbb{A}|)$ nodes, and then applying the $2^{O(|V|)}$-time algorithm for hypergraph automorphism due to Luks (1999).

Corollary 10. *For games in which the number of actions per player is bounded, we can compute* $\mathrm{Sym}(\Gamma)$ *in polytime.*

This is because then $|\mathbb{A}| = \mathcal{O}(N)$, making the algorithm of Theorem 5 run in time $2^{\mathcal{O}(N)}$, which is is polytime in the size of the payoff tensors of the game. It gives a new perspective on the GA- and GI-hardness results we proved so far, since they hold even for a bounded ($= 2$) number of players: the computational hardness arises from a growing number of actions of multiple players simultaneously.

We further utilize the reduction idea to hypergraph automorphism for games with player symmetries: A game symmetry ϕ is called a *player symmetry* if it keeps the action labels "fixed", that is, if it sends action $k \in \{1, \dots, m^i\}$ of player i to action k of player $\pi(i)$.

Theorem 6. *We can compute (a generator set of) the group of player symmetries in polytime.*

For this proof, we must be particularly careful that the number of nodes in the constructed hypergraph does not increase linearly with the size of an individual player's action set. This is accomplished by creating $\mathcal{O}(\log m^i)$ action nodes for each player (instead of m^i as in Theorem 1), and associating to each action a *subset* of these nodes.

Corollary 11. *We can determine whether a game is totally symmetric (Def. 4), implicitly assuming $A^* = A^1 = \cdots = A^N$ with the current action numbering, in polytime.*

The positive results of Corollary 10 and Theorem 6 rely on the fact that the game is given in explicit form. In other, more concise game representations, we might find that these computational problems become hard again. In graphical games (Kearns, Littman, and Singh 2001), for example, we have easy-to-obtain hardness simply because such games are already conveniently *represented* as graphs.

Proposition 12. *The game automorphism (resp. isomorphism) problem for graphical games is* GA- *(resp.* GI-*)hard, even in team games with* 2 *actions per player.*

4 Preliminaries on Nash Equilibria That Respect Game Symmetries

Beyond giving background definitions in this section, we also study how "respecting" symmetries relate to action orbits, and what that implies for the the complexity considerations of Nash equilibrium computation.

4.1 Strategies, Nash Equilibria, Respecting Symmetries

As usual, we allow the players to randomize over their actions. That is, they can choose a probability distribution—called *strategy*—over A^i. The strategy sets are denoted by $S^i = \Delta(A^i)$. A strategy profile $\mathbf{s}$ and the strategy profile set S are defined similarly to their counterpart for actions: $\mathbf{s} = (\mathbf{s}_1, \dots, \mathbf{s}_N) \in S^1 \times \cdots \times S^N =: S$. Utilities naturally extend to S by taking the expectation $u^i(\mathbf{s}) := \sum_{\mathbf{a} \in A} \mathbf{s}_1(\mathbf{a}_1) \cdot \ldots \cdot \mathbf{s}_N(\mathbf{a}_N) \cdot u^i(\mathbf{a})$. For notational convenience, $\mathbf{s}_{-i} \in \bigtimes_{j \neq i} S^j =: S^{-i}$ abbreviates the strategies that all players are playing but i.

Definition 13. *A strategy profile* $\mathbf{s} \in S$ *is called a* Nash equilibrium *of* Γ *if for all player* $i \in \mathcal{N}$ *and all alternative strategies* $s \in S^i$ *we have* $u^i(\mathbf{s}) = u^i(\mathbf{s}_i, \mathbf{s}_{-i}) \geq u^i(s, \mathbf{s}_{-i})$.

That is, every player plays their optimal strategy taken as given what the other players have chosen. It is well-known that any game admits a Nash equilibrium (Nash 1950). Next, we will discuss Nash's follow-up work that further shows that symmetries-respecting Nash equilibria always exist.

Working towards that result, we first observe that a game symmetry mapping ϕ naturally extends to probability distributions over actions, *i.e.*, strategies. Thus, we can overload notation and write $S \ni \mathbf{s} \mapsto \phi(s)$. Symmetry ϕ will then also satisfy the invariance $u^i(\mathbf{s}) = u^{\pi(i)}(\phi(\mathbf{s}))$. Furthermore, we have:

Remark 14. *For any symmetry* ϕ *of* Γ*, we have that strategy profile* $\mathbf{s} \in S$ *is a Nash equilibrium if and only if* $\phi(\mathbf{s})$ *is.*

As we have argued in the introduction of this paper, game symmetries may indicate what actions ought to be played with the same likelihood; *cf.*, *e.g.*, the discussions of Figures 1 and 2. Let $\Sigma \subseteq \mathrm{Sym}(\Gamma)$ be a particular set of symmetries that we want to respect. This could be the trivial set $\{\mathrm{Id}\}$, in which case no symmetries need to be respected, or the full set $\mathrm{Sym}(\Gamma)$. This could also be *any* subset of symmetries that are readily available to us for a particular game, for example, because they are immediately exposed from a verbal description of the game.

Definition 15. *A strategy profile* $\mathbf{s} \in S$ *is said to* respect the symmetries $\Sigma \subseteq \mathrm{Sym}(\Gamma)$ *if for all* $\phi \in \Sigma$ *we have* $\phi(\mathbf{s}) = \mathbf{s}$.

Theorem 7 (Nash 1951)**.** *Any game* Γ *admits a Nash equilibrium that respects* all *symmetries* $\mathrm{Sym}(\Gamma)$. *Hence, it admits a Nash equilibrium that respects any particular set* $\Sigma \subseteq \mathrm{Sym}(\Gamma)$ *of symmetries.*

Nash obtains this result via a Brouwer fixed point argument, and the proof contains a nonconstructive analysis of the set of symmetries-respecting strategy profiles. In order to make this proof constructive and computational in Proposition 18, we introduce action orbits next.

4.2 Orbits Are All You Need

If we are interested in respecting a set Σ of symmetries, it suffices to know what actions are mapped to another action under some symmetry in the subgroup $\langle \Sigma \rangle \leq \mathrm{Sym}(\Gamma)$ *generated* by Σ. This is called the *orbit* of an action $a \in \mathbb{A}$ under group $\langle \Sigma \rangle$, denoted by $\langle \Sigma \rangle\, a := \{\phi(a) : \phi \in \langle \Sigma \rangle\} \subseteq \mathbb{A}$. The orbits $W(\langle \Sigma \rangle) = \{\langle \Sigma \rangle\, a : a \in \mathbb{A}\}$ partition the total set of actions $\mathbb{A}$. We obtain a characterization that has already been noted in prior work that studied player symmetries.

Lemma 16 (*cf.* Emmons et al. 2022)**.** *Profile* $\mathbf{s}$ *respects a set of symmetries* $\Sigma \subseteq \mathrm{Sym}(\Gamma)$ *if and only if it respects* $\langle \Sigma \rangle$ *if and only if for all orbits* $\omega \in W(\langle \Sigma \rangle)$ *and actions* $a, a' \in \omega$ *of respective players* i, i' *we have* $\mathbf{s}_i(a) = \mathbf{s}_{i'}(a')$.

4.3 Computational Considerations

In games of three players or more, the only (symmetry-respecting) Nash equilibrium might take on irrational values (Nash 1951) even though the game payoffs are integers. In order to represent solutions in finite bit length, we allow approximate solutions up to some precision error $\epsilon > 0$. An ϵ-Nash equilibrium must then satisfy $u^i(\mathbf{s}) \geq u^i(s, \mathbf{s}_{-i}) - \epsilon$ in Definition 13. We want ϵ to be 'small' relative to the range of utility payoffs, which—by shifting and rescaling (Tewolde and Conitzer 2024)—we can w.l.o.g. assume to be $[0, 1]$. Then, ϵ is given in binary, *i.e.*, we seek algorithms that depend polynomially on $\log(1/\epsilon)$.

Since (1) symmetry-respecting Nash equilibria always exist, and (2) we can check whether a strategy profile is indeed a Nash equilibrium that respects a given set of symmetries, we enter the complexity theory landscape of total NP search problems when it comes to *finding* such equilibria. Its subclasses are characterized by the proof technique used to show that each problem instance admits a solution. We will be interested in the subclasses PPAD and CLS, which both lie somewhere in between FP and FNP (the direct analogues to P and NP when we deal with search problems). PPAD ("Polynomial Parity Arguments on Directed graphs", Papadimitriou 1994) contains the problems where a solution is guaranteed to exist via a fixed-point argument, and CLS ("Continuous Local Search", Daskalakis and Papadimitriou 2011) is based on gradient dynamics on a compact polyhedral domain always admitting a solution.

Last but not least, we will consider three representation schemes for the symmetries we require to be respected, in decreasing order of explicitness: (1) *Explicit form*: A set $\Sigma \subseteq \text{Sym}(\Gamma)$ is given as a list of symmetries, each given in explicit form. (2) *Orbit form*: A partition W of actions $\mathbb{A}$ into orbits, with the promise that W is induced by an unknown set of symmetries $\Sigma \subseteq \text{Sym}(\Gamma)$. (3) No symmetries are specified and we require that the full set $\text{Sym}(\Gamma)$ of symmetries of Γ is respected. We think the explicit form is the natural first inclination for a computational analysis. We introduce the orbit form for games that have a concise description in verbal form, for example, using phrases such as "if either player 1 plays A or player 2 plays B, then X happens". Clearly, computing an equilibrium with the last "representation scheme" is hardest: A Nash equilibrium that respects *all* symmetries in particular respects any subset of symmetries (even if given in orbit form). Moreover, computation with the orbit form cannot be easier than with the explicit form because we can obtain the former efficiently from the latter:

Lemma 17. *Given symmetries Σ in explicit form we can compute the orbits $W(\langle\Sigma\rangle)$ it induces in time $\mathcal{O}(|\Sigma| \cdot |\mathbb{A}|)$.*

5 Finding Symmetries-Respecting Equilibria

In this section, we analyze how hard it is to find a Nash equilibrium that respects symmetries.

5.1 Complexity Results

To start with the general case, let SYM-NASH denote the search problem that takes a game Γ in explicit form, a precision parameter $\epsilon > 0$ in binary, and symmetries of Γ in orbit form. Given that, it asks for a strategy profile μ of Γ that respects the said symmetries and that is an ϵ-Nash equilibrium.

Proposition 18. SYM-NASH *is* PPAD*-complete.* PPAD*-hardness already holds for two-player games, where the symmetries are given in explicit form $\Sigma \subseteq \text{Sym}(\Gamma)$, and Σ contains $\{$just / more than$\}$ the identity symmetry.*

Proof Idea. Section 1.2 discussed the well-known idea for proving hardness. We obtain membership by fitting Nash's function to Etessami and Yannakakis (2010) framework for showing that a Brouwer fixed point problem is in PPAD. □

The membership—which, to the best of our knowledge, forms the novel contribution—shows that we can find symmetries-respecting Nash equilibria with fixed-point solvers and path-following methods, just as it is the case with finding *any* Nash equilibrium in a normal-form game. Hence, this is a positive algorithmic result. Garg et al. (2018) proved a related FIXP_a-completeness result for exact computation of a player-symmetric Nash equilibrium in a totally symmetric game of constant number of players.

Next, we narrow down our interest to SYM-NASH-TEAM, which we define as the restriction of SYM-NASH to the special case of team games, *i.e.*, games with $u^1 = \ldots = u^N$.

Theorem 8. SYM-NASH-TEAM *is* CLS*-complete.* CLS*-hardness already holds for totally symmetric team games of five players where the player symmetries that show total symmetry are given in explicit form.*

Proof Idea. We leverage a strong connection between single-player decision making under imperfect recall and decision making in a team under symmetry constraints (Lambert, Marple, and Shoham 2019), inheriting known CLS-hardness results for problems in the former setting (Tewolde et al. 2023, 2024). For membership, we show that symmetries-respecting Nash equilibria correspond to first-order stationary points (formally, *Karush-Kuhn-Tucker* (KKT) points) of the following polynomial optimization problem: Maximize the team's utility function over the polyhedral domain of symmetries-respecting strategy profiles. Fearnley et al. (2023) have shown that finding an approximate KKT point of such a problem is in CLS. □

The CLS-membership shows that first-order methods are suited to find a symmetries-respecting Nash equilibria in team games. This has already been observed for player symmetries (1) by Emmons et al. (2022) for the exact gradient descent dynamics and (2) by Ghosh and Hollender (2024) for the two-player case. Our CLS-hardness result, on the other hand, shows that gradient descent is the most efficient algorithm—modulo polynomial time speedups and barring major complexity theory breakthroughs—that is available for this problem. The main result of Ghosh and Hollender's concurrent work shows that CLS-hardness already holds for totally symmetric *two-player* games. We remark that this time the trivial symmetry set $\Sigma = \{\text{Id}_{\mathbb{A}}\}$ does not suffice for hardness, because we can find an arbitrary Nash equilibrium of a team game in linear time by going through the payoff tensor and selecting the payoff-maximizing action profile. This is different in game representations that are not normal-form: Babichenko and Rubinstein (2021)—and to a lesser extent Daskalakis and Papadimitriou (2011)—study the concisely represented *polymatrix games* and *c-polytensor games* for $c \in \mathbb{N}$. They also obtain a CLS-completeness result for the team game case, but this time it is for finding any Nash equilibrium.

5.2 Efficient Computation

In view of the hardness results in Proposition 18 and Theorem 8, we may ask why it is so popular to leverage game symmetries in game solvers, as discussed in the introduction. Indeed, restricting our attention to symmetries-respecting strategy profiles does allow for a significant dimensionality reduction in the to-be-studied profile space. Unfortunately, one (or a few) game symmetries do not allow for enough of a reduction to affect the asymptotic computational complexity. However, if the number of symmetries is vast, or equivalently, the number of distinct orbits is low, then we can show that equilibrium computation becomes easier. This generalizes a similarly derived result by Papadimitriou and Roughgarden (2008) for games with total symmetry, and we have illustrated that in the discussion of the Rock-Paper-Scissors extensions in Figure 2. Furthermore, the color coordination game in Figure 1 reduces to the simple, yet nontrivial polynomial optimization problem $\max_{r,\bar{r}\geq 0, r+\bar{r}=1} 10r^2 + 4\bar{r}^2$, where variables r and $\bar{r}$ denote the probabilities assigned to the "red" and "the other" orbit.

Theorem 9. SYM-NASH *can be solved in time* $\mathsf{poly}\big(|\Gamma|, \log(1/\epsilon), (|W|N)^{|W|}\big)$. *In two-player games, this can be improved to exact equilibrium computation in time* $\mathsf{poly}\big(|\Gamma|, 2^{|W|}\big)$.

This runtime is polynomial in the input size whenever the symmetries-induced number of orbits $|W|$ is bounded.

Proof Idea. We show that symmetries-respecting strategy profiles correspond one-to-one to "orbit profiles" in $\mathbb{R}^{|W|}$, which indicate with what probability a player plays any action in a particular orbit. This lower-dimensional space can be described efficiently, and the Nash equilibrium conditions now make a system of $\mathcal{O}(|W|)$ additional *polynomial* (in-)equalities, where each polynomial has degree at most N. Therefore, we can invoke known algorithms for solving such a system from the *the existential theory of the reals* (Renegar 1992), which will achieve the desired runtime. If it is a two-player game, we can use *support enumeration* (Dickhaut and Kaplan 1993) on orbit profiles instead. □

Lastly, we move our attention to two-player zero-sum games, which can be solved for a Nash equilibrium in polytime (von Neumann 1928; Dantzig 1951; Adler 2013). We establish that we can even ensure that *all* symmetries present in the game are respected *without* having to compute all / any symmetries of the game (which we know is as hard as GI and GA due to Proposition 9 and Theorem 1).

Theorem 10. *Given a two-player zero-sum game, we can compute a Nash equilibrium that respects all symmetries present in the game in polytime.*

Proof Idea. The set of Nash equilibria of a two-player zero-sum game is a convex polytope that can be described efficiently via a system of linear (in-)equalities (*cf.* minimax theorem, von Neumann 1928). Hence, we can solve any convex quadratic objective over this domain to *exact precision* in polytime (Kozlov, Tarasov, and Khachiyan 1980). Intuitively, we recognize that changing a strategy profile towards respecting symmetries equates to increasing its probability entropy. Formally, we analyze the "symmetric" regularizer objective $S^1 \times S^2 \to \mathbb{R}, \mathbf{s} \mapsto \sum_{a \in \mathbb{A}} f(\mathbf{s}(a))$ for an arbitrary strictly convex function $f : [0,1] \to \mathbb{R}$, *e.g.*, $x \mapsto x^2$. We prove that a Nash equilibrium that minimizes this objective also respects all symmetries present in the game. □

We believe this result can generalize to other representations or solution concepts, as long as the solution space is an efficiently describable convex compact polytope.

6 Conclusion

The concept of symmetry is rich, with many applications across the sciences, and in AI in particular. For game theory, the situation is no different. Indeed, a typical course in game theory conveys the most basic concept of a symmetric (two-player) game; to check whether it applies, no more needs to be done than taking the transpose of a matrix. But there are other, significantly richer symmetry concepts as well, ones that require relabeling players' actions or which do not allow arbitrary players to be swapped. We have studied these richer concepts. First, we studied the problem of *identifying* symmetries in games, and exhibited close connections to graph iso- and automorphism problems. We also devised performant algorithms for this task, and discussed special cases that have polytime guarantees. Second, we studied the problem of computing solutions to games that *respect* their symmetries. We have shown that requiring to respect them does not worsen the algorithmic complexity (significantly), and that it improves the complexity when the number of symmetries is vast. We also gave a strongly positive result for two-player zero-sum games.

There are many directions for future research, including the following. (1) We have focused on normal-form games. What about other ways to represent games, such as extensive-form games, stochastic games, and compact representations such as action-graph games (Jiang, Leyton-Brown, and Bhat 2011) and MAIDs (Koller and Milch 2003)? (2) We have focused on exact symmetries; what about approximate symmetries and other informative relations between players and strategies? (3) There are many conceptual questions regarding symmetries. For example, in many games, the players would benefit from being able to break the symmetries, such as in the color coordination game in the introduction, or from adopting distinct roles (say, on a soccer team). What are effective and robust ways to break symmetries to achieve better outcomes?

Acknowledgements

We are grateful to the anonymous reviewers for their valuable improvement suggestions for this paper. Emanuel Tewolde, Caspar Oesterheld, and Vincent Conitzer thank the Cooperative AI Foundation, Polaris Ventures (formerly the Center for Emerging Risk Research) and Jaan Tallinn's donor-advised fund at Founders Pledge for financial support. Caspar Oesterheld's work is also supported by an FLI PhD Fellowship. Tuomas Sandholm is supported by the Vannevar Bush Faculty Fellowship ONR N00014-23-1-2876, National Science Foundation grants RI-2312342 and RI-1901403, ARO award W911NF2210266, and NIH award A240108S001. Brian Hu Zhang's work is supported in part by the CMU Computer Science Department Hans Berliner PhD Student Fellowship.

References

Adler, I. 2013. The equivalence of linear programs and zero-sum games. *International Journal of Game Theory*, 42(1): 165–177.

Alós-Ferrer, C.; and Kuzmics, C. 2013. Hidden symmetries and focal points. *Journal of Economic Theory*, 148(1): 226–258.

Babai, L. 2016. Graph isomorphism in quasipolynomial time. In *ACM Symposium on Theory of Computing (STOC)*, 684–697.

Babichenko, Y.; and Rubinstein, A. 2021. Settling the complexity of Nash equilibrium in congestion games. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 1426–1437. ACM.

Brandt, F.; Conitzer, V.; Endriss, U.; Lang, J.; and Procaccia, A. D. 2015. *Handbook of Computational Social Choice*. Cambridge University Press.

Brown, N.; and Sandholm, T. 2018. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374): 418–424.

Brown, N.; and Sandholm, T. 2019. Superhuman AI for multiplayer poker. *Science*, 365(6456): 885–890.

Cao, Z.; and Yang, X. 2018. Symmetric games revisited. *Mathematical Social Sciences*, 95: 9–18.

Chen, X.; Deng, X.; and Teng, S. 2009. Settling the Complexity of Computing Two-Player Nash Equilibria. *J. ACM*, 56(3): 14:1–14:57.

Dantzig, G. B. 1951. A proof of the equivalence of the programming problem and the game problem. In *Activity analysis of production and allocation*, 330–335. Cowles Commission Monograph No.13.

Daskalakis, C.; Goldberg, P. W.; and Papadimitriou, C. H. 2009. The Complexity of Computing a Nash Equilibrium. *SIAM Journal on Computing*, 39(1): 195–259.

Daskalakis, C.; and Papadimitriou, C. 2007. Computing Equilibria in Anonymous Games. In *48th Symposium on Foundations of Computer Science (FOCS'07)*, 83–93.

Daskalakis, C.; and Papadimitriou, C. 2011. Continuous Local Search. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11, 790–804. Society for Industrial and Applied Mathematics.

Dickhaut, J.; and Kaplan, T. 1993. A program for finding Nash equilibria. In *Economic and Financial Modeling with Mathematica®*, 148–166. Springer.

Emmons, S.; Oesterheld, C.; Critch, A.; Conitzer, V.; and Russell, S. 2022. For Learning in Symmetric Teams, Local Optima are Global Nash Equilibria. In *International Conference on Machine Learning, ICML 2022*, volume 162, 5924–5943. PMLR.

Etessami, K.; and Yannakakis, M. 2010. On the Complexity of Nash Equilibria and Other Fixed Points. *SIAM J. Comput.*, 39(6): 2531–2597.

Fabrikant, A.; Papadimitriou, C.; and Talwar, K. 2004. The Complexity of Pure Nash Equilibria. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing (STOC '04)*, 604–612. ACM.

Fearnley, J.; Goldberg, P.; Hollender, A.; and Savani, R. 2023. The Complexity of Gradient Descent: CLS = PPAD $\cap$ PLS. *Journal of the ACM*, 70(1): 7:1–7:74.

Gabarró, J.; García, A.; and Serna, M. J. 2011. The complexity of game isomorphism. *Theoretical Computer Science*, 412(48): 6675–6695.

Gale, D.; Kuhn, H. W.; and Tucker, A. W. 1952. *On Symmetric Games*, 81–88. Princeton University Press. ISBN 9780691079349.

Garg, J.; Mehta, R.; Vazirani, V. V.; and Yazdanbod, S. 2018. $\exists\mathbb{R}$-Completeness for Decision Versions of Multi-Player (Symmetric) Nash Equilibria. *ACM Transactions on Economics and Computation*, 6(1): 1:1–1:23.

Ghosh, A.; and Hollender, A. 2024. The Complexity of Symmetric Bimatrix Games with Common Payoffs. In *20th Conference on Web and Internet Economics, WINE 2024*.

Gilpin, A.; and Sandholm, T. 2007. Lossless Abstraction of Imperfect Information Games. *Journal of the ACM*, 54(5).

Ham, N. 2021. Notions of Anonymity, Fairness and Symmetry for Finite Strategic-Form Games. *CoRR*, abs/1311.4766.

Harsanyi, J. C.; and Selten, R. 1988. *A General Theory of Equilibrium Selection in Games*. MIT Press Classics. MIT Press.

Hu, H.; Lerer, A.; Peysakhovich, A.; and Foerster, J. N. 2020. "Other-Play" for Zero-Shot Coordination. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119, 4399–4410. PMLR.

Jiang, A. X.; Leyton-Brown, K.; and Bhat, N. A. R. 2011. Action-Graph Games. *Games and Economic Behavior*, 71(1): 141–173.

Kearns, M.; Littman, M.; and Singh, S. 2001. Graphical models for game theory. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 253–260.

Koller, D.; and Milch, B. 2003. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1): 181–221.

Kozlov, M. K.; Tarasov, S. P.; and Khachiyan, L. G. 1980. The Polynomial Solvability of Convex Quadratic Programming. *USSR Computational Mathematics and Mathematical Physics*, 20(5): 223–228.

Lambert, N. S.; Marple, A.; and Shoham, Y. 2019. On equilibria in games with imperfect recall. *Games and Economic Behavior*, 113: 164–185.

Liu, S.; Marris, L.; Piliouras, G.; Gemp, I.; and Heess, N. 2024. NfgTransformer: Equivariant Representation Learning for Normal-form Games. In *The Twelfth International Conference on Learning Representations, ICLR 2024*.

Lozano, A.; and Torán, J. 1992. On the Non-Uniform Complexity of the Graph Isomorphism Problem. In *Complexity Theory: Current Research*, 245–271.

Luks, E. M. 1999. Hypergraph isomorphism and structural equivalence of boolean functions. In *ACM Symposium on Theory of Computing (STOC)*, 652–658.

Marris, L.; Gemp, I.; Anthony, T.; Tacchetti, A.; Liu, S.; and Tuyls, K. 2022. Turbocharging Solution Concepts: Solving NEs, CEs and CCEs with Neural Equilibrium Solvers. In *Advances in Neural Information Processing Systems 35, NeurIPS 2022*.

Mathon, R. 1979. A Note on the Graph Isomorphism Counting Problem. *Information Processing Letters*, 8(3): 131–132.

Miller, G. L. 1977. Graph isomorphism, general remarks. In *ACM Symposium on Theory of Computing (STOC)*, 143–150.

Moulin, H. 2004. *Fair Division and Collective Welfare*. The MIT Press. ISBN 9780262633116.

Nash, J. 1951. Non-Cooperative Games. *Annals of Mathematics*, 54(2): 286–295.

Nash, J. F. 1950. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1): 48–49.

Nisan, N.; Roughgarden, T.; Tardos, É.; and Vazirani, V. V., eds. 2007. *Algorithmic Game Theory*. Cambridge University Press. ISBN 9780511800481.

Papadimitriou, C. H. 1994. On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence. *Journal of Computer and System Sciences*, 48(3): 498–532.

Papadimitriou, C. H.; and Roughgarden, T. 2008. Computing correlated equilibria in multi-player games. *J. ACM*, 55(3).

Rawls, J. 1971. *A Theory of Justice*. The Belknap press of Harvard University Press.

Renegar, J. 1992. On the computational complexity and geometry of the first-order theory of the reals. Part I: Introduction. Preliminaries. The geometry of semi-algebraic sets. The decision problem for the existential theory of the reals. *Journal of symbolic computation*, 13(3): 255–299.

RPS. 2024. Rock-Paper-Scissors with additional weapons. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Rock_paper_scissors#Additional_weapons, retrieved 2024-12-19.

Samuel, A. L. 1959. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3): 210–229.

Schelling, T. C. 1960. *The Strategy of Conflict*. Harvard University Press.

Schöning, U. 1988. Graph isomorphism is in the low hierarchy. *Journal of Computer and System Sciences*, 37(3): 312–323.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T. P.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; and Hassabis, D. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587): 484–489.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; Chen, Y.; Lillicrap, T. P.; Hui, F.; Sifre, L.; van den Driessche, G.; Graepel, T.; and Hassabis, D. 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676): 354–359.

Sims, C. C. 1978. Some group-theoretic algorithms. In *Topics in Algebra: Proceedings, 18th Summer Research Institute of the Australian Mathematical Society*, 108–124. Springer.

Tewolde, E.; and Conitzer, V. 2024. Game Transformations That Preserve Nash Equilibria or Best-Response Sets. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024*.

Tewolde, E.; Oesterheld, C.; Conitzer, V.; and Goldberg, P. W. 2023. The Computational Complexity of Single-Player Imperfect-Recall Games. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*, 2878–2887.

Tewolde, E.; Zhang, B. H.; Oesterheld, C.; Zampetakis, M.; Sandholm, T.; Goldberg, P. W.; and Conitzer, V. 2024. Imperfect-Recall Games: Equilibrium Concepts and Their Complexity. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024*.

Treutlein, J.; Dennis, M.; Oesterheld, C.; and Foerster, J. N. 2021. A New Formalism, Method and Open Issues for Zero-Shot Coordination. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, 10413–10423. PMLR.

van Damme, E. 1997. *Equilibrium Selection in Team Games*, 100–110. Springer. ISBN 978-3-642-60495-9.

von Neumann, J. 1928. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100: 295–320.

von Neumann, J.; and Morgenstern, O. 1944. *Theory of Games and Economic Behavior*. Princeton University Press.