Playing Repeated Coopetitive Polymatrix Games with Small Manipulation Cost

Shivakumar Mahesh[1], Nicholas Bishop[2], Le Cong Dinh[3], and Long Tran-Thanh[1]

[1] University of Warwick, UK
long.tran-thanh@warwick.ac.uk
[2] University of Oxford, UK
nicholas.bishop@cs.ox.ac.uk
[3] University of Southampton, UK
lcd1u18@soton.ac.uk

Abstract. Repeated coopetitive games capture the situation when one must efficiently balance between cooperation and competition with the other agents over time in order to win the game (e.g., to become the player with highest total utility). Achieving this balance is typically very challenging or even impossible when explicit communication is not feasible (e.g., negotiation or bargaining are not allowed). In this paper we investigate how an agent can achieve this balance to win in repeated coopetitive polymatrix games, without explicit communication. In particular, we consider a 3-player repeated game setting in which our agent is allowed to (slightly) manipulate the underlying game matrices of the other agents for which she pays a manipulation cost, while the other agents satisfy weak behavioural assumptions. We first propose a payoff matrix manipulation scheme and sequence of strategies for our agent that provably guarantees that the utility of any opponent would converge to a value we desire. We then use this scheme to design winning policies for our agent. We also prove that these winning policies can be found in polynomial running time. We then turn to demonstrate the efficiency of our framework in several concrete coopetitive polymatrix games, and prove that the manipulation costs needed to win are bounded above by small budgets. For instance, in the social distancing game, a polymatrix version of the lemonade stand coopetitive game, we showcase a policy with an infinitesimally small manipulation cost per round, along with a provable guarantee that, using this policy leads our agent to win in the long-run. Note that our findings can be trivially extended to n-player game settings as well (with $n > 3$).

1 Introduction

Repeated coopetitive games play a central role in multi-agent learning [11,23,18,3], as well as in many other areas of multi-agent systems (MAS) [14,16,24,**?**]. They capture the situation in which a number of competing agents repeatedly playing an underlying multi-player game. The goal of each agent is not just simply maximizing their total payoff, but also to have the highest one (a.k.a. to win

the game). The agents, however, cannot achieve this by just solely focusing on their own policies, but they need to coordinate with some of their competitors to play against the rest (hence the term coopetition, which is a portmanteau of the words cooperation and competition). When communication between agents is explicitly feasible, many MAS based approaches can be used to initiate and maintain these cooperation, ranging from negotiation and bargaining theory, to coalitional game theory and coordination [14,22].

However, when explicit communication is not feasible, achieving those necessary cooperative behaviours becomes a significantly more difficult situation. Recently, there has been a line of research investigating whether such cooperative behaviours can emerge by just observing and reacting to the played strategies of the opponents [6,4,16]. A key challenge here is not just to identify the appropriate opponents to cooperate, but to know when to switch sides as well. For example, in the lemonade stand game [25], three players simultaneously place their stands on one of the twelve positions uniformly distributed on the shore of a circle-shaped island. The payoff of each player is the sum of the distances between their stand and that of their opponents (a more detailed description of its polymatrix game version, called social distancing Game, can be found in section 7). The goal of each agent is then to win the game, that is, to be the one with the highest total payoff over a finite period of time. Now, in order to achieve this, the agent must pick one of the two opponents and start cooperating (e.g., by placing their stands at the opposite positions of the circle). By doing so, one can easily prove that the average payoff of the two cooperating players will be significantly higher than that of the third one. However, this cooperation alone would not provide a guaranteed win (as the cooperating partner can still get higher payoffs). Thus, a key step in this game is to know the right time to switch the team and start cooperating with the third player (by doing so, one might be able to become the one with the highest payoff in the long run).

Note that although the LSG has rather a simplistic setting, it captures the essence of many real-world applications, ranging from technological battles (e.g., the high-definition optical disc format war between Blu-ray and DVD) and R&D alliances [2], to environmental politics [5], and multiplayer video gaming [19], where strategically switching side and its timing are critical.

This paper seeks to address this problem in the following way: we relax the original setting by considering the case when one of the players is keen to sacrifice a (small) portion of their received payoff to modify the others' payoff value (e.g., the player donates some of their payoffs to the opponents, or makes some costly effort to reduce the others' payoff). We refer to this type of actions as payoff manipulation, the corresponding cost as manipulation cost, and the player who performs this as the manipulator (or as player 1 in the technical sections, we will make this clear later in the paper). We also assume that the opponents of the manipulator satisfy a series of stronger and stronger assumptions for which we present different policies for the manipulator that exploit these behavioural assumptions. The weakest of which is learning to play a strictly dominant action over time, and the strongest of which is being no-regret (for a more detailed

discussion of the behaviour of the opponents, see Section 3). Note that even the strongest assumption we make is mild and widely used in the game theory and online learning/optimization communities. To focus on the essence of the problem, we only deal with the 3-player setting in this paper. Note that our findings can be extended to the generic n-player setting. In addition, we assume that the game is a polymatrix game. Against this background, our contributions are as follows:

- First we propose a number of winning policies for the manipulator. In particular, we show that there exist a set of dominance solvable policies that can guarantee the win for the manipulator (Theorem 1) and they can be calculated in polynomial running time (Theorem 2).
- We then further improve these results by proposing another novel class of methods called batch coordination policies that can provably guarantee low manipulation cost (Theorem 3), which can also be calculated in polynomial time (Theorem 4).
- We also investigate a number of additional objectives, apart from just aiming to win the game (e.g., winning with the largest possible margin, or achieving socially good outcome, etc,).
- Finally we further refine our findings to a number of concrete polymatrix games. In particular, we show that for these games, the total manipulation cost the manipulator needs to spend is very small. For example, in the Social Distancing Game, the manipulator can already achieve guaranteed win by just using an infinitesimally small amount of manipulation (Section 7).

Note that due to page limitations, we have deferred all the proofs, example game analyses, and numerical results to the online ArXiv version of this paper (under the same title).

1.1 Related Work

From the manipulating agent's perspective, our setting can be viewed as a mechanism design (MD) problem [13,9]. In particular, we can consider the game matrix chosen by the manipulator (i.e., the designer) as the mechanism, and the actions chosen by each participant as the information they choose to report. In this domain, perhaps the most similar to our problem setting is the online MD framework [15,7], in which a central mechanism must make decisions over time as different agents arrive and depart at different time steps. However, our setting deals with agents which do not depart or arrive, but rather gain knowledge about the central mechanism as time moves on. Secondly, the goal of the designer is distinct from typical MD settings. Rather than standard solution concepts such as incentive compatibility or social welfare, we aim for the goal of guiding players into playing specific strategies. Such solution concepts are common amongst the online learning community in which the problem of playing a repeated game against another agent is explored under various conditions.

The problem of constructing zero-sum games with a pre-specified (strictly) dominant strategy is similar to designing games with unique minimax equilibrium [20,8,1,?] (for a more detailed description of this topic, we refer the reader to Appendix ??).
While the work above only focuses on the existence of unique equilibria, methods for constructing games with unique equilibria were also developed in tandem. Following the aforementioned work of [20], a parameterized construction for bimatrix games was proposed by [17], which subsumes an earlier construction proposed by [10]. It is worth noting that the closest to our setting is the work from [4], which also considers the problem of payoff matrix manipulation so that the unique Nash equilibrium of the new game is a predefined strategy profile. To the best of our knowledge, neither this work nor the other above-mentioned settings have considered manipulation cost (as we do in our paper), and therefore might not be able to find winning policies with small manipulation costs in our setting.

2 Preliminaries

To begin, we introduce some basic definitions from game theory through which our problem setting will be formally described. We define a finite normal form three-player general-sum game, Γ, as a tuple $(\mathcal{N}, \mathcal{A}, u)$. We denote the set of players by $\mathcal{N} = \{1, 2, 3\}$. Each player $i \in \mathcal{N}$ must simultaneously select an action from a finite set $\mathcal{A}_i$. For the sake of brevity, we use n, m and l to denote the cardinalities of $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$ respectively. We denote by $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \mathcal{A}_3$ the set of all possible combinations of actions that may be chosen by the players. Furthermore, each player is allowed to randomize their choice of action. In other words, player i can select any probability distribution $s \in \Delta(\mathcal{A}_i)$ over her action set. An action is then selected by randomly sampling according to this distribution. We refer to this set of probability distributions as the set of strategies available to the player. We say that a strategy is pure if it corresponds to the deterministic choice a single action, otherwise we say that a strategy is mixed. Hereafter we refer to the manipulating agent as player 1. We denote the strategy chosen by player 1 by the vector $\mathbf{x}$, where $\mathbf{x}(i)$ indicates the probability that player 1 selects action i. Similarly, we use $\mathbf{y}$ and $\mathbf{z}$ to denote the strategies chosen by players 2 and 3 respectively.
After strategies have been selected, player i receives a reward given by her utility function $u_i : \mathcal{A} \to \mathbb{R}$, which we consider to be a random variable under the probability space $(\mathcal{A}, \mathcal{F}, \mathbb{P})$ where we define the event space $\mathcal{F}$ to be the power set of $\mathcal{A}$ along with the probability measure $\mathbb{P}$ to be the real-valued function $\mathbb{P} : \mathcal{F} \to [0, 1]$ such that for any $a_1 \in \mathcal{A}_1$, $a_2 \in \mathcal{A}_2$ and $a_3 \in \mathcal{A}_3$, $\mathbb{P}(\{(a_1, a_2, a_3)\}) = \mathbf{x}(a_1) \cdot \mathbf{y}(a_2) \cdot \mathbf{z}(a_3)$.
In this paper, we restrict our focus to polymatrix games. That is we assume that the utility function for each agent is of the form $u_i = \sum_{i \neq j} u_{ij}$, where $u_{ij} : \mathcal{A}_i \times \mathcal{A}_j \to \mathbb{R}$ describes the payoff player i receives from its interaction with player j. Observe that any three-player polymatrix game can be succinctly

represented by six payoff matrices $A^{(i,j)}$, which each correspond to a function u_{ij}. Additionally, we let $\|A\|_\infty := \max_{k,l}|A(k,l)|$ denote the infinity norm of a given payoff matrix.

In what follows, we consider a direct extension of the three-player polymatrix setting, in which player 1 takes the role of a manipulator, and is allowed to alter the payoff matrices $A^{(2,1)}$ and $A^{(3,1)}$. In other words, we assume that player 1 has control over the payoffs other players receive when interacting with her. Thus, in addition to selecting a strategy, player 1 is also tasked with specifying the payoff matrices $A^{(2,1)}$ and $A^{(3,1)}$. We refer to the joint submission of a strategy and payoff matrices as player 1's complete strategy. We denote player 1's complete strategy by the tuple $(\mathbf{x}, (A^{(i,j)})_{(i,j)\in P})$, where P is the index set $\{(2,1),(3,1)\}$. We use A_0 to denote the original payoff matrices of the game before they are altered by player 1. One can interpret A_0 as a description of the dynamics of interaction between players, before the manipulator has implemented rules and restrictions. In a realistic setting, player 1 should not be able to modify the original game wherever there is interaction between player 2 and 3 alone. We clearly capture this notion in polymatrix games by specifying that player 1 cannot modify the matrices $A_0^{(2,3)}$ and $A_0^{(3,2)}$.

We assume that there is an associated cost for modifying the payoff matrices, which takes the form $\sum_{(i,j)\in P}\|A^{(i,j)} - A_0^{(i,j)}\|_\infty$. This cost has a natural interpretation when the manipulator uses monetary incentives in attempt to alter the behaviour of fellow players. More specifically, the cost corresponds to the sum of the maximum monetary payments (or fines) each player can receive, and thus represents, in the worst case, how much the manipulator may need to pay (or charge) in order to implement an altered version of the game.

With this cost in mind, observe that the expected payoff (or utility) of player 1 is given by the expected payoff it receives when participating in the altered polymatrix game, minus the cost it incurs for altering payoff matrices:

$$\mathbf{x}^T A^{(1,2)}\mathbf{y} + \mathbf{x}^T A^{(1,3)}\mathbf{z} - \sum_{(i,j)\in P}\left\|A^{(i,j)} - A_0^{(i,j)}\right\|_\infty.$$

In contrast, the expected utility of player 2 is simply given by the expected payoff it receives from participating in the altered game: $\mathbf{x}^T A^{(2,1)}\mathbf{y} + \mathbf{y}^T A^{(2,3)}\mathbf{z}$. Similarly, the expected payoff of player 3 is given by $\mathbf{x}^T A^{(3,1)}\mathbf{z} + \mathbf{y}^T A^{(3,2)}\mathbf{z}$.

Note that since all three players are employing mixed strategies, the payoff observed by each player may not be the same as the expected payoff. For example, if the players sample actions (a_1, a_2, a_3) from the distributions ($\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$), then the utility player 1 observes is

$$u_1(\mathbf{x},\mathbf{y},\mathbf{z}) = A^{(1,2)}(a_1,a_2) + A^{(1,3)}(a_1,a_3) - \sum_{(i,j)\in P}\left\|A^{(i,j)} - A_0^{(i,j)}\right\|_\infty$$

Similarly, the utility for player 2 is $u_2(\mathbf{x},\mathbf{y},\mathbf{z}) = A^{(2,1)}(a_1,a_2) + A^{(2,3)}(a_2,a_3)$ and the observed utility for player 3 follows in a analogous manner. When the strategies used are clear from context we will drop them from notation and use

u_1, u_2, u_3. We say player 1 has won the game if her utility is higher than the utilities of other players.

3 Problem Setting

In many cases, a manipulator will engage repeatedly with the same system participants. Additionally, aside from the manipulator, players are often unaware of their own, and others, utility functions and must learn them over time. With these concerns in mind, we consider a repeated version of the setting described above. More specifically, we consider a setting in which players engage in the aforementioned polymatrix game repeatedly for T time steps. At each time step t, each player is required to commit to a strategy, $\mathbf{x}_t$, $\mathbf{y}_t$ and $\mathbf{z}_t$. In addition, player 1, in her role as manipulator, must select the set of payoff matrices $A_t^{(i,j)}$, for $(i,j) \in P$, at each time step. We assume that players 2 and 3 have no initial knowledge of A_0, but receive feedback, at the end of every time step detailing the payoff they received. More precisely, player 2 receives feedback $u_{2,t} = u_2(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)$, at the end of time step t. Player 3 receives feedback in a similar fashion. Therefore, when selecting their strategy in round $t+1$, players 2 and 3 have access to a history of feedback (and a history of their own strategy choices) up to time step t to inform their decision. In contrast, whilst also receiving feedback at the end of each time step, we assume that player 1 has full knowledge of A_0 prior to the start of play.

We use $H_t = (u_{1,t'}, \mathbf{x}_{t'})_{t'=1}^{t}$ to denote the history observed by player 1 up to time step t. We use the notation $\mathcal{H}_t$ to denote the set of all observable histories of length t. Given a time horizon T, we define a policy $\rho = (\rho_t)_{t=1}^{T}$ as a sequence of, potentially randomized, mappings $\rho_t : \mathcal{H}_t \to \Delta(\mathcal{A}_1) \times \mathbb{R}^{n \times m} \times \mathbb{R}^{n \times l}$ from feedback histories to complete strategies. In other words, a policy ρ is a specification of which complete strategy to choose given the feedback observed so far.

Generalizing from the single-shot setting, we define the utility of each player in the repeated setting as the time average of their respective utilities in each round. That is: $U_i(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{T} := \frac{1}{T} \sum_{t=1}^{T} u_i(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)$. As before, when the sequence of strategies used by each player is clear from context, we write U_1, U_2 and U_3 for the sake of brevity. We say that player 1 has won the game, if her utility is the highest. We assume that player 1 participates in the game with the aim of winning. On the other hand, we assume that players 2 and 3 are 'consistent' agents.

Definition 1. *(Consistent Agent) Suppose that for an agent there exists an action a^* that is the unique best response for her for every round of the game. Suppose that within T rounds of the game, the number of rounds the agent plays action a^* is T^*. If $\mathbb{P}\Big(\lim_{T\to\infty} \frac{T^*}{T} = 1\Big) = 1$ then the agent is 'consistent'.*

In other words, if an agent has a single action that performs the best in all rounds, and the proportion of time she plays that action converges to 1 almost surely, then we say she is consistent. If a consistent agent does not have an action that always performs the best, we make no assumption on the behaviour of the player.

Unless stated otherwise, we restrict our focus to players who are consistent, no matter the strategies submitted by the other players. This assumption is much weaker that the standard assumption of rationality in full information mechanism design settings. In the sections that follow, we will develop a number of policies which guarantee player 1 a winning outcome with high probability under this assumption.

4 Winning Policies

In this section, we present a number of policies which guarantee player 1 a winning outcome with high probability. Before describing these policies in detail, we first present a brief conceptual argument showcasing the underlying idea behind all the policies we present.

Consider the policy where player 1 plays the same action i^* in every round. Assume that player 2 and player 3 each have strictly dominant actions j^* and k^* respectively against, the action i^* of player 1. That is, $u_2(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_k) > u_2(\mathbf{e}_{i^*}, \mathbf{e}_j, \mathbf{e}_k)$ for all $j \neq j^*$ and k and $u_3(\mathbf{e}_{i^*}, \mathbf{e}_j, \mathbf{e}_{k^*}) > u_3(\mathbf{e}_{i^*}, \mathbf{e}_j, \mathbf{e}_k)$ for all j and $k \neq k^*$. Since both players are consistent, the proportion of time each of them plays their strictly dominant action converges to 1 almost surely. Therefore, if $u_1(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}) \geq \max\{u_2(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}), u_3(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*})\}$, then intuitively, player 1 will eventually win if T is large enough. Unfortunately such an action i^*, which satisfies the above assumptions, may not exist in the original game. However, player 1 can always guarantee the existence of such an action by altering payoff matrices. If player 1 can find a low cost alteration, then she can win the game with high probability.

We then present another policy of a similar flavor where player 1 plays the same action i^* in every round. We assume that player 2 has a strictly dominant action j^* against the action i^* of player 1, but player 3 only has a strictly dominant action k^* against the action i^* of player 1 and the action j^* of player 2. Therefore if player 3, is an agent who is willing to wait for some action to eventually become her unique best response, then the manipulator can modify the payoff matrices appropriately to ensure that she wins if T is large enough. Therefore in order for such a policy to work successfully, we must make a slightly stronger behavioural assumption on player 3, which leads us to the definition of a 'persistent' agent.

All of the policies we present here combine games constructed to satisfy assumptions similar to those above, with a simple time-dependent deterministic policy. First in Section 4.1 we show how to construct payoff matrices such that actions j^* and k^* are strictly dominant for players 2 and 3, under the assumption that player 1 uses action i^*. In Section 4.2, we present the class of dominance solvable policies, which consist of stationary policies leveraging the methodologies developed in the previous section. Lastly, we present the class of batch coordination policies, which spend half the time horizon cooperating with one player, and half of the time horizon cooperating with the other.

4.1 Designing Dominance Solvable Games

Here, we describe several constructions of three player games which will be used extensively in our definitions for different kinds of policies. In particular, we show how to find a matrix $A^{(2,1)}$ (or $A^{(3,1)}$) such that a particular action for player 2 (or 3) is strictly dominant against all actions of player 3 (or 2) and a particular action of the manipulator. We also show how to find a matrix $A^{(3,1)}$ (or $A^{(2,1)}$) such that a particular action for player 3 (or 2) is strictly dominant against a particular action of player 2 (or 3) and a particular action of the player 1. For the sake of brevity we refer to players 1, 2 and 3 by P1, P2 and P3 respectively. Let $\mathbf{x}$ be the fixed strategy of the manipulator. To ensure that P2 has a strictly dominant strategy $\mathbf{e}_{j^*}$ against $\mathbf{x}$ and all actions of P3, For some $v_2 \in \mathbb{R}^l$ we must choose a matrix $A^{(2,1)}$ that satisfies the system

$$\begin{aligned} [\mathbf{x}^T A^{(2,1)} \mathbf{e}_j + \mathbf{e}_j^T A^{(2,3)} \mathbf{e}_k] &= v_{2,k} \quad \forall k \in [l] \text{ and } j = j^* \\ [\mathbf{x}^T A^{(2,1)} \mathbf{e}_j + \mathbf{e}_j^T A^{(2,3)} \mathbf{e}_k] &< v_{2,k} \quad \forall k \in [l] \text{ and } j \neq j^* \end{aligned} \tag{1}$$

By symmetry, to ensure that P3 has a strictly dominant strategy $\mathbf{e}_{k^*}$ against $\mathbf{x}$ and all actions of P2, for some $v_3 \in \mathbb{R}^m$ we must choose a matrix $A^{(3,1)}$ that satisfies the system

$$\begin{aligned} [\mathbf{x}^T A^{(3,1)} \mathbf{e}_k + \mathbf{e}_j^T A^{(3,2)} \mathbf{e}_k] &= v_{3,j} \quad \forall j \in [m] \text{ and } k = k^* \\ [\mathbf{x}^T A^{(3,1)} \mathbf{e}_k + \mathbf{e}_j^T A^{(3,2)} \mathbf{e}_k] &< v_{3,j} \quad \forall j \in [m] \text{ and } k \neq k^* \end{aligned} \tag{2}$$

Now further suppose that P2 plays the fixed strategy $\mathbf{y}$. In order to make $\mathbf{e}_{k^*}$ the dominant strategy against the strategies $\mathbf{x}$ and $\mathbf{y}$ of P1 and P2 respectively, for some $v_0 \in \mathbb{R}$ we must choose a matrix $A^{(2,1)}$ that satisfies the system

$$\begin{aligned} [\mathbf{x}^T A^{(3,1)} \mathbf{e}_k + \mathbf{y}^T A^{(3,2)} \mathbf{e}_k] &= v_0 \quad k = k^* \\ [\mathbf{x}^T A^{(3,1)} \mathbf{e}_k + \mathbf{y}^T A^{(3,2)} \mathbf{e}_k] &< v_0 \quad k \neq k^* \end{aligned} \tag{3}$$

With the following lemma, we show that strategy profiles satisfying systems (1) and (2) always exist.

Proposition 1. *Fix $i^* \in [n]$, $j^* \in [m]$ and $k^* \in [l]$ with $\mathbf{x} = \boldsymbol{e}_{i^*}$. Matrices $A^{(2,1)}$ and $A^{(3,1)}$ that satisfy the systems (1) and (2) exist.*

Proof. Set the entries of $A^{(2,1)}$ to

$$\begin{aligned} A^{(2,1)}(i^*, j) &:= 2\|A_0^{(2,3)}\|_\infty + 1 \quad \text{for } j = j^* \\ A^{(2,1)}(i^*, j) &:= 0 \quad \text{for } j \neq j^* \end{aligned}$$

and the entries of $A^{(3,1)}$ to

$$\begin{aligned} A^{(3,1)}(i^*, k) &:= 2\|A_0^{(3,2)}\|_\infty + 1 \quad \text{for } k = k^* \\ A^{(3,1)}(i^*, k) &:= 0 \quad \text{for } k \neq k^* \end{aligned}$$

now both of these matrices together satisfy systems (1) and (2).

In addition, this result clearly extends to system (3), as any matrix satisfying system (2) satisfies system (3).

Corollary 11. *Fix $i^* \in [n]$, $j^* \in [m]$ and $k^* \in [l]$ with $\mathbf{x} = \boldsymbol{e}_{i^*}$ and $\mathbf{y} = \boldsymbol{e}_{j^*}$. Matrices $A^{(2,1)}$ and $A^{(3,1)}$ that satisfy the systems systems (1) and (3) exist.*

Proof. The same as the proof of Proposition 1. If system (2) is satisfied, so is system (3).

In what follows, we will develop policies based on payoff matrices which satisfy systems (1), (2) and (3).

4.2 Dominance Solvable Policies

In this section, we introduce the class of dominance solvable policies. In short, dominance solvable policies consist of player 1 playing a constant complete strategy which satisfies a number of the linear systems. We first introduce type-I dominance solvable policies.

Definition 2. *(Dominance Solvable Type-*I *Policy)*
*Let $\left(A^{(2,1)}, A^{(3,1)}\right)$ satisfy systems (1) and (2) for some $i^* \in [n]$, $j^* \in [m]$, $k^* \in [l]$. Then, the policy $\rho_t(H_t) = \left(e_{i^*}, A^{(2,1)}, A^{(3,1)}\right)$ for $t \in \mathbb{N}$ is a dominance solvable type-*I *policy.*

In words, a dominance solvable type-I policy is one in which player 1 plays a constant complete strategy which satisfies systems (1) and (2). Similarly, we define dominance solvable type-II policies as those in which player 1 plays a constant complete strategy which satisfies systems (1) and (3).

Definition 3. *(Dominance Solvable Type-*II *Policy)*
*Let $\left(A^{(2,1)}, A^{(3,1)}\right)$ satisfy systems (1) and (3) for some $i^* \in [n]$, $j^* \in [m]$, $k^* \in [l]$. Then, the policy $\rho_t(H_t) = \left(e_{i^*}, A^{(2,1)}, A^{(3,1)}\right)$ for $t \in \mathbb{N}$ is a dominance solvable type-*II *policy.*

If one uses iterated elimination of strictly dominated strategies and there is only one strategy left for each player, the game is called dominance solvable [12]. We name the policies described above dominance solvable since the underlying single-shot game that results from these policies is almost dominance solvable. In the game that results from these policies, if we eliminate all the actions of player 1 except i^* and then implement iterated elimination of strictly dominated strategies, there will be only one strategy left for each player.
We say that a dominance solvable policy is winning if player 1 wins the corresponding single-shot game when $(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*})$ is played:

$$u_1(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}) \geq u_2(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}) \quad \text{and} \quad u_1(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}) \geq u_3(\mathbf{e}_{i^*}, \mathbf{e}_{j^*}, \mathbf{e}_{k^*}) \tag{4}$$

Winning dominance solvable type-I policies are highly attractive as they allow the manipulator to win in the long-run against consistent agents. This claim is formalized in the following theorem.

Theorem 1. *If the manipulator uses a winning dominance solvable type-*I *policy against consistent agents in an infinitely repeated game then,*

$$\mathbb{P}\Big(U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \geq U_2(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \text{ and } U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \geq U_3(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty}\Big) = 1$$

At times, for the sake of brevity, we use U_1^{∞}, U_2^{∞} and U_3^{∞} to denote the long-run utilities $U_1(\mathbf{x}_t,\mathbf{y}_t,\mathbf{z}_t)_{t=1}^{\infty}$, $U_2(\mathbf{x}_t,\mathbf{y}_t,\mathbf{z}_t)_{t=1}^{\infty}$ and $U_3(\mathbf{x}_t,\mathbf{y}_t,\mathbf{z}_t)_{t=1}^{\infty}$ respectively.
For the analogous guarantee on type-II policies we assume a slightly stronger behavioural assumption than being 'consistent' on one of the agents. We assume that player 3 is 'persistent', i.e. if there is some finite-time cutoff point after which there exists an action that always remains the unique best-response in hindsight then she will play that action a large fraction of time.

Definition 4. *(Persistent Agent) Suppose that the action k^* is the best action in hindsight for player 3 eventually, with probability 1. That is,*

$$\mathbb{P}\Big(\boldsymbol{e}_{k^*} = \arg\max_{\boldsymbol{z}\in\Delta_l} U_3(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z})_{t=1}^{T} \text{ eventually}\Big) = 1$$

Let T^* denote the number of rounds within T rounds, that player 3 plays action k^*. If $\mathbb{P}\Big(\lim_{T\to\infty}\frac{T^*}{T} = 1\Big) = 1$ then player 3 is 'persistent'.

Note that every persistent agent is consistent. We prove this in Proposition 2. The guarantee of winning when using type-II policies is exactly the same as type-I policies except that we assume one of the players is persistent.

Theorem 2. *If the manipulator uses a winning dominance solvable type-*II *policy against a consistent player 2 and a persistent player 3 in an infinitely repeated game then,*

$$\mathbb{P}\Big(U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \geq U_2(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \text{ and } U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} \geq U_3(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty}\Big) = 1$$

Observe that any constant complete strategy is dominance solvable as long as it satisfies the linear systems (1) and (2) (or (3)) for a given triple of actions (i^*, j^*, k^*). Furthermore, a dominance solvable policy is winning if and only if it satisfies the pair of linear inequalities in system (4). As a result, winning dominance solvable policies, if they exist, can be found in polynomial time by solving a sequence of linear feasibility problems, where each linear feasibility problem corresponds to a different triple of actions.

Theorem 3. *If winning dominance solvable policies exist, then there exists an algorithm that can find such policies with running time that is polynomial in the number of actions of the players.*

If player 1 uses a type-I policy, she can make a very weak behavioural assumption on the other players to guarantee winning in the long run. On the other hand, type-II policies are guaranteed to be at least as cost-effective as type-I policies, as all type-I policies are also type-II policies.

4.3 Batch Coordination Policies

Note that, even if a winning dominance solvable policy exists, it may be very costly to alter the payoff matrix. However, the manipulator may be able to beat one player through very cheap alterations whilst losing to the other, and vice versa. In this case it makes sense for player 1 to divide the time horizon, spending half the horizon winning over one player, and spending the other half winning over the other, using cheap alterations to the original payoff matrices in the process. This is the central idea behind batch coordination policies. The following definition makes this idea rigorous.

Definition 5. *(Winning Batch Coordination policy) Suppose the matrices $\hat{A}^{(2,1)}$ and $\hat{A}^{(3,1)}$ satisfy systems (1) and (2) for some $i_2 \in [n]$, $j_2 \in [m]$, $k_2 \in [l]$ and that the matrices $\tilde{A}^{(2,1)}$ and $\tilde{A}^{(3,1)}$ satisfy systems (1) and (2) for some $i_3 \in [n]$, $j_3 \in [m]$, $k_3 \in [l]$ such that for $i \neq 1$*

$$\mathbb{E}[u_1(\boldsymbol{e}_{i_2}, \boldsymbol{e}_{j_2}, \boldsymbol{e}_{k_2})] + \mathbb{E}[u_1(\boldsymbol{e}_{i_3}, \boldsymbol{e}_{j_3}, \boldsymbol{e}_{k_3})] > \mathbb{E}[u_i(\boldsymbol{e}_{i_3}, \boldsymbol{e}_{j_3}, \boldsymbol{e}_{k_3})] + \mathbb{E}[u_i(\boldsymbol{e}_{i_3}, \boldsymbol{e}_{j_3}, \boldsymbol{e}_{k_3})]$$

then the policy

$$\rho_t = \begin{cases} (\boldsymbol{e}_{i_1}, \hat{A}^{(2,1)}, \hat{A}^{(3,1)}) & \text{if } 1 \leq t \leq T/2 \\ (\boldsymbol{e}_{i_2}, \tilde{A}^{(2,1)}, \tilde{A}^{(3,1)}) & \text{if } T/2 < t \leq T \end{cases}$$

is called a winning batch coordination policy.

Winning batch coordination policies can be interpreted as following different dominance solvable policies for each half of the game. Therefore, winning batch coordination policies are more general than winning dominance solvable policies. Note that the dominance solvable polices played in each half of the time horizon may not be winning by themselves. However, when combined, these sub-policies must form a winning policy for the overall batch coordination policy to be winning.

Before we present the guarantee for player 1 when using winning batch coordination policies, we make a slightly stronger behavioural assumption on both players than the assumption of being 'persistent'. We now assume that both players aim to maximize their expected utility. We use the well-established notion of regret as a metric for measuring the performance of players 2 and 3 with respect to the payoffs they accumulate over time.

Definition 6. *The regret of any sequence of strategies $(\boldsymbol{y}_1, ..., \boldsymbol{y}_T)$ chosen by player 2 with respect to a fixed strategy $\boldsymbol{y}$ is given by*

$$\mathcal{R}_{T,\boldsymbol{y}} = \sum_{t=1}^{T} \boldsymbol{x}_t^T A_t^{(2,1)} \boldsymbol{y}_t + \boldsymbol{y}_t^T A_0^{(2,3)} \boldsymbol{z}_t - \sum_{t=1}^{T} \boldsymbol{x}_t^T A_t^{(2,1)} \boldsymbol{y} + \boldsymbol{y}^T A_0^{(2,3)} \boldsymbol{z}_t$$

That is, the regret is the difference between the payoff accumulated by the sequence $(\mathbf{y}_1, ..., \mathbf{y}_T)$ and the payoff accumulated by the sequence where a given

fixed mixed strategy $\mathbf{y}$ is chosen at each time step. A similar notion of regret is defined for the player 3. We say that a player is 'no-regret' if her regret with respect to the sequence of strategies chosen by the other two players is sublinear in T: $\lim_{T\to\infty} \max_{\mathbf{y}\in\Delta_m} \frac{\mathcal{R}_{T,\mathbf{y}}}{T} = 0$.
Note that every no-regret player is persistent. We prove this in Proposition 2. If the manipulator uses a winning batch coordination policy against no-regret players, then the probability that there exists some finite number of rounds in which she wins is 1. This result is formalized in the following theorem.

Theorem 4. *If the manipulator uses a winning batch coordination policy against no-regret players then*

$$\mathbb{P}\Big(U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^T \geq U_2(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^T \text{ and } U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^T \geq U_3(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^T \text{ eventually}\Big) = 1$$

It is worth noting that this guarantee on the convergence of utilities is stronger than the one given for winning dominance solvable policies in Theorem 1. This is because of the strict inequality on the utilities of the players in Definition 5. By presenting this guarantee instead of a guarantee on the infinite horizon utilities, we are ensured that the guarantee of winning in a finite number of rounds with probability 1 implies that the manipulator can use one set of game matrices for half the rounds and another set for the second half.
Similarly to winning dominance solvable policies, winning batch coordination policies can be found in polynomial time by solving a number of linear feasibility problems.

Theorem 5. *If winning batch coordination policies exist, then there exists an algorithm that can find such policies with running time that is polynomial in the number of actions of the players.*

We present the following proposition that states all persistent players are consistent, and that all no-regret players are persistent.

Proposition 2. *All persistent players are consistent. Further, all no-regret players are persistent.*

That is, each assumption on the behaviour of the agents is successively stronger. To emphasize this, we prove that there exists a type of player who is persistent but not no-regret.

Proposition 3. *If an agent uses the Follow the Leader algorithm, then she is persistent but not no-regret.*

For the remainder of the paper we use the weakest assumption, that players are consistent but not necessarily persistent.

5 Additional Objectives

The manipulator may have additional goals and objectives aside from simply winning the game. For example, the manipulator may want to win by a large

margin, or win by making the smallest alterations to the payoff matrices possible, or even have a goal completely different to winning, such as maximizing the egalitarian social welfare. For each of the policy classes from Section 4, the manipulator can solve a sequence of linear feasibility problems in order to find a winning policy if one exists. As long as the linear constraints of one of these problems are satisfied, the manipulator is guaranteed to win (i.e. she has found a winning policy). Therefore, the manipulator can specify any additional objectives she may have as a linear function to optimize with respect to the linear constraints imposed by the policy class. In other words, the manipulator may choose a linear objective function which captures her additional goals, and solve a sequence of linear programs (LPs), instead of a sequence of linear feasibility problems. For example let d_2 and d_3 be the cost of altering matrices $A_0^{(2,1)}$ and $A_0^{(3,1)}$ respectively. If we consider a minimization problem with objective $d_2 + d_3$, then this amounts to finding a winning policy which makes the least cost modification possible. Similarly, let v_2 be the payoff for player 2 and v_3 be the payoff for player 3 in the strategy profile of consideration. Setting v_2 as a maximization objective amounts to winning whilst ensuring player 2 does as well as possible. We could also act adversarially against player 2, by instead minimizing v_2. Meanwhile, setting $v_2 + v_3$ as a maximization objective corresponds to winning whilst maximizing the utilitarian welfare of the other players.
In what follows, we investigate additional objectives and goals of wider interest. In Section 5.1 we investigate how player 1 can maximize her margin of victory, in Section 5.2 we investigate how the manipulator may win in the most cost efficient way possible. Meanwhile, in Section 5.3 we investigate how the manipulator may maximizing the egalitarian social welfare.

5.1 Winning by the Largest Margin

In strictly competitive settings, it is often desirable for players to win, whilst ensuring that their long run utility is much higher than the other players. This motivates the following definition:

Definition 7. *The margin of a policy* $\rho_t(H_t) = (\mathbf{x}_t, (A_t^{(i,j)})_{(i,j)\in P})$ *for* $t \in \mathbb{N}$ *when playing against player 2 and player 3's no-regret sequence of strategies* $(\boldsymbol{y}_t)_{t=1}^{\infty}$ *and* $(\boldsymbol{z}_t)_{t=1}^{\infty}$ *is defined to be*

$$\min\left\{\mathbb{E}\left[U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} - U_2(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty}\right], \mathbb{E}\left[U_1(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty} - U_3(\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{z}_t)_{t=1}^{\infty}\right]\right\}$$

That is, the margin is the minimum difference between the long run expected utility of player 1 and another player. Any winning dominance solvable policy will have a margin of at least zero. Additionally, for any of the policy classes discussed above, if a winning policy exists, then a winning policy with the largest margin can be found efficiently via the addition of a linear objective and a small number of linear constraints and variables.

Theorem 6. *If winning dominance solvable policies exist, then there exists an algorithm that can find the largest margin dominance solvable policy, with running time that is polynomial in the number of actions of the players.*

5.2 Winning with the Lowest Inefficiency Ratio

In many scenarios, it is only sensible to make changes to payoff matrices if one would see a large relative improvement compared to the cost of alteration. We characterize the notion of relative improvement using the following definition.

Definition 8. *The Inefficiency Ratio of a policy* $\rho_t(H_t) = (\mathbf{x}_t, (A_t^{(i,j)})_{(i,j)\in P})$ *for* $t \in \mathbb{N}$ *when playing against player 2 and player 3's no-regret sequence of strategies* $(\boldsymbol{y}_t)_{t=1}^{\infty}$ *and* $(\boldsymbol{z}_t)_{t=1}^{\infty}$ *is defined to be*

$$\frac{\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{(i,j)\in P} \|A_t^{(i,j)} - A_0^{(i,j)}\|_\infty}{\mathbb{E}\Big[\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \Big(\mathbf{x}_t^T A_t^{(1,2)} \mathbf{y}_t + \mathbf{x}_t^T A_t^{(1,3)} \mathbf{z}_t\Big)\Big] - K}$$

where $K = \min_{i,j,k} \left(A^{(1,2)}(i,j) + A^{(1,3)}(j,k)\right)$ *is the minimum revenue for player 1.*

In other words, the inefficiency ratio is the ratio between the cost for modifying the payoff matrices and the expected increase in long run payoffs from the worst case payoff. Note that this fraction must converge for the definition to be meaningful. In a similar fashion to maximizing the margin of victory, policies which minimize the inefficiency ratio can be found in polynomial time.

Theorem 7. *If winning dominance solvable policies exist, then there exists an algorithm that can find the winning dominance solvable policy with the lowest inefficiency ratio, with running time that is polynomial in the number of actions of the players.*

5.3 Maximizing the Egalitarian Social Welfare

We now consider an altruistic goal for the manipulator that is different from the original goal of winning. Here, we relax the original goal of winning and develop a policy that ensures the utility of all players are as large as possible. To further this notion, we define the quantity we call the egalitarian social welfare, which we aim to maximize.

Definition 9. *The Egalitarian Social Welfare of a strategy profile* $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ *is defined to be*

$$\mathcal{S}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) := \min\Big\{U_1(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}), U_2(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}), U_3(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})\Big\}$$

We can find the dominance solvable policy that maximizes egalitarian social welfare in polynomial running time. Note that such a policy will always exist.

Theorem 8. *There exists an algorithm that can find the dominance solvable policy that maximizes egalitarian social welfare with running time that is polynomial in the number of actions of the players.*

Now, we present a number of examples of the theory we have developed. Each example highlights a different aspect of the theory.

6 Three-Player Iterated Prisoner's Dilemma

The first example we consider is a three-player version of the iterated prisoner's dilemma. As in the two-player version, each player must choose from a set of two actions $\mathcal{A} = \{\text{C, D}\}$ which stand for cooperate and defect respectively. The payoff matrices for each player are defined as follows:

$$A_0^{(i,j)} = \begin{bmatrix} 3 & 0 \\ 5 & 1 \end{bmatrix} \text{ if } i < j \text{ and } A_0^{(i,j)} = \begin{bmatrix} 3 & 5 \\ 0 & 1 \end{bmatrix} \text{ if } i > j$$

6.1 Winning Strategy for a Manipulator

Note that defection is a strictly dominant strategy for each player. Moreover, the payoff awarded to each player is the same when everyone defects. As a result, by Theorem 1, player 1 can win the game with high probability by repeatedly defecting, and never altering payoff matrices. Note that this policy is zero cost in the sense that the manipulator never needs to alter any payoff matrices. However, the margin is also zero. We now illustrate how alterations to the payoff matrices can result in a winning policy for the manipulator, which has positive margin, and encourages cooperation between players. In particular, we outline a policy which the manipulator may use to converge to the strategy profile (D, C, C). For $0 \leq \epsilon \leq 7/6$ set

$$\hat{A} = \begin{bmatrix} 3 & 5 \\ 3/2 + \epsilon & -1/2 \end{bmatrix}.$$

Let player 1 adopt the policy $\rho_t = \left(\mathbf{e}_2, \hat{A}, \hat{A}\right)$. Note that the mixed strategy of any player is characterized by the probability that they cooperate. If player 3 cooperates with probability λ then the expected utility player 2 receives from cooperating is $3/2 + \epsilon + 3\lambda$. Meanwhile, the expected utility player 2 receives by defecting is $1/2 + 4\lambda$. Since, $\lambda \in [0, 1]$, this implies that cooperation is a strictly dominant strategy for player 2. By symmetry, cooperation is also a strictly dominant strategy for player 3.
The single shot utility under the profile (D, C, C) for player 1 is $7 - 2\epsilon$. Meanwhile the utilities of players 2 and 3 are both $4.5 + \epsilon$. By Theorem 4.2, this implies

$$\mathbb{P}\Big(U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_2(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \text{ and } U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_3(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty}\Big) = 1$$

since $\epsilon \leq 7/6$.
Observe that the policy ρ has a much improved margin relative to the trivial policy of repeated defection we first considered. In fact, the margin of policy ρ is $2.5 - \epsilon$, which is the maximum margin achievable by a dominance solvable policy as $\epsilon \to 0$.

7 Social Distancing Game

Next, we consider a more practical application of the theory above. More precisely, we consider the social distancing game which is a small variation of the lemonade

stand game introduced by [25]: It is summer on a remote island, and you need to survive. You decide to set up camp on the beach (which you may shift anywhere around the island), as do two others. There are twelve places to set up around the island like the numbers on a clock. The game is repeated. Every night, everyone moves under cover of darkness (simultaneously). There is no cost to move. The pandemic is eternal, so the game is infinitely repeated. The utility of the repeated game is the time-averaged utility of the single-shot games. The only person that survives is the one with the highest total utility at the end of the game.

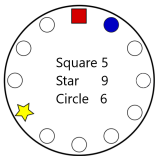

Fig. 1. Example Social Distancing Game

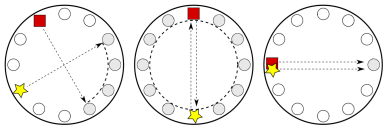

Fig. 2. Best-responses for different opponent configurations: The dashed and shaded segment indicates the third player's best-response actions, and arrows point to the action opposite each opponent. (Figures reworked from [21])

The utility of a player in a single round of the social distancing game is the sum of its distances from the other two players. The distance between two players is the length of the shortest path between them along the circumference of the clock. More formally, the distance between two positions is defined as follows:

$$d(i,j) = \begin{cases} |i-j| & |i-j| \leq 6 \\ 12 - |i-j| & otherwise \end{cases}$$

For example, if Alice sets up at the 3 o'clock location, Bob sets up at 10 o'clock, and Candy sets up at 6 o'clock, then the utility of Alice is $d(3,10) + d(3,6) = 5 + 3 = 8$, the utility of Bob is $d(10,3) + d(10,6) = 5 + 4 = 9$, and the utility of Candy is $d(6,3) + d(6,10) = 3 + 4 = 7$. If all the camps are set up in the same spot, everyone gets 0. If exactly two camps are located at the same spot, the two collocated camps get the distance to the non-collated camp as their utility and the non-collated camp gets twice the same distance as her utility. In contrast to the lemonade stand game, the social distancing game is not constant-sum.

However, it is a polymatrix game, and thus the techniques developed above can be applied. In what follows, we consider a three-player, infinitely repeated version of the social distancing game . Each player i has 12 actions, $\mathcal{A}_i = \{1, \ldots, 12\}$, each corresponding to a number on the clock. The payoff matrix for each pair of players (i, j) is derived directly from the distance function d. That is, $A_0^{(i,j)}(k, l) = d(k, l)$ for all $k, l \in \mathcal{A}_i$.

7.1 Winning Strategy for a Manipulator

We now present a winning dominance solvable type-I policy for the social distancing game. By definition, for any pair of positions (k, l) on the clock, $d(k, l) \leq 6$. This implies that the maximum utility achievable by any player is 12. In addition, a player i only achieves their maximum payoff when both remaining players place themselves directly opposite of player i. Thus, there are only 12 combinations of pure strategies which maximize the utility of player 1, each corresponding to a single number on the clock. In particular, we choose to work with one such strategy profile, $(\mathbf{e}_{12}, \mathbf{e}_6, \mathbf{e}_6)$. Consider the following dominance solvable policy. Set

$$\hat{A}(k, l) = \begin{cases} d(k, l) - \epsilon & \text{if } d(k, l) < 6 \\ d(k, l) + \epsilon & \text{if } d(k, l) = 6 \end{cases}$$

and let player 1 adopt the policy $\rho_t = \left(\mathbf{e}_{12}, \hat{A}, \hat{A}\right)$. First, observe that, under policy ρ, $\mathbf{e}_6$ is a dominant strategy for player 2 against the fixed strategy $\mathbf{e}_{12}$ of player 1. Additionally, by symmetry, $\mathbf{e}_6$ is also a dominant strategy for player 3 against the fixed strategy of player 1. Moreover, note that player 1's utility under the strategy profile $(\mathbf{e}_{12}, \mathbf{e}_6, \mathbf{e}_6)$ is $12 - 2\epsilon$. Meanwhile, the utilities of both players 2 and 3 is $6 + \epsilon$. Thus, by Theorem 1, for sufficiently small ϵ we have

$$\mathbb{P}\Big(U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_2(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \text{ and } U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_3(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty}\Big) = 1$$

Note that such a result implies that player 1 can guarantee her maximum payoff in the long run by only making an infinitesimal change to the payoff matrices!

7.2 Maximizing Egalitarian Social Welfare

We now present a socially good solution a manipulator can guide the players to converge to by using a winning dominance solvable policy. In the standard version of the game without a manipulator, one of the "socially optimal" strategy profiles is $(\mathbf{e}_{12}, \mathbf{e}_4, \mathbf{e}_8)$, since in this profile, all the players are spread out evenly around the clock. It is possible for a manipulator to guide the players to an approximately optimal solution, in the sense that she can enable convergence to the strategy profile $(\mathbf{e}_{12}, \mathbf{e}_5, \mathbf{e}_7)$.
Consider the following dominance solvable policy. Set

$$\hat{A}(k, l) = \begin{cases} d(k, l) & \text{if } k \neq 12 \\ d(k, l) - 1 - 2\epsilon & \text{if } k = 12 \text{ and } l \neq 5 \\ d(k, l) + 1 - \epsilon & \text{if } k = 12 \text{ and } l = 5 \end{cases}$$

and

$$\tilde{A}(k,l) = \begin{cases} d(k,l) & \text{if } k \neq 12 \\ d(k,l) - 1 + \epsilon & \text{if } k = 12 \text{ and } l \neq 7 \\ d(k,l) + 1 - \epsilon & \text{if } k = 12 \text{ and } l = 7 \end{cases}$$

and let player 1 adopt the policy $\rho_t = \left(\mathbf{e}_{12}, \hat{A}, \tilde{A}\right)$. First, observe that, under policy ρ, $\mathbf{e}_5$ is a dominant strategy for player 2 against the fixed strategy $\mathbf{e}_{12}$ of player 1. Additionally, $\mathbf{e}_7$ is a dominant strategy for player 3 against the fixed strategy of player 1. Moreover, note that player 1's utility under the strategy profile $(\mathbf{e}_{12}, \mathbf{e}_5, \mathbf{e}_6)$ is $10 - (2 + \epsilon) = 8 - \epsilon$. Meanwhile, the utilities of both players 2 and 3 is also $8 - \epsilon$. Thus, by Theorem 1, for sufficiently small $\epsilon > 0$ we have

$$\mathbb{P}\Big(U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_2(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \text{ and } U_1(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty} \geq U_3(\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t)_{t=1}^{\infty}\Big) = 1$$

Note that such a result implies that player 1 can guarantee that the game converges to an approximately socially optimal solution whilst ensuring that she still wins the game!

8 Conclusions

In this paper, we considered a 3-player repeated polymatrix game setting in which our agent is allowed to (slightly) manipulate the underlying game matrices of the other agents for which she pays a manipulation cost, while the other agents are 'consistent'. In our framework, two examples of consistent agents are those that use follow-the-leader or any no-regret algorithm to play the game. We first proposed a payoff matrix manipulation scheme and sequence of strategies for our agent that provably guarantees that the utility of any consistent opponent would converge to a value we desire. Using this theory we developed winning dominance solvable policies and winning batch coordination policies, both of which have strong theoretical guarantees such as tractability and the ability to win in a finite number of rounds almost surely. In addition, we showed that these policies can be found efficiently by solving a sequence of linear feasibility problems. We then considered additional objectives the manipulator may have, such as winning by the largest margin or whilst seeing a large improvement relative to the cost of modifying the payoff matrices. We then considered a socially good objective different from winning, namely maximization of the egalitarian social welfare. We showed that our framework could be extended to capture such objectives via linear objective functions. After this, we considered a social distancing game and showed that, by making only infinitesimal changes to the payoff matrices, the manipulator can maximize her payoff i.e. maximize her distance from the other players. The manipulator can also guide the utilities of all players to converge to a socially optimal solution. Note that due to page limitations, we have deferred all the proofs, example game analyses, and numerical results to the online ArXiv version of this paper (under the same title). Therefore we refer readers interested in more detailed discussions to that longer version of our paper.

References

1. Appa, G.: On the uniqueness of solutions to linear programs. Journal of the Operational Research Society **53**(10), 1127–1132 (2002)
2. Baglieri, D., Carfì, D., Dagnino, G.B.: Asymmetric r&d alliances and coopetitive games. In: International conference on information processing and management of uncertainty in knowledge-based systems. pp. 607–621. Springer (2012)
3. Bansal, T., Pachocki, J., Sidor, S., Sutskever, I., Mordatch, I.: Emergent complexity via multi-agent competition. In: International Conference on Learning Representations (2018)
4. Bishop, N., Dinh, L.C., Tran-Thanh, L.: How to guide a non-cooperative learner to cooperate: Exploiting no-regret algorithms in system design. In: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems (2021)
5. Carfì, D., Schiliro, D.: Global green economy and environmental sustainability: a coopetitive model. In: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems. pp. 593–606. Springer (2012)
6. Dinh, L.C., Nguyen, T.D., Zemhoho, A.B., Tran-Thanh, L., et al.: Last round convergence and no-dynamic regret in asymmetric repeated games. In: Algorithmic Learning Theory. pp. 553–577. PMLR (2021)
7. Gerding, E.H., Robu, V., Stein, S., Parkes, D.C., Rogers, A., Jennings, N.R.: Online mechanism design for electric vehicle charging. In: AAMAS. pp. 811–818 (2011)
8. Goldman, A., Tucker, A.: Theory in linear programming. In: Kuhn, H., Tucker, A. (eds.) Linear Inequalities and Related Systems, pp. 53–97. Princeton University Press (1956)
9. Grzeundefined, M.: Reward shaping in episodic reinforcement learning. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems. p. 565–573. AAMAS '17, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2017)
10. Heuer, G.A.: Uniqueness of equilibrium points in bimatrix games. International Journal of Game Theory **8**(1), 13–25 (Mar 1979). https://doi.org/10.1007/BF01763049, https://doi.org/10.1007/BF01763049
11. Lowe, R., WU, Y., Tamar, A., Harb, J., Pieter Abbeel, O., Mordatch, I.: Multi-agent actor-critic for mixed cooperative-competitive environments. Advances in Neural Information Processing Systems **30**, 6379–6390 (2017)
12. Myerson, R.B.: Game Theory: Analysis of Conflict. Harvard University Press (1991), http://www.jstor.org/stable/j.ctvjsf522
13. Nisan, N.: Introduction to mechanism design (for computer scientists). In: Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V. (eds.) Algorithmic Game Theory, chap. 9, pp. 209–242. Cambridge University Press (2007)
14. Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. Autonomous agents and multi-agent systems **11**(3), 387–434 (2005)
15. Parkes, D.C.: Online mechanisms. In: Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V. (eds.) Algorithmic Game Theory, chap. 16, pp. 411–442. Cambridge University Press (2007)
16. Phan, T., Gabor, T., Sedlmeier, A., Ritz, F., Kempter, B., Klein, C., Sauer, H., Schmid, R., Wieghardt, J., Zeller, M., et al.: Learning and testing resilience in cooperative multi-agent systems. In: Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems. pp. 1055–1063 (2020)

17. Quintas, L.G.: Uniqueness of nash equilibrium points in bimatrix games. Economics Letters **27**(2), 123 – 127 (1988). https://doi.org/https://doi.org/10.1016/0165-1765(88)90083-3, http://www.sciencedirect.com/science/article/pii/0165176588900833
18. Ryu, H., Shin, H., Park, J.: Cooperative and competitive biases for multi-agent reinforcement learning. In: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems. pp. 1091–1099 (2021)
19. Samvelyan, M., Rashid, T., de Witt, C.S., Farquhar, G., Nardelli, N., Rudner, T.G., Hung, C.M., Torr, P.H., Foerster, J.N., Whiteson, S.: The starcraft multi-agent challenge. In: AAMAS (2019)
20. Shapley, L., Karlin, S., Bohnenblust, H.: Solutions of discrete, two-person games. Contributions to the Theory of Games **1**, 51–72 (1950)
21. Sykulski, A., Chapman, A., Munoz de Cote, E., Jennings, N.: Ea^2: The winning strategy for the inaugural lemonade stand game tournament. Frontiers in Artificial Intelligence and Applications **215** (01 2010). https://doi.org/10.3233/978-1-60750-606-5-209
22. Tan, M.: Multi-agent reinforcement learning: Independent vs. cooperative agents. In: Proceedings of the tenth international conference on machine learning. pp. 330–337 (1993)
23. Wen, C., Xu, M., Zhang, Z., Zheng, Z., Wang, Y., Liu, X., Rong, Y., Xie, D., Tan, X., Yu, C., et al.: A cooperative-competitive multi-agent framework for auto-bidding in online advertising. arXiv e-prints pp. arXiv–2106 (2021)
24. Yuan, Q., Li, S., Wang, C., Xie, G.: Cooperative-competitive game based approach to the local path planning problem of distributed multi-agent systems. In: 2020 European Control Conference (ECC). pp. 680–685. IEEE (2020)
25. Zinkevich, M.A., Bowling, M., Wunder, M.: The lemonade stand game competition: solving unsolvable games. ACM SIGecom Exchanges **10**(1), 35–38 (2011)