Beyond Symmetry in Repeated Games with Restarts

Henry Fleischmann[1] , **Kiriaki Fragkia**[1] , **Ratip Emin Berker**[1,2]
[1]Carnegie Mellon University
[2]Foundations of Cooperative AI Lab (FOCAL)
{hfleisch, kiriakif, rberker}@cs.cmu.com

Abstract

Infinitely repeated games support equilibrium concepts beyond those present in one-shot games (*e.g.*, cooperation in the prisoner's dilemma). Nonetheless, repeated games fail to capture our real-world intuition for settings with many anonymous agents interacting in pairs. Repeated games with restarts, introduced by Berker and Conitzer, address this concern by giving players the option to restart the game with someone new whenever their partner deviates from an agreed-upon sequence of actions. In their work, they studied symmetric games with symmetric strategies. We significantly extend these results, introducing and analyzing more general notions of equilibria in asymmetric games with restarts. We characterize which goal strategies players can be incentivized to play in equilibrium, and we consider the computational problem of finding such sequences of actions with minimal cost for the agents. We show that this problem is NP-hard in general. However, when the goal sequence maximizes social welfare, we give a pseudo-polynomial time algorithm.

1 Introduction

Social dilemmas often arise when individuals aim to satisfy their own incentives, which often may prohibit cooperation. In fact, in many games, although cooperating could yield better payoffs for both players, it does not yield a Nash equilibrium and is hence unlikely to occur. Repeated games can circumvent this concern by capturing more complex and realistic notions of equilibria, where mutual cooperation can be incentivized. For example, consider the game in Table 1.

Notice that in the single-shot version of the game, actions C_1 and C_2 ("cooperate") are strictly dominated by action D ("defect"). This leads to (D, D) being a dominant strategy Nash equilibrium, even though each player could obtain more value by cooperating. However, when playing this game repeatedly, cooperation *ad infinitum* can be an equilibrium given that players are sufficiently patient: both players can agree to cooperate by playing C_1 until their opponent defects, at which point they start playing action D forever [Friedman, 1971]. In this case, no player is incentivized to deviate from C_1, since any additional payoff they could receive from deviating would be offset by the subsequent punishment.

	C_1	C_2	D
C_1	$8, 8$	$0, 8$	$0, 17$
C_2	$8, 0$	$2, 2$	$0, 11$
D	$17, 0$	$11, 0$	$1, 1$

Table 1: Symmetric repeated game

However, this type of collaboration fails in many real-world anonymous settings, in which players can choose to leave the game and restart with someone new. Consider an infinite collection of agents playing a repeated game in pairs either forever or until one of the players chooses to leave, in which case they are assigned a new partner. If there is no way for a player to check their new partner's history, a malicious agent could hop from partner to partner, defecting and then immediately leaving before suffering any punishment. In real-life relationships, such as ones between colleagues, freelancers with clients, or among romantic partners, agents tend to more gradually build up trust to avoid repeated exploitation. How can we formalize this game-theoretically?

One way is to consider repeated games *with restarts* [Berker and Conitzer, 2024], in which pairs of anonymous agents play an infinitely repeated game with the option to restart the game with a new player at any point. Consider then the strategy of everyone agreeing on a common sequence of actions to take, and if either player in a pair ever deviates from the sequence, the sequence is restarted. This simulates the agent punishing a defector by leaving the relationship and seeking a new partner. Ideally, such a sequence would incentivize agents to follow it at the risk of initiating a relationship with a new partner, which might come at a high cost.

Concretely, consider again the game in Table 1 and let $(D, D), (C_2, C_2), (C_1, C_1), (C_1, C_1), \ldots$ be a sequence of action pairs that both players commit to playing. Notice that the first action pair (D, D) is a dominant strategy Nash equilibrium, so no player has an incentive to deviate. In the second round, an agent can guarantee an additional payoff of $+9$ by deviating to action D. However, this results in their partner ending the relationship, at which point the deviating player will have to restart the sequence. This results in a $(1 + 11)/2 = 6$ per-round average payoff, whereas that player could have eventually guaranteed an average payoff of 8 by choosing to follow the sequence as is. Deviating to D on round three yields an additional payoff of 9, but this only amounts to a $(1+2+17)/3 \approx 6.66$ average, compared to the 8 they could have gained following the existing sequence. A similar reasoning applies for future rounds, ensuring stability.

In previous work, Berker and Conitzer [2024] formalize this subclass of Nash equilibria in repeated games with restarts and analyze its computational complexity, in the restricted setting of symmetric games and symmetric strategies. However, even in a simple example such as the one above, having players alternate between two actions (an asymmetric strategy) can yield a higher per-round average payoff. For example, say from the third round onward, players follow the strategy $(C_1, D), (D, C_1), \ldots$ (alternating between actions C_1 and D). Then, each will receive a per-round average payoff of $(0 + 17)/2 = 8.5$, compared to the 8 that the best symmetric strategy (C_1, C_1) could yield. This sequence is also stable: when each player plays action D, they receive a payoff of 17 and have no incentive to deviate. When playing C_1, they could deviate to D for a $+1$ additional payoff, but this is once again offset by the cost of restarting the sequence. This shows that, *even in symmetric games*, equilibria with asymmetric strategies improve outcomes for both players. Therefore, in our work we aim to answer the following:

How can we optimize payoff of a (possibly asymmetric) equilibrium sequence in (possibly asymmetric) repeated games with restarts?

1.1 Related Work

Infinitely repeated games *without* restarts are well studied in the literature. For a thorough treatment, see Mailath and Samuelson [2006] and Mertens *et al.* [2015]. In particular, there are numerous characterizations of equilibria, referred to as Folk Theorems (see, for example, Friedman [1971] and Fudenberg and Maskin [1986]). One interpretation of a Folk theorem is that, for each action pair $(a^{(1)}, a^{(2)})$ where players receive strictly more utility than their minmax payoff, there is a strategy and a sufficiently large discount factor such that $(a^{(1)}, a^{(2)})$ is repeated forever in equilibrium. (Recall that in repeated games it is typical to introduce a discount factor $\beta \in (0, 1)$ such that the round i utility is scaled by β^i.) Here, the minmax payoff refers to the maximum payoff a player gets if their opponent plays the action minimizing the first player's maximum payoff. The key idea is that either player can punish their opponent for deviating by playing the action minimizing the opponent's (maximum) utility.

The Folk theorem result most relevant to our work is that of Fudenberg and Maskin [1986]. In their setting, the mere threat of punishment motivates players to adhere to Nash equilibria, since leaving your partner is not allowed. In comparison, as we will see, agents in our setting must be *hazed* upfront to prevent serial defectors. This distinction arises as a result of our model capturing anonymity among players, a feature common to many interactions in the real world. Another difference between our works is that our focus is not whether equilibria with a given stable sequence exist (the direct analogue of typical Folk theorem guarantees), but, given that they do, we aim to find the "best" such equilibrium among them. We view this as finding the "least severe" punishment for deviation that still ensures an equilibrium.

The negative impacts of anonymity on establishing cooperative outcomes are well-documented and have garnered significant scientific interest. For example, see Adar and Huberman [2000] and Hughes *et al.* [2005] for discussions of the rise of free-riding agents ultimately resulting in the decline of the peer-to-peer file sharing network Gnutella. This can be viewed as an instance of the Tragedy of the Commons [Hardin, 1968]. Several research strands have consequently analyzed game-theoretic approaches to encourage cooperation in anonymous settings [Ngan *et al.*, 2010; Yang *et al.*, 2012]. We take a different perspective, focusing not necessarily on how to incentivize cooperation, but, rather, how to understand and compute the most cooperative stable outcome under the restrictions imposed by the game at hand.

Cooperation among near-anonymous agents interacting in pairs has also been studied in repeated games in the context of partner selection rules [Zhang *et al.*, 2016; Rand *et al.*, 2011; Wang *et al.*, 2012]. In prior work [Anastassacos *et al.*, 2020; Leung and Turrini, 2024; Leung *et al.*, 2024], they find the emergent dominance of "equivalent retaliation rules" akin to Tit-for-Tat. The latter two works demonstrate that learning agents both learn and (as a majority) adopt the Out-for-Tat rule, in which players leave partners who deviate against them. This provides strong empirical support for our model, which assumes this behavior.

Our starting point is a framework introduced by Berker and Conitzer [2024], who study repeated symmetric games with restarts in which all agents follow an identical sequence of moves. They prove several fundamental results on equilibrium sequences in this restricted setting.

Theorem 1. *(Informal version of Proposition 1, Lemma 1, and Lemma 2 of Berker and Conitzer [2024]) In repeated symmetric games Γ with restarts and discount factor β, where all agents follow an identical sequence of actions (*i.e.*, the strategy is symmetric), we have each of the following:*

1. *If there is some equilibrium sequence for Γ, then there exists an equilibrium sequence maximizing the agents' payoffs, which we call an* optimal sequence.
2. *Any optimal sequence will eventually reach a step in which both agents achieve a single payoff for the rest of the sequence. Call that payoff the* goal value.
3. *For large enough β, the* goal value *of any optimal sequence will be the highest payoff of an action in Γ.*

Theorem 1 has a number of implications. It is natural to seek equilibria where the agents have the best cumulative outcomes. The first property says that such equilibria in fact exist (at least, for example, in any game with a pure Nash equilibrium), making it reasonable to study such equilibria. The second and third properties characterize the general structure of optimal equilbrium sequences for sufficiently large discount factor β. They begin with a *hazing period*, in which the agents sacrifice utility to build mutual trust, followed by the agents reaping the reward of their camaraderie by receiving the goal value utility in each round thereafter.

Among such equilibrium sequences, some sequences require less hazing than others. Consider the game shown in Table 1. It is easy to check that both (D, D), (C_2, C_2), (C_1, C_1), $(C_1, C_1), \ldots$ and (D, D), (C_2, C_2), $(C_2, C_2), \ldots$, (C_2, C_2), (C_1, C_1), (C_1, C_1), $\ldots$ are equilibrium sequences for sufficiently large β. However, the latter sequence delays the socially optimal action C_1 unnecessarily.

Therefore, Berker and Conitzer [2024] define an equivalence relation among sequences, yielding a more granular view of optimality and capturing the notion of optimality of a sequence also with respect to the amount of required hazing (see Section 4 of Berker and Conitzer [2024], limit-utility equivalence classes). This in turn motivates a natural computational problem: given a symmetric game, compute an optimal symmetric strategy sequence with minimal hazing. They show that this problem is (weakly) NP-hard, while also giving a pseudo-polynomial time algorithm.

In showing these results, the authors utilize a number of properties of optimal sequences in this restricted setting. For instance, they exploit the so-called "threshold monotonicity" (see Lemma 3 of [Berker and Conitzer, 2024]) property, which intutively states we can restrict our attention to sequences that order actions in terms of how much hazing they need before they can be played. This does not hold when extending to asymmetric strategies, as playing an action pair might require different amounts of hazing for each player.

In this work, we significantly relax the structural assumptions of [Berker and Conitzer, 2024], considering repeated games in which the players need not play the same action in each round and in which the game itself may be asymmetric. In the case of symmetric games and asymmetric strategies, this raises the question of which strategy each player will be assigned to when rematching. We primarily consider the case in which players follows the same strategy every time they rematch with a new partner. In Section 6 we permit players to switch roles when rematching. Our main complexity and algorithmic results easily extend to this setting.

1.2 Motivating Examples

We begin by giving several examples to motivate our results, differentiate them from those in [Berker and Conitzer, 2024], and highlight the complex behaviors that arise in this setting.

Symmetric games with asymmetric strategies. Consider the game of two agents working on a series of group projects. Two distinct tasks must be done to complete each project, T_1 and T_2, and the agents only get utility 1 if the project is complete. This symmetric game is represented in Table 2. Note that, even though this is a symmetric game, no sequence of pairs of actions with both players always playing the same action will be stable: both players are incentivized to deviate when playing (T_2, T_2) or playing (T_1, T_1). Hence, no stable sequence exists in this game under the model considered in [Berker and Conitzer, 2024]. However, (T_1, T_2) is a pure strategy Nash equilibrium, and, hence, repeating this action pair forever is a stable sequence in our model.

	T_1	T_2
T_1	$0, 0$	$1, 1$
T_2	$1, 1$	$0, 0$

Table 2: Group Project

	C	D	H_1	H_2
C	$99, 99$	$0, 100$	$0, 0$	$0, 0$
D	$100, 0$	$0, 0$	$0, 0$	$0, 0$
H_1	$0, 0$	$0, 0$	$0, 0$	$5, 50$
H_2	$0, 0$	$0, 0$	$50, 5$	$0, 0$

Table 3: Nose Goes

Some (even symmetric) games are unfair. Can we always distribute the hazing cost or utility fairly between agents? It turns out that sometimes agents must be hazed unequally to achieve minimum total hazing.

Consider the game in Table 3. The maximum social welfare outcome consists of (C, C) repeating *ad infinitum*, and, for large enough discount factor, it is possible to haze enough to disincentivize deviation from (C, C) in only a single round. However, the minimum hazing sequence must include only one of either (H_1, H_2) or (H_2, H_1), leading to uneven total hazing. Inherently unfair games are perhaps less surprising in the asymmetric setting, but which games this holds for is not immediately obvious. We explore this question in Section 3.

Due to the complexity of characterizing the "fairness" of stable sequences, we focus on formulating and solving corresponding optimization problems, the focus of Section 5.

1.3 Organization of the Paper

In Section 3, we formalize the notions of (possibly asymmetric) equilibrium sequences in repeated games with restarts. In Section 4, we characterize the conditions under which finite sequences of action pairs can form the "goal sequences" of stable sequences. In Section 5 we consider this problem in the limit as the discount factor becomes negligible. In this regime, we define two optimization problems related to finding minimum hazing stable sequences and show that both are NP-hard. We also show that when the goal sequence is composed of maximum social welfare action pairs, there is a pseudo-polynomial time algorithm for solving the problem. Section 6 addresses an alternative model where agents can change roles after restarting the game. In Section 7, we discuss several directions for future work. All omitted proofs can be found in the appendix of the full version of the paper.

2 Preliminaries

Say Γ is a two-player normal-form game, with a set of action pairs $A = A^{(1)} \times A^{(2)}$, where $A^{(i)} = \{a_1^{(i)}, a_2^{(i)}, \ldots, a_{n^{(i)}}^{(i)}\}$ is the set of actions available to player i.

- We let $p^{(i)} : A \to \mathbb{Z}$ be the *payoff* function of player i, taking as input a pair of actions of the two players and outputting an integer value. As shorthand, we also let $p^{(1)+(2)}$ denote $p^{(1)} + p^{(2)}$.
- Let $\beta \in (0, 1)$ be the *discount factor* such that if player i receives payoff $p_t^{(i)}$ in timestep t, then her total discounted utility will be $\sum_{t=0}^{\infty} \beta^t p_t^{(i)}$.

	R	C
r	$1, 0$	$-100, -100$
c	$-100, -100$	$0, 1$

Table 4: Tightrope walking

We will denote a game as a tuple, $\Gamma = (p^{(1)}, p^{(2)}, A)$. When i refers to a player, we use $-i$ to refer to their opponent. The sequences of action pairs are 0-indexed for consistency with the powers of the discount factor β. With $\mathbb{N}$ we denote the non-negative integers.

3 Equilibrium Sequences

We focus on strategies corresponding to sequences $\sigma = (\sigma_t^{(1)}, \sigma_t^{(2)})_{t\in\mathbb{N}}$, where $\sigma \in A^{\mathbb{N}}$ is a sequence of action pairs in Γ such that player i commits to playing $(\sigma_t^{(i)})_{t\in\mathbb{N}}$. Player 1 will restart the sequence if player 2 deviates from $\sigma^{(2)}$ and vice versa. We also allow either player to restart the game after any round (even without deviating). This is a subtlety that does not arise in [Berker and Conitzer, 2024]. We illustrate the possibility of agents wanting to restart the game without deviating through the following example.

Example 1 (Restarting without Deviating)**.** *Consider the game in Table 4. Consider any sequence (tightrope) $\sigma \in \{(r, R), (c, C)\}^{\mathbb{N}}$. Neither player can ever deviate or they receive -100 utility (fall off the tightrope). Hence, if players could only restart upon deviations of their opponents, any such σ would be stable. However, if the players are permitted to restart after any round, one of the players can always ensure they receive utility* 1*. Namely, if $\sigma_0 = (r, R)$, the row player can restart the game after the first round and similarly for the column player if $\sigma_0 = (c, C)$. We view the options for players to restart after any round as akin to typical assumptions of individual rationality.*

Below we formally define the notion of a stable sequence of action pairs (*i.e.*, a Nash equilibrium), in which no player can gain more utility by deviating or restarting.

Definition 2 (Stable Sequences)**.** We call a sequence $\sigma = (\sigma_t^{(1)}, \sigma_t^{(2)})_{t\in\mathbb{N}} \in A^{\mathbb{N}}$ *stable* for discount factor β if no player can increase their discounted utility by deviating or restarting the game at any timestep. Concretely, for player 1 we have, for all $k \in \mathbb{N}$ and $a^{(1)} \in A^{(1)}$:

$$\sum_{t=0}^{k-1} \beta^t p^{(1)}(\sigma_t^{(1)}, \sigma_t^{(2)}) + \beta^k p^{(1)}(a^{(1)}, \sigma_k^{(2)}) \\ + \sum_{t=0}^{\infty} \beta^{k+1+t} p^{(1)}(\sigma_t^{(1)}, \sigma_t^{(2)}) \le \sum_{t=0}^{\infty} \beta^t p^{(1)}(\sigma_t^{(1)}, \sigma_t^{(2)}),$$

The analogous inequalities must also hold for player 2.

Remark 3. A careful reader might notice that Definition 2 only seems to consider players deviating a single time. This is because, if deviating only once cannot increase a player's utility, deviating more than once cannot either (this is an application of the "one-shot deviation principle"). An analogous observation was made in [Berker and Conitzer, 2024].

Proposition 4. *It benefits a player to deviate at least once if and only if it benefits a player to deviate once.*

The proof follows by a simple inductive argument, using that, post-deviation, the remaining game becomes a scaled version of the initial game.

Notice that there can be infinitely many stable sequences for a given game (*e.g.*, stable sequences of the form (D, D), (C_2, C_2), (C_2, C_2), $\dots$, (C_1, C_1), (C_1, C_1), $\dots$ in the game in Table 1). Therefore, we would like to be able to (1) distinguish these sequences and (2) compute the most desirable among them. To do so, we formalize the notions of *Pareto-optimality*, *welfare maximization*, and *limit-utility fairness*. In Section 4, we describe stable sequences in symmetric games with asymmetric strategies that satisfy all three properties.

Remark 5. In our study of stable sequences, we restrict our attention to sequences of the following form: a finite length prefix followed by an infinite periodic sequence of action pairs. We call the initial prefix the *hazing period* and the finite sequence repeated infinitely thereafter the *goal sequence*. A finite description length is a prerequisite for efficient computation, and general sequences need not necessarily admit one —there is an uncountably infinite number of sequences but only a countably infinite number of finite descriptions. Moreover, periodicity allows us to take limits of the sums of differences of payoffs in sequences without concern for sequence convergence issues. This permits a natural way to compare the "quality" of sequences and formally define the related optimization problems of finding "optimal" stable sequences.

Definition 6 (Pareto-Optimal Sequence)**.** Given a game $\Gamma = (p^{(1)}, p^{(2)}, A)$ and $\sigma, \tilde{\sigma} \in A^{\mathbb{N}}$, we say that σ *surpasses* $\tilde{\sigma}$ if there exists $i \in [2]$ such that:

$$\lim_{\beta\to 1} \sum_{t=0}^{\infty} \beta^t \left(p^{(i)}(\sigma_t) - p^{(i)}(\tilde{\sigma}_t)\right) > 0, \text{ while}$$
$$\lim_{\beta\to 1} \sum_{t=0}^{\infty} \beta^t \left(p^{(-i)}(\sigma_t) - p^{(-i)}(\tilde{\sigma}_t)\right) \ge 0.$$

A sequence $\sigma \in A^{\mathbb{N}}$ is *Pareto-optimal* (in $\beta \to 1$) if (1) it is stable for all sufficiently large β and (2) it is not surpassed by any other stable sequence.

A notion stronger than Pareto optimality is welfare maximization, which we define for our context below.

Definition 7 (Welfare maximization)**.** Given a game $\Gamma = (p^{(1)}, p^{(2)}, A)$, a stable sequence $\sigma \in A^{\mathbb{N}}$ is *welfare maximizing* (in the $\beta \to 1$ limit) if, for any other stable sequence $\tilde{\sigma} \in A^{\mathbb{N}}$, it holds that:

$$\lim_{\beta\to 1} \sum_{t=0}^{\infty} \beta^t (p^{(1)+(2)}(\sigma_t) - p^{(1)+(2)}(\tilde{\sigma}_t)) \ge 0.$$

We give a final desirable property of stable sequences.

Definition 8 (Limit-utility fairness)**.** Given a game $\Gamma = (p^{(1)}, p^{(2)}, A)$, a stable sequence $\sigma \in A^{\mathbb{N}}$ is *limit-utility fair* if there exists $T \in \mathbb{N}$ such that $\lim_{\beta\to 1} \sum_{t=T}^{\infty} \beta^t \left(p^{(1)}(\sigma_t) - p^{(2)}(\sigma_t)\right) = 0$.

Remark 9. Limit-utility fairness is not always possible in asymmetric games. For example, it is not possible in games where one player always receives strictly more utility than the other. Even when the players' utilities are normalized to be between 0 and 1, say by shifting their minimum utilities to each be 0 and then scaling down, there are inherently "unfair" examples, such as the one shown in Table 5.

	P	S
P	$1, 0$	$0, 1$
S	$0, 1$	$0, 1$

Table 5: Doomed to suffer

In this game, if either player plays S, the row player "suffers," receiving utility 0. Indeed, the row player only receives utility 1 if both players play P ("seek and receive pity"). But, the sequence repeating (S, S) forever is stable, and no stable sequence can ever include (P, P) since the column player will always be incentivized to deviate to (P, S).

4 Existence Results

Our starting point is the following:

> *For which pairs (Γ, γ), for Γ a game and $\gamma \in A^r$, can γ be the goal sequence of a stable sequence of Γ?*

Not all (even symmetric) games admit stable sequences, *e.g.*, Rock-Paper-Scissors. Moreover, although all 2×2 symmetric games have a stable sequence, this does not hold for asymmetric games. We discuss these nuances in the appendix of the full version of the paper.

Theorem 12 characterizes which goal sequences can arise in stable sequences. For convenience, we define the goal value of a goal sequence γ as the average per-round payoffs obtained in the goal sequence when $\beta \to 1$. We also introduce notation for the deviation payoffs for an action pair.

Definition 10 (Goal value)**.** Given a game $\Gamma = (p^{(1)}, p^{(2)}, A)$ and a goal sequence $\gamma \in A^r$, its corresponding goal value is

$$v_\gamma := (v_\gamma^{(1)}, v_\gamma^{(2)}) = \left(\tfrac{1}{r}\textstyle\sum_{j=1}^{r} p^{(1)}(\gamma_j), \tfrac{1}{r}\sum_{j=1}^{r} p^{(2)}(\gamma_j)\right).$$

Definition 11 (Deviation payoff)**.** Given a game $\Gamma = (p^{(1)}, p^{(2)}, A)$, and action pair $a = (a^{(1)}, a^{(2)}) \in A$, define

$$d_a^{(1)} := \max_{\tilde{a}^{(1)}} p^{(1)}(\tilde{a}^{(1)}, a^{(2)}), \quad d_a^{(2)} := \max_{\tilde{a}^{(2)}} p^{(2)}(a^{(1)}, \tilde{a}^{(2)}).$$

Theorem 12. *Let $\Gamma = (p^{(1)}, p^{(2)}, A)$ be a game and $\gamma \in A^r$.*

1. *Suppose there exists $a \in A$ such that $d_a^{(1)} < v_\gamma^{(1)}$ and $d_a^{(2)} < v_\gamma^{(2)}$. Then, for large enough $\beta \in (0, 1)$ and $T \in \mathbb{N}$, the sequence σ repeating a for T time steps and then repeating γ forever is stable.*
2. *If for all $a \in A$ we have $d_a^{(1)} > v_\gamma^{(1)}$ or $d_a^{(2)} > v_\gamma^{(2)}$ then, for any β, no stable sequence has γ as its goal sequence.*

The proof of the first part is inspired by the Folk theorem. The second part follows from considering stability at the first action pair in a candidate stable sequence.

Corollary 13. *Let Γ be symmetric and suppose there exists $a_* = (a_*^{(1)}, a_*^{(2)}), a \in A$ with a_* maximum social welfare and $d_a^{(1)}, d_a^{(2)} < (p^{(1)}(a_*) + p^{(2)}(a_*))/2$. Then $\gamma = ((a_*^{(1)}, a_*^{(2)}), (a_*^{(2)}, a_*^{(1)}))$ is the goal sequence of some stable sequence in Γ by Theorem 12. Moreover, this stable sequence is Pareto-optimal, limit-utility fair, and welfare-maximizing.*

5 Computing Minimum Hazing Sequences

We next consider the problem of computing stable sequences with minimum hazing in the $\beta \to 1$ limit. We begin by defining the *hazing cost* and *threshold* for a given action pair.

Definition 14 (Hazing Cost, Threshold)**.** For a game $\Gamma = (p^{(1)}, p^{(2)}, A)$, goal sequence $\gamma \in A^r$, $a \in A$, and $i \in [2]$, we define the *hazing cost* $h_a^{(i)} := v_\gamma^{(i)} - p^{(i)}(a)$ and the *threshold* $t_a^{(i)} := d_a^{(i)} - v_\gamma^{(i)}$ for player i.

The *hazing cost* of an action pair for a player defines how much utility that player loses in the long run by playing that action compared to her average utility in the goal sequence.[1] Intuitively, we want the hazing sequence to have a sufficiently high cost for both players to disincentivize them from taking an action that would result in restarting the sequence. As we will see, the *threshold* of an action pair for a given player defines the amount of total hazing that that player must have accumulated before playing that action in order to guarantee that they will not deviate. In Theorem 17, we will make use of these two definitions to give a sufficient and necessary condition for stability in the $\beta \to 1$ limit.

For conciseness, we also define notation for total hazing and the threshold for a goal sequence.

Definition 15 (Total Hazing)**.** Let $\sigma \in A^{\mathbb{N}}$ be a stable sequence with goal sequence $\gamma \in A^r$ for a game $\Gamma = (p^{(1)}, p^{(2)}, A)$. For each $k \in \mathbb{N}$, define the total hazing up to time k, $H_k := (H_k^{(1)}, H_k^{(2)}) = \sum_{t=0}^{k} h_{\sigma_t}$, to be the sum of hazing costs of the first $k + 1$ actions in the hazing period.

Definition 16 (Threshold for a goal sequence)**.** For a goal sequence $\gamma \in A^r$ and $i \in [2]$, define its *threshold* as:

$$\theta_\gamma^{(i)} = \max_{k \in [r]} \left(t_{\gamma_k}^{(i)} - \textstyle\sum_{t=1}^{k-1} h_{\gamma_t}^{(i)}\right). \tag{1}$$

In words, for each $k \in [r]$, we need to surpass the threshold for γ_k upon reaching it, which is only possible by accumulating $\theta_\gamma^{(i)}$ hazing for each player i by the time the goal sequence is reached. The summation in equation (1) accounts for the change in total hazing (after the goal sequence begins) up to the $(k - 1)^{\text{th}}$ action pair in the goal sequence.

Finally, Theorem 17 defines the stability of a sequence in the $\beta \to 1$ limit.

Theorem 17. *Let $\Gamma = (p^{(1)}, p^{(2)}, A)$ be a game. A sequence $\sigma \in A^{\mathbb{N}}$ with finite hazing period and goal sequence $\gamma \in A^r$ is stable for all sufficiently large $\beta \in (0, 1)$ if and only if, for all $k \in \mathbb{N}$ and $i \in [2]$, we have $H_{k-1}^{(i)} > t_{\sigma_k}^{(i)}$.*

The intuition for the strict inequality here is that ties break in favor of deviating, since the deviation payoff comes earlier.

We can also characterize stability in the limit $\beta \to 1$ using the language of thresholds of the goal sequence.

Corollary 18. *A sequence σ formed by a hazing period of length T and a repeated goal sequence $\gamma \in A^r$ is stable in the $\beta \to 1$ limit if and only if σ is stable in the hazing period in the $\beta \to 1$ limit and, for $i \in [2]$, $H_T^{(i)} > \theta_\gamma^{(i)}$.*

[1] Note that the hazing cost could be negative for some actions.

Using the necessary and sufficient conditions for stability from Corollary 18, we now define the computational problem of finding sequences inducing the minimum possible hazing.

Definition 19 (MinHazing)**.** Denote the hazing and threshold tuples for each action pair and each player in a game as $\left\{ \left(h_a^{(1)}, h_a^{(2)}, t_a^{(1)}, t_a^{(2)}\right) \right\}_{a \in A} \in \left((1/r \cdot \mathbb{Z})^4\right)^{|A|}$. Also let $(\theta^{(1)}, \theta^{(2)}) \in (\frac{1}{r} \cdot \mathbb{Z})^2$ be the thresholds for a goal sequence for each player. Given $\Delta > 0$, MinHazing asks to find a sequence $\sigma \in A^\ell$ (for any finite ℓ) such that the sum of the total hazings satisfies $H_{\ell-1}^{(1)+(2)} = \sum_{t=0}^{\ell-1} h_{\sigma_t}^{(1)+(2)} \leq \Delta$, subject to:

1. $H_{\ell-1}^{(i)} > \theta^{(i)}, \quad \forall i \in [2]$
2. $H_{k-1}^{(i)} > t_{\sigma_k}^{(i)}, \quad \forall k \in \{0, \cdots, \ell-1\}, \quad \forall i \in [2]$

Notice that, by Theorem 12, in some games it is easy to compute hazing sequences that induce stable sequences with goal sequence γ. In fact, if there exists some action pair $a \in A$ that satisfies the conditions of Theorem 12, repeating a sufficiently many times makes for such a hazing sequence. Moreover, the sum of the players' total hazing will be at most:

$$B := \theta_\gamma^{(1)} + \theta_\gamma^{(2)} + h_a^{(1)} + h_a^{(2)} \leq (r+2)\kappa,$$

where κ is the difference between the largest and smallest possible payoff values in Γ. However, this trivial upper bound, B, could be arbitrarily larger than the minimum hazing possible, which is what we explore in this section.

Remark 20. In the instance of MinHazing induced by Γ and a finite length goal sequence $\gamma \in A^r$, we have $(\theta_\gamma^{(1)}, \theta_\gamma^{(2)}) \in (\frac{1}{r} \cdot \mathbb{Z})^2$ and $\{(h_\sigma^{(1)}, h_\sigma^{(2)}, t_\sigma^{(1)}, t_\sigma^{(2)})\}_{\sigma \in A} \in ((\frac{1}{r} \cdot \mathbb{Z})^4)^{|A|}$. This follows directly from Definitions 14 and 16. We make heavy use of this fact in Algorithm 1.

Theorem 21. MinHazing *is (weakly)* NP*-hard.*

Proof Sketch. The key idea is to reduce from the Unbounded Subset-Sum Problem with non-negative integers. We define a symmetric game in which: all but the threshold for the goal sequence action pair are trivially met and only the main diagonal action pairs are viable in a minimum hazing stable sequence. The payoffs on the main diagonal can then be chosen such that their hazing costs correspond to the integers from the instance of Unbounded Subset-Sum.

Although MinHazing is well-defined in broad generality, we are mostly interested in the problem of computing minimum hazing sequences for welfare-maximizing stable sequences with infinitely repeated finite-length goal sequences. So, we define the following computational problem.

Definition 22 (MaxWelfareMinHazing)**.** Consider a game, $\Gamma = (p^{(1)}, p^{(2)}, A)$, and a goal sequence, $\gamma \in A^r$, where for each $t \in [r]$, $\gamma_t \in A$ is a maximum social welfare action pair. Suppose also that γ is the goal sequence of a stable sequence, σ, that achieves total sum of hazings B. Then MaxWelfareMinHazing(Γ, γ, B) is the instance of MinHazing induced by Γ, γ, and total hazing bound B.

Remark 23. Since $\gamma_t \in A$ is maximum social welfare for each $t \in [r]$, we know that each $a \in A$ has $h_a^{(1)} + h_a^{(2)} \geq 0$. Indeed, if not, then a would induce higher social welfare than the per-round average payoff in γ, a contradiction.

We will use the structure of MaxWelfareMinHazing to show that for any stable sequence with goal sequence $\gamma \in A^r$, there exists another highly structured stable sequence with goal sequence γ and minimum hazing (Lemma 24). This insight will allow us to solve MaxWelfareMinHazing in pseudo-polynomial time (Theorem 27).

Lemma 24. *For a game* Γ*, let* $\gamma \in A^r$ *be a maximum social welfare goal sequence, such that there exists a stable sequence with goal sequence* γ *and total hazing* B*. Then, there exists a* minimum hazing *stable sequence* σ *with goal sequence* γ*, such that for* k *and* $k_1 < k_2$ *in the hazing period:*

1. Total hazing bound: $0 \leq H_k^{(1)}, H_k^{(2)}, H_k^{(1)+(2)} \leq B$.

2. Monotonicity of total hazing: $H_{k_2}^{(1)+(2)} - H_{k_1}^{(1)+(2)} \geq 0$.

3. Injectivity of total hazing: $(H_{k_1}^{(1)}, H_{k_1}^{(2)}) \neq (H_{k_2}^{(1)}, H_{k_2}^{(2)})$.

Proof Sketch. The first two properties follow from stability and the fact that each $\gamma_t \in \gamma$ is maximum social welfare. The third property exploits the observation that we can remove segments of the hazing period between repeated pairs of total hazing values without compromising stability.

Although MinHazing is not obviously in NP in general (*e.g.*, optimal stable sequences can have exponentially long hazing periods), MaxWelfareMinHazing is indeed in NP under some weak additional structural assumptions.

Theorem 25. MaxWelfareMinHazing(Γ, γ, B) *(in its decision version) is in* NP *for classes of games* Γ*, goal sequence* $\gamma \in A^r$ *with* $|A| = n$ *and* $r = \mathrm{poly}(n)$*, and* B *such that either one of the following conditions is satisfied:*

1. *There is a stable sequence with goal sequence* γ *with total hazing* $\mathrm{poly}(n)$*,* e.g.*,* $B = \mathrm{poly}(n)$.
2. *There exists a stable sequence with at most* $\mathrm{poly}(n)$ *action pairs with negative hazing value for either player.*

Proof Sketch. The first sufficient condition follows from the second and third properties of Lemma 24 (or Theorem 27). The second condition follows from the fact that, as long as the threshold for some $a \in A$ is met at some time step t and it does not contribute negative hazing to either player, it can be inserted at time step t without disrupting stability. This allows for clustering all such $a \in A$ into consecutive runs, yielding hazing sequences with succinct representations.

Remark 26. The first part of Theorem 25 holds when the utilities in Γ are bounded by $\mathrm{poly}(n)$ and there exists $a \in A$ such that $v_\gamma^{(i)} > d_a^{(i)}$ for $i \in [2]$. Indeed, by Theorem 12, since the threshold for γ is $\mathrm{poly}(n)$, repeating $a \in A$ for $\mathrm{poly}(n)$ iterations, followed by cycling through γ, results in a stable sequence with goal sequence γ and $\mathrm{poly}(n)$ total hazing.

Finally, we give a dynamic programming algorithm for solving MaxWelfareMinHazing (Algorithm 1) and prove that it runs in pseudo-polynomial time (Theorem 27).

We start by giving an overview of Algorithm 1. Given as input the tuples of thresholds and hazing costs for each action pair, the threshold for the goal sequence, and the upper bound B on the total hazing, the algorithm will first construct an empty queue, Q, and a matrix, `memo`, indexed by all possible pairs of hazing values (rationals with denominator r between 0 and B from Remark 20). The algorithm will start with a total hazing value of 0 for each player (*i.e.*, starts at the upper left entry of `memo`), and, from that pair of hazing values, it will consider all possible action pairs available to the agents. For each action pair that satisfies the conditions of Lemma 24 and whose threshold is met, the algorithm will compute the new reachable total hazing for each player by adding the hazing cost of that action pair to the the total hazing of each player so far. That will "move" the algorithm to a different entry in `memo`, and the algorithm will enqueue that entry to be explored later. The algorithm then proceeds in a breadth-first-search manner, dequeuing pairs of hazing values from Q (*i.e.*, "visiting" entries in `memo`) and considering all other entries in `memo` that can be reached by taking "valid" action pairs. While visiting each entry in `memo`, the algorithm keeps track of the minimum total hazing value pair encountered so far that satisfies the goal threshold, as well as the appropriate information to reconstruct the minimum hazing sequence. The algorithm terminates when Q is empty. Recall that we use $h^{(1)+(2)}$ as shorthand denoting $h^{(1)}+h^{(2)}$ (see Section 2).

Theorem 27. MAXWELFAREMINHAZING(Γ, γ, B) *is solvable in* $O(\text{poly}(n) r^2 B^2)$ *time and* $O(r^2 B^2)$ *space, where* $|A| = n$ *and* $\gamma \in A^r$.

Proof Sketch. We use the structure of the min-hazing sequence shown in Lemma 24 to upper bound the size of `memo` that needs to be searched to find that sequence. This allows us to bound the time and space complexity of the algorithm.

Note that while Algorithm 1 minimizes $H^{(1)+(2)}$ as written, it can be modified to minimize any function of the hazing costs $H^{(1)}, H^{(2)}$ by modifying Steps 10 and 24 accordingly.

6 Random Player Reassignment

In this section, we consider a variant of our framework, where the roles of players may switch after a rematch. Indeed, in other settings, it has been shown that such role-switching can promote cooperation [Moon and Conitzer, 2016]. As before, we assume each player is required to play at least one round of the game with their partner, and can choose to leave after any round. When reassigned to a new relationship, they take on each of the two possible player roles with probability $1/2$. We introduce the analogous notion of stability below.

Definition 28 (Stable Sequences, Random Player Reassignment)**.** We call a sequence $\sigma = (\sigma_t^{(1)}, \sigma_t^{(2)})_{t \in \mathbb{N}} \in A^{\mathbb{N}}$ stable if no player can increase their expected discounted utility by deviating at any timestep. This means for all $k \in \mathbb{N}$ and $i \in [2]$ we have $\sum_{t=0}^{k-1} \beta^t p^{(i)}(\sigma_t) + \beta^k d_{\sigma_k}^{(i)} + \sum_{t=0}^{\infty} \beta^{k+1+t} \frac{p^{(i)}(\sigma_t) + p^{(-i)}(\sigma_t)}{2} \leq \sum_{t=0}^{\infty} \beta^t p^{(i)}(\sigma_t)$.

With Definition 2, defecting multiple times was profitable only if defecting once was as well. This remains true here and follows from an argument similar to Proposition 4.

Just as before, instances of MAXWELFAREMINHAZING and MINHAZING are induced by this variant of the repeated games framework (each action pair has a threshold value, hazing value, etc.). It is not hard to show that the same NP-hardness result holds (by the same construction as in Theorem 21), and Algorithm 1 still solves MAXWELFAREMINHAZING in pseudo-polynomial time.

Algorithm 1 Dynamic Programming Algorithm for MAXWELFAREMINHAZING

```
1: Input: (1.) Goal sequence threshold $\theta := (\theta^{(1)}, \theta^{(2)}) \in (\frac{1}{r} \cdot \mathbb{Z})^2$, (2.) action pair hazing costs and threshold $\{(h_a^{(1)}, h_a^{(2)}, t_a^{(1)}, t_a^{(2)})\}_{a \in A} \in ((\frac{1}{r} \cdot \mathbb{Z})^4)^{|A|}$, and (3.) initial total hazing bound $B$.
2: Output: Hazing sequence, $\sigma_* \in A^\ell$, and total hazing $H_*^{(1)} + H_*^{(2)} \in \frac{1}{r} \cdot \mathbb{Z}$.
3: Create an empty queue, $Q$
4: memo ← [] × [] {Indexed by pairs in $[0, B] \times [0, B]$}
5: $(H_*^{(1)}, H_*^{(2)}) \leftarrow (B, B)$ {Min. hazing above $\theta$ so far}
6: Enqueue(0, 0, none, none) {Hazing pairs to process}
7: while $Q$ is not empty do
8:     $(H^{(1)}, H^{(2)}, p, a_p) \leftarrow$ Dequeue($Q$) {$H^{(i)}$ is current hazing for player $i$}
9:     memo$[H^{(1)}, H^{(2)}] = (p, a_p)$ {Parent (hazing, action)}
10:    if $H^{(i)} > \theta^{(i)} \; \forall i \in [2]$ and $H^{(1)+(2)} < H_*^{(1)+(2)}$ then
11:        $(H_*^{(1)}, H_*^{(2)}) \leftarrow (H^{(1)}, H^{(2)})$ {Update optimal}
12:        continue
13:    for $a := (a^{(1)}, a^{(2)}) \in A$ do
14:        if $\exists i \in [2]$ s.t. $H^{(i)} < t_a^{(i)}$ or $(H^{(1)} + h_a^{(1)}, H^{(2)} + h_a^{(2)}) \in$ memo then
15:            continue {Thresh. unsatisfied or redundant}
16:        if $H^{(1)+(2)} + h_a^{(1)+(2)} > B$ or $\exists i \in [2]$ s.t. $H^{(i)} + h_a^{(i)} < 0$ then
17:            continue {Invalid or irrelevant hazing pair}
18:        Enqueue($(H^{(1)} + h_a^{(1)}, H^{(2)} + h_a^{(2)}, (H^{(1)}, H^{(2)}), a)$)
19: $\sigma_* \leftarrow ()$
20: $(H^{(1)}, H^{(2)}) \leftarrow (H_*^{(1)}, H_*^{(2)})$
21: while memo$[H^{(1)}, H^{(2)}][2] \neq$ none do
22:     $\sigma_*$.prepend(memo$[H^{(1)}, H^{(2)}][3]$) {Rebuild $\sigma_*$}
23:     $(H^{(1)}, H^{(2)}) \leftarrow$ memo$[H^{(1)}, H^{(2)}][2]$
24: return: $\sigma_*, H_*^{(1)+(2)}$
```

7 Directions for Future Work

There are several interesting directions for future work. One avenue is to permit mixed strategies. How would agents verify whether their opponent adhered to their assigned strategy or deviated to another strategy with the same support? Cryptographic protocols, such as those in Blum [1983], are a good candidate approach for strategy verification.

Another direction is to extend our algorithmic results beyond maximum social welfare goal sequences. The main difficulty in this general setting is that net "unhazing" is now possible, so total hazing is no longer monotonic.

A third direction is to extend our work to fixed discount factors β. While some of our results hold for large enough β (e.g., Theorem 12 and Corollary 13), our algorithmic approach does not; for fixed β, the thresholds for actions depend on the time they are played.

Our results extend to games with any number of players, although Algorithm 1's runtime scales exponentially.

Acknowledgements

We are grateful to Tuomas Sandholm and Brian Hu Zhang for their valuable feedback on an early manuscript and George Z. Li for helpful discussions in the early stages of this project. We also thank our IJCAI'25 reviewers for their generous and detailed reviews.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE2140739. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

R.E.B. thanks the Cooperative AI Foundation, Macroscopic Ventures (formerly Polaris Ventures / the Center for Emerging Risk Research) and Jaan Tallinn's donor-advised fund at Founders Pledge for financial support. R.E.B. is also supported by the Cooperative AI PhD Fellowship.

References

[Adar and Huberman, 2000] Eytan Adar and Bernardo A. Huberman. Free riding on gnutella. *First Monday*, 5(10), Oct. 2000.

[Anastassacos *et al.*, 2020] Nicolas Anastassacos, Stephen Hailes, and Mirco Musolesi. Partner selection for the emergence of cooperation in multi-agent systems using reinforcement learning. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*, 2020.

[Berker and Conitzer, 2024] Ratip Emin Berker and Vincent Conitzer. Computing optimal equilibria in repeated games with restarts. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, 2024.

[Blum, 1983] Manuel Blum. Coin flipping by telephone a protocol for solving impossible problems. *ACM SIGACT News*, 15(1):23–27, 1983.

[Friedman, 1971] James W. Friedman. A non-cooperative equilibrium for supergames. *The Review of Economic Studies*, 38(1):1–12, 1971.

[Fudenberg and Maskin, 1986] Drew Fudenberg and Eric Maskin. The folk theorem in repeated games with discounting or with incomplete information. *Econometrica*, 54(3):533–554, 1986.

[Hardin, 1968] Garrett Hardin. The tragedy of the commons: the population problem has no technical solution; it requires a fundamental extension in morality. *Science*, 162(3859):1243–1248, 1968.

[Hughes *et al.*, 2005] Daniel Hughes, Geoff Coulson, and James Walkerdine. Free riding on gnutella revisited: the bell tolls? *IEEE distributed systems online*, 6(6), 2005.

[Leung and Turrini, 2024] Chin-wing Leung and Paolo Turrini. Learning partner selection rules that sustain cooperation in social dilemmas with the option of opting out. In *Proceedings of the Twenty-Third International Conference on Autonomous Agents and Multiagent Systems (AAMAS-24)*, 2024.

[Leung *et al.*, 2024] Chin-wing Leung, Tom Lenaerts, and Paolo Turrini. To promote full cooperation in social dilemmas, agents need to unlearn loyalty. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, 2024.

[Mailath and Samuelson, 2006] George J Mailath and Larry Samuelson. *Repeated games and reputations: long-run relationships*. Oxford university press, 2006.

[Mertens *et al.*, 2015] Jean-François Mertens, Sylvain Sorin, and Shmuel Zamir. *Repeated games*, volume 55. Cambridge University Press, 2015.

[Moon and Conitzer, 2016] Catherine Moon and Vincent Conitzer. Role assignment for game-theoretic cooperation. In *Proceedings of the Fifteenth International Conference on Autonomous Agents and Multiagent Systems (AAMAS-16)*, pages 1413–1414, 2016.

[Ngan *et al.*, 2010] Tsuen-Wan Johnny Ngan, Roger Dingledine, and Dan S Wallach. Building incentives into tor. In *Proceedings of the Fourteenth International Conference on Financial Cryptography and Data Security (FC-10)*, 2010.

[Rand *et al.*, 2011] David G Rand, Samuel Arbesman, and Nicholas A Christakis. Dynamic social networks promote cooperation in experiments with humans. *Proceedings of the National Academy of Sciences*, 108(48):19193–19198, 2011.

[Wang *et al.*, 2012] Jing Wang, Siddharth Suri, and Duncan J Watts. Cooperation and assortativity with dynamic partner updating. *Proceedings of the National Academy of Sciences*, 109(36):14363–14368, 2012.

[Yang *et al.*, 2012] Mu Yang, Vladimiro Sassone, and Sardaouna Hamadou. A game-theoretic analysis of cooperation in anonymity networks. In *Proceedings of the First International Conference on Principles of Security and Trust: First International Conference (POST-12), Held as Part of the European Joint Conferences on Theory and Practice of Software (ETAPS-12)*, 2012.

[Zhang *et al.*, 2016] Bo-Yu Zhang, Song-Jia Fan, Cong Li, Xiu-Deng Zheng, Jian-Zhang Bao, Ross Cressman, and Yi Tao. Opting out against defection leads to stable coexistence with cooperation. *Scientific reports*, 6(1):35902, 2016.