For Learning in Symmetric Teams, Local Optima are Global Nash Equilibria

Anonymous Authors[1]

Abstract

Although it has been known since the 1970s that a *globally* optimal strategy profile in a common-payoff game is a Nash equilibrium, global optimality is a strict requirement that limits the result's applicability. In this work, we show that any *locally* optimal symmetric strategy profile is also a (global) Nash equilibrium. Furthermore, we show that this result is robust to perturbations to the common payoff and to the local optimum. Applied to machine learning, our result provides a global guarantee for any gradient method that finds a local optimum in symmetric strategy space. While this result indicates stability to *unilateral* deviation, we nevertheless identify broad classes of games where mixed local optima are unstable under *joint*, asymmetric deviations. We analyze the prevalence of instability by running learning algorithms in a suite of symmetric games, and we conclude by discussing the applicability of our results to multi-agent RL, cooperative inverse RL, and decentralized POMDPs.

1. Introduction

We consider *common-payoff* games (also known as *identical interest* games (Ui, 2009)), in which the payoff to all players is always the same. Such games model a wide range of situations involving cooperative action towards a common goal. Under the heading of *team theory*, they form an important branch of economics (Marschak, 1955; Marschak & Radner, 1972). In cooperative AI (Dafoe et al., 2021), the common-payoff assumption holds in *Dec-POMDPs* (Oliehoek et al., 2016), where multiple agents operate independently according to policies designed centrally to achieve a common objective. Many applications of *multiagent reinforcement learning* also assume a common payoff (Foerster et al., 2016; 2018; Gupta et al., 2017). Finally, in *assistance games* (Russell, 2019) (also known as cooperative inverse reinforcement learning or CIRL games (Hadfield-Menell et al., 2017)), which include at least one human and one or more "robots," it is assumed that the robots' payoffs are exactly the human's payoff, even if the robots must learn it.

Our focus is on symmetric strategy profiles in common-payoff games. Loosely speaking, a symmetric strategy profile is one in which some subset of players share the same strategy; Section 3 defines this in a precise sense. For example, in Dec-POMDPs, an offline solution search may consider only symmetric strategies as a way of reducing the search space. (Notice that this does not lead to identical *behavior*, because strategies are state-dependent.) In common-payoff multiagent reinforcement learning, each agent may collect percepts and rewards independently, but the reinforcement learning updates can be pooled to learn a single parameterized policy that all agents share; prior work has found experimentally that "parameter sharing is crucial for reaching the optimal protocol" (Foerster et al., 2016). In team theory, it is common to develop a strategy that can be implemented by every employee in a given category and leads to high payoff for the company. In civic contexts, symmetry commonly arises through notions of fairness and justice. In treaty negotiations and legislation that mandates how parties behave, for example, there is often a constraint that all parties be treated equally.

For the purposes of this paper, we consider Nash equilibria—strategy profiles for all players from which no individual player has an incentive to deviate—as a reasonable solution concept. Marschak & Radner (1972) make the obvious point that a globally optimal (possibly asymmetric) strategy profile—one that achieves the highest common payoff—is necessarily a Nash equilibrium. Moreover, it can be found in time linear in the size of the payoff matrix.

In any sufficiently complex game, however, we should not expect to be able to find a *globally* optimal strategy profile. For example, matrix games have size exponential in the number of players, and the matrix representation of a game tree has size exponential in the depth of the tree. Therefore, global search over all possible contingency plans is infeasible for all but the smallest of games. This is why some

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

		Bot L	Bot W
Rob	L	1	2
	W	0	1

(a) Bot's laundry basket is broken

		Bot L	Bot W
Rob	L	1	2
	W	2	1

(b) Bot's laundry basket is fixed

		Bot L	Bot W
Rob	L	3	2
	W	2	3

(c) Synergy working together

Table 1: Three versions of the laundry/washing up game. Solutions are described in the text.

of the most effective methods in machine learning, such as gradient methods, employ *local* search over strategy space.

Lacking global guarantees, local search methods may converge only to *locally* optimal strategy profiles. Roughly speaking, a *locally optimal* strategy profile is a strategy profile from which no group of players has an incentive to slightly deviate. Obviously, a locally optimal profile may not be a Nash equilibrium, as a player may still have an incentive to deviate to some more distant point in strategy space. Nonetheless, Ratliff et al. (2016) argue that a local Nash equilibrium may still be stable in a practical sense if agents are computationally unable to find a better strategy.

The central question of this work is: what can we say about the (global) properties of locally optimal symmetric strategy profiles? Our first main result, informally stated, is that *in a symmetric, common-payoff game, every local optimum in symmetric strategies is a (global) Nash equilibrium*. Section 4 states the result more precisely and gives an example illustrating its generality. Section 4.2 shows that the result is robust to perturbations to the common payoff and to the local optimum. Section 3.5 elaborates on the symmetry required by the result, illustrating how the theorem applies even when the physical environment is asymmetric and when players have differing capabilities. Complete proofs for all of our results are in the appendices.

Despite many decades of research on symmetry in common-payoff games, our result appears to be novel. There are some echoes of the result in the literature on single-agent decision making (Piccione & Rubinstein, 1997; Briggs, 2010; Schwarz, 2015), which can be connected to symmetric solutions of common-payoff games by treating all players jointly as a single agent, but our result appears more general than published results. Perhaps closest to our work is Piccione & Rubinstein (1997), which establishes an equilibrium-of-sorts among the "modified multi-selves" of a single player's information set. The proof we give of our result also contains elements similar to the proof (of a related but different result) in Taylor (2016).

In the second half of our paper, we turn to the thorny question of *stability*. Instability, if not handled carefully, might lead to major coordination failures in practice (Bostrom et al., 2016). While it is already known that local strict optima in a totally symmetric team game attain one type

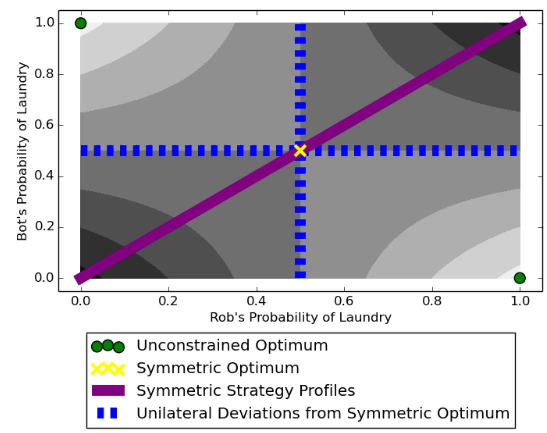


Figure 1: The strategy profile landscape of the symmetric laundry game (Table 1b). Lighter color indicates higher expected utility. Although the symmetric optimum has lower expected utility than the unrestricted optima, total symmetry of the game implies that the symmetric optimum is a Nash equilibrium; this is a special case of Theorem 4.0.2.

of stability, the issue is complex because there are several ways of enforcing (or not enforcing) strict symmetries in payoffs and strategies (Milchtaich, 2016). Whereas our first main result implies stability to *unilateral* deviation, our second main result establishes when stability exists to *joint*, possibly-asymmetric, deviation. We prove for a non-degenerate class of games that local optima in symmetric strategy space fail to be local optima in asymmetric strategy space if and only if at least one player is mixing, and we experimentally quantify how often mixing occurs for learning algorithms in the GAMUT suite of games (Nudelman et al., 2004).

2. Motivating examples

To gain some intuition for these concepts and claims, let us consider a situation in which two robots, Rob and Bot, have to do some housework—specifically, laundry (L) and washing up (W). Here, the common payoff is to the owners. It is evident that a symmetric strategy profile—both doing the laundry or both doing the washing up—is not ideal, because the other task will not get done.

The first version of the game, whose payoffs U are shown in Table 1a, is asymmetric: while Rob can do both tasks, Bot's built-in laundry basket is broken and cannot hold clothes. Here, as Marschak and Radner (Marschak & Radner, 1972) pointed out, the strategy profile (L, W) is both globally optimal and a Nash equilibrium. If we posit a mixed (randomized) strategy profile in which Rob and Bot have laundry probabilities p and q respectively, the gradients $\partial U/\partial p$ and $\partial U/\partial q$ are $+1$ and -1, driving the solution to (L, W).

In the second version of the game (Table 1b), Bot's built-in laundry basket has been repaired, and symmetry is restored. The pure profiles (L, W) and (W, L) are (asymmetric) globally optimal solutions and hence Nash equilibria. Figure 1 shows the entire payoff landscape as a function of p and q: looking just at symmetric strategy profiles, it turns out that there is a local optimum at $p = q = 0.5$, i.e., where Rob and Bot toss fair coins to decide what to do. Although the expected payoff of this solution is lower than that of the asymmetric optima, the local optimum is, nonetheless, a Nash equilibrium. All unilateral deviations from the symmetric local optimum result in the same expected payoff because if one robot is tossing a coin, the other robot can do nothing to improve the final outcome.

In the third version of the game (Table 1c), the robots are able to do the best quality of work when they work together on a task. In this case, there is again a Nash equilibrium at $p = q = 0.5$, but it is a local minimum rather than a local maximum in symmetric strategy space. Thus, not all symmetric Nash equilibria are symmetric local optima; this is because Nash equilibria depend on *unilateral* deviations, whereas symmetric local optima depend on *joint* deviations that maintain symmetry.

2.1. Complex coordination example where a simple symmetric strategy is best

Consider 10 robots that must each choose between 3 actions, a, b, and c. If all robots play action a, they receive a reward of 1. If exactly one robot plays action b while the rest play action c, they receive a reward of $1 + \epsilon$. Otherwise, the reward is 0. For small enough ϵ, the optimal symmetric policy is for all robots to play action a. Here, trying to coordinate in symmetric strategies to reach the asymmetric optimum is suboptimal—the best symmetric strategy is the simple one. Furthermore, our subsequent theory shows that the best symmetric strategy is stable; it is locally optimal even when considering *joint* (possibly asymmetric) deviations.

3. Preliminaries: games and symmetries

3.1. Normal-form games

Throughout, we consider *normal-form games* $\mathcal{G} = (N, A, u)$ defined by a finite set N with $|N| = n$ players, a finite set of action profiles $A = A_1 \times A_2 \times \ldots \times A_n$ with A_i specifying the actions available to player i, and the utility function $u = (u_1, u_2, \ldots, u_n)$ with $u_i : A \to \mathbb{R}$ giving the utility for each player i (Shoham & Leyton-Brown, 2008). We call $\mathcal{G}$ *common-payoff* if $u_i(a) = u_j(a)$ for all action profiles $a \in A$ and all players i, j. In common-payoff games we may omit the player subscript i from utility functions.

We model each player as employing a (mixed) *strategy* $s_i \in \Delta(A_i)$, a probability distribution over actions. We denote the support of the probability distribution s_i by $\text{supp}(s_i)$. Given a (mixed) *strategy profile* $s = (s_1, s_2, \ldots, s_n)$ that specifies a strategy for each player, player i's expected utility is $EU_i(s) = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j)$. If a strategy s_i for player i maximizes expected utility given the strategies s_{-i} of all the other players, i.e., if $s_i \in \text{argmax}_{s'_i \in \Delta(A_i)} EU_i(s'_i, s_{-i})$, we call s_i a *best response* to s_{-i}. If each strategy s_i in a strategy profile s is a best response to s_{-i}, we call s a *Nash equilibrium*. A Nash equilibrium s is *strict* if every s_i is the unique best response to s_{-i}.

Note that, while we have chosen to use the normal-form game representation for simplicity, normal-form games are highly expressive. Normal-form games can represent mixed strategies in all finite games, including games with sequential actions, stochastic transitions, and partial observation such as imperfect-information extensive form games with perfect recall, Markov games, and Dec-POMDPs. To represent a sequential game in normal form, one simply lets each normal-form action be a complete strategy (contingency plan) accounting for every potential game decision.

3.2. Symmetry in game structure

We adopt the fairly general group-theoretic notions of symmetry introduced by von Neumann & Morgenstern (1944) and Nash (1951), and we borrow notation from Plan (2017). More recent work has analyzed *narrower* notions of symmetry (Reny, 1999; Vester, 2012; Milchtaich, 2016). For example, Daskalakis & Papadimitriou (2007) study "anonymous games" and show that anonymity substantially reduces the complexity of finding solutions. Additionally, Ham (2013) generalizes the player-based notion of symmetry to include further symmetries revealed by renamings of actions. We conjecture our results extend to this more general case, at some cost in notational complexity, but we leave this to future work.

Our basic building block is a *symmetry of a game*:

Definition 3.2.1. *Call a permutation of player indices* $\rho : \{1, 2, ..., n\} \to \{1, 2, ..., n\}$ *a symmetry of a game* $\mathcal{G}$ *if, for all strategy profiles* $(s_1, s_2, ..., s_n)$*, permuting the strategy profile permutes the expected payoffs:* $EU_{\rho(i)}((s_1, s_2, ..., s_n)) = EU_i((s_{\rho(1)}, s_{\rho(2)}, ..., s_{\rho(n)})),\ \forall i.$

Note that, when we speak of a symmetry of a game, we implicitly assume $A_i = A_j$ for all i, j with $\rho(i) = j$ so that permuting the strategy profile is well-defined.[1]

We characterize the symmetric structure of a game by its set of game symmetries:

Definition 3.2.2. *Denote the set of all symmetries of a game $\mathcal{G}$ by: $\Gamma(\mathcal{G}) = \{\rho : \{1, 2, ..., n\} \to \{1, 2, ..., n\}$ a symmetry of $\mathcal{G}\}$.*

A spectrum of game symmetries is possible. On one end of the spectrum, the identity permutation might be the only symmetry for a given game. On the other end of the spectrum, all possible permutations might be symmetries for a given game. Following the terminology of von Neumann & Morgenstern (1944), we call the former case *totally unsymmetric* and the latter case *totally symmetric*:

Definition 3.2.3. *If $\Gamma(\mathcal{G}) = S_n$, the full symmetric group, we call the game $\Gamma(\mathcal{G})$ totally symmetric. If $\Gamma(\mathcal{G})$ contains only the identity permutation, we call the game totally unsymmetric.*

Let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be any subset of the game symmetries. Because $\Gamma(\mathcal{G})$ is closed under composition, we can repeatedly apply permutations in $\mathcal{P}$ to yield a group of game symmetries $\langle \mathcal{P} \rangle$:

Definition 3.2.4. *Let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries. The group generated by $\mathcal{P}$, denoted $\langle \mathcal{P} \rangle$, is the set of all permutations that can result from (possibly repeated) composition of permutations in $\mathcal{P}$: $\langle \mathcal{P} \rangle = \{\rho_1 \circ \rho_2 \circ \ldots \circ \rho_m \mid m \in \mathbb{N}, \rho_1, \rho_2, \ldots, \rho_m \in \mathcal{P}\}$.*

Group theory tells us that $\langle \mathcal{P} \rangle$ defines a closed binary operation (permutation composition) including an identity and inverse maps, and $\langle \mathcal{P} \rangle$ is the closure of $\mathcal{P}$ under function composition.

With a subset of game symmetries $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ in hand, we can use the permutations in $\mathcal{P}$ to carry one player index to another. For each player i, we give a name to the set of player indices to which permutations in $\mathcal{P}$ can carry i: we call it player i's *orbit*.

Definition 3.2.5. *Let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. The orbit of player i under $\mathcal{P}$ is the set of all other player indices that $\langle \mathcal{P} \rangle$ can assign to i: $\mathcal{P}(i) = \{\rho(i) \mid \rho \in \langle \mathcal{P} \rangle\}$.*

By standard group theory, the orbits of a group action on a set partition the set's elements, so:

Proposition 3.2.6. *Let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$. The orbits of $\mathcal{P}$ partition the game's players.*

[1] We make this choice to ease notational burden, but we conjecture that our results can be generalized to allow for mappings between actions (Ham, 2013), which we leave for future work.

Proposition 3.2.6 tells us each $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ yields an equivalence relation among the players. To gain intuition for this equivalence relation, consider two extreme cases. In a totally unsymmetric game, $\Gamma(\mathcal{G})$ contains only the identity permutation, in which case each player is in its own orbit of $\Gamma(\mathcal{G})$; the equivalence relation induced by the orbit partition shows that no players are equivalent. In a totally symmetric game, by contrast, every permutation is an element of $\Gamma(\mathcal{G})$, i.e., $\Gamma(\mathcal{G}) = S_n$, the full symmetric group; now, all the players share the same orbit of $\Gamma(\mathcal{G})$, and the equivalence relation induced by the orbit partition shows that all the players are equivalent.

We leverage the orbit structure of an arbitrary $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ to define an equivalence relation among players because it adapts to however much or little symmetry is present in the game. Between the extreme cases of no symmetry (n orbits) and total symmetry (1 orbit) mentioned above, there could be any intermediate number of orbits of $\mathcal{P}$. Furthermore, it might not be the case that players who share an orbit can be swapped in arbitrary ways. For an example of this, see Appendix C.

3.3. Symmetry in strategy profiles

Having formalized a symmetry of a game in the preceding section, we follow Nash (1951) and define symmetry in strategy profiles with respect to symmetry in game structure:

Definition 3.3.1. *Let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. We call a strategy profile $s = (s_1, s_2, ..., s_n)$ $\mathcal{P}$-invariant if $(s_1, s_2, ..., s_n) = (s_{\rho(1)}, s_{\rho(2)}, ..., s_{\rho(n)})$ for all $\rho \in \langle \mathcal{P} \rangle$.*

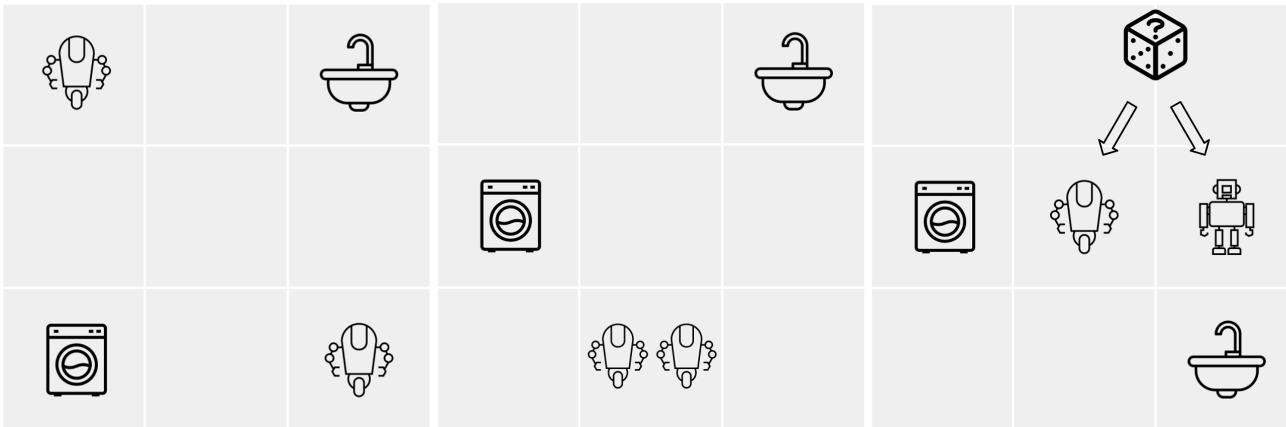
The equivalence relation among players induced by the orbit structure of $\mathcal{P}$ is fundamental to our definition of symmetry in strategy profiles by the following proposition:

Proposition 3.3.2. *A strategy profile $s = (s_1, s_2, ..., s_n)$ is $\mathcal{P}$-invariant if and only if $s_i = s_j$ for each pair of players i and j with $\mathcal{P}(i) = \mathcal{P}(j)$.*

To state Proposition 3.3.2 another way, a strategy profile is $\mathcal{P}$-invariant if all pairs of players i and j that are equivalent under the orbits of $\mathcal{P}$ play the same strategy.

3.4. Symmetry via the veil of ignorance

Sometimes strategies must be specified for all players before knowing the players' roles and initial conditions. Consider writing laws or programming household robots; all players are treated equally in specifying situation-dependent contingency plans. When all players have equal likelihood of ending up in any given situation (e.g., when all players have the same initial state distribution), the game of choosing contingency plans *a priori* is totally symmetric. (Appendix A gives an example.) For its analog in the philosophy of Rawls



(a) Symmetric agents & environment (b) Same initial conditions (c) Veil of ignorance

Figure 2: Various laundry / washing up grid-world games that satisfy our symmetry requirement. (a) Symmetric agents in a symmetric environment. (b) Although the environment is asymmetric, the game is still symmetric because the robots have the same initial condition. (c) When agents must be programmed *before* knowing their initial conditions (e.g. location, morphology), symmetry holds behind the *veil of ignorance* (Section 3.4) even with asymmetric agents and environments.

(1971) and Harsanyi (1975), we call this situation the *veil of ignorance*.

3.5. What do symmetric games look like?

To illustrate types of symmetry in games, Figure 2 presents symmetric variants of a laundry / washing up grid-world game inspired by the motivating example of Section 2. The robots are on a team to do the laundry / washing up, and their movement and interaction is restricted to adjacent grid cells.

An idealized symmetric environment is shown in Figure 2a. Here, the robots are identical, and the environment is perfectly symmetric; the symmetry of the game is clear. This is the sort of symmetry that might be found in highly controlled environments such as factories.

A commonplace, asymmetric environment is shown in Figure 2b. Because the robots are identical and have the same initial condition, their action sequences can be swapped without changing the outcome of the game. Thus, *the game is symmetric even though the environment is asymmetric*. While it is impossible for real-world robots to have the exact same physical location, it suffices for them to have the same *distribution* over initial conditions. Furthermore, we expect that *virtual* agents (such as customer service chatbots or nodes in a compute cluster) may have identical initial conditions.

Asymmetric agents in an asymmetric environment are shown in Figure 2c. If we assume that the morphology and / or the initial location of each robot is equally random, then the game of choosing contingency plans behind the veil of ignorance (Section 3.4) is totally symmetric. We expect this case of symmetry to be common when AI uses the same source code or the same learned parameters. In fact, weight sharing is already common practice in multi-agent RL (Foerster et al., 2016).

4. Local symmetric optima are (global) Nash equilibria

After the formal definitions of symmetry in the previous section, we are almost ready to formally state the first of our main results. The only remaining definition is that of a local symmetric optimum:

Definition 4.0.1. *Call s a locally optimal $\mathcal{P}$-invariant strategy profile of a common-payoff game if: (i) s is $\mathcal{P}$-invariant, and (ii) for some $\epsilon > 0$, no $\mathcal{P}$-invariant strategy s' with $EU(s') > EU(s)$ can be formed by adding or subtracting at most ϵ to the probability of taking any given action $a_i \in A_i$. If, furthermore, condition (ii) holds for all $\epsilon > 0$, we call s a globally optimal $\mathcal{P}$-invariant strategy profile or simply an optimal $\mathcal{P}$-invariant strategy profile.*

Now we can state our first main theorem, that local symmetric optima are (global) Nash equilibria:

Theorem 4.0.2. Let $\mathcal{G}$ be a common-payoff normal-form game, and let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. Any locally optimal $\mathcal{P}$-invariant strategy profile is a Nash equilibrium.

Proof. We provide a sketch here and full details in Appendix B. Suppose, for the sake of contradiction, that an individual player i could beneficially deviate to action a_i

(if a beneficial deviation exists, then there is one to a pure strategy). Then, consider instead a collective change to a symmetric strategy profile in which all the players in i's orbit shift slightly more probability to a_i. By making the amount of probability shifted ever smaller, the probability that this change affects exactly one agent's realized action (making it a_i when it would not have been before) can be arbitrarily larger than the probability that it affects multiple agents' realized actions. Moreover, if this causes exactly one agent's realized action to change, this must be in expectation beneficial, since the original unilateral deviation was in expectation beneficial. Hence, the original strategy profile cannot have been locally optimal. □

4.1. Applications of the theorem

First, we provide an example of applying Theorem 4.0.2 to multi-agent RL.

Example 4.1.1. *Consider a cooperative multi-agent RL environment where all agents have the same initial state distribution. Suppose, as is typical practice (Foerster et al., 2016), that we use a gradient method to train the parameters of a policy that all agents will share. Assume that the gradient method reaches a symmetric local optimum in mixed strategy space. If we wanted to improve upon this symmetric local optimum, we might lift the symmetry requirement and perform iterative best response, i.e., continue learning by updating the parameters of just one agent. However, by Theorem 4.0.2, the symmetric local optimum is a Nash equilibrium. Thus, updating the parameters of a single agent cannot improve the common payoff; updating the parameters of at least two agents is necessary.*

The preceding example assumes that a gradient method in multi-agent RL reaches a symmetric local optimum in mixed strategy space. In practice, agents may employ behavioral strategies, and it may not be possible to verify how close a symmetric strategy profile is to a local optimum.

In Appendix C, we give another example that shows how Theorem 4.0.2 is more general than the case of total symmetry. The example illustrates the existence of rotational symmetry without total symmetry, and it illustrates how picking different $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ leads to different optimal $\mathcal{P}$-invariant strategies and thus different $\mathcal{P}$-invariant Nash equilibria by Theorem 4.0.2.

4.2. Robustness to payoff and strategy perturbations

Theorem 4.0.2 assumes that all players' payoffs are exactly the same, and it applies to strategy profiles that are exact local optima. If we relax these assumptions, the theorem still holds approximately. If all players' payoffs are equal $\pm\epsilon$, or if a strategy profile is ϵ distance away from a symmetric local optimum, then a robust version of Theorem 4.0.2 guarantees a $k\epsilon$-Nash equilibrium for some game-dependent constant k. See Appendix D for a precise treatment of these robustness results.

While the results of this section concern Nash equilibria, we note that Nash equilibria, by definition, consider the possibility of only a *single* agent deviating. In the next section, we investigate when multiple agents might have an incentive to *simultaneously* deviate by studying the optimality of symmetric strategy profiles in possibly-asymmetric strategy space.

5. When are local optima in symmetric strategy space also local optima in possibly-asymmetric strategy space?

Our preceding theory applies to locally optimal $\mathcal{P}$-invariant, i.e., symmetric, strategy profiles. This leaves open the question of how well locally optimal symmetric strategy profiles perform when considered in the broader, possibly-asymmetric strategy space. When are locally optimal $\mathcal{P}$-invariant strategy profiles also locally optimal in possibly-asymmetric strategy space? This question is important in machine learning (ML) applications where users of symmetrically optimal ML systems might be motivated to make modifications to the systems, even for purposes of a common payoff.

To address this precisely, we formally define a *local optimum in possibly-asymmetric strategy space*:

Definition 5.0.1. *A strategy profile $s = (s_1, s_2, \ldots, s_n)$ of a common-payoff normal-form game is locally optimal among possibly-asymmetric strategy profiles, or, equivalently, a local optimum in possibly-asymmetric strategy space, if for some $\epsilon > 0$, no strategy profile s' with $EU(s') > EU(s)$ can be formed by changing s in such a way that the probability of taking any given action $a_i \in A_i$ for any player i changes by at most ϵ.*

Definition 5.0.1 relates to notions of *stability* under dynamics, such as those with perturbations or stochasticity, that allow multiple players to make asymmetric deviations. In particular, if s is not a local maximum in asymmetric strategy space, this means that there is some set of players C and strategy s'_C arbitrarily close to s, such that if players C were to play s'_C (by mistake or due to stochasticity), some Player $i \in N - C$ would develop a strict preference over the support of s_i. To illustrate this, we return to the laundry/washing up game of the introduction.

Example 5.0.2. *Consider again the game of Table 1b. As Figure 1 illustrates, the symmetric optimum is for both Rob and Bot to randomize uniformly between W and L. While this is a Nash equilibrium, it is not a local optimum in possibly-asymmetric strategy space. If one player deviates from uniformly randomizing, the other player develops a* *strict preference for either W or L.*

To generalize the phenomenon of Example 5.0.2, we use the following *degeneracy*[2] condition:

Definition 5.0.3. *Let s be a Nash equilibrium of a game $\mathcal{G}$: (i) If s is deterministic, i.e., if every s_i is a Dirac delta function on some a_i, then s is degenerate if at least two players i are indifferent between a_i and some other $a'_i \in A_i - \{a_i\}$. (ii) Otherwise, if s is mixed, then s is degenerate if for all players i and all $a_{-i} \subseteq \operatorname{supp}(s_{-i})$, the term $EU_i(a_i, a_{-i})$ is constant across $a_i \in \operatorname{supp}(s_i)$.*

We call a game $\mathcal{G}$ degenerate if it has at least one degenerate Nash equilibrium.

Intuitively, our definition says that a deterministic Nash equilibrium is non-degenerate when it is strict or almost strict (excepting of at most one player who may be indifferent over available actions). A mixed Nash equilibrium, on the other hand, is non-degenerate when *mixing matters*.

In non-degenerate games, our next theorem shows that a local symmetric optimum is a local optimum in possibly-asymmetric strategy space if and only if it is deterministic. Formally:

Theorem 5.0.4. Let $\mathcal{G}$ be a non-degenerate common-payoff normal-form game, and let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. A locally optimal $\mathcal{P}$-invariant strategy profile is locally optimal among possibly-asymmetric strategy profiles if and only if it is deterministic.

To see why the non-degeneracy condition is needed in Theorem 5.0.4, we provide an example of a degenerate game:

	a	b	c
a	1	1	1
b	1	-10	$1+\epsilon$
c	1	$1+\epsilon$	-10

Example 5.0.5. *Consider the 3x3 symmetric game shown above. Here, (a, a) is the unique global optimum in symmetric strategy space. By Theorem 4.0.2, it is therefore also a Nash equilibrium. However, it is a degenerate Nash equilibrium and not locally optimal in asymmetric strategic space. The payoff can be improved by, e.g., the row player shifting small probability to b, and the column player shifting small probability to c.*

We have already seen an example of a non-degenerate deterministic equilibrium. The symmetric optimum from Section 2.1, even though it is not the *global* asymmetric optimum, is nevertheless *locally* optimal in possibly-asymmetric strategy space by Theorem 5.0.4.

[2]We note that "degnerate" is already an established term in the game-theoretic literature where it is often applied only to two-player games (see, e.g, von Stengel, 2007, Definition 3.2). While similar to the established notion of degeneracy, our definition of degeneracy is stronger, which makes our statements about non-degenerate games more general. (See Appendix E for details.)

6. Learning symmetric strategies in GAMUT

Theorem 5.0.4 shows that, in non-degenerate games, a locally optimal symmetric strategy profile is stable in the sense of Section 5 if and only if it is pure. For those concerned about stability, this raises the question: how often are optimal strategies pure, and how often are they mixed?

To answer this question, we present an empirical analysis of learning symmetric strategy profiles in the GAMUT suite of game generators (Nudelman et al., 2004). We are interested both in how centralized optimization algorithms (such as gradient methods) search for symmetric strategies and in how decentralized populations of agents evolve symmetric strategies. To study the former, we run Sequential Least SQuares Programming (SLSQP) (Kraft, 1988; Virtanen et al., 2020), a local search method for constrained optimization. To study the latter, we simulate the replicator dynamics (Fudenberg & Levine, 1998), an update rule from evolutionary game theory with connections to reinforcement learning (Börgers & Sarin, 1997; Tuyls et al., 2003a;b). (See Appendix F.3 for details.)

6.1. Experimental setup

We ran experiments in all three classes of symmetric GAMUT games: RandomGame, CoordinationGame, and CollaborationGame. (While other classes of GAMUT games, such as the prisoner's dilemma, exist, they cannot be turned into a symmetric, common-payoff game without losing their essential structure.) Intuitively, a RandomGame draws all payoffs uniformly at random, whereas in a CoordinationGame and a CollaborationGame, the highest payoffs are always for outcomes where all players choose the same action. (See Appendix F.1 for details.) Because CoordinationGame and CollaborationGame have such similar game structures, our experimental results in the two games are nearly identical. To avoid redundancy, we only include experimental results for CoordinationGame.

For each game class, we sweep the parameters of the game from 2 to 5 players and 2 to 5 actions, i.e., with $(|N|, |A_i|) \in \{2, 3, 4, 5\} \times \{2, 3, 4, 5\}$. We sample 100 games at each parameter setting and then attempt to calculate the global symmetric optimum using (i) 10 runs of SLSQP and (ii) 10 runs of the replicator dynamic (each with a different initialization drawn uniformly at random over the simplex), resulting in 10 + 10 = 20 solution attempts per game. Because we do not have ground truth for the globally optimal solution of the game (which is NP-hard to compute), we instead use the best of our 20 solution attempts, which we call the "best solution."

6.2. How often are symmetric optima local optima among possibly-asymmetric strategies?

Here, we try to get a sense for how often symmetric optima are stable in the sense that they are also local optima in possibly-asymmetric strategy space (see Section 5). In Appendix Table 3b, we show in what fraction of games the best solution of our 20 optimization attempts is mixed; by Theorem 5.0.4 and Proposition F.2.1 from the Appendix, this is the fraction of games whose symmetric optima are not local optima in possibly-asymmetric strategy space. In CoordinationGames, the symmetric optimum is always (by construction) for all players to choose the same action, leading to stability. By contrast, we see that 36% to 60% of RandomGames are *unstable*. We conclude that if real-world games do not have the special structure of CoordinationGames, then instability may be common.

6.3. How often do SLSQP and the replicator dynamic find an optimal solution?

As sequential least squares programming and the replicator dynamic are not guaranteed to converge to a global optimum, we test empirically how often each run converges to the best solution of our 20 optimization runs. In Appendix Table 4 / Table 6, we show what fraction of the time any single SLSQP / replicator dynamics run finds the best solution, and in Appendix Table 5 / Table 7, we show what fraction of the time at least 1 of 10 SLSQP / replicator dynamics runs finds the best solution. First, we note that the tables for SLSQP and the replicator dynamics are quite similar, differing by no more than a few percentage points in all cases. So the replicator dynamics, which are used as a model for how populations evolve strategies, can also be used as an effective optimization algorithm. Second, we see that individual runs of each algorithm are up to 93% likely to find the best solution in small RandomGames, but they are less likely (as little as 24% likely) to find the best solution in larger RandomGames and in CoordinationGames. The best of 10 runs, however, finds the best solution $\geq 87\%$ of the time; so random algorithm restarts benefit symmetric strategy optimization.

7. Conclusion

There are a variety of reasons we expect to see symmetric games in machine learning systems. The first is mass hardware production, which will proliferate identical robots such as self-driving cars, that require ad-hoc cooperation (Stone et al., 2010). The second is interaction over the internet, where websites treat all users equally. The third is anonymous protocols, such as voting, which depend on symmetry. As Figure 2 shows, symmetric *games* can still arise even when *agents* and the *environment* are asymmetric.

Similarly, there are a variety of reasons we expect to see symmetric strategies in practice. The first is software copies: we expect many artificial agents will run the same source code. The second is optimization - enforcing symmetric strategies exponentially reduces the joint-strategy space. The third is parameter sharing between different neural networks, which can be critical to success in multi-agent RL (Foerster et al., 2016) and may occur as a result of pretraining on large datasets (Dasari et al., 2020). The fourth is communication: symmetry (and symmetry breaking) is a key component of zero-shot coordination with other agents and humans (Hu et al., 2020; Treutlein et al., 2021). The fifth is that a single-player game with imperfect recall can be interpreted as a multi-agent game in symmetric strategies (Aumann et al., 1997).

When cooperative AI is deployed in the world with symmetric strategy profiles, it raises questions about the properties of such profiles. Would individual agents (or the users they serve) want to deviate from these profiles? Are they robust to small changes in the game or in the executed strategies? Could there be better asymmetric strategy profiles nearby?

Our results yield a mix of good and bad news. Theorems 4.0.2 and D.0.3 are good news for stability, showing that even local optima in symmetric strategy space are (global) Nash equilibria in a robust sense. So, with respect to *unilateral* deviations among team members, symmetric optima are relatively stable. On the other hand, this may be bad news for optimization because unilateral deviation cannot improve on a local symmetric optimum (Example 4.1.1). Furthermore, Theorem 5.0.4 is perhaps bad news, showing that a broad class of symmetric local optima are unstable when considering *joint* deviations in asymmetric strategy space (Section 5). Empirically, our results with learning algorithms in GAMUT suggest that these unstable solutions may not be uncommon in practice (Section 6.2).

Future work could build on our analysis in a couple ways. First, we focus on *mixed* strategy space. However, future work may wish to deal with *behavioral* strategy space. Second, our experimental results focus on the normal-form representation of games in GAMUT (Nudelman et al., 2004). It would be interesting to see what experimental properties symmetric optima have in sequential decision making benchmarks.

References

Aumann, R. J., Hart, S., and Perry, M. The absent-minded driver. *Games and Economic Behavior*, 20:102–116, 1997.

Börgers, T. and Sarin, R. Learning through reinforcement and replicator dynamics. *Journal of economic theory*, 77 (1):1–14, 1997.

Bostrom, N., Douglas, T., and Sandberg, A. The unilateralist's curse and the case for a principle of conformity. *Social epistemology*, 30(4):350–371, 2016.

Briggs, R. Putting a value on beauty. In *Oxford Studies in Epistemology*, volume 3, pp. 3–24. Oxford University Press, 2010.

Dafoe, A., Bachrach, Y., Hadfield, G., Horvitz, E., Larson, K., and Graepel, T. Cooperative ai: machines must learn to find common ground, 2021.

Dasari, S., Ebert, F., Tian, S., Nair, S., Bucher, B., Schmeckpeper, K., Singh, S., Levine, S., and Finn, C. Robonet: Large-scale multi-robot learning. In *Conference on Robot Learning*, pp. 885–897. PMLR, 2020.

Daskalakis, C. and Papadimitriou, C. H. Computing equilibria in anonymous games. In *FOCS*, 2007.

Foerster, J., Assael, I. A., De Freitas, N., and Whiteson, S. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in neural information processing systems*, pp. 2137–2145, 2016.

Foerster, J. N., Farquhar, G., Afouras, T., Nardelli, N., and Whiteson, S. Counterfactual multi-agent policy gradients. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

Fudenberg, D. and Levine, D. K. *The theory of learning in games*, volume 2. MIT press, 1998.

Gupta, J. K., Egorov, M., and Kochenderfer, M. Cooperative multi-agent control using deep reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems*, pp. 66–83. Springer, 2017.

Hadfield-Menell, D., Dragan, A. D., Abbeel, P., and Russell, S. J. Cooperative inverse reinforcement learning. In *Advances in Neural Information Processing 29*, 2017.

Ham, N. Notions of symmetry for finite strategic-form games. arXiv:1311.4766, 2013.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

Harsanyi, J. C. Can the maximin principle serve as a basis for morality? a critique of john rawls's theory. *American political science review*, 69(2):594–606, 1975.

Hu, H., Lerer, A., Peysakhovich, A., and Foerster, J. "other-play" for zero-shot coordination. In *International Conference on Machine Learning*, pp. 4399–4410. PMLR, 2020.

Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

Kraft, D. A software package for sequential quadratic programming. Technical Report DFVLR-FB 88-28, DLR German Aerospace Center – Institute for Flight Mechanics, Koln, Germany, 1988.

Marschak, J. Elements for a theory of teams. *Management Science*, 1(2):127–137, 1955.

Marschak, J. and Radner, R. *Economic Theory of Teams*. Yale University Press, 1972.

Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J. K., Singh, S., Rathnayake, T., Vig, S., Granger, B. E., Muller, R. P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., Curry, M. J., Terrel, A. R., Roučka, v., Saboo, A., Fernando, I., Kulal, S., Cimrman, R., and Scopatz, A. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, January 2017. ISSN 2376-5992. doi: 10.7717/peerj-cs.103. URL https://doi.org/10.7717/peerj-cs.103.

Milchtaich, I. Static stability in symmetric and population games. Technical report, Bar-Ilan University, November 2016. URL https://www.biu.ac.il/soc/ec/wp/2008-04/2008-04.pdf.

Nash, J. Non-cooperative games. *Annals of Mathematics*, pp. 286–295, 1951.

Nudelman, E., Wortman, J., Shoham, Y., and Leyton-Brown, K. Run the gamut: a comprehensive approach to evaluating game-theoretic algorithms. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, 2004. AAMAS 2004.*, pp. 880–887. IEEE, 2004.

Oliehoek, F. A., Amato, C., et al. *A Concise Introduction to Decentralized POMDPs*. Springer, 2016.

Piccione, M. and Rubinstein, A. On the interpretation of decision problems with imperfect recall. *Games and Economic Behavior*, 20(1):3–24, 1997.

Plan, A. Symmetric n-player games. Working paper, available at asafplan.com, 2017.

Ratliff, L. J., Burden, S. A., and Sastry, S. S. On the characterization of local nash equilibria in continuous games. *IEEE Transactions on Automatic Control*, 61 (8):2301–2307, Aug. 2016. ISSN 0018-9286. doi: 10.1109/TAC.2016.2583518.

Rawls, J. *A Theory of Justice*. Belknap Press, Cambridge, Massachusetts, 1971.

Reback, J., McKinney, W., jbrockmendel, den Bossche, J. V., Augspurger, T., Cloud, P., Hawkins, S., gfyoung, Sinhrks, Roeschke, M., Klein, A., Petersen, T., Tratner, J., She, C., Ayd, W., Naveh, S., patrick, Garcia, M., Schendel, J., Hayden, A., Saxton, D., Jancauskas, V., Gorelli, M., Shadrach, R., McMaster, A., Battiston, P., Seabold, S., Dong, K., chris b1, and h vetinari. pandas-dev/pandas: Pandas 1.2.4, April 2021. URL https://doi.org/10.5281/zenodo.4681666.

Reny, P. J. On the existence of pure and mixed strategy Nash equilibria in discontinuous games. *Econometrica*, 67(5):1029–1056, 1999.

Russell, S. *Human Compatible: Artificial Intelligence and the Problem of Control*. Viking Press, 2019.

Schwarz, W. Lost memories and useless coins: revisiting the absentminded driver. *Synthese*, 192:3011–3036, 2015.

Shoham, Y. and Leyton-Brown, K. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008. ISBN 0521899435. URL http://www.masfoundations.org/.

Stone, P., Kaminka, G., Kraus, S., and Rosenschein, J. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, 2010.

Taylor, J. In memoryless Cartesian environments, every UDT policy is a CDT+SIA policy, June 2016. URL https://www.alignmentforum.org/posts/5bd75cc58225bf06703751b2/in-memoryless-cartesian-environments-every-udt-policy-is-a-cdt-sia-policy.

Treutlein, J., Dennis, M., Oesterheld, C., and Foerster, J. A new formalism, method and open issues for zero-shot coordination. In *Proceedings of the Thirty-eighth International Conference on Machine Learning (ICML'21)*. 2021.

Tsybakov, A. *Introduction to Nonparametric Estimation*. Springer, 2009.

Tuyls, K., Heytens, D., Nowe, A., and Manderick, B. Extended replicator dynamics as a key to reinforcement learning in multi-agent systems. In *European Conference on Machine Learning*, pp. 421–431. Springer, 2003a.

Tuyls, K., Verbeeck, K., and Lenaerts, T. A selection-mutation model for q-learning in multi-agent systems. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pp. 693–700, 2003b.

Ui, T. Bayesian potentials and information structures: Team decision problems revisited. *International Journal of Economic Theory*, 5(3):271–291, 2009.

Vester, S. *Symmetric Nash Equilibria*. PhD thesis, Ecole Normale Superieure de Cachan, 2012.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

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

von Stengel, B. Equilibrium computation for two-player games in strategic and extensive form. In *Algorithmic Game Theory*. Cambridge University Press, 2007.

Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman (eds.), *Proceedings of the 9th Python in Science Conference*, pp. 56 – 61, 2010. doi: 10.25080/Majora-92bf1922-00a.

A. Veil of ignorance example

Two robots arrive at a resource that can be used by only one of them. They can choose as their action either Cautious or Aggressive. If both choose C, one of them gets the resource at random. If exactly one chooses A, that one gets the resource. If both choose A, the resource is destroyed and neither gets it (utility 0).

Each robot privately knows whether it has High or Low need for the resource (each type occurs independently with probability $1/2$). A robot that has High need values the resource at 6; one that has Low need values it at 4. Robots are on the same team and care about the sum of utilities.

From behind the veil of ignorance, the optimal symmetric strategy (contingency plan) is: when having type L, always play C; when having type H, play A with probability $p = 1/6$ (and C otherwise). Note, as guaranteed by Theorem 4.0.2, that this is a Nash equilibrium. To verify this, observe that from the perspective of a robot with type H, the expected team utility for playing A (when the other follows the given strategy with p) is $(1/2) \cdot 6 + (1/2)(1-p) \cdot 6 = 6 - 3p$, and for playing C it is $(1/2)((4+6)/2) + (1/2) \cdot 6 = 5.5$, and if $p = 1/6$ these are equal. In contrast, from the perspective of a robot with type L, the expected team utility for playing A (when the other follows the given strategy) is $(1/2) \cdot 4 + (1/2)(1-p) \cdot 4 = 4 - 3p$, and for playing C it is $(1/2) \cdot 4 + (1/2)(p \cdot 6 + (1-p) \cdot (4+6)/2) = 4.5 + p/2$, so C is strictly preferred.

Overall, this optimal symmetric strategy results in an expected team utility of $(1/4) \cdot 4 + (1/2) \cdot ((5/6) \cdot 4 + (1/6) \cdot 6) + (1/4) \cdot ((5/6)(5/6) \cdot 4 + 2(1/6)(5/6) \cdot 6 + (1/36) \cdot 0) = 77/18 \approx 4.28$. Compare this with an asymmetric strategy where robot 1 plays A when it has type H but otherwise C is always played by both robots, which results in a team utility of $(4+5+6+6)/4 = 21/4 = 5.25$. (If types were not private knowledge, $22/4 = 11/2 = 5.5$ would be possible.)

In this example, we see how players can coordinate using symmetric strategies from behind the veil of ignorance. Although it is possible to achieve a higher payoff using asymmetric strategies, the optimal symmetric strategy is nonetheless a Nash equilibrium by Theorem 4.0.2.

B. Proofs of Section 4 results

Theorem 4.0.2. Let $\mathcal{G}$ be a common-payoff normal-form game, and let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. Any locally optimal $\mathcal{P}$-invariant strategy profile is a Nash equilibrium.

Proof. We proceed by contradiction. Suppose $s = (s_1, s_2, \ldots, s_n)$ is locally optimal among $\mathcal{P}$-invariant strategy profiles that is not a Nash equilibrium. We will construct an s' arbitrarily close to s with $EU(s') > EU(s)$.

Without loss of generality, suppose s_1 is not a best response to s_{-1} but that the pure strategy of always playing a_1 is a best response to s_{-1}. For an arbitrary probability $p > 0$, consider the modified strategy s_1' that plays action a_1 with probability p and follows s_1 with probability $1-p$. Now, construct $s' = (s_1', s_2', \ldots, s_n')$ as follows:

$$s_i' = \begin{cases} s_i' = s_1' & \text{if } i \in \mathcal{P}(1) \\ s_i' = s_i & \text{otherwise.} \end{cases}$$

In words, s' modifies s by having the members of player 1's orbit mix in a probability p of playing a_1. We claim for all sufficiently small p that $EU(s') > EU(s)$.

To establish this claim, we break up the expected utility of s' according to cases of how many players in 1's orbit play the action a_1 because of mixing in a_1 with probability p. In particular, we observe

$$\begin{aligned} EU(s') = {} & B(m{=}0, p) EU(s) \\ & + B(m{=}1, p) EU((s_1', s_2, \ldots, s_n)) \\ & + B(m{>}1, p) EU(\ldots), \end{aligned}$$

where $B(m, p)$ is the probability of m successes for a binomial random variable on m independent events that each have success probability p and where $EU(\ldots)$ is arbitrary. Note that the crucial step in writing this expression is grouping the terms with the coefficient $B(m{=}1, p)$. We can do this because for any player $j \in \mathcal{P}(1)$, there exists a symmetry $\rho \in \Gamma(\mathcal{G})$ with $\rho(j) = 1$.

Now, to achieve $EU(s') > EU(s)$, we require

$$EU(s) < \frac{B(m=1,p)}{B(m>0,p)} EU((s'_1, s_2, \ldots, s_n)) + \frac{B(m>1,p)}{B(m>0,p)} EU(\ldots).$$

We know $EU((s'_1, s_2, ..., s_n)) > EU(s)$, but we must deal with the case when $EU(...)$ is arbitrarily negative. Because $\lim_{p\to 0} B(m>1,p)/B(m=1,p) = 0$, by making p sufficiently small, $B(m=1,p)/B(m>0,p)$ can be made greater than $B(m>1,p)/B(m>0,p)$ by an arbitrarily large ratio. The result follows. □

C. Example of general symmetry in Theorem 4.0.2

Example C.0.1. *There are four groups of partygoers positioned in a square. We number these 1,2,3,4 clockwise, such that, e.g., 1 neighbors 4 and 2. There is also a robot butler at each vertex of the square. The partygoers can fetch refreshments from the robot butler at their vertex of the square and from the robot butler at adjacent vertices of the square, but it is too far of a walk for them to fetch refreshments from the robot at the opposite vertex.*

The game has each robot butler choose what refreshment to hold. For simplicity, suppose each robot butler can hold food or drink. The common payoff of the game is the sum of the utilities of the four groups of partygoers. For each group, if the group cannot fetch drink, the payoff for that group is 0. If the group can only fetch drink, the payoff is 1, and if the group can fetch food and drink, the group's payoff is 2.

The symmetries of the game $\Gamma(\mathcal{G})$ include the set of permutations generated by rotating the robot butlers once clockwise. In standard notation for permutations, $\{(1,2,3,4),(2,3,4,1),(3,4,1,2),(4,1,2,3)\} \subset \Gamma(\mathcal{G})$.

First, consider applying the theorem to $\mathcal{P} = \Gamma(\mathcal{G})$. In this case, the constraint of $\mathcal{P}$-invariance requires all the robot butlers play the same strategy because all of them are in the same orbit. As we show in the proof below, the optimal $\mathcal{P}$-invariant strategy is then for each robot to hold food with probability $\sqrt{2}-1$. Theorem 4.0.2 tells us that this optimal $\mathcal{P}$-invariant strategy profile is a Nash equilibrium. The proof below also shows how to verify this without the use of Theorem 4.0.2.

Second, consider applying the theorem to the case where $\mathcal{P}$ consists only of the rotation twice clockwise, i.e., the permutation which maps each robot onto the robot on the opposite vertex of the square. In standard notation for permutations, $\mathcal{P} = \{(3,4,1,2)\}$. Now, the constraint of $\mathcal{P}$-invariance requires robot butlers at opposite vertices of the square to play the same strategy. However, neighboring robots can hold different refreshments. The optimal $\mathcal{P}$-invariant strategy is for one pair of opposite-vertex robots, e.g., 1 and 3, to hold food and for the other pair of robots, 2 and 4, to hold drink. While it turns out to be immediate that this optimal $\mathcal{P}$-invariant strategy is a Nash equilibrium because it achieves the globally optimal outcome, we could have applied Theorem 4.0.2 to know that this optimal $\mathcal{P}$-invariant strategy profile is a Nash equilibrium even without knowing what the optimal $\mathcal{P}$-invariant strategy was.

Proof. We here calculate the optimal $\Gamma(\mathcal{G})$-invariant strategy profile for Example C.0.1. Let p be the probability of holding drink. By symmetry of the game and linearity of expectation, the expected utility given p is simply four times the expected utility of any one group of partygoers. The utility of one group of partygoers is 0 with probability $(1-p)^3$, is 1 with probability p^3 and is 2 with the remaining probability. Hence, the expected utility of a single group of partygoers is

$$p^3 + (1-(1-p)^3 - p^3)\cdot 2 = 2 - 2(1-p)^3 - p^3.$$

The maximum of this term (and thus the maximum of the overall utility of all neighborhoods) can be found by any computer algebra system to be $p = 2-\sqrt{2}$, which gives an expected utility of $4(\sqrt{2}-1) \approx 1.66$.

To double-check, we can also calculate the symmetric Nash equilibrium of this game. It's easy to see that that Nash equilibrium must be mixed and therefore must make each robot butler indifferent about what to hold. So let p again be the probability with which each robot butler holds drink. The expected utility of holding drink relative to holding nothing for any of the three relevant neighborhoods is $2(1-p)^2$. (Holding drink lifts the utility of a group of partygoers from 0 to 2 if they can not already fetch drink. Otherwise, it doesn't help to hold drink.) The expected utility of holding food relative to broadcasting nothing is simply p^2. We can find the symmetric Nash equilibrium by setting

$$2(1-p)^2 = p^2,$$

which gives us the same solution for p as before. □

D. Robustness of Theorem 4.0.2 to payoff and strategy perturbations

The first type of robustness we consider is robustness to perturbations in the game's payoff function. Formally, we define an *ϵ-perturbation* of a game as follows:

Definition D.0.1. *Let $\mathcal{G}$ be a normal-form game with utility function μ. For some $\epsilon > 0$, we call $\mathcal{G}'$ an ϵ-perturbation of $\mathcal{G}$ if $\mathcal{G}'$ has utility function μ' satisfying* $\max_{i\in N, a\in A}|u_i'(a) - u_i(a)| \leq \epsilon$.

There are a variety of reasons why ϵ-perturbations might arise in practice. Our game model may contain errors such as the game not being perfectly symmetric; the players' preferences might fluctuate over time; or we might have used function approximation to learn the game's payoffs. With Proposition D.0.2, we note a generic observation about Nash equilibria showing that our main result, Theorem 4.0.2, is robust in the sense of degrading *linearly* in the payoff perturbation's size:

Proposition D.0.2. *Let $\mathcal{G}$ be a common-payoff normal-form game, and let s^* be a locally-optimal $\mathcal{P}$-invariant strategy profile for some subset of game symmetries $\mathcal{P} \subseteq \Gamma(\mathcal{G})$. Suppose G' is an ϵ-perturbation of $\mathcal{G}$. Then s^* is a 2ϵ-Nash equilibrium in $\mathcal{G}'$.*

Proof. By Theorem 4.0.2, s^* is a Nash equilibrium in $\mathcal{G}$. After perturbing $\mathcal{G}$ by ϵ to form $\mathcal{G}'$, payoffs have increased / decreased at most $\pm\epsilon$, so the difference between any two actions' expected payoffs has changed by at most 2ϵ. □

The second type of robustness we consider is robustness to symmetric solutions that are only approximate. For example, we might try to find a symmetric local optimum through an approximate optimization method, or the evolutionary dynamics among players' strategies might lead them to approximate local symmetric optima. Again, a generic result about Nash equilibria shows that the guarantee of Theorem 4.0.2 degrades linearly in this case:

Theorem D.0.3. Let $\mathcal{G}$ be a common-payoff normal-form game, and let s^* be a locally-optimal $\mathcal{P}$-invariant strategy profile for some subset of game symmetries $\mathcal{P} \subseteq \Gamma(\mathcal{G})$. Suppose s is a strategy profile with total variation distance $TV(s, s^*) \leq \delta$. Then s is an ϵ-Nash equilibrium with $\epsilon = 4\delta \max_{i\in N, a\in A}|u_i(a)|$.

Proof. Consider the perspective of an arbitrary player i. The difference in expected utility of playing any action a_i between the opponent strategy profiles s_{-i} and s^*_{-i} is given by:

$$\begin{aligned}
&\left|EU_i(a_i, s_{-i}) - EU_i(a_i, s^*_{-i})\right| \\
&= \Bigg|\sum_{a_{-i}\in A_{-i}} s_{-i}(a_{-i})u_i(a_i, a_{-i}) \\
&\quad - \sum_{a_{-i}\in A_{-i}} s^*_{-i}(a_{-i})u_i(a_i, a_{-i})\Bigg| \\
&\leq \sum_{a_{-i}\in A_{-i}} |u_i(a_i, a_{-i})| \left|s_{-i}(a_{-i}) - s^*_{-i}(a_{-i})\right| \\
&\leq 2TV(s, s^*) \max_{i\in N, a\in A}|u_i(a)| \\
&\leq 2\delta \max_{i\in N, a\in A}|u_i(a)|.
\end{aligned}$$

In particular, let a_i be an action in the support of s_i^*, and let a_i' be any other action. Then, using the above, we have:

$$\begin{aligned}
&EU_i(a_i', s_{-i}) - EU_i(a_i, s_{-i}) \\
&= EU_i(a_i', s_{-i}) - EU_i(a_i', s_{-i}^*) + EU_i(a_i', s_{-i}^*) \\
&\quad - EU_i(a_i, s_{-i}^*) + EU_i(a_i, s_{-i}^*) - EU_i(a_i, s_{-i}) \\
&\leq EU_i(a_i', s_{-i}) - EU_i(a_i', s_{-i}^*) \\
&\quad + EU_i(a_i, s_{-i}^*) - EU_i(a_i, s_{-i}) \\
&\leq \left|EU_i(a_i', s_{-i}) - EU_i(a_i', s_{-i}^*)\right| \\
&\quad + \left|EU_i(a_i, s_{-i}) - EU_i(a_i, s_{-i}^*)\right| \\
&\leq 4\delta \max_{i \in N, a \in A} |u_i(a)|,
\end{aligned}$$

where $EU_i(a_i', s_{-i}^*) - EU_i(a_i, s_{-i}^*) \leq 0$ because s_i^* is a Nash equilibrium by Theorem 4.0.2. □

By Theorem D.0.3, we have a robustness guarantee in terms of the total variation distance between an approximate local symmetric optimum and a true local symmetric optimum. Without much difficulty, we can also convert this into a robustness guarantee in terms of the Kullback-Leibler divergence:

Corollary D.0.4. *Let $\mathcal{G}$ be a common-payoff normal-form game, and let s^* be a locally-optimal $\mathcal{P}$-invariant strategy profile for some subset of game symmetries $\mathcal{P} \subseteq \Gamma(\mathcal{G})$. Suppose s is a strategy profile with Kullback-Leibler divergence satisfying $D_{KL}(s||s^*) \leq \nu$ or $D_{KL}(s^*||s) \leq \nu$. Then s is an ϵ-Nash equilibrium with $\epsilon = 2\sqrt{2\nu} \max_{i \in N, a \in A} |u_i(a)|$.*

Proof. By Pinsker's inequality (Tsybakov, 2009), we have

$$TV(s, s^*) \leq \sqrt{\frac{1}{2} D_{KL}(s||s^*)}.$$

As $TV(s, s^*) = TV(s^*, s)$ and with a similar application of Pinsker's inequality, we have by assumption that $TV(s, s^*) \leq \sqrt{\nu/2}$. Applying Theorem D.0.3 with $\delta = \sqrt{\nu/2}$ yields the result. □

E. Proof of Section 5 results

First, we clarify how our notion of non-degeneracy compares to the existing literature. If a two-player game $\mathcal{G}$ is non-degenerate in the usual sense from the literature, it is non-degenerate in the sense of Definition 5.0.3. Moreover, if $\mathcal{G}$ is common-payoff, then for each player i, we can define a two-player game played by i and another single player who controls the strategies of $N - \{i\}$. If for all i these two-player games are non-degenerate in the established sense, then $\mathcal{G}$ is non-degenerate in the sense of Definition 5.0.3.

Now, we proceed with the proof of Section 5 results:

Theorem 5.0.4. Let $\mathcal{G}$ be a non-degenerate common-payoff normal-form game, and let $\mathcal{P} \subseteq \Gamma(\mathcal{G})$ be a subset of the game symmetries $\Gamma(\mathcal{G})$. A locally optimal $\mathcal{P}$-invariant strategy profile is locally optimal among possibly-asymmetric strategy profiles if and only if it is deterministic.

Proof. Let s be a locally optimal $\mathcal{P}$-invariant strategy profile. By Theorem 4.0.2, s is a Nash equilibrium. Because $\mathcal{G}$ is non-degenerate, so is s. We prove the claim by proving that (1) if s is deterministic, it is locally optimal in asymmetric strategy space; and (2) if s is mixed then it is not locally optimal in asymmetric strategy space.

(1) The deterministic case: Let s be deterministic. Now consider a potentially asymmetric strategy profile s'. We must show as s' becomes sufficiently close to s that $EU(s') \leq EU(s)$.

Let $\epsilon_1, \epsilon_2, ..., \epsilon_n$ and $\hat{s}_1, ..., \hat{s}_n$ be such that for $i \in N$, s_i' can be interpreted as following s_i with probability $1 - \epsilon_i$ and following $\hat{s}_i$ with probability ϵ_i, where $s_i \notin \operatorname{supp}(\hat{s}_i)$. Then (similar to the proof of Theorem 4.0.2), we can write

$$\begin{aligned}
&EU(s') \\
&= \left(\prod_{i\in N}(1-\epsilon_i)\right) EU(s) \\
&+ \sum_{j\in N} \epsilon_j \left(\prod_{i\in N-\{j\}} 1-\epsilon_i\right) \cdot EU(\hat{s}_j, s_{-j}) \\
&+ \sum_{j,l\in N: j\neq l} \epsilon_j \epsilon_i \left(\prod_{i\in N-\{j,l\}} 1-\epsilon_i\right) \cdot EU(\hat{s}_j, \hat{s}_l, s_{-j-l}) \\
&+ \ldots
\end{aligned}$$

The second line is the expected value if everyone plays s, the third line is the sum over the possibilities of one player j deviating to $\hat{s}_j$, and so forth. We now make two observations. First, because s is a Nash equilibrium, the expected utilities $EU(\hat{s}_j, s_{-j})$ in the third line are all at most as big as $EU(s)$. Now consider any later term corresponding to the deviation of some set C, containing at least two players i, j. Note that it may be $EU(\hat{s}_C, s_{-C}) > EU(s)$. However, this term is multiplied by $\epsilon_i\epsilon_j$. Thus, as the ϵ go to 0, the significance of this term in the average vanishes in comparison to that of both the terms corresponding to the deviation of just i and just j, which are multiplied only by ϵ_i and ϵ_j, respectively. By non-degeneracy, it is $EU(\hat{s}_i, s_{-i}) < EU(s)$ or $EU(\hat{s}_j, s_{-j}) < EU(s)$. Thus, if the ϵ_i are small enough, the overall sum is less than $EU(s)$.

(2) The mixed case: Let s be mixed. We proceed by constructing a strategy profile s' that is arbitrarily close to s with $EU(s') > EU(s)$.

Let m be the largest integer where for all subsets of players $C \subseteq N$ with $|C| \leq m$, the expected payoff is constant across all joint deviations to $a_i \in \operatorname{supp}(s_i)$ for all $i \in C$, i.e., where $EU(a_C, s_{-C}) = EU(s)$ for all $a_C \in \operatorname{supp}(s_C)$. As s is a non-degenerate Nash equilibrium, $1 \leq m < n$.

By definition of m, there exists a subset of players $C \subset N$ with $|C| = m$ and choice of actions $a_C \in \operatorname{supp}(s_C)$ where $EU(a_j, a_C, s_{-j-C})$ is not constant across the available actions $a_j \in A_j$ for some player $j \in N - C$. Denote player j's best response to the joint deviation a_C as $a_j^* \in \operatorname{argmax}_{a_j} EU(a_j, a_C, s_{-j-C})$, and note $EU(a_j, a_C, s_{-j-C}) > EU(a_C, s_{-C}) = EU(s)$.

To construct s', modify s by letting player j mix according to s_j with probability $(1-\epsilon)$ and play action a_j with probability ϵ. Similarly, let each player $i \in C$ mix according to s_i with probability $(1-\epsilon)$ and play their action a_i specified by a_C with probability ϵ. Because we allow $\epsilon > 0$ to be arbitrarily small, all we have left to show is $EU(s') > EU(s)$.

Observe as before that we can break $EU(s')$ up into cases based on the number of players who deviate according to the modified probability ϵ:

$$\begin{aligned}
&EU(s') \\
&= \left(\prod_{k\in C\cup\{j\}} (1-\epsilon)\right) EU(s) \\
&+ \sum_{l\in C\cup\{j\}} \epsilon \left(\prod_{k\in C\cup\{j\}: k\neq l} 1-\epsilon\right) EU(a_l, s_{-l}) \\
&+ \ldots \\
&+ \left(\prod_{k\in C\cup\{j\}} \epsilon\right) EU(a_j, a_C, s_{-j-C}).
\end{aligned}$$

By construction, every value in the expected value calculation $EU(s')$ is equal to $EU(s)$ except for the last value $EU(a_j, a_C, s_{-j-C})$, which is greater than $EU(s)$. We conclude $EU(s') > EU(s)$. □

		Player 2	
		α	β
Player 1	α	$u_{\alpha\alpha}$	$u_{\alpha\beta}$
	β	$u_{\alpha\beta}$	$u_{\beta\beta}$

Table 2: A payoff matrix with $|N| = 2$ and $A_1 = A_2 = \{\alpha, \beta\}$ to illustrate GAMUT games. In a RandomGame, $u_{\alpha\alpha}$, $u_{\alpha\beta}$, and $u_{\beta\beta}$ are i.i.d. draws from $Unif(-100, 100)$. In a CoordinationGame, $u_{\alpha\alpha}$ and $u_{\beta\beta}$ are i.i.d. draws from $Unif(0, 100)$ while $u_{\alpha\beta}$ is a draw from $Unif(-100, 0)$. In a CollaborationGame, $u_{\alpha\alpha} = u_{\beta\beta} = 100$, and $u_{\alpha\beta}$ is a draw from $Unif(-100, 99)$.

F. GAMUT details and additional experiments

F.1. GAMUT games

In Section 6.1, we analyzed all three classes of symmetric GAMUT games: RandomGame, CoordinationGame, and CollaborationGame. Below, we give a formal definiton of these game classes:

Definition F.1.1. *A RandomGame with $|N|$ players and $|A|$ actions assumes $A_i = A_j$ for all i, j and draws a payoff from $Unif(-100, 100)$ for each unordered action profile $a \in A$.*

Definition F.1.2. *A CoordinationGame with $|N|$ players and $|A|$ actions assumes $A_i = A_j$ for all i, j. For each unordered action profile $a \in A$ with $a_i = a_j$ for all i, j, it draws a payoff from $Unif(0, 100)$; for all other unordered action profiles, it draws a payoff from $Unif(-100, 0)$.*

Definition F.1.3. *A CollaborationGame with $|N|$ players and $|A|$ actions assumes $A_i = A_j$ for all i, j. For each unordered action profile $a \in A$ with $a_i = a_j$ for all i, j, the payoff is 100; for all other unordered action profiles, it draws a payoff from $Unif(-100, 99)$.*

Note that these games define payoffs for each *unordered* action profile because the games are totally symmetric (Definition 3.2.3). Table 2 gives illustrative examples.

F.2. Proof of non-degeneracy in GAMUT

Proposition F.2.1. *Drawing a degenerate game is a measure-zero event in RandomGames, CoordinationGames, and CollaborationGames, i.e., these games are almost surely non-degenerate.*

Proof. By Definition 5.0.3, in order for a game to be degenerate, there must exist a player i, a set of actions for the other players a_{-i}, and a pair of actions $a_i \neq a_i'$ with $EU(a_i, a_{-i}) = EU(a_i', a_{-i})$. In RandomGames, CoordinationGames, and CollaborationGames, $EU(a_i, a_{-i}) = \mu(a_i, a_{-i})$ and $EU(a_i', a_{-i}) = \mu(a_i', a_{-i})$ are continuous random variables that are independent of each other. (Or, in the case of a CollaborationGame, $\mu(a_i, a_{-i})$ may be a fixed value outside of the support of $\mu(a_i', a_{-i})$.) So $EU(a_i, a_{-i}) = EU(a_i', a_{-i})$ is a measure-zero event. □

F.3. Replicator dynamics

Consider a game where all players share the same action set, i.e., with $A_i = A_j$ for all i, j, and consider a totally symmetric strategy profile $s = (s_1, s_1, \ldots, s_1)$. In the replicator dynamic, each action can be viewed as a species, and s_1 defines the distribution of each individual species (action) in the overall population (of actions). At each iteration of the replicator dynamic, the prevalence of an individual species (action) grows in proportion to its relative fitness in the overall population (of actions). In particular, the replicator dynamic evolves $s_1(a)$ over time t for each $a \in A_1$ as follows:

$$\frac{d}{dt} s_1(a) = s_1(a) \left[EU(a, s_{-1}) - EU(s) \right].$$

To simulate the replicator dynamic with Euler's method, we need to choose a stepsize and a total number of iterations. Experimentally, we found the fastest convergence with a stepsize of 1, and we found that 100 iterations sufficed for convergence; see Figure 3. For good measure, we ran 10,000 iterations of the replicator dynamic in all of our experiments.

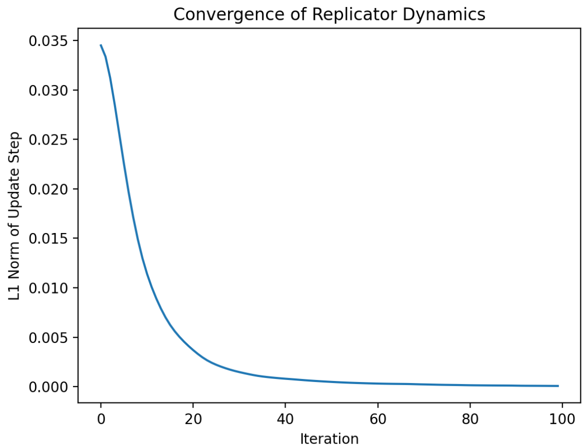


Figure 3: The magnitude of the replicator dynamics update step averaged over 10,000 RandomGames[3] with 2 players and 2 actions. Although this plot indicates that the replicator dynamics converge by 100 iterations, we ran 10,000 iterations for good measure in all of our experiments.

A N	2	3	4	5
2	0.36	0.44	0.44	0.50
3	0.38	0.49	0.59	0.60
4	0.42	0.45	0.46	0.46
5	0.45	0.48	0.49	0.47

(a) RandomGame

A N	2	3	4	5
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0

(b) CoordinationGame

Table 3: The fraction of games whose symmetric optima are mixed. By Theorem 5.0.4, these symmetric equilibria are the ones *unstable* in the sense of Section 5. Numbers in the table were empirically determined from 100 randomly sampled games per GAMUT class.

We are interested in the replicator dynamic for two reasons. First, it is a model for how agents in the real world may collectively arrive at a symmetric solution to a game (e.g., through evolutionary pressure). Second, it is a learning algorithm that performs local search in the space of symmetric strategies. In our experiments of Appendix F.5, we find that using the replicator dynamic as an optimization algorithm is competitive with Sequential Least SQuares Programming (SLSQP), a local search method from the constrained optimization literature (Kraft, 1988; Virtanen et al., 2020).

F.4. What fraction of symmetric optima are local optima in possibly-asymmetric strategy space?

As discussed in Section 6.2, we would like to get a sense for how often symmetric optima are stable in the sense that they are also local optima in possibly-asymmetric strategy space (see Section 5). Table 3 shows in what fraction of games the best solution we found is *unstable*.

F.5. How often do SLSQP and the replicator dynamic find an optimal solution?

As discussed in Section 6.3, Table 4 and Table 5 show how often SLSQP finds an optimal solution, while Table 6 and Table 7 show how often the replicator dynamic finds an optimal solution.

[3] In this simulation only we rescaled the RandomGames so that each payoff is a draw from $Unif(0, 1)$.

A N	2	3	4	5
2	0.92	0.81	0.70	0.64
3	0.80	0.69	0.57	0.48
4	0.75	0.57	0.40	0.35
5	0.70	0.45	0.36	0.31

(a) RandomGame

A N	2	3	4	5
2	0.59	0.50	0.40	0.33
3	0.53	0.38	0.28	0.29
4	0.53	0.37	0.29	0.26
5	0.53	0.36	0.33	0.25

(b) CoordinationGame

Table 4: The fraction of single SLSQP runs that achieve the best solution found in our 20 total optimization attempts. Numbers in the table were empirically determined from 100 randomly sampled games per GAMUT class.

A N	2	3	4	5
2	1.00	0.99	0.99	0.98
3	1.00	0.99	1.00	0.96
4	1.00	0.96	0.94	0.88
5	0.98	0.90	0.88	0.91

(a) RandomGame

A N	2	3	4	5
2	0.99	1.00	0.98	0.97
3	1.00	0.99	0.93	0.95
4	1.00	0.97	0.97	0.93
5	0.99	1.00	0.95	0.92

(b) CoordinationGame

Table 5: The fraction of games in which at least 1 of 10 SLSQP runs achieves the best solution found in our 20 total optimization attempts. Numbers in the table were empirically determined from 100 randomly sampled games per GAMUT class.

A N	2	3	4	5
2	0.93	0.81	0.68	0.65
3	0.81	0.70	0.58	0.46
4	0.76	0.58	0.36	0.34
5	0.69	0.43	0.36	0.30

(a) RandomGame

A N	2	3	4	5
2	0.58	0.45	0.40	0.33
3	0.57	0.35	0.29	0.27
4	0.53	0.37	0.28	0.25
5	0.51	0.33	0.33	0.24

(b) CoordinationGame

Table 6: The fraction of single replicator dynamics runs that achieve the best solution found in our 20 total optimization attempts. Numbers in the table were empirically determined from 100 randomly sampled games per GAMUT class.

A N	2	3	4	5
2	1.00	1.00	1.00	1.00
3	0.99	1.00	0.95	0.96
4	1.00	0.98	0.91	0.91
5	0.98	0.97	0.92	0.87

(a) RandomGame

A N	2	3	4	5
2	1.00	1.00	0.99	0.94
3	1.00	0.97	0.93	0.96
4	0.99	1.00	0.93	0.92
5	1.00	0.98	0.96	0.90

(b) CoordinationGame

Table 7: The fraction games in which at least 1 of 10 replicator dynamics runs achieves the best solution found in our 20 total optimization attempts. Numbers in the table were empirically determined from 100 randomly sampled games per GAMUT class.

A N	2	3	4	5
2	58.9%	55.9%	61.8%	64.6%
3	73.7%	70.9%	73.4%	73.7%
4	74.1%	77.4%	78.4%	82.5%
5	77.4%	84.9%	89.9%	87.5%

(a) RandomGame

Table 8: The average decrease in expected utility that worst-case infinitesimal asymmetric payoff perturbations cause to unstable symmetric optima. To get these numbers, we first perturb payoffs in the 100 RandomGames from Section 6.2 whose symmetric optima s are not local optima in possibly-asymmetric strategy space. Then, in each perturbed game, we compute a simultaneous best-response update to s and record its decrease in expected utility.

F.6. How costly is payoff perturbation under the simultaneous best response dynamic?

When a game is not stable in the sense of Section 5, we would like to understand how costly the worst-case ϵ-perturbation of the game can be. (See Definition D.0.1 for the definition of an ϵ-perturbation of a game.) In particular, we study the case when individuals simultaneously update their strategies in possibly-asymmetric ways by defining the following *simultaneous best response dynamic*:

Definition F.6.1. *The simultaneous best response dynamic at s updates from strategy profile $s = (s_1, s_2, \ldots, s_n)$ to strategy profile $s' = (s'_1, s'_2, \ldots, s'_n)$ with every s'_i a best response to s_{-i}.*

For each of the RandomGames in Section 6.2 whose symmetric optimum s is not a local optimum in possibly-asymmetric strategy space, we compute the worst-case ϵ payoff perturbation for infinitesimal ϵ. Then, we update each player's strategy according to the simultaneous best response dynamic at s. This necessarily leads to a decrease in the original common payoff because the players take simultaneous updates on an objective that, after payoff perturbation, is no longer common. Table 8 reports the average percentage decrease in expected utility, which ranges from 55% to 89%. Our results indicate that simultaneous best responses after payoff perturbation in RandomGames can be quite costly.

G. Code and computational resources

All of our code is available at [URL redacted for blind peer review; please find the code uploaded as supplementary material] under the MIT License. With a reduced number of random seeds, we guess that it would be possible to reproduce the experiments in this paper on a modern laptop. To test a large number of random seeds, we ran our experiments for a few days on an Amazon Web Services `c5.24xlarge` instance.

Our code uses the following Python libraries:

- Matplotlib (Hunter, 2007), released under "a nonexclusive, royalty-free, world-wide license,"
- NumPy (Harris et al., 2020), released under the BSD 3-Clause "New" or "Revised" License,
- pandas (Reback et al., 2021; Wes McKinney, 2010), released under the BSD 3-Clause "New" or "Revised" License,
- SciPy (Virtanen et al., 2020), released under the BSD 3-Clause "New" or "Revised" License, and
- SymPy (Meurer et al., 2017), released under the New BSD License.