# 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

## 1.1 Motivation

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, sometimes interactions with symmetries can be described more concisely
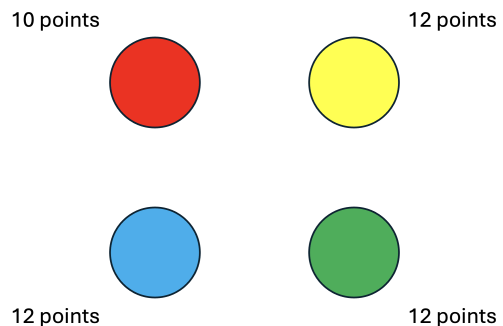


Figure 1: A two-player coordination game. If both players pick the same color, they each receive the associated utility points. If they miscoordinate, both receive 0 points. Without knowing who you are playing with, what color would you choose?

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.

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 forms 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; Emmons et al. 2022)). Ham (2021), and the 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

---

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

| | Game symmetries | | Game isomorphisms | | Computing a symmetric equilibrium |
|---|---|---|---|---|---|
| General games | GA-c; (Th. 1) | XP$\left(\frac{\#actions}{\#players}\right)$ (Th. 5) | GI-c; (GGS11) | XP$\left(\frac{\#actions}{\#players}\right)$ (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 (Prop. 9) | | | | 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. XP$(k)$ stands for runtimes in which the only exponent is $k$.

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 leveraging the player symmetry in chess and Go by always orienting the board from the moving player's perspective, Silver et al. also exploit 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, for example, 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.

---

[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.

Some strategic interactions may also force symmetric play across different decision points. This could be, for instance, because multiple agents (say, self-driving cars) run the same software for taking decisions. In another example, an agent may not recall being in the same situation before (*absentmindedness*) because, *e.g.*, the agent does not retain any record of its history. In this case, it will necessarily act in the same fashion as it did before. Theorem 8 makes precise and exploits that there does not appear to be a sharp distinction between (1) being absentminded, (2) being multiple copies of the same agent, and (3) symmetric agents playing symmetries-respecting profiles. We illustrate the underlying argument for it in Appendix B on a variant of an example that Kovařík et al. [2023, 2025] give for games with simulation of other players. A broader discussion of the game-theoretic aspects of these AI settings can be found in (Conitzer 2019; Conitzer and Oesterheld 2023).

Last but not least, 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 Appendix C.

## 1.2 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 10 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 computational 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 1.3, and accompany it with additional insights.

Full proofs can be found in the appendix.

## 1.3 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 $\Gamma$, 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 re-

---

[3]Concretely, it requires parsing the full payoff matrix while recognizing that payoffs are in $[0, 1]$ in A. This takes linear time.
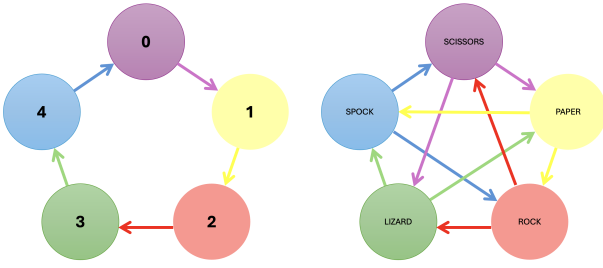
Figure 2: Two extensions to Rock-Paper-Scissors. In both, there is only one symmetries-respecting strategy profile. The right game is known as Rock-Paper-Scissors-Lizard-Spock.

spects 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 when an action wins in this game for $m = 5$, and it shows another variant that is prominently referenced in pop culture (Wikipedia contributors 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.

**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 $\Gamma$ *consists of*

1. *A finite set of* players $\mathcal{N} := \{1, \ldots, N\}$, *where* $N \geq 2$ *denotes the number of players,*
2. *A finite set of* actions $A^i := \{1, \ldots, m^i\}$ *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, $\mathbf{a} = (\mathbf{a}_1, \ldots, \mathbf{a}_N) \in A^1 \times \cdots \times A^N =: A$. 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* $\Gamma$ *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 $\Gamma$.

**Definition 3** (Nash 1951). *A* (game) symmetry *of a game* $\Gamma$ *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 $\phi$ 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 $\pi$ is henceforth referred to as a *player permutation*. By abuse of notation, $\phi$ can then be considered to map an action profile $\mathbf{a} \in A$ to action profile $\phi(\mathbf{a}) := (\phi(\mathbf{a}_{\pi^{-1}(j)}))_{j \in \mathcal{N}} \in A$.[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* $\Gamma$ *is called* totally symmetric *if each player has the same action set* $A^*$, *and if for any player permutation* $\pi$ *we have* $u^i(\mathbf{a}) = u^{\pi(i)}((\mathbf{a}_{\pi^{-1}(j)})_{j \in \mathcal{N}})$ *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 $\phi$ is represented in *explicit form*, which, again, means as a look-up table of evaluations of $\phi$. Unlike with games, this explicit representation does not blow up exponentially with the number of players since $\phi$ only has $|\mathbb{A}|$ evaluations.

Note that the identity map $\text{Id}_{\mathbb{A}}$ is always a game symmetry (henceforth the *trivial symmetry*), and that two symmetries

---

[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 $\pi(j)$.

$\phi$ and $\phi'$ compose to a third symmetry. Moreover, if $\phi$ is a symmetry, then $\phi^{-1}$ is one as well. Therefore:

**Remark 5.** *The set* $\mathrm{Sym}(\Gamma)$ *of symmetries of a game* $\Gamma$ *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 nontrivial 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 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 consider two simple cases. We start by observing that a game is totally symmetric if and only if swapping any two players $i, j \in \mathcal{N}$ forms a game symmetry. This yields the following result.

**Proposition 8.** *We can determine whether a game is totally symmetric, for $A^* = A^1 = \cdots = A^N$ with the current action numbering, in polytime $\mathcal{O}(N^2 \cdot |A|)$.*

Next, we study a type of symmetry we have not discussed yet. Let us call $\phi$ a *1PL-actions* symmetry of $\Gamma$ if it merely permutes the actions of a single player, *i.e.*, there is $i \in \mathcal{N}$ such that $\phi|_{\mathbb{A}\setminus A^i} = \mathrm{Id}_{\mathbb{A}\setminus A^i}$. We show that those are generated by symmetries $\phi'$ that expose action duplicates, *i.e.*, that swap two actions $a, a' \in A^i$ for a player $i$ and keep the rest fixed. There are $\mathcal{O}(N \cdot \max_i(m^i)^2)$ such symmetries.

**Proposition 9.** *We can compute (a generator set of) the group of 1PL-actions symmetries of a game $\Gamma$ in polytime $\mathcal{O}(N \cdot \max_i(m^i)^2 \cdot |A|)$.*

---

[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 $Z$.

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

### 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 $\phi$ whose $\pi = \mathrm{Id}_{\mathcal{N}}$. We will show that such symmetries are already hard to characterize, let alone the whole set of symmetries $\mathrm{Sym}(\Gamma)$—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 $\{(i, a) : i \in \mathcal{N}, a \in A^i\} \cup \{(a, \mathbf{a}) : a \in \mathbf{a} \in A\} \cup \{(i, \mathbf{a}) : i \in \mathcal{N}, \mathbf{a} \in A\}$. 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. $\square$

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 $\Gamma$ and $\Gamma'$ and are asking for a player- and utility-preserving map $\phi : \mathbb{A} \to \mathbb{A}'$; see Appendix D for a formal definition. 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 $\mathrm{Sym}(\Gamma)$ and automorphisms $\mathrm{Aut}(G)$ of associated game-graph pair $(\Gamma, G)$ are isomorphic in a group-theoretic sense. Therefore, we can inherit further known results about graph automorphisms for our setting (Mathon 1979).

**Proposition 10.** *The following problems for a game $\Gamma$ are polynomial-time Turing-equivalent to* GI*: (a) determining a generator set of* $\mathrm{Sym}(\Gamma)$*, and (b) determining the cardinality of* $\mathrm{Sym}(\Gamma)$ *. 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 $\Gamma$ has a symmetry $\phi$ 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 $\Gamma$ is* name-irrelevant symmetric, *that is, whether for all possible player permutations $\pi : \mathcal{N} \to \mathcal{N}$ there is symmetry $\phi \in$ Sym$(\Gamma)$ of $\Gamma$ 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 Appendix F for the treatment of game isomorphisms.

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

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 $2^{\mathcal{O}(|V|)}$-time algorithm for hypergraph automorphism due to Luks (1999).

**Corollary 11.** *For games in which the number of actions per player is bounded, we can compute* 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 $\phi$ is called a *player symmetry* if it keeps the action labels "fixed", that is, if it sends action $k \in \{1, \ldots, m^i\}$ of player $i$ to action $k$ of player $\pi(i)$.

**Theorem 6.** *We can compute (a generator set of) the group of player symmetries of a game $\Gamma$ 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.

The positive results of Corollary 11 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: $\mathbf{s} = (\mathbf{s}_1, \ldots, \mathbf{s}_N) \in S^1 \times \cdots \times S^N =: S$. 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, $\mathbf{s}_{-i} \in \times_{j \neq i} S^j =: S^{-i}$ 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 $\Gamma$ 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 $\phi$ 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 $\phi$ will then also satisfy the invariance $u^i(\mathbf{s}) = u^{\pi(i)}(\phi(\mathbf{s}))$. Furthermore, we have:

**Remark 14.** *For any symmetry $\phi$ of $\Gamma$, 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 $\Sigma \subseteq$ Sym$(\Gamma)$ be a particular set of symmetries that we want to respect. This could be the trivial set $\{\text{Id}\}$, in which case no symmetries need to be respected, or the full set 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 $\Sigma \subseteq$ Sym$(\Gamma)$ if for all $\phi \in \Sigma$ we have $\phi(\mathbf{s}) = \mathbf{s}$.*

**Theorem 7** (Nash 1951). *Any game $\Gamma$ 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 $\Sigma$ 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 $\Sigma$. 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 $\epsilon$-Nash equilibrium must then satisfy $u^i(\mathbf{s}) \geq u^i(s, \mathbf{s}_{-i}) - \epsilon$ in Definition 13. We want $\epsilon$ 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, $\epsilon$ 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 $\Sigma \subseteq \mathrm{Sym}(\Gamma)$ 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 $\Sigma \subseteq \mathrm{Sym}(\Gamma)$. (3) No symmetries are specified and we require that the full set $\mathrm{Sym}(\Gamma)$ of symmetries of $\Gamma$ 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 $\Sigma$ 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 $\Gamma$ in explicit form, a precision parameter $\epsilon > 0$ in binary, and symmetries of $\Gamma$ in orbit form. Given that, it asks for a strategy profile $\mu$ of $\Gamma$ that respects the said symmetries and that is an $\epsilon$-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 $\Sigma \subseteq \mathrm{Sym}(\Gamma)$, and $\Sigma$ contains $\{$just / more than$\}$ the identity symmetry.*

*Proof Idea.* Section 1.3 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 $\mathrm{FIXP}_a$-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 $\Sigma = \{\mathrm{Id}_{\mathbb{A}}\}$ 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* $\mathrm{poly}\big(|\Gamma|, \log(1/\epsilon), (|W|N)^{|W|}\big)$. *In two-player games, this can be improved to exact equilibrium computation in time* $\mathrm{poly}\big(|\Gamma|, 2^{|W|}\big)$.

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 10 and Theorem 1).

**Theorem 10.** *Given a two-player zero-sum game, we can compute an exact 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 (and briefly, graphical 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

## 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.

Bilò, V.; and Mavronicolas, M. 2021. Existential-R-Complete Decision Problems about (Symmetric) Nash Equilibria in (Symmetric) Multi-player Games. *ACM Transactions on Economics and Computation*, 9(3).

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

Brandt, F.; Fischer, F.; and Holzer, M. 2009. Symmetries and the complexity of pure Nash equilibrium. *Journal of Computer and System Sciences*, 75(3): 163–177.

Brandt, F.; Fischer, F. A.; and Holzer, M. 2011. Equilibria of graphical games with symmetries. *Theor. Comput. Sci.*, 412(8-10): 675–685.

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.

Cheng, S.-F.; Reeves, D. M.; Vorobeychik, Y.; and Wellman, M. P. 2004. Notes on Equilibria in Symmetric Games. In *The 6th International Workshop On Game Theoretic And Decision Theoretic Agents (GTDT 2004)*, 71–78.

Conitzer, V. 2019. Designing Preferences, Beliefs, and Identities for Artificial Intelligence. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, 9755–9759.

Conitzer, V.; and Oesterheld, C. 2023. Foundations of Cooperative AI. In *Thirty-Seventh AAAI Conference on Artificial Intelligence*, 15359–15367. AAAI Press.

Conitzer, V.; and Sandholm, T. 2008. New complexity results about Nash equilibria. *Games and Economic Behavior*, 63(2): 621–641.

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 ∩ 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. ∃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.

Kontogiannis, S. C.; and Spirakis, P. G. 2011. Approximability of Symmetric Bimatrix Games and Related Experiments. In Pardalos, P. M.; and Rebennack, S., eds., *Experimental Algorithms - 10th International Symposium, SEA 2011, Kolimpari, Chania, Crete, Greece, May 5-7, 2011. Proceedings*, volume 6630 of *Lecture Notes in Computer Science*, 1–20. Springer.

Kovařík, V.; Oesterheld, C.; and Conitzer, V. 2023. Game Theory with Simulation of Other Players. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023*, 2800–2807. ijcai.org.

Kovařík, V.; Sauerberg, N.; Hammond, L.; and Conitzer, V. 2025. Game Theory with Simulation in the Presence of Unpredictable Randomisation. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025*.

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.

Mehta, R.; Vazirani, V. V.; and Yazdanbod, S. 2015. Settling Some Open Problems on 2-Player Symmetric Nash Equilibria. In *8th International Symposium on Algorithmic Game Theory, SAGT 2015*, volume 9347, 272–284. Springer.

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.

Reny, P. J. 1999. On the Existence of Pure and Mixed Strategy Nash Equilibria in Discontinuous Games. *Econometrica*, 67(5): 1029–1056.

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.

Wikipedia contributors. 2024. Rock paper scissors with additional weapons — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/wiki/Rock_paper_scissors#Additional_weapons. [Online; retrieved 2024-12-19].

## A  More Proof Idea Sketches

Here, we give some additional proof idea sketches that are missing in the main body due to space constraints.

*Proof Idea of Theorem 3.* Construct the graph $G$ with node set $\mathcal{N} \sqcup \mathbb{A} \sqcup A$ as in Theorem 1, and then check whether $G$ has an isomorphism that maps any player $i \in \mathcal{N}$ to a different player. We show this to reduce to GI.

We show hardness with an independent proof: Given graphs $G_1$ and $G_2$, create a zero-sum game of two players, each with action set $V_1 \cup V_2$. When both players pick vertices from different graphs, nothing happens. In the case where both choose vertices from the same graph $G_i$, player $i$ will strongly prefer to choose a neighbor of the other player's chosen vertex, and gets slightly punished instead if they do not manage to do so. □

*Proof Idea of Theorem 4.* Hardness follows from Theorem 3. For inclusion, check whether, for every pair of players $i$ and $j$, the graph $G$ constructed in Theorem 1 has an automorphism $\phi$ whose restriction to $\mathcal{N}$ is the transposition $(ij)$. We can show that this reduces to GI. □

## B  On Simulation and Imperfect Recall

Let us describe the connections between (1) being absentminded, (2) being multiple copies of the same agent, and (3) symmetric agents playing symmetries-respecting profiles. We illustrate it on a variant of an example that Kovařík et al. [2023, 2025] give for games with simulation of other players: Consider the perspective of an AI agent that handles a user's financial assets. It can either invest them, giving the agent utility 1, or steal them, giving the agent utility 3. However, the user is not naïve, and decides to first run the agent *in simulation*, but only once. If the user catches the agent stealing in simulation, the agent will not actually be given control of the assets, and will walk away with utility 0. Crucially, the agent is unable to tell the difference between being run in simulation and being run in the real world. Therefore, it must steal with some probability $p$ whenever called upon to act, and will thus receive expected utility $(1-p)^2 + 3p(1-p) = -2p^2 + p + 1$. Hence, it is optimal for the agent to steal with probability $1/4$ for an expected utility of $9/8$. We can capture this decision problem by modeling the agent as being absentminded, since it will not remember having been in the same situation before. Alternatively, we can also model the decision problem as a game between two distinct players that form a team, one acting first in simulation and one acting second in the real world. In order to do so, we shall randomly assign the two roles (simulated or real) to the two players without revealing the assignment to them. This creates a symmetry between the two players. If we then require that the players respect this symmetry, they will both choose to steal with the same probability $p$. Indeed, without the random and secret role assignment, the team can achieve utility 3 by letting the player in simulation invest and the player in the real world steal. Without the constraint to respect the symmetry, the team prefers to have one player invest and the other player steal, because

then they steal successfully 50% of the time, achieving utility $3/2 > 9/8$. Hence, both modeling steps are needed to accurately reflect the decision problem described.

## C  Other Related Work

In this section, we describe further related work.

First, let us give more context on the open conjecture we resolved in Theorem 4: Cao and Yang (2018), more generally, studied total symmetric, *renaming symmetric*, and name-irrelevant symmetric games from a structural lens. A game is called renaming symmetric if one can "relabel" the action set of each action set, such that—if one considers them as the same action set $A^*$—the relabeled game is totally symmetric. They derive GI-completeness for deciding whether a two-player team game with $\{0,1\}$ utility payoffs is renaming symmetric.

Conitzer and Sandholm (2008), Kontogiannis and Spirakis (2011), Garg et al. (2018), and Bilò and Mavronicolas (2021) study various decision and approximation problems involving symmetric Nash equilibria in totally symmetric games, such as counting or maximizing payoffs and many more. Mehta, Vazirani, and Yazdanbod (2015) show that it is NP-complete to decide whether a non-symmetric Nash equilibrium exists in a totally symmetric (bimatrix) game. Brandt, Fischer, and Holzer (2009) study concisely represented totally symmetric games, anonymous games, and minor variants, in terms of the complexity of finding a pure Nash equilibria. They continue this study in follow-up work (Brandt, Fischer, and Holzer 2011) with a focus on graphical games. Cheng et al. (2004) show that (possibly asymmetric) pure Nash equilibrium exist in 2-action totally symmetric games, and how symmetries can boost equilibrium solvers based on Nelder-Mead variants or replicator dynamics methods. Related to another existence result of theirs, Reny (1999) shows existence of symmetric Nash equilibrium in games that have particular kinds of player symmetries, and whose action spaces $A^i$ are compact subsets of a topological vector spaces (among other conditions).

## D  Further Background on Section 2

In this section, we define the game isomorphism problem, and give more background on the graph isomorphism and automorphism literature.

### D.1  Notation

Analogous to strategies, we write $\mathbf{a}_{-i} \in A^{-i}$ for the parts in $\mathbf{a}$ and $A$ that are associated to the other players from $i$.

### D.2  Game Isomorphisms

**Definition 19** (Gabarró, García, and Serna 2011)**.** *A* (game) symmetry *between two games* $\Gamma = (\mathcal{N}, (A^i)_{i \in \mathcal{N}}, (u^i)_{i \in \mathcal{N}})$ *and* $\Gamma' = (\mathcal{N}', (A'^i)_{i \in \mathcal{N}}, (u'^i)_{i \in \mathcal{N}})$ *is a bijective map* $\phi : \mathbb{A} \to \mathbb{A}'$ *satisfying*

1. *actions of the same player are mapped to the same player, that is, for each* $i \in \mathcal{N}$, *there is* $\pi(i) \in \mathcal{N}'$ *with* $a, a' \in A^i \implies \phi(a), \phi(a') \in A'^{\pi(i)}$

*[In particular, bijective $\phi$ then induces a bijective player map $\pi : \mathcal{N} \to \mathcal{N}$ and bijective action set maps $\phi^i := \phi|_{A^i} : A^i \to A'^{\pi(i)}$. In particular, action profiles $\mathbf{a} \in A$ are mapped to action profiles $\phi(\mathbf{a}) := \big(\phi(\mathbf{a}_{\pi^{-1}(j)})\big)_{j \in \mathcal{N}'} \in A'$.*

*2. payoffs are invariant under the isomorphism, that is, $u^i(\mathbf{a}) = u'^{\pi(i)}(\phi(\mathbf{a}))$ for all $i \in \mathcal{N}$ and $\mathbf{a} \in A$.*

Just as with game symmetries, we assume a game isomorphism is represented in *explicit form*, that is, as a look-up table of evaluations of $\phi$.

## D.3 Graph Isomorphisms and Automorphism

First, a comment on why there are indeed generator sets of logarithmic size: If set $Z = \{z_1, \ldots, z_k\}$ is an inclusion-wise minimal generating set of group $G$, then $Z$ generates at least $2^k$ distinct elements, namely $z_S := \prod_{i \in S} z_i$ for each subset $S \subseteq [k]$.

An *edge-labeled* (hereafter simply *labeled*) *graph* is a tuple $G = (V, E_1, \ldots, E_k)$, where $E_i$ is the set of edges with label $i$. Two labeled graphs are isomorphic if there is an edge-label-preserving isomorphism between them. In this section, we state some known or simple results about various problems related to edge-labeled graphs.

**Proposition 20.** *The following decision problems are* GI-*complete:*

1. *The isomorphism problem for edge-labeled graphs*
2. *Given $k$ pairs of (labeled) graphs $(G_1, H_1), \ldots, (G_k, H_k)$, decide whether $G_i \cong H_i$ for all $i$*
3. *Given $k$ pairs of (labeled) graphs $(G_1, H_1), \ldots, (G_k, H_k)$, decide whether $G_i \cong H_i$ for any $i$*
   *(Note that for $k = 1$ both of the previous problems are just graph isomorphism for labeled graphs)*
4. *Given two (labeled) graphs $G, H$ and pairs $(u_1, v_1), \ldots, (u_k, v_k) \in V_G \times V_H$, decide whether there exists an isomorphism $\phi : V_G \to V_H$ such that $\phi(u_i) = v_i$ for all $i$.*

*Proof.* In all cases, hardness follows because GI is a special case. So lets move to membership:

1. This is a special case of Theorem 2 of Miller (1977), where the edge sets $E_i$ form what they call *relations $R_i$*.
2.+3. These problems are known to be in GI for simple graphs (Lozano and Torán 1992). The labeled version then follows by combining with the previous item.
4. Assume that the $u_i$'s are all different and that the $v_j$'s are all different; otherwise, either the instance is trivially false or there are duplicate pairs, which can be removed. Let $G^{u_1, \ldots, u_k}$ be the graph $G$ except with a long "chain" / "line" of nodes (say, of length $\ell_i \geq V(G)$ for $1 \leq i \leq k$, with the $\ell_i$'s all different) of new vertices attached to $u_i$ for each $i$. Then an isomorphism with the desired property exists if and only if $G^u \cong G^v$. $\square$

**Proposition 21.** *The automorphism problem for edge-labeled graphs is* GA-*complete.*

*Proof.* Follows immediately from a construction identical to Theorem 2 of Miller (1977). We also note that the association between automorphism in the graph and and automorphisms in the corresponding edge-labeled graph is a group isomorphism itself, hence questions about the cardinality and generator sets of the automorphism groups correspond as well. $\square$

## E On Section 3 up until Section 3.1

In this section, we provide proofs to the results of Section 3 excluding Section 3.2.

**Proposition 8.** *We can determine whether a game is totally symmetric, for $A^* = A^1 = \cdots = A^N$ with the current action numbering, in polytime $\mathcal{O}(N^2 \cdot |A|)$.*

*Proof.* First, we can test whether indeed $m^1 = \cdots = m^N$ in linear time. Next, observe that a game is totally symmetric iff all player permutations $\pi \in \mathcal{S}_{\mathcal{N}}$ are player symmetries iff all player swaps $\sigma_{ij}$ for $i, j \in \mathcal{N}$ are player symmetries (since they generate $\mathcal{S}_{\mathcal{N}}$). The latter can be checked in time $\mathcal{O}(N^2 \cdot |A|)$. $\square$

**Proposition 9.** *We can compute (a generator set of) the group of 1PL-actions symmetries of a game $\Gamma$ in polytime $\mathcal{O}(N \cdot \max_i(m^i)^2 \cdot |A|)$.*

*Proof.* Suppose $\phi$ is a 1PL-actions symmetry for player $i$ that maps action $a \in A^i$ to $a' \in A^i$. Let $\mathbf{a}_{-i} \in A^{-i}$ be arbitrary. Then

$$u^i(a, \mathbf{a}_{-i}) = u^{\pi(i)}(\phi(a, \mathbf{a}_{-i})) = u^i(a', \mathbf{a}_{-i}).$$

This implies that $a$ and $a'$ are duplicate actions, always achieving the same payoffs. In particular, the map $\phi'$ that exposes the duplicates $a, a'$ is also a symmetry. But then, any 1PL-actions symmetry for player $i$ is identified by how it permutes $A^i$, which is generated by the swaps of any two actions in $A^i$ (transpositions). Of those swaps, there are $N \cdot \max_i(m^i)^2$ many. Checking if a single swap makes a symmetry takes $\mathcal{O}(|A|)$ time. $\square$

**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.*

*Proof.*

**Membership** Given a game $\Gamma$, create a labeled graph $G$ with node set $\mathcal{N} \cup \mathbb{A} \cup A$. Let $\lambda \neq \lambda' \in \mathbb{N}$ be values that do not occur as payoffs anywhere in $\Gamma$. We want the following edges in the graph:

1. $\{i, a\}$ with label $\lambda$ for any pair $i \in \mathcal{N}$ and $a \in A^i$
2. $\{a, \mathbf{a}\}$ with label $\lambda'$ for any pair $\mathbf{a} \in A$ and $a \in \mathbf{a}$, and
3. $\{i, \mathbf{a}\}$ with label $u^i(\mathbf{a})$ for any pair $i \in \mathcal{N}$ and $\mathbf{a} \in A$.

We can define a map $M$ that maps a symmetry $\phi$ of $\Gamma$ to an automorphism $M(\phi) := \psi$ of $G$. It is naturally defined by $\psi(i) := \pi(i)$, $\psi(a) := \phi(a)$, and $\psi(\mathbf{a}) := \phi(\mathbf{a})$. Then the labeled edges remain invariant under $\psi$, with the only point to note that this association indeed yields for any edge $\{i, \mathbf{a}\}$

that $u^i(\mathbf{a}) = u^{\pi(i)}(\phi(\mathbf{a}))$, and thus $\{\psi(i), \psi(\mathbf{a})\}$ is the same labeled edge in $G$.

We can show that $M$ is a bijection from symmetries of $\Gamma$ to automorphisms of $G$. Injectivity follows immediately. $M$ is also surjective. Say $\psi$ is a graph automorphism of $G$. We first show that $\psi(\mathcal{N}) \subseteq \mathcal{N}$, $\psi(A^i) \subseteq \mathbb{A}$, and $\psi(A) \subseteq A$, while simultaneously defining the preimage symmetry candidate $\phi$ of $\Gamma$. Let $i \in \mathcal{N}$. Then for $a \in A^i$ and $\mathbf{a} \in A$, we have edges $\{i, a\}$ and $\{i, \mathbf{a}\}$. Hence, $\{\psi(i), \psi(a)\}$ and $\{\psi(i), \psi(\mathbf{a})\}$ are also edges in $G$ with the same respective labels $\lambda$ and $u^i(\mathbf{a})$. The only vertices in $G$ that has neighbors as $\psi(i)$ are the ones in $\mathcal{N}$, hence $\psi(i) \in \mathcal{N}$. Hence, also $\psi(a) \in A^{\psi(i)} \subset \mathbb{A}$, and $\psi(\mathbf{a}) \in A$. Since $\psi$ is a bijection, we get that $\psi$ induces bijections $\pi := \mathcal{N} \to \mathcal{N}$, $\phi := \mathbb{A} \to \mathbb{A}$, and $\phi|_{A^i} = A^i \to A^{\pi(i)}$. Through edges $\{a, \mathbf{a}\}$ and $\{\psi(a), \psi(\mathbf{a})\}$, we also obtain that if $a \in A^i$, then $A^{\pi(i)} \ni \phi(a) = \psi(a) \in \psi(\mathbf{a})$. Hence, $\psi(\mathbf{a})$ is equal to $\phi(\mathbf{a}) := \big( \phi(\mathbf{a}_{\pi^{-1}(j)}) \big)_{j \in \mathcal{N}} \in A$. It is clear that $\phi$ is indeed the preimage of $psi$ under $M$, i.e., $M(\phi) = \psi$. In order to complete showing that $\phi$ indeed makes a symmetry, it remains to highlight that for $i$ and $\mathbf{a}$, we have $u^i(\mathbf{a}) = u^{\pi(i)}(\phi(\mathbf{a}))$ because $\{i, \mathbf{a}\}$ and $\{\psi(i), \{\psi(\mathbf{a})\} = \{\pi(i), \{\phi(\mathbf{a})\}$ have the same edge labels.

Overall, we have that $M$ is a bijective, and we observe that we have correspondence $M(\mathrm{Id}_{\mathbb{A}}) = \mathrm{Id}_V$ for the identity symmetry and identity automorphism. Using Proposition 21, we therefore get GA-membership.

**Hardness** Let $G = (V, E)$ be a graph. Create a game $\Gamma$ of two players. PL1's action set shall be $V$, and PL2's action set $V \cup \{\alpha\}$, where action 0 is added as a dummy action so that all symmetries of $\Gamma$ will have to keep player identities fixed, as we will see later on. In terms of utility payoffs, we have quite some flexibility: PL1 will receive one of four distinct utility payoffs $\{c_k\}_{k=1}^4 \subset \mathbb{R}$, and PL2 will receive one of four distinct utility payoffs $\{d_k\}_{k=1}^4 \subset \mathbb{R}$. To give specific examples, if we want $\Gamma$ to be a team game, we might choose $c_k = k = d_k$ for all $k$; if we want $\Gamma$ to be to be a zero-sum game, we might choose $c_k = k = -d_k$ for all $k$. The players receive payoffs as follows:

(1) a payoff of $c_1$ and $d_1$ respectively if PL2 chooses action $\alpha$,

(2) a payoff of $c_2$ and $d_2$ respectively if the players choose distinct vertices $u, v \in V$ that are not neighbouring ($\{u, v\} \notin E$),

(3) a payoff of $c_3$ and $d_4$ respectively if the players choose distinct vertices $u, v \in V$ that are neighbouring ($\{u, v\} \in E$), and

(4) a payoff of $c_4$ and $d_4$ respectively if the players choose the same vertex $v \in V$.

We can define a map $M$ that maps a symmetry $\psi$ of $G$ to a symmetry $M(\psi) := \phi$ that maps player $i$ to $i$, action $v \in V$ to $\psi(v)$ for both players, and action $\alpha$ to $\alpha$ for PL2. Then each case of payoffs are mapped to themselves: With cases (1) and (4) this is by definition, with cases (2) and (3) this follows from vertices being neighbors if and only if they are after applying $\psi$. So $M(\psi)$ is a game symmetry of $\Gamma$.

We can again show that $M$ is a bijection. Injectivity again follows immediately. Let us consider surjectivity. We claim that if $\phi$ is a symmetry of the game $\Gamma$, then $\phi|_{A^1}$ is an automorphism of $G$ such that $M(\psi^1) = \phi$. So let $\phi$ be a symmetry of $\Gamma$. Then:

1. The players have action sets of different cardinality. This eliminates the possibility that $\pi(1) = 2$ because in that case $\phi|_{A^1}$ would have to be a bijection between the action sets. Thus, $\pi$ Thus, $\pi$ maps $i$ to $i$.

2. In particular, $\phi|_{A^1}$ is therefore a permutation of the vertex set $V$.

3. The players have to receive a utility of $c_1$ and $d_1$ again when PL2 plays $\phi(\alpha)$. This would only be the case for action $\alpha$, thus $\phi(\alpha) = \alpha$.

4. The players have to receive a utility of $c_4$ and $d_4$ again when, for any $v \in V$, the players play $\phi|_{A^1}(v)$ and $\phi|_{A^2}(v)$. This implies $\phi|_{A^1}(v) = \phi|_{A^2}(v)$ in $V$.

5. The players have to receive a utility of $c_3$ and $d_3$ (resp. $c_2$ and $d_2$) again when, for any distinct $u, v \in V$ with $\{u, v\} \in E$ (resp. $\{u, v\} \notin E$), the players play $\phi|_{A^1}(u)$ and $\phi|_{A^2}(v) = \phi|_{A^1}(v)$. Therefore, we must have $\{\phi|_{A^1}(u), \phi|_{A^1}(v)\} \in E$ (resp. $\{\phi|_{A^1}(u), \phi|_{A^1}(v)\} \notin E$) in that case as well.

Together, this implies that $\phi|_{A^1}$ is a symmetry of $G$, and by definition of $M$, $M(\phi|_{A^1}) = \psi$.

Overall, we have that $M$ is a bijective, and we observe that we have correspondence $M(\mathrm{Id}_V) = \mathrm{Id}_{\mathbb{A}}$ for the identity automorphism and identity symmetry. This yields GA-hardness. $\square$

**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.*

*Proof.* We can use the same proofs as for Theorem 1. Except now, we start with two games $\Gamma, \Gamma'$ (resp. graphs $G, G'$), and construct each of their corresponding graphs $G, G'$ (resp. games $\Gamma, \Gamma'$). Then, by similar reasoning to the previous proof, we have that the latter pair is isomorphic if and only if the former pair is. In the membership proof, we need to make sure that we use the same $\lambda, \lambda'$ in $G$ and $G'$ and they are different from any payoffs in $G$ and $G'$. In the hardness proof, we need to choose the same values $\{c_k, d_k\}_{k=1}^4$ for both $\Gamma$ and $\Gamma'$. $\square$

**Proposition 10.** *The following problems for a game $\Gamma$ are polynomial-time Turing-equivalent to* GI: *(a) determining a generator set of* $\mathrm{Sym}(\Gamma)$, *and (b) determining the cardinality of* $\mathrm{Sym}(\Gamma)$ . *Hardness already holds for two-player {zero-sum / team} with only player-separable symmetries.*

*Proof.* Taking another look at the proof of Theorem 1, we have that $M$ commutes with composition of game symmetries and graph automorphisms, that is, $M(\phi' \circ \phi) = M(\phi') \circ M(\phi)$ in the membership proof and $M(\psi' \circ \psi) = M(\psi') \circ M(\psi)$ in the hardness proof. This together with the previously observed characteristics of $M$, implies both

$M$'s are group isomorphisms between the symmetry group $\text{Sym}(\Gamma)$ and the automorphism group $\text{Aut}(G)$ of corresponding $\Gamma$ and $G$ in these two proofs. Therefore, they have equal cardinality and generator sets. Next, transition to the same problems for standard (unlabeled) graphs with Proposition 21. Those are polynomial-time Turing-equivalent to GI by Mathon (1979). □

**Theorem 3.** *Deciding whether $\Gamma$ has a symmetry $\phi$ 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.*

*Proof.*

**Membership** Take the same graph construction $G$ as in Theorem 1. We will have $N^2$ instances of the graph isomorphism problem variant where we ask whether $G$ is isomorphic to $G$ via mapping $i \in V$ to $j \in V$, for any $i, j \in \mathcal{N} \subset V$. From Proposition 20, we have that each of these questions reduce to GI, and thus also, that it reduces to GI to decide whether any are true.

**Hardness** Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs for which we would like to decide whether they are isomorphic. Construct a two-player zero-sum game $\Gamma$ with action sets $A^1 = V_1 \cup V_2 = A^2$ and payoffs

(1) $u^1(v, v') = 0 = u^2(v, v')$ if either ($v \in V_1$ and $v' \in V_2$) or ($v \in V_2$ and $v' \in V_1$) player,

(2) $u^1(v, v') = 2 = -u^2(v, v')$ if $v, v' \in V_1$ and $\{v, v'\} \in E_1$,

(3) $u^1(v, v') = -1 = -u^2(v, v')$ if $v, v' \in V_1$ and $\{v, v'\} \notin E_1$,

(4) $u^1(v, v') = -2 = -u^2(v, v')$ if $v, v' \in V_2$ and $\{v, v'\} \in E_2$,

(5) $u^1(v, v') = 1 = -u^2(v, v')$ if $v, v' \in V_2$ and $\{v, v'\} \notin E_2$.

So in the case where both choose vertices are from the same graph $G_i$, player $i$ will strongly prefer to choose a neighbor of the other player's chosen vertex, and gets slightly punished instead if they do not manage to do so. Nonetheless, these exact effects are not crucial to this construction; intuitively, it is only important that the players receive neighboring-dependent payoffs that are reversed across the two graphs.

Let $\psi$ be an isomorphism $G_1 \to G_2$. Then $\Gamma$ has the player swapping symmetry $\pi : 1 \leftrightarrow 2$ and $\phi|_{A^1}(v) :=$
$\phi|_{A^1}(v) := \begin{cases} \psi(v) & \text{if } v \in V_1 \\ \psi^{-1}(v) & \text{if } v \in V_2 \end{cases}$. Then this makes a game symmetry. This follows immediately for case (1), and it follows for the other cases, because $\psi$ exactly preserves neighboring vertices, and because cases (2) and (4) as well as cases (3) and (5) have swapped payoffs.

Let $\phi$ be a symmetry of $\Gamma$ that is not player-separable, that is, where $\pi$ swaps PL1 and PL2. We claim that then $\phi$ maps $V_1 \subset A^1$ to $V_2 \subset A^2$, and that $\psi := \phi|_{V_1}$ is the

graph isomorphism $G_1 \to G_2$. Let $v \in V_1$. Then, taking an arbitrary $v' \in V_1$, we have that

$$u^2(\phi(v), \phi(v')) = u^1(v, v') = \begin{cases} 2 & \text{if } \{v, v'\} \in E_1 \\ -1 & \text{if } \{v, v'\} \notin E_1 \end{cases}. \tag{1}$$

PL2 can only receives those payoffs when $\phi(v), \phi(v') \in V_2$. Analogous reasoning yields that $\phi(V_2) \subseteq V_1$, and hence, by bijectivity of $\phi|_{A^1}$, we have that $\phi|_{V_1}$ bijectively maps $V_1$ to $V_2$. Moreover, above equations (1) also yields for $v, v' \in V_1$ that

$$\{v, v'\} \in E_1 \iff u^1(v, v') = 2 \iff u^2(\phi(v), \phi(v')) = 2$$
$$\iff \{\phi(v), \phi(v')\} \in E_2.$$

So $\psi = \phi|_{V_1}$ is a group homomorphism. □

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

*Proof.* Hardness for two-player zero-sum games follows from Theorem 3, where the only nontrivial player permutation is swapping PL1 with PL2. For membership, we first observe the following fact: Each player permutation $\pi$ is induced by some symmetry $\phi$ of $\Gamma$ if and only if all the transpositions $(i, j)$ for $i, j \in \mathcal{N}$ as player permutations are induced by some symmetries of $\Gamma$. The forward direction is immediate, and the backward direction uses that any player permutation can be constructed by composing transpositions appropriately (and hence, it suffices to compose the underlying symmetries appropriately to induce a desired player permutation). With that fact, we can conclude the proof with analogous steps to the membership proof of Theorem 3; except now, we invoke the part of Proposition 20 that allows us to ask whether multiple graph isomorphism questions are *all* true. Each individual graph isomorphism question, indexed by $(i, j) \in \mathcal{N}$ will now be as follows: If $G$ and $G'$ are two identical copies of the graph construction of $\Gamma$ as in Theorem 1, check whether they are isomorphic to each other under the constraint that $i \in G$ is mapped to $j \in G'$, $j \in G$ is mapped to $i \in G'$, and every other player node in $G$ is mapped to itself in $G'$. □

## F   On Section 3.2

In this section, we establish Theorems 5 and 6, and also develop their counterpart results for game isomorphisms between two games in Theorem 12 and Theorem 13. Throughout this section, we will assume $m^i \geq 2$ for all players.

Let $\mathcal{S}_{\mathcal{N}}$ denote the *symmetry group* of permutations on a finite set $\mathcal{N}$. Later on, $\mathcal{N}$ will stand for the player set. We begin by specifying a few lemmas about computation in general symmetry groups and cosets. For the purpose of discussing algorithms, subgroups $G \leq \mathcal{S}_{\mathcal{N}}$ are specified by generating sets of size $\text{poly}(N)$, and cosets $xG = \{xg : g \in G\}$ are specified as pairs $(x, G)$. Later on, $x$ will represent a game isomorphism from game $\Gamma$ to game $\Gamma'$.

First, we will present a few known results. We start with the observation that we can efficiently shrink a generator set to $\mathsf{poly}(N)$ size, and therefore do not need to worry about algorithms that produce larger generator sets.

**Lemma 22** (Sims 1978). *Let $Z \subseteq \mathcal{S}_N$. There exists a $\mathsf{poly}(N, |Z|)$-time algorithm that outputs a set $Z' \subseteq Z$ with $|Z'| \leq \log_2(N!)$ and $\langle Z' \rangle = \langle Z \rangle$.*

Next, we turn our attention to computing coset intersections sufficiently fast. Say $G, H \leq \mathcal{S}_N$ and $x, y \in \mathcal{S}_N$. Then it is not hard to show that there exists an element $z \in xG \cap yH$ if and only if $xG \cap yH = z(G \cap H)$. Hence, in order to compute a coset intersection, it suffices to determine one element $z$ in the coset intersection and a generator set for the group intersection $G \cap H$.

**Lemma 23** (Luks 1999). *Let $G, H \leq \mathcal{S}_N$, and $x, y \in \mathcal{S}_N$. There exists a $2^{\mathcal{O}(N)}$-time algorithm that computes the coset intersection $xG \cap yH$.*[7]

Next, we recall the concept of a *player symmetry*. A permutation $\pi \in \mathcal{S}_N$ acts on game $\Gamma$ by permuting its players according to $\pi$ and leaving action numbering fixed. This does not yet necessarily make a game symmetry of $\Gamma$ if, *e.g.*, some action sets are not equally sized or the utility payoffs do not match. If it does, however, we call it a player symmetry. It preserves utility payoffs in the sense that

$$u^i(\mathbf{a}) = u^{\pi(i)}(\phi_\pi(\mathbf{a})) = u^{\pi(i)}(\mathbf{a}_{\pi^{-1}(1)}, \ldots, \mathbf{a}_{\pi^{-1}(N)})$$

for all $i \in \mathcal{N}$ and $\mathbf{a} \in A$. The set of player symmetries form a subgroup of $\mathcal{S}_N$, and is denoted as $\mathrm{Sym}_\mathcal{N}(\Gamma)$.

For efficient encoding, we will work with hypergraphs. A *hypergraph* is a pair $G = (V, E)$ of vertices $V$ and *hyperedges* $E \subseteq 2^V$. The notion of an automorphism extends to hypergraphs: It is a bijective mapping $\pi : V \to V$ such that if $V \supseteq W \in E$, then $\pi(W) \in E$.

**Theorem 11** (Luks 1999). *The automorphism group for a given hypergraph $G$ can be computed in time $2^{\mathcal{O}(|V|)}$.*

Note that this bound is tight up to the constant hidden by the big-O because $G$ might have $|E| = 2^{\Theta(|V|)}$ many edges that are included in its encoding in the first place. Next, we show that we can efficiently compute the group of player symmetries $\mathrm{Sym}_\mathcal{N}(\Gamma)$.

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

*Proof.* Consider the following hypergraph. There are several types of nodes:

- Create one node $i$ for each player $i \in \mathcal{N}$.
- Create one node $b_{ik}$ for each $k = 1, 2, \ldots, k_i := \lceil \log_2(m^i) \rceil$. The actions of player $i$ will be, for this proof, identified with subsets of $B_i := \{b_{i1}, \ldots, b_{ik_i}\}$.
- Let $P := \{u^i(\mathbf{a}) : i \in \mathcal{N}, \mathbf{a} \in A\}$ be the set of unique payoff values across all players. Create one node $u_k$ for each $k = 1, 2, \ldots, \lceil \log_2 |P| \rceil$. As before, the unique utility values $u \in P$ will be identified with subsets of $U := \{u_k\}_k$.

---

[7]Faster algorithms exist, *e.g.*, $\exp(\log^{\mathcal{O}(1)}(N))$ due to Babai (2016), but this will be enough for our purposes.

Now for each player $i \in \mathcal{N}$ and strategy profile $\mathbf{a} \in A$, create a hyperedge containing the node of player $i$, the subset of $B_j$ identified with action $\mathbf{a}_j$ for every player $j$, and the subset of $U$ identified with the utility value $u^i(\mathbf{a})$. Let $V := \mathcal{N} \sqcup B_1 \sqcup \cdots \sqcup B_N \cup U$ be the vertex set of this hypergraph.

Let $G \leq \mathcal{S}_V$ be the automorphism group of the hypergraph. Furthermore, define the subgroup $H \leq \mathcal{S}_V$ of vertex permutations as $\{\psi_\pi\}_{\pi \in H_\mathcal{N}}$, where (1) the player permutation subgroup $H_\mathcal{N} \leq \mathcal{S}_N$ contains exactly those $\pi$ that satisfy $m^i = m^{\pi(i)}$, *i.e.*, we map players with the same action set sizes to each other, and where (2) vertex permutation $\psi_\pi$ for $\pi \in H_\mathcal{N}$ maps $\mathcal{N} \ni i \mapsto \pi(i)$, $b_{ik} \mapsto b_{\pi(i)k}$, and $u_k \mapsto u_k$.

**Observation 1:** $\{\psi_\pi : \pi \in \mathrm{Sym}_\mathcal{N}(\Gamma)\} = G \cap H$. The set inclusion "$\subseteq$" is clear, since any player symmetry $\pi$ of $\Gamma$ indeed induces an automorphism of the form $\psi_\pi$ of this hypergraph construction. "$\supseteq$": Let $\psi \in G \cap H$, that is, $\psi = \psi_\pi$ for some $\pi \in H_\mathcal{N}$ and $\psi_\pi$ is a hypergraph automorphism. Then for any $i \in \mathcal{N}$ and $\mathbf{a} \in A$, since there is a hyperedge to $(i, \mathbf{a}, u^i(\mathbf{a}))$ in $E$, there will also be the hyperedge $(\pi(i), \psi_\pi(\mathbf{a}), u^i(\mathbf{a}))$ in $E$. This translates to $u^{\pi(i)}(\mathbf{a}_{\pi^{-1}(1)}, \ldots, \mathbf{a}_{\pi^{-1}(N)}) = u^i(\mathbf{a})$. Hence, $\pi$ indeed forms a player symmetry, that is, $\pi \in \mathrm{Sym}_\mathcal{N}(\Gamma) \leq H_\mathcal{N}$.

**Observation 2:** Above observation implies that a generator set of $G \cap H$, restricted to $\mathcal{N} \subset V$, forms a generator set of $\mathrm{Sym}_\mathcal{N}(\Gamma)$.

**Observation 3:** Intersection $G \cap H$ can be computed in polytime. To start, we can compute $G$ with Theorem 11. To that end, we observe that

$$|V| \leq N + \sum_{i=1}^N \log_2 m^i + N + \log_2 |P| + 1$$
$$= \mathcal{O}\left(\sum_{i=1}^N \log_2 m^i\right) = \mathcal{O}\left(\log_2 \prod_{i=1}^N m^i\right)$$
$$= \mathcal{O}(\log_2 |A|),$$

which yields a running time of

$$2^{\mathcal{O}(|V|)} = 2^{\mathcal{O}(\log_2 |A|)} = \mathsf{poly}(|A|).$$

Next, we note that $H$ is generated by the vertex permutations $\psi_{\sigma_{ij}}$ where $i, j \in \mathcal{N}$ are players with equal action set sizes $m^i = m^j$, and $\sigma_{ij} \in \mathcal{S}_N$ is the player permutation that swaps $i$ with $j$ and keeps the rest fixed. This generator set can be computed in time $\mathcal{O}(N^2 \cdot |V|) = \mathsf{poly}(|A|)$. Finally, Lemma 23 for $x = y = \mathrm{Id} \in \mathcal{S}_V$ allows us to compute the intersection $G \cap H$ in time $2^{\mathcal{O}(|V|)} = \mathsf{poly}(|A|)$. $\square$

How does this generalize to game isomorphisms? First, note that we can apply any player permutation $\pi \in \mathcal{S}_N$ to a game $\Gamma$ to get a new game $\pi(\Gamma)$ which has reordered players according to $\pi$, and the action sets are permuted accordingly. By construction, $\pi$ forms a game isomorphism between $\Gamma$ and $\pi(\Gamma)$. Next, if we have two arbitrary games $\Gamma$ and $\Gamma'$ with the same number of players, and upon fixing a numbering of the players, we can w.l.o.g. identify the players

sets as the same $\mathcal{N} = \{1, \ldots, N\}$. With this, we define the set of *player isomorphisms* $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma') \leq \mathcal{S}_N$ as the set of player permutations for which $\pi(\Gamma) = \Gamma'$. Any player isomorphism in $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ then naturally induces a game isomorphism between $\Gamma$ and $\Gamma'$. Upon also fixing a numbering of the player's action sets, a player isomorphism $\pi$ is characterized by satisfying for all $i \in \mathcal{N}$ and $\mathbf{a} \in A$:

$$u^i(\mathbf{a}) = (u')^{\pi(i)}(\phi_\pi(\mathbf{a})) = (u')^{\pi(i)}(\mathbf{a}_{\pi^{-1}(1)}, \ldots, \mathbf{a}_{\pi^{-1}(N)}).$$

Notice that when $\Gamma = \Gamma'$, then $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma') = \text{Sym}_{\mathcal{N}}(\Gamma)$.

**Lemma 24.** *For any two games $\Gamma, \Gamma'$, either $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ is empty, or it is a left coset $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma') = \pi \text{Sym}_{\mathcal{N}}(\Gamma)$ for $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ chosen arbitrarily.*

*Proof.* Suppose there is a player isomorphism $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$. Then we show that $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma') = \pi \text{Sym}_{\mathcal{N}}(\Gamma)$. If $\sigma \in \text{Sym}_{\mathcal{N}}(\Gamma)$, then by definition we have $\pi(\sigma(\Gamma)) = \pi(\Gamma) = \Gamma'$, so $\pi \circ \sigma \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$. Conversely, if $\pi' \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ then $\pi^{-1}(\pi'(\Gamma)) = \Gamma$, so $\sigma := \pi^{-1} \circ \pi' \in \text{Sym}_{\mathcal{N}}(\Gamma)$ satisfies $\pi \circ \sigma = \pi'$. $\square$

With this, we can get a similar result to Theorem 6 for game isomorphisms.

**Theorem 12.** *We can compute the set of player isomorphisms $\text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ between two games $\Gamma$ and $\Gamma'$ in polytime.*

*Proof.* By Lemma 24, it suffices to determine an element $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$, if existent, and $\text{Sym}_{\mathcal{N}}(\Gamma)$. We can do the latter in polytime via Theorem 6. So let us study the former.

Construct a game $\hat{\Gamma}$ that incorporates both $\Gamma$ and $\Gamma'$ as follows. $\hat{\Gamma}$ has $2N$ players: the $N$ players originating from $\Gamma$, which we will call $\mathcal{N} = \{1, 2, \ldots, N\}$ as usual; and the $N$ players originating from $\Gamma'$, for which we introduce a separate copy of players $\mathcal{N}' := \{1', 2', \ldots, N'\}$. Each player $i \in \hat{\mathcal{N}} := \mathcal{N} \cup \mathcal{N}'$ has action set $\hat{A}^i := A^i \cup \{\bot\}$, where $\bot$ is a special symbol. The utilities are as follows. Let $u^\bot$ be any number that does not appear as the utility of any player in $\Gamma$ or $\Gamma'$, and let $\bot(\mathbf{a}) = \{i : \mathbf{a}_i = \bot\} \subseteq \hat{\mathcal{N}}$ be the players that play $\bot$ in $\mathbf{a}$. If $\bot(\mathbf{a}) \notin \{\mathcal{N}, \mathcal{N}'\}$ then all players get utility $u^\bot$. If $\bot(\mathbf{a}) = \mathcal{N}'$, then players $i \in \mathcal{N}$ get utility $u^i(\mathbf{a}_1, \ldots, \mathbf{a}_N)$, and players $i' \in \mathcal{N}'$ get utility $u^\bot$. If $\bot(\mathbf{a}) = \mathcal{N}$, then players $i' \in \mathcal{N}'$ get utility $(u')^{i'}(\mathbf{a}_{1'}, \ldots, \mathbf{a}_{N'})$, and players $i \in \mathcal{N}$ get utility $u^\bot$.

To analyze this construction, associate to each existent player isomorphism $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$ the player permutation $\tau_\pi \in \mathcal{S}_{\hat{\mathcal{N}}}$ defined as $\tau_\pi(i) = \pi(i)$ for $i \in \mathcal{N} \subset \hat{\mathcal{N}}$ and $\tau_\pi(i') = \pi^{-1}(i')$ for $i' \in \mathcal{N}' \subset \hat{\mathcal{N}}$.

**Observation 1:** If $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$, then $\tau_\pi$ is a player symmetry of $\hat{\Gamma}$ with $\tau_\pi(\mathcal{N}) = \mathcal{N}'$. Indeed, action set sizes match thanks to $\pi$. Moreover, since $\tau_\pi : \mathcal{N} \overset{\text{bij}}{\leftrightarrow} \mathcal{N}'$, we have that players who received $u^\bot$ before applying the symmetry will continue to do so after applying the symmetry. Last but not least, players who received a payoff other than $u^\bot$ will continue to receive the same payoff after applying the player symmetry $\tau_\pi$ since $\pi$ and $\pi^{-1}$ are player isomorphisms.

**Observation 2:** Any player symmetry $\tau$ of $\hat{\Gamma}$ must map $\tau(\mathcal{N}) = \mathcal{N}$ or $\tau(\mathcal{N}) = \mathcal{N}'$. This is because of the case $\bot(\mathbf{a}) = \mathcal{N}$, in which, after applying $\tau$, players $i' \in \mathcal{N}'$ will again need to receive payoffs $\neq u^\bot$, implying $\tau(\mathcal{N}) \in \{\mathcal{N}, \mathcal{N}'\}$.

**Observation 3:** If there is a player symmetry $\tau$ of $\hat{\Gamma}$ mapping $\tau(\mathcal{N}) = \mathcal{N}'$, then, identifying $\mathcal{N} \hat{=} \mathcal{N}'$, permutation $\pi : \mathcal{N} \to \mathcal{N}, i \mapsto \tau(i)$ forms a player isomorphism from $\Gamma$ to $\Gamma'$. Indeed, action set sizes have to match since $\tau$ is a player symmetry. Therefore, it remains to show that for any $i \in \mathcal{N}$ and $\mathbf{a} \in A$, we have

$$
\begin{aligned}
&(u')^{\pi(i)}(\mathbf{a}_{\pi^{-1}(1)}, \ldots, \mathbf{a}_{\pi^{-1}(N)}) \\
&= (u')^{\tau(i)}(\mathbf{a}_{\tau^{-1}(1')}, \ldots, \mathbf{a}_{\tau^{-1}(N')}) \\
&= \hat{u}^{\tau(i)}(\bot, \ldots, \bot, \mathbf{a}_{\tau^{-1}(1')}, \ldots, \mathbf{a}_{\tau^{-1}(N')}) \\
&\overset{(*)}{=} \hat{u}^i(\mathbf{a}_1, \ldots, \mathbf{a}_N, \bot, \ldots, \bot) \\
&= u^i(\mathbf{a}_1, \ldots, \mathbf{a}_N),
\end{aligned}
$$

where in $(*)$ we used that $\tau$ is a player symmetry of $\hat{\Gamma}$.

**Observation 4:** If there exists a player symmetry $\tau$ of $\hat{\Gamma}$ mapping $\tau(\mathcal{N}) = \mathcal{N}'$, then any generator set of $\text{Sym}_{\hat{\mathcal{N}}}(\hat{\Gamma})$ must contain at least one such player symmetry, due to Observation 2.

**Conclusion:** In order to determine a player isomorphism $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$, if existent, compute a generator set of the automorphism group $\text{Sym}_{\hat{\mathcal{N}}}(\hat{\Gamma})$ via Theorem 6. This takes polytime in $|\Gamma|$ and $|\Gamma'|$ because $|\hat{A}| \leq (\max\{|A|, |A'|\})^2$. If any generator $\tau$ satisfies $\tau(\mathcal{N}) = \mathcal{N}'$, use Observation 3 to obtain and return the desired player isomorphism $\pi \in \text{Iso}_{\mathcal{N}}(\Gamma, \Gamma')$. If none satisfies $\tau(\mathcal{N}) = \mathcal{N}'$, then by Observation 4, there is no player symmetry $\tau'$ of $\hat{\Gamma}$ satisfying $\tau'(\mathcal{N}) = \mathcal{N}'$. By Observation 1, this means in particular that there cannot be a player isomorphism from $\Gamma$ to $\Gamma'$, and we can return that. $\square$

Next, we discuss parameterized algorithms for computing general game symmetries and isomorphisms, allowing for "nontrivial" mappings between action sets.

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

As argued in the main body, this implies the following.

**Corollary 11.** *For games in which the number of actions per player is bounded, we can compute $\text{Sym}(\Gamma)$ in polytime.*

*Proof of Theorem 5.* We construct the same hypergraph as in Theorem 6, except now, we do not introduce a logsize encoding of action sets by subsets of nodes. This is because we are interested in game symmetries that map actions flexibly to actions. Therefore, we instead introduce one node $b_{ik}$ for *each* action $a_k \in A^i$; we will also refer to it simply as $a_k$. In previous terms, we now have $k_i = m^i$. Note that the set of unique payoffs $P$ continue to be encoded as subsets of $U := \{u_k\}_k$. In the third observation below, we investigate the complexity of constructing this graph. Before

that, we will (again) analyze $\mathrm{Sym}(\Gamma)$ through permutations of the vertices $V$ of this hypergraph. For that, associate any game symmetry $\phi \in \mathrm{Sym}(\Gamma)$ with the vertex permutation $\psi_\phi : V \to V$ defined as $\mathcal{N} \ni i \mapsto \phi(i)$, $A^i \ni a_k \mapsto \phi(a_k)$, and $u_k \mapsto u_k$.

**Observation 1:** $\{\psi_\phi : \phi \in \mathrm{Sym}(\Gamma)\} = G \cap H$, where $G$ is the automorphism group of the hypergraph, and $H$ is the group of permutations that resemble game mappings, that is, all permutations $\psi : V \to V$ such that $\psi : \mathcal{N} \overset{\mathrm{bij}}{\to} \mathcal{N}$, $\psi : A^i \overset{\mathrm{bij}}{\to} A^{\psi(i)}$ for all $i \in \mathcal{N}$, and $\psi|_P = \mathrm{Id}_P$. Indeed, for "$\supseteq$" it is not hard to see that any $\psi \in G \cap H$ satisfies $\psi = \psi_\phi$ for a mapping $\phi$ defined as follows, forming a game symmetry: map $i \mapsto \psi(i)$ and $A^i \ni a_k \mapsto \psi(a_k)$.

**Observation 2:** Above observation implies that a generator set of $G \cap H$, restricted to $\mathcal{N} \cup \mathbb{A} \subset V$, forms a generator set of $\mathrm{Sym}(\Gamma)$.

**Observation 3:** The hypergraph can be constructed in time $2^{\mathcal{O}(|\mathbb{A}|)}$. Let us prove that. It takes $\mathcal{O}((N \cdot |A|)^2)$ time to determine the set of unique payoffs $P$ and then construct the corresponding nodes $U$. Moreover, there are an additional $N + |\mathbb{A}|$ nodes and an additional $N \cdot |A|$ hyperedges to be created. Hence, the hypergraph construction takes $\mathcal{O}(|A|^4)$ time. Notice that the quantity $|A|$ is largest for any given quantity $|\mathbb{A}|$ exactly when we have the same action set sizes $m^1 = \cdots = m^N = |\mathbb{A}|/N$. Therefore, we derive

$$\log_2 |A| \le \log_2 \left( \frac{|\mathbb{A}|}{N} \right)^N = N \log_2 \frac{|\mathbb{A}|}{N} \le N \cdot \frac{|\mathbb{A}|}{N} = |\mathbb{A}| \,.$$

This yields a running time

$$\mathcal{O}(|A|^4) = \mathcal{O}\left( \left( 2^{\log_2 |A|} \right)^4 \right) = \mathcal{O}\left( \left( 2^{|\mathbb{A}|} \right)^4 \right) = \mathcal{O}\left( 2^{4|\mathbb{A}|} \right)$$
$$= 2^{\mathcal{O}(|\mathbb{A}|)} \,.$$

**Observation 4:** The automorphism group $G$ can be computed in time $2^{\mathcal{O}(|\mathbb{A}|)}$. This is because of Theorem 11 since the hypergraph has

$$|V| = N + |\mathbb{A}| + \lceil \log_2 |P| \rceil \le 2|\mathbb{A}| + \lceil \log_2 |A| \rceil = \mathcal{O}(|\mathbb{A}|)$$

many vertices by the estimation from Observation 3.

**Observation 5:** The permutation group $H$ is generated by the set of action swaps and the set of player swaps, that is, $\{(b_{ik}b_{il}) \mid i \in \mathcal{N}, 1 \le k, l \le m^i\}$ and $\{(ij)(b_{i1}b_{j1})\ldots(b_{im^i}b_{jm^i}) \mid i,j \in \mathcal{N} \text{ with } m^i = m^j\}$. We can list those generators in time $\mathcal{O}(|\mathbb{A}|^3 \cdot |V|) = 2^{\mathcal{O}(|\mathbb{A}|)}$.

**Conclusion** Compute $G$ and $H$ as described in Observations 4 and 5, the intersection $G \cap H$, and its intersection with Lemma 23. All of this takes $2^{\mathcal{O}(|\mathbb{A}|)}$ time. Observation 2 then yields $\mathrm{Sym}(\Gamma)$. □

The following result is proven analogously to Lemma 24.

**Lemma 25.** *For any two games $\Gamma, \Gamma'$, either $\mathrm{Iso}(\Gamma, \Gamma')$ is empty, or it is a left coset $\mathrm{Iso}(\Gamma, \Gamma') = \psi\mathrm{Sym}(\Gamma)$ for $\psi \in \mathrm{Iso}(\Gamma, \Gamma')$ chosen arbitrarily.*

**Theorem 13.** *Given two games $\Gamma$ and $\Gamma'$, the coset of isomorphisms $\mathrm{Iso}(\Gamma, \Gamma')$ is computable in time $2^{\mathcal{O}(|\mathbb{A}|)}$.*

*Proof.* The proof is analogous to Theorem 12, except that we have to use Theorem 5 and Lemma 25, and that we have to allow and adapt for actions now being permuted as well. For example, to any game isomorphism $\psi \in \mathrm{Iso}(\Gamma, \Gamma')$, we now have to associate a mapping $\phi_\psi$ for the constructed game $\hat{\Gamma}$ that still maps $\phi_\psi(i) = \psi(i)$ for $i \in \mathcal{N} \subset \hat{\mathcal{N}}$ and $\phi_\psi(a') = \psi^{-1}(a')$ for $a' \in A^{i'}$, but additionaly also maps $\phi_\psi(a) = \psi(a)$ for $a \in A^i$ and $\phi_\psi(i') = \psi^{-1}(i')$ for $i' \in \mathcal{N}' \subset \hat{\mathcal{N}}$. Below we adjust two parts of the proof to our new setting:

**Observation 1.5:** Prior to Observation 2, we now need to first observe that in any gamy symmetry $\psi$ of $\hat{\Gamma}$, an action $\perp$ must always be mapped to action $\perp$ again. To see that, assume there exists $i \in \hat{\mathcal{N}}$, w.l.o.g. $i \in \mathcal{N} \subset \hat{\mathcal{N}}$, with $\phi(\perp \in A^i) \ne \perp \in A^{\phi(i)}$. Then, because of $\phi$ being a symmetry and $m^i \ge 2$, we have that there must in return be two actions $\bar{a}, \bar{a}' \in A^i$ with $\phi(\bar{a}) = \perp \in A^{\phi(i)}$ and $\phi(\bar{a}') \ne \perp \in A^{\phi(i)}$. We note that in game $\hat{\Gamma}$, player $i$ receives a payoff $\ne u^\perp$ under profiles $\bar{\mathbf{a}} := (\bar{a}, \mathbf{a}_{-i}, \perp_{A'})$ and $\bar{\mathbf{a}}' := (\bar{a}', \mathbf{a}_{-i}, \perp_{A'})$ for any $\mathbf{a}_{-i} \in A^{-i}$, because $\perp(\bar{\mathbf{a}}) = \perp(\bar{\mathbf{a}}') = \mathcal{N}'$. Because $\phi$ is a symmetry, this implies that player $\phi(i)$ must receive a payoff $\ne u^\perp$ under profile $\phi(\bar{\mathbf{a}})$ and $\phi(\bar{\mathbf{a}}')$, which implies that $\perp(\phi(\bar{\mathbf{a}})) = \perp(\phi(\bar{\mathbf{a}}')) \in \{\mathcal{N}, \mathcal{N}'\}$. However, this cannot be true since $\bar{\mathbf{a}}$ and $\bar{\mathbf{a}}'$ only differ in one entry whose image under $\phi$ is $= \perp$ and $\ne \perp$ respectively. Hence, a contradiction. Analogous reasoning also works for if $i$ was element in $\mathcal{N}'$.

**Observation 3':** The derivations here, for game symmetry $\phi$ of $\hat{\Gamma}$ and associated isomorphism $\psi$ from $\Gamma$ to $\Gamma'$ will now look as follows for $i \in \mathcal{N}$ and $\mathbf{a} \in A$:

$$(u')^{\psi(i)}(\psi(\mathbf{a})) = (u')^{\phi(i)}(\phi(\mathbf{a})) = \hat{u}^{\phi(i)}(\perp, \ldots, \perp, \phi(\mathbf{a}))$$
$$= \hat{u}^{\phi(i)}(\phi(\mathbf{a}, \perp, \ldots, \perp)) = \hat{u}^i(\mathbf{a}, \perp, \ldots, \perp) = u^i(\mathbf{a}) \,.$$
□

### F.1 Search for Particular Properties

We can leverage the computation from Section 3.2 (Theorems 5, 6, 12 and 13) to further efficiently compute the symmetries and isomorphisms satisfying certain properties. Below, we merely give an example of this:

**Proposition 26.** *Given game $\Gamma$ and, optionally, another game $\Gamma'$, we can compute the following groups / cosets:*

1. *In time $2^{\mathcal{O}(|\mathbb{A}|)}$ the game symmetries $\phi$ of $\Gamma$, or the game isomorphisms $\phi : \Gamma \to \Gamma'$, that map a particular action $a \in \mathbb{A}$ to another action $\phi : a \mapsto b$, and*
2. *In polytime the game symmetries $\phi$ of $\Gamma$, or the game isomorphisms $\phi : \Gamma \to \Gamma'$, that map a particular player $i \in \mathcal{N}$ to another player $\phi : i \mapsto j$.*

*Proof.* These results are all proved in a similar fashion, so let us consider the case of game isomorphisms $\phi : \Gamma \to \Gamma'$ mapping $a \mapsto b$. Call this coset $\mathcal{C}$. As before, fix an ordering of players in $\Gamma$ and $\Gamma'$ such that we can identify the player sets with each other and have same respective action set sizes. (If

that is not possible, $\Gamma$ and $\Gamma'$ cannot be isomorphic at all.) Then fix an order of the actions to identify the action sets with each other as well.

Let $\mathsf{Stab}(a)) < \mathcal{S}_\mathbb{A}$ be the (stabilizing) subgroup of permutations $\pi : \mathbb{A} \to \mathbb{A}$ that fix $a$, *i.e.*, $\pi(a) = a$. Moreover, let $\sigma_{ab}$ be the permutation $\mathbb{A} \to \mathbb{A}$ that swaps actions $a$ and $b$, and keeps the other actions fixed. Then it is not hard to see that

$$\mathcal{C} = \mathrm{Iso}(\Gamma, \Gamma') \cap (\sigma_{ab} \circ \mathsf{Stab}(a)) .$$

$\mathsf{Stab}(a))$ is generated by all action swaps $(a', b')$ where $a' \neq a \neq b'$. With that, and Theorem 13 and Lemma 23, this yields a $2^{\mathcal{O}(|\mathbb{A}|)}$ time method to compute $\mathcal{C}$.

For the player isomorphisms case, we are now working with permutations on the player set $\mathcal{N}$ and we instead need to observe that

$$\mathcal{C}' = \mathrm{Iso}_\mathcal{N}(\Gamma, \Gamma') \cap (\sigma_{ij} \circ \mathsf{Stab}(i)) .$$

$\square$

## F.2 Excursion to Graphical Games

**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.*

*Proof.* Given a graph $G = (V, E)$, view it as a graphical game in which each node / player $v \in V$ has two actions $a_0, a_1$. First, we define player $v$'s payoffs for the corner cases when $v$ has no neighbors (=is an isolated node): Then, define its payoffs as $-1$ if that player plays $a_0$, and $-2$ if that player plays $a_1$. The purpose of these values is simply to be distinct from the payoffs a node with neighbors might receive. Namely, in that case, define $v$'s payoffs as

$$\sum_{v' \text{ adjacent to } v} 1 \left[ v \text{ and } v' \text{ play } a_1 \right] ,$$

where $1[P]$ is the indicator function evaluating as 1 if property $P$ is satisfied, otherwise evaluating as 0.

We can observe that a graph automorphism of $G$ forms a player symmetry of the graphical game (as defined above Theorem 6). We can also show a reverse direction of this statement, starting from a game symmetry $\phi$ of the graphical game. First, we note that the payoffs force $\pi$ to map isolated nodes to isolated nodes, and non-isolated nodes to non-isolated nodes. This, in turn, forces $\phi$ to map action $a_i$ to $a_i$. In particular, $\phi$ is a player symmetry. With this knowledge, we obtain that if $v$ and $v'$ are neighbors, then the action of $v'$ has an effect on the payoff of $v$ if $v$ plays $a_1$, hence there is a same effect of $\pi(v')$ on $\pi(v)$, and therefore $\pi(v')$ and $\pi(v)$ are neighbors. Thus, $\pi$ of $\phi$ is a graph automorphism of $G$.

Analogous reasoning works for reducing the graph isomorphism problem for two graphs to the game isomorphism problem for two graphical games. $\square$

## G  Further background on Section 4

In this section, we give more context to Section 3 and prove statements from it.

First, we observe that any game symmetry naturally extends to strategy profiles: A strategy profile $\mathbf{s}$ is mapped to $\phi(\mathbf{s})$ that plays an action $a \in A^j$ of player $j$ with the probability that $\mathbf{s}$ plays the associated action $\phi^{-1}(a)$ of associated player $\pi^{-1}(j)$. Satisfying the second property in Definition 3 (as it is phrased for action profiles $\mathbf{a} \in A$) is already enough to ensure that the analogous statement holds for the bigger set of all strategy profiles $\mathbf{s} \in S$, because in that case

$$
\begin{aligned}
u^i(\mathbf{s}) &= \sum_{\mathbf{a} \in A} \mathbf{s}_1(\mathbf{a}_1) \cdot \ldots \cdot \mathbf{s}_N(\mathbf{a}_N) \cdot u^i(\mathbf{a}) \\
&= \sum_{\mathbf{a} \in A} \phi(\mathbf{s})_{\pi(1)}(\phi(\mathbf{a}_1)) \cdot \ldots \cdot \phi(\mathbf{s})_{\pi(N)}(\phi(\mathbf{a}_N)) \\
&\qquad \cdot u^{\pi(i)}(\phi(\mathbf{a})) \\
&\overset{\mathbf{a} = \phi^{-1}(\mathbf{a}')}{=} \sum_{\mathbf{a}' \in A} \phi(\mathbf{s})_{\pi(1)}(\phi(\phi^{-1}(\mathbf{a}')_1)) \cdot \ldots \\
&\qquad \cdot \phi(\mathbf{s})_{\pi(N)}(\phi(\phi^{-1}(\mathbf{a}')_N)) \cdot u^{\pi(i)}(\mathbf{a}') \\
&= \sum_{\mathbf{a}' \in A} \phi(\mathbf{s})_{\pi(1)}(\mathbf{a}'_{\pi(1)}) \cdot \ldots \cdot \phi(\mathbf{s})_{\pi(N)}(\mathbf{a}'_{\pi(N)}) \\
&\qquad \cdot u^{\pi(i)}(\mathbf{a}') \\
&\overset{\text{reorder}}{=} \sum_{\mathbf{a}' \in A} \phi(\mathbf{s})_1(\mathbf{a}'_1) \cdot \ldots \cdot \phi(\mathbf{s})_N(\mathbf{a}'_N) \cdot u^{\pi(i)}(\mathbf{a}') \\
&= u^{\pi(i)}(\phi(\mathbf{s})) .
\end{aligned}
$$

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

*Proof.* For action $a \in A^i$, we have

$$u^i(a, \mathbf{s}_{-i}) - u^i(\mathbf{s}) = u^{\pi(i)}(\phi(a), \phi(\mathbf{s})_{-\pi(i)}) - u^{\pi(i)}(\phi(\mathbf{s})) \tag{2}$$

and therefore, there will be a player (say, $i$) with positive deviation incentives under $\mathbf{s}$ if and only if there is a player (then, $\pi(i)$) with positive deviation incentives under $\phi(\mathbf{s})$. $\square$

**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 $\mathbf{s}_i(a) = \mathbf{s}_{i'}(a')$.*

*Proof.* If a profile respects symmetries $\phi$ and $\phi'$ in $\Sigma$, then it must also respect symmetry $\phi' \circ \phi$. Since $\mathrm{Sym}(\Gamma)$ is finite, this suffices to say that respecting $\Sigma$ implies respecting its generated set $\langle \Sigma \rangle$. But strategy profile $\mathbf{s}$ respecting $\langle \Sigma \rangle$ means no more than $\mathbf{s}_i(a) = \mathbf{s}_{\pi(i)}(\phi(a))$ for all $\phi \in \langle \Sigma \rangle$ for all $a \in A^i$. This is equivalent to requiring $\mathbf{s}_i(a) = \mathbf{s}_{i'}(a')$ for all $a' \in A^{i'}$ ($i' \in \mathcal{N}$) that is in the same orbit as $a \in A^i$, where $a \in \mathbb{A}$ arbitrary. This yields the second statement. $\square$

At this point, we might highlight the following two results from the literature for an interested reader: Since $\langle \Sigma \rangle$ and $\mathbb{A}$ are finite, we get from the orbit-stabilizer theorem and Lagrange's theorem that the size $|\langle \Sigma \rangle a|$ of an orbit is exactly equal to the number of symmetries $|\langle \Sigma \rangle|$ divided by the number of symmetries $\phi \in \langle \Sigma \rangle$ that map $a$ to itself

("fixing $a$"). Thanks to Burnside's Lemma we have that the number of distinct orbits equals the average number of actions that a symmetry of $\langle \Sigma \rangle$ keeps fixed:

$$W(\langle \Sigma \rangle) = \frac{1}{|\langle \Sigma \rangle|} \sum_{\phi \in \langle \Sigma \rangle} |\{a \in \mathbb{A} : \phi(a) = a\}| .$$

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

*Proof.* We build the orbits orbit by orbit. Create a queue $Q$ consisting of all actions $\mathbb{A}$. Take the next element $a$ from the queue $Q$. If it is not marked yet, create a new (yet uncompleted orbit) set $O = \{a\}$, mark $a$ as "orbit already identified", and add a pointer from it to $O$ that indicates that $O$ will be $a$'s orbit. Moreover, create an additional queue $Q_O$ for adding to $O$, initially consisting only of $\{a\}$. Enter this subloop that fully builds $O$, only after which we will go back to outer loop queue $Q$. The subloop: Take the next element $a'$ from queue $Q_O$ and loop through the list of symmetries. For each such symmetry $\phi$, check if $\phi(a')$ is already marked. If not, add it $O$, mark it, add a pointer from it to $O$, and add it to the queue $Q_O$. If it was already marked, then it must have already been in $O$ and we can ignore it. Once done looping through the symmetries, do the same looping again with the next action in queue $Q_O$, all up until $Q_O$ becomes empty. At that point, we have found all ways one may apply compositions of symmetries to $a$ to get to new actions in $\mathbb{A}$. So $O$ is fully constructed, and we can go back to the outer loop and get the next element $a$. If it has not been marked yet, progress as described above. If it has been marked though, we can skip it because we have already identified its orbit in a previous encounter. Running time analysis: Each action is dealt with once, and when being dealt with at most once in queue $Q$ and once in a queue $Q_O$. The former process takes constantly many operations, the latter operation takes $\mathcal{O}(|\Sigma|)$ many operations. Hence $\mathcal{O}(|\Sigma| \cdot |\mathbb{A}|)$. □

# H  On Section 5

In this section, we give proofs to the results in Section 5.

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

*Proof.* PPAD-hardness is straight-forward: In cases where $\Sigma = \{\mathrm{Id}_\mathbb{A}\}$, we are facing the problem of computing any $\epsilon$-Nash equilibrium of the game, which is PPAD-complete (Daskalakis, Goldberg, and Papadimitriou 2009; Chen, Deng, and Teng 2009). We also obtain this result for a nontrivial set of symmetries through a well-known trick that is related to not disclosing to participants which player identity they might play as (Gale, Kuhn, and Tucker 1952): Nash equilibria of a bimatrix game $(A, B)$—which we can w.l.o.g. assume to only have positive entries by shifting payoffs—correspond to player-symmetric Nash equilibria of the totally symmetric bimatrix game defined by $M := \begin{pmatrix} 0 & A \\ B^T & 0 \end{pmatrix}$, see Nisan et al. (2007)[Theorem 2.4].

We develop PPAD-membership using the general framework of Etessami and Yannakakis (2010) for showing that finding a fixed point of a Brouwer function $F : D \to D$ is in PPAD. Our domain $D$ will be the set of strategy profiles respecting the symmetries. Our Brouwer function will be Nash's function (Nash 1951), which was defined for the larger set $S$ of all strategy profiles: Given a profile $\mathbf{s} \in S$, player $i \in \mathcal{N}$, and action $a \in A^i$, define the advantage of $a$ (over $\mathbf{s}_i$) as

$$g(\mathbf{s}, i, a) := u^i(a, \mathbf{s}_{-i}) - u^i(\mathbf{s})$$

and profile $F(\mathbf{s})$ as

$$F(\mathbf{s})_i(a) := \frac{\mathbf{s}_i(a) + \max\{0, g(\mathbf{s}, i, a)\}}{1 + \sum_{a' \in A^i} \max\{0, g(\mathbf{s}, i, a)\}} .$$

Nash shows that $F(\mathbf{s})$ for $\mathbf{s} \in S$ indeed makes a profile in $S$ again, *i.e.*, $F : S \to S$. He also gives a short and high-level argument for why $F$ maps symmetries-respecting profiles to symmetries-respecting profiles, that is, $F|_D : D \to D$. For completeness, we work out the details of this observation at the end of this proof. Etessami and Yannakakis moreover show that (1) the functions $F$ constructed from a game $\Gamma$ are nicely Lipschitz continuous and therefore satisfy the requirement of being *polynomially continuous*, and that (2) an $\epsilon'$-fixed point $\mathbf{s} \in S$ of $F$ makes an $\epsilon$-Nash equilibrium of the $\Gamma$ for some $\epsilon$ in poly encoding size of the encoding of $\Gamma$ and $\epsilon'$. Naturally, the functions $F_D$ inherit both of these properties from the functions $F$. It remains to show that $D$ is a convex polytope efficiently describable by linear (in-)equalities; because then, we have that functions $F_D$ are also *polynomially computable*, implying the PPAD result by invoking (Etessami and Yannakakis 2010)[Proposition 2.2].

$D$ **is easy to describe:** First, get the symmetries in orbit form $W$ via Lemma 17 (if not already given in that form). Next, choose one representative $a_\omega$ of each orbit $\omega \in W$. Then, we can describe $D$ through the following linear (in-)equality system:

$$\begin{aligned}
\mathbf{s}_i(a) &\geq 0 & \forall i \in \mathcal{N}, a \in A^i \\
\sum_{a \in A^i} \mathbf{s}_i(a) &= 1 & \forall i \in \mathcal{N} \qquad\qquad (3)\\
\mathbf{s}_{i(a)}(a) &= \mathbf{s}_{i(a_\omega)}(a_\omega) & \forall \omega \in W, a \in \omega \setminus \{a_\omega\} ,
\end{aligned}$$

where we use the notation $i(a')$ to denote the player $i$ with $a' \in A^i$. These $2|\mathbb{A}| + N$ many (in-)equalities together with their coefficients can be computed in polytime in the encoding of $\Gamma$ and $W$.

$F$ **maps** $D$ **to** $D$**:** Let $\mathbf{s} \in D$, $i \in \mathcal{N}$, and $a \in A^i$. Then, get $\omega \in W$ such that $a \in w$, and a symmetry $\phi$ with $\phi(a) = a_\omega \in A^{i(a_\omega)}$ such that there exists $\phi \in \Sigma \subseteq \mathrm{Sym}(\Gamma)$ with $W = W(\langle \Sigma \rangle)$. We have to show that $F(\mathbf{s})_{i(a_\omega)}(a_\omega) = F(\mathbf{s})_i(a)$. Then $F(\mathbf{s})$ satisfies the system in (3), yielding membership in $D$. To that end, we observe for all $a' \in A^i$ that

$$g(\mathbf{s}, i, a') \overset{(2)}{=} g(\phi(\mathbf{s}), \pi(i), \phi(a'))$$
$$\overset{\mathbf{s} \in D}{=} g(\mathbf{s}, i(a_\omega), \phi(a')) .$$

In particular, we have

$$g(\mathbf{s}, i, a) = g(\mathbf{s}, i(a_\omega), a_\omega),$$

and

$$\sum_{a' \in A^i} \max\{0, g(\mathbf{s}, i, a')\}$$
$$= \sum_{a' \in A^i} \max\{0, g(\mathbf{s}, i(a_\omega), \phi(a'))\}$$
$$= \sum_{\tilde{a} \in A^{i(a_\omega)}} \max\{0, g(\mathbf{s}, i(a_\omega), \tilde{a})\}.$$

Therefore, we can combine these observations to conclude that indeed we have

$$F(\mathbf{s})_{i(a_\omega)}(a_\omega) = \frac{\mathbf{s}_{i(a_\omega)}(a_\omega) + \max\{0, g(\mathbf{s}, i(a_\omega), a_\omega)\}}{1 + \sum_{\tilde{a} \in A^{i(a_\omega)}} \max\{0, g(\mathbf{s}, i(a_\omega), \tilde{a})\}}$$
$$= \frac{\mathbf{s}_i(a) + \max\{0, g(\mathbf{s}, i, a)\}}{1 + \sum_{a' \in A^i} \max\{0, g(\mathbf{s}, i, a')\}} = F(\mathbf{s})_i(a).$$

$\square$

**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.*  We will begin with proving CLS-membership.

**Reduction to KKT**  We show that SYM-NASH-TEAM reduces to the CLS-complete problem of finding a KKT point of a nice enough function over a convex compact polytope. We assume familiarity with the definition of that problem KKT here, which can be found in (Fearnley et al. 2023)[Page 21]. Let a SYM-NASH-TEAM instance be given, that is, a team game $\Gamma$ in explicit form, a precision parameter $\epsilon > 0$, and symmetries $W$ of $\Gamma$ in orbit form.

Analogous to the proof of Proposition 18, choose one representative $a_\omega$ of each orbit $\omega \in W$, and let $D$ be the nonempty convex compact polytope as described in (3) that captures the set of strategy profiles that respect the given symmetries. Denoting with $w(a) \in W$ the orbit of an action $a \in A^i$, we can rewrite the description of $D$ to:

$$\mathbf{s}_i(a) \geq 0 \qquad \forall i \in \mathcal{N}, a \in A^i$$
$$\sum_{a \in A^i} \mathbf{s}_i(a) = 1 \qquad \forall i \in \mathcal{N}$$
$$\mathbf{s}_i(a) = \mathbf{s}_{i(a_{\omega(a)})}(a_{\omega(a)}) \quad \forall i \in \mathcal{N}, a \in A^i \text{ s.t. } a \neq a_{\omega(a)}.$$

Each equality constraint can be converted into two inequality constraints. The team's utility function $u$ and its partial derivatives are the following multivariate polynomial functions:

$$u(\mathbf{s}) = \sum_{\mathbf{a} \in A} u(\mathbf{a}) \prod_{i \in \mathcal{N}} \mathbf{s}_i(\mathbf{a}_i)$$

and for $i \in \mathcal{N}, a \in A^i$:

$$\nabla_{i,a} u(\mathbf{s}) = \sum_{\mathbf{a}^- \in A^{-i}} u(a, \mathbf{a}^-) \prod_{j \neq i} \mathbf{s}_j(\mathbf{a}_j^-)$$

for $i \in \mathcal{N}$ and $a \in A^i$. They can be constructed in polytime, with monomial coefficients derived from the payoffs $\{u(\mathbf{a})\}_\mathbf{a}$. A Lipschitz constant $L > 0$ that shows that both $u$ and $\nabla u$ are Lipschitz continuous over $D$ (via derivatives of $u$ and $\nabla u$) can also be constructed in polytime. Both functions—as simple-to-describe polynomials—make *well-behaved arithmetic circuits*. Finally, choose the precision parameter of KKT as $\delta := \epsilon/2 > 0$. Then, we satisfy the conditions of the computational problem KKT. Since the gradient and Lipschitz constant are computed correctly, the problem returns a $\delta$-KKT point of the optimization problem of $u$ over $D$. A $\delta$-KKT point $\mathbf{s} \in \mathbb{R}^{|A|}$ for our problem is characterized by the following conditions: there exist KKT-multipliers $(\mu_{i,a})_{i \in \mathcal{N}, a \in A^i}$, $(\kappa_i)_{i \in \mathcal{N}}$, and $(\tau_{i,a})_{i \in \mathcal{N}, a \in A^i : a \neq a_{\omega(a)}}$ in $\mathbb{R}$ such that

$$\mathbf{s}_i(a) \geq 0 \qquad \forall i \in \mathcal{N}, a \in A^i$$
$$\sum_{a \in A^i} \mathbf{s}_i(a) = 1 \qquad \forall i \in \mathcal{N}$$
$$\mathbf{s}_i(a) = \mathbf{s}_{i(a_{\omega(a)})}(a_{\omega(a)}) \quad \forall i \in \mathcal{N}, \forall a \in A^i : a \neq a_{\omega(a)}$$
$$\mu_{i,a} \geq 0 \qquad \forall i \in \mathcal{N}, \forall a \in A^i$$
$$\mu_{i,a} = 0 \quad \text{or} \quad \mathbf{s}_i(a) = 0 \quad \forall i \in \mathcal{N}, \forall a \in A^i$$
$$|\nabla_{i,a} u(\mathbf{s}) + \mu_{i,a} - \kappa_i - \tau_{i,a}| \leq \delta$$
$$\forall i \in \mathcal{N}, \forall a \in A^i : a \neq a_{\omega(a)}$$
$$\left|\nabla_{i,a} u(\mathbf{s}) + \mu_{i,a} - \kappa_i + \sum_{a' \in w(a) \setminus \{a\}} \tau_{i(a'),a'}\right| \leq \delta \qquad (4)$$
$$\forall i \in \mathcal{N}, \forall a \in A^i : a = a_{\omega(a)}.$$

We will show that $\mathbf{s}$ must then be an $\epsilon$-Nash equilibrium of $\Gamma$.

**Well-Supportedness**  In fact, we show a stronger property which is that $\mathbf{s}$ is an $\epsilon$-well-supported Nash equilibrium. This is the case if $\forall i \in \mathcal{N}, a \in A^i$ with $\mathbf{s}_i(a) > 0$, we have that for all $\tilde{a} \in A^i$:

$$u(a, \mathbf{s}_{-i}) \geq u(\tilde{a}, \mathbf{s}_{-i}) - \epsilon.$$

As a first step, we will argue now that in order to check the above condition, because of $\mathbf{s} \in D$, it suffices to check for all $\omega \in W$ with $\mathbf{s}_{i_\omega}(a_w) > 0$, that for all $\tilde{a} \in A^{i_\omega}$

$$u(a_\omega, \mathbf{s}_{-i_\omega}) \geq u(\tilde{a}, \mathbf{s}_{-i_\omega}) - \epsilon. \qquad (5)$$

And indeed, with any symmetry $\phi : a \mapsto a_\omega$ that is element of some $\Sigma \subseteq \text{Sym}(\Gamma)$ with $W = W(\langle\Sigma\rangle)$, such a $\phi$ implies for any $\mathbf{s} \in D$ that

1. $\mathbf{s}_i(a) > 0 \iff \mathbf{s}_{i_\omega}(a_\omega) > 0$,
2.
$$\nabla_{i(a),a} u(\mathbf{s}) = u(a, \mathbf{s}_{-i}) = u(a_\omega, \mathbf{s}_{-i_\omega}) = \nabla_{i_\omega, a_\omega} u(\mathbf{s}), \qquad (6)$$
3. $u(a, \mathbf{s}_{-i}) - u(\tilde{a}, \mathbf{s}_{-i}) = u(a_\omega, \mathbf{s}_{-i_\omega}) - u(\phi(\tilde{a}), \mathbf{s}_{-i_\omega}).$

An advantage of well-supportedness is the following: For any orbit $\omega$ with $\mathbf{s}_{i_\omega}(a_w) > 0$, we also have $\mathbf{s}_i(a) > 0$ for all $a \in \omega$. By the KKT-conditions, this implies

$$\sum_{a \in \omega} \mu_{i,a} = \sum_{a \in \omega} 0 = 0. \qquad (7)$$

**Cardinality** Compute the orbit cardinalities $c(\omega) = |A^{i_\omega} \cap \omega| > 0$ for all $\omega \in W$. This takes $\mathcal{O}(|\mathbb{A}| \cdot |W|)$ time to compute. The cardinality can be interpreted as the number of actions an orbit contains from a(ny) single player $i$ with $\omega \cap A^i \neq \emptyset$, and it is indeed the same, independent of the chosen representative $a_\omega$: Suppose $a \in \omega \cap A^i$ and $a' \in \omega \cap A^{i'}$ are two possible representatives. If $i = i'$, then we immediately obtain the same cardinality. If $i \neq i'$, then a game symmetry $\phi$ that maps $a \mapsto a'$ and for which there exists $\phi \in \Sigma \subseteq \mathrm{Sym}(\Gamma)$ with $W = W(\langle \Sigma \rangle)$, bijectively associates any $b \in \omega \cap A^i$ with a $b' = \phi(b) \in \omega \cap A^{i'}$.

**Simplifying** First, we can combine the two $\delta$-inequalities of (4) to obtain for each orbit $\omega \in W$ that

$$\left| |\omega| \cdot \nabla_{i_\omega, a_\omega} u(\mathbf{s}) + \sum_{a \in \omega} \mu_{i(a), a} - \sum_{a \in \omega} \kappa_{i(a)} \right| \tag{8}$$

$$\overset{(6)}{=} \left| \nabla_{i_\omega, a_\omega} u(\mathbf{s}) + \mu_{i_\omega, a_\omega} - \kappa_{i_\omega} + \sum_{a \in \omega \setminus \{a_\omega\}} \tau_{i(a), a} \right.$$

$$\left. + \sum_{a \in \omega \setminus \{a_\omega\}} \left( \nabla_{i(a), a} u(\mathbf{s}) + \mu_{i(a), a} - \kappa_{i(a)} - \tau_{i(a), a} \right) \right|$$

$$\overset{\Delta\text{-Ineq.}}{=} \left| \nabla_{i_\omega, a_\omega} u(\mathbf{s}) + \mu_{i_\omega, a_\omega} - \kappa_{i_\omega} + \sum_{a \in \omega \setminus \{a_\omega\}} \tau_{i(a), a} \right|$$

$$+ \sum_{a \in \omega \setminus \{a_\omega\}} \left| \left( \nabla_{i(a), a} u(\mathbf{s}) + \mu_{i(a), a} - \kappa_{i(a)} - \tau_{i(a), a} \right) \right|$$

$$\overset{(4)}{\leq} \delta + \sum_{a \in \omega \setminus \{a_\omega\}} \delta = |\omega| \cdot \delta \,.$$

For orbits $\omega \in W$ with $\mathbf{s}_{i_\omega}(a_w) > 0$, this, in particular, implies

$$\nabla_{i_\omega, a_\omega} u(\mathbf{s}) = \frac{1}{|\omega|} \cdot \left( |\omega| \cdot \nabla_{i_\omega, a_\omega} u(\mathbf{s}) + \sum_{a \in \omega} \mu_{i(a), a} \right)$$

$$\overset{(7)}{\geq} \frac{1}{|\omega|} \left( \sum_{a \in \omega} \kappa_{i(a)} - |\omega| \delta \right) = \frac{1}{|\omega|} \sum_{a \in \omega} \kappa_{i(a)} - \delta \,. \tag{9}$$

Secondly, consider another action $a_\omega \neq \tilde{a} \in A^{i_\omega}$, and denote its orbit with $\tilde{\omega} := \omega(\tilde{a})$. Then, we can further show that

$$\frac{1}{|\omega|} \sum_{a \in \omega} \kappa_{i(a)} = \frac{1}{|\tilde{\omega}|} \sum_{a \in \tilde{\omega}} \kappa_{i(a)} \,. \tag{10}$$

To see this, notice that the $\kappa_{i(a)}$'s only depend on the player index $i$. Define $\mathcal{I}(\omega) = \{ i \in \mathcal{N} : \omega \cap A^i \neq \emptyset \}$ as the player set that $\omega$ intersects, and $\mathcal{I}(\tilde{\omega})$ analogously. Then, $i_\omega \in \mathcal{I}(\omega) \cap \mathcal{I}(\tilde{\omega})$. This non-emptiness, in fact, already implies $\mathcal{I}(\omega) = \mathcal{I}(\tilde{\omega})$: Any symmetry $\phi : a_\omega \mapsto a' \in A^{i'}$ for some $i' \in \mathcal{N}$ implies $\phi(\tilde{a}) \in A^{i'}$ as well. Therefore, we can

derive

$$\frac{1}{|\omega|} \sum_{a \in \omega} \kappa_{i(a)} = \frac{c(\omega)}{|\omega|} \sum_{i \in \mathcal{I}(\omega)} \kappa_i = \frac{1}{|\mathcal{I}(\omega)|} \sum_{i \in \mathcal{I}(\omega)} \kappa_i$$

$$= \frac{1}{|\mathcal{I}(\tilde{\omega})|} \sum_{i \in \mathcal{I}(\tilde{\omega})} \kappa_i = \frac{1}{|\tilde{\omega}|} \sum_{a \in \tilde{\omega}} \kappa_{i(a)} \,.$$

**Concluding Membership** Finally, let us show that $\mathbf{s}$ is an $\epsilon$-well-supported Nash equilibrium. As argued earlier, we have to show the following: For $\omega \in W$ with $\mathbf{s}_{i_\omega}(a_w) > 0$ and $\tilde{a} \in A^{i_\omega}$, we have

$$u(a_\omega, \mathbf{s}_{-i_\omega}) = \nabla_{i_\omega, a_\omega} u(\mathbf{s})$$

$$\overset{(9)}{\geq} \frac{1}{|\omega|} \sum_{a \in \omega} \kappa_{i(a)} - \delta \overset{(10)}{=} \frac{1}{|\tilde{\omega}|} \sum_{a \in \tilde{\omega}} \kappa_{i(a)} - \delta$$

$$\overset{\mu \geq 0}{\geq} \frac{1}{|\tilde{\omega}|} \left( -\sum_{a \in \tilde{\omega}} \mu_{i(a), a} + \sum_{a \in \tilde{\omega}} \kappa_{i(a)} \right) - \delta$$

$$\overset{(8)}{\geq} \frac{1}{|\tilde{\omega}|} \left( |\tilde{\omega}| \cdot \nabla_{i_{\tilde{\omega}}, a_{\tilde{\omega}}} u(\mathbf{s}) - |\tilde{\omega}|\delta \right) - \delta$$

$$= \nabla_{i_{\tilde{\omega}}, a_{\tilde{\omega}}} u(\mathbf{s}) - 2\delta \overset{(6)}{=} \nabla_{i_\omega, \tilde{a}} u(\mathbf{s}) - \epsilon$$

$$= u(\tilde{a}, \mathbf{s}_{-i_\omega}) - \epsilon$$

by the choice of $\delta$.

**Hardness** We refer to Ghosh and Hollender (2024)'s concurrent work, which—as mentioned in the main body—develops a stronger CLS-hardness result than us. Arguably, their proof approach is more fundamental than ours. More precisely, they leverage that the reverse connection holds to what we have shown in the membership proof, namely, that the search for a KKT point in a quadratic optimization problem over the simplex can be reduced to finding a Nash equilibrium of a two-player game that is totally symmetric. Nonetheless, we will describe our proof approach for totally symmetric 5-player games (or more players) below.

We reduce hardness from a CLS-completeness result by Tewolde et al. (2023, 2024) for single-player *imperfect-recall games*. Those are extensive-form decision problems (*i.e.*, single-player games), where the player's decision nodes are arbitrarily partitioned into infosets. Then it might be the case that the player forgets information at some decision node that it held at a previous decision node. Lambert, Marple, and Shoham (2019) show that a particular kind of equilibrium (*Causal Decision Theory* (CDT) equilibrium) always exists in such games by showing that they correspond to Nash equilibria that respect a particular set of player symmetries in an associated perfect-recall game. We will argue below that with careful consideration, this construction gives rise to a polytime reduction for the computational search problems of these equilibria. Upon establishing that, we can combine it with Tewolde et al. (2023)[Theorem 2], which shows that finding a CDT equilibrium is CLS-complete, even in single-player imperfect-recall games $G$ with a highly regular game tree: it has a constant depth of 6 for every root-leaf paths, every nonterminal node (*i.e.*, not leaf) is a decision node (*i.e.*, not a chance node), belonging

to the same single infoset of the game, having $m$ actions. In words, the player has to has to take one of $m$ actions 5 times in a row without remembering whether it has taken a decision before or not (or what decision it took).

So let us revisit the construction of the associated perfect-recall game $\Gamma$ to $G$ of the structure described above. Since the infoset in $G$ is visited 5 times in a row by the player, we introduce a team of 5 players PL1 to PL5 to $\Gamma$. Game $\Gamma$ then first uniformly randomly draws an ordering $\sigma$ among those 5 players, without revealing the ordering to the players. Next, the players are asked one after the other to choose one of $m$ actions, without knowing whether other players have acted before them, or what action they chose. Note that we can freely permute the player identities in this game without any affects on the game, because of the uniform random draw of the player order at game start. In this paper's formalism, we can make this precise by looking at the normal-form representation of $\Gamma$ and observing that it is totally symmetric. What Lambert, Marple, and Shoham then showed is that CDT equilibria in $G$ correspond to Nash equilibria of $\Gamma$ that respect this total symmetry, *i.e.*, where all players play the same probability distribution over their $m$ actions. Since we are interested in approximation, they show that the Nash deviation incentives in $\Gamma$ are equal to a multiple of the CDT deviation incentives in $G$. For us, it is a multiple of 5; in general, the multiple equals the expected frequency with which the infoset is visited. Hence, an $\frac{\epsilon}{5}$-Nash equilibrium of $\Gamma$ that respect its total symmetry makes an $\epsilon$-CDT equilibria of $G$.

The construction is polytime: $\Gamma$ is 5! times as big as $G$, and $\Gamma$'s normal-form game representation is a payoff tensor of size $m^5$. Lastly, we enumerate the 5! (player) symmetries in explicit form. $\qquad\square$

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

*Proof.* We show how to perform the equilibrium computation on the orbits instead of strategy profiles. Let $D$ be the set of strategy profiles that respect the given symmetries, which we may assume are in orbit form. For each orbit $\omega \in W$, let $a(\omega) \in \omega$ be a designated action representative of $\omega$ and $i(\omega)$ the player that $a(\omega)$ belongs to. Also compute the orbit cardinalities $c(\omega) = |A^{i(\omega)} \cap \omega| > 0$ for all $\omega \in W$, as in the proof of Theorem 8.

We define the orbit profile set $E$ as any $\mathbf{r} \in \mathbb{R}^{|W|}$ that satisfies the following linear (in-)equality system:

$$
\begin{aligned}
\mathbf{r}_\omega &\geq 0 & \forall \omega \in W \\
\mathbf{r}_\omega &\leq 1 & \forall \omega \in W \\
\sum_{w' \in W \text{ with } w' \cap A^{i(\omega)} \neq \emptyset} \mathbf{r}_{\omega'} &= 1 & \forall \omega \in W.
\end{aligned}
\tag{11}
$$

We show that there is a one-to-one correspondance between $D$ and $E$, that is, between strategy profiles $\mathbf{s}$ that respect the orbits and orbit profiles. Given $\mathbf{s} \in D$ define $\mathbf{r} = \tilde{\mathbf{r}}(\mathbf{s})$ as

$$
\mathbf{r}_\omega := \sum_{a \in A^{i(\omega)} \cap \omega} \mathbf{s}_{i(\omega)}(a) = c(\omega) \cdot \mathbf{s}_{i(\omega)}(a(\omega)).
$$

This works because since $\mathbf{s}$ respects the orbits, the sum above will be the same independent of the choice of representative $a(\omega)$ and, in fact, each summand will be equal, which also yields the latter equality. In particular, $\tilde{\mathbf{r}}(\mathbf{s})$ satisfies (11). To see why the last equality holds, for each $\omega \in W$, we can imagine that each of these $\omega'$ have had their representitives chosen as $i(\omega') = i(\omega)$, which simplifies the sum to $\sum_{a \in A^{i(\omega)}} \mathbf{s}_{i(\omega)}(a) = 1$. Next, given an orbit profile $\mathbf{r} \in E$, define $\mathbf{s} = \tilde{\mathbf{s}}(\mathbf{r})$ as $\mathbf{s}_i(a) = \frac{\mathbf{r}_\omega}{c(\omega)}$ where $\omega$ satisfies $a \in \omega$. By similar reasoning, $\mathbf{s}$ makes a strategy profile. It also respects the orbits because each action of the same orbit is assigned the same probability. We have that those two maps $\tilde{\mathbf{r}} : D \to E$ and $\tilde{\mathbf{s}} : E \to D$ are each other inverse (both-sided). Moreover, with precomputed representatives $a(\omega)$, cardinalities $c(\omega)$, and pointers that tell us to what orbit an action $a \in A^i$ belongs, we can compute $\tilde{\mathbf{s}}(\mathbf{r})$ and $\tilde{\mathbf{r}}(\mathbf{s})$ in linear time.

Now, construct all polynomial function $u^i : S \to \mathbb{R}$ in linear time. Next, construct utility functions $v^i := u^i \circ \tilde{\mathbf{s}}$. We remark a difference between $u^i$ and $v^i$: function $u^i$ is *multilinear* in each $S^i$, *i.e.*, in each monomial in $u^i$ at most one action probability $\mathbf{s}_i(a)$ occurs out of $\{\mathbf{s}_i(a')\}_{a' \in A^i}$ for each player $i$. Functions $v^i$, on the other hand, are not, for example, they may have a $\mathbf{r}_\omega^k$ occur in a monomial with a higher degree $k \geq 2$ if $\omega$ covers actions from different players. Next, construct for each orbit $\omega \in W$ the deviation incentive function

$$
g_\omega(\mathbf{r}) := u^{i(\omega)}\big(a(\omega), \tilde{\mathbf{s}}(\mathbf{r})_{-i(\omega)}\big),
$$

which again is a polynomial function over $E$. With all of this, consider the additional inequalities:

$$
g_\omega(\mathbf{r}) \leq v^{i(\omega)}(\mathbf{r}) \quad \forall \omega \in W
\tag{12}
$$

Then we claim that symmetries-respecting $\mathbf{s} \in D$ is a Nash equilibrium if and only if its associated $\mathbf{r} = \tilde{\mathbf{r}}(\mathbf{s}) \in E$ satisfies (12). This is because for any $a \in A^i$ we have (if $\omega$ denotes the orbit it belongs to):

$$
\begin{aligned}
u^i(a, \mathbf{s}_{-i}) - u^i(\mathbf{s}) &\overset{(2)}{=} u^{i(\omega)}\big(a(\omega), \mathbf{s}_{-i(\omega)}\big) - u^{i(\omega)}(\mathbf{s}) \\
&= g_\omega(\mathbf{r}) - v^{i(\omega)}(\mathbf{r}).
\end{aligned}
$$

Thus, the former is nonpositive if and only if the latter is nonpositive.

Therefore, we reduced in polytime the task of computing a symmetries-respecting Nash equilibrium $\mathbf{s}$ to the task of finding an element in $\mathbb{R}^{|W|}$ that satisfies the joint system of (11) and (12). This system consists of $4|W|$ polynomial (in-)equalities, with the highest degree of an occuring polynomial being that of $v^i$ which is at most $N$ (for when an orbit covers at least one action of each player). The algorithm by Renegar (1992) for solving such a system takes polytime in all parameters of such a system except in $(\#(\text{in-})\text{equalities} \cdot \#\text{degree})^{\#(\text{in-})\text{equalities}}$. We can turn this into an inverse-exponential approximation of a solution in $E$ with a subdivision algorithm on $E$ with query access to Renegar (for details, compare with Tewolde et al. (2024)[Proposition 9]). Since the polynomial functions are nicely Lipschitz-continuous, inverse-exponential approximation in $E$ also

implies inverse-exponentially decreasing deviation incentives $\epsilon$ in the definition of an $\epsilon$-Nash equilibrium. All in all, that makes a poly$\big(|\Gamma|, \log(1/\epsilon), (|W|N)^{|W|}\big)$ to compute an orbit profile $\mathbf{r}$ whose associated strategy profile $\mathbf{s} = \tilde{\mathbf{s}}(\mathbf{r})$ makes an $\epsilon$-Nash equilibrium of $\Gamma$ that respects the given symmetries.

Lastly, if we only consider the case of two players ($N = 2$), then we can do better: use the *support enumeration* approach (Dickhaut and Kaplan 1993) instead, but in the orbit profile space $E$. For that, we note that the viable supports for a symmetries-respecting strategy profile $\mathbf{s} \in D$ are exactly all possible supports of associated orbit profile $\mathbf{r} = \tilde{\mathbf{r}}(\mathbf{s})$. There are $2^{|W|}$ of such supports, and checking whether there exists an orbit profile with a given support that is also associated to a Nash equilibrium $\mathbf{s} \in D$ can be done in polytime as usual. $\qquad\square$

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

For this result we will first prove a intermediate result:

**Lemma 27.** *Let $\Gamma$ be a two-player zero-sum game. Let $f : [0, 1] \to \mathbb{R}$ be any strictly convex function, and define the regularizer $\rho : S^1 \times S^2 \to \mathbb{R}$ by*

$$\rho(\mathbf{s}) = \sum_{a \in \mathbb{A}} f(\mathbf{s}(a)).$$

*Then the Nash equilibrium $\mathbf{s}^* \in S$ that minimizes the regularizer $\rho$ also respects all symmetries.*

*Proof.* Let $\phi : \mathbb{A} \to \mathbb{A}$ be a symmetry of $\Gamma$. Note that $\phi$ is a permutation, and let $n$ be the order of $\phi$. That is, $n$ is the smallest positive integer such that $\phi^n$ is the identity. Define the sequence of strategies $\mathbf{s}^1 = \phi(\mathbf{s}^*), \mathbf{s}^2 = \phi^2(\mathbf{s}^*), \ldots, \mathbf{s}^n = \phi^n(\mathbf{s}^*) = \mathbf{s}^*$. By Remark 14 each $\mathbf{s}^k$ is itself a Nash equilibrium of $\Gamma$. Since the set of Nash equilibria of any zero-sum game is convex, the average strategy profile $\bar{\mathbf{s}} := \frac{1}{n} \sum_{k=1}^n \mathbf{s}^k$ is also a Nash equilibrium. Moreover, $\rho$ is a function that respects symmetries in the sense that $\rho(\mathbf{s}) = \rho(\phi(\mathbf{s}))$, because $\phi$ only reorders the probability values in $\mathbf{s}$ and $\rho$ only depends on the present values. Therefore, since $\rho$ is also strictly concave, we have

$$\rho(\bar{\mathbf{s}}) \leq \frac{1}{n} \sum_{k=1}^n \rho(\mathbf{s}^k) = \rho(\mathbf{s}^*)$$

with equality only when $\bar{\mathbf{s}} = \mathbf{s}^k$ for all $k$. Since $\mathbf{s}^*$ minimizes $\rho$ among all Nash equilibria, we thus have that all the $\bar{\mathbf{s}}^k$'s are the same, *i.e.*, $\mathbf{s}^*$ obeys the symmetry $\phi$. As symmetry $\phi$ was chosen arbitrarily, $\mathbf{s}^*$ obeys all symmetries of $\Gamma$. $\qquad\square$

Theorem 10 now follows by noting that finding a $\rho$-minimizing Nash equilibrium can be expressed as a convex quadratic program.

*Proof of Theorem 10.* Let $\Gamma$ be a two-player zero-sum game described by P1's payoff matrix $U \in \mathbb{R}^{m_1 \times m_2}$. Thanks to the minimax theorem (von Neumann 1928), $\Gamma$ then has a

well-defined *value* $v \in \mathbb{R}$, and its Nash equilibrium set can be determined via the following linear (in-)equality system

$$\mathbf{s}_1 \in S^1, \mathbf{s}_2 \in S^2$$
$$(U\mathbf{s}_2)_a \leq v \quad \forall a \in A^1$$
$$(\mathbf{s}_1^T U)_a \leq v \quad \forall a \in A^2 \, .$$

Now add the quadratic objective function $\rho(\mathbf{s}_1, \mathbf{s}_2)$ to it that is defined as in Lemma 27 by, for example, $f(x) := x^2$. This makes it a convex quadratic program which can be solved in polytime (Kozlov, Tarasov, and Khachiyan 1980). $\qquad\square$