arXiv:2208.10469v3 [cs.AI] 17 Aug 2023

Get It in Writing: Formal Contracts Mitigate Social Dilemmas in Multi-Agent RL[*]

Phillip J.K. Christoffersen[†]
MIT CSAIL
Cambridge, MA, USA
philljkc@mit.edu

Andreas A. Haupt[†]
MIT CSAIL
Cambridge, MA, USA
haupt@csail.mit.edu

Dylan Hadfield-Menell
MIT CSAIL
Cambridge, MA, USA
dhm@csail.mit.edu

ABSTRACT

Multi-agent reinforcement learning (MARL) is a powerful tool for training automated systems acting independently in a common environment. However, it can lead to sub-optimal behavior when individual incentives and group incentives diverge. Humans are remarkably capable at solving these social dilemmas. It is an open problem in MARL to replicate such cooperative behaviors in selfish agents. In this work, we draw upon the idea of formal contracting from economics to overcome diverging incentives between agents in MARL. We propose an augmentation to a Markov game where agents voluntarily agree to binding state-dependent transfers of reward, under pre-specified conditions. Our contributions are theoretical and empirical. First, we show that this augmentation makes all subgame-perfect equilibria of all fully observed Markov games exhibit socially optimal behavior, given a sufficiently rich space of contracts. Next, we complement our game-theoretic analysis with experiments running deep RL on the contracting augmentation for various social dilemmas. We discuss some practical issues with learning in the contracting augmentation, and provide a training methodology that leads to high-welfare outcomes, Multi-Objective Contract Augmentation Learning (MOCA). We test our methodology in static, single-move games, as well as dynamic domains that simulate traffic, pollution management and common pool resource management.

KEYWORDS

Social Dilemma, Decentralized Training, Formal Contracts

ACM Reference Format:
Phillip J.K. Christoffersen, Andreas A. Haupt, and Dylan Hadfield-Menell. 2023. Get It in Writing: Formal Contracts Mitigate Social Dilemmas in Multi-Agent RL. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 12 pages.

1 INTRODUCTION

We study the problem of how to get selfishly motivated agents to act pro-socially through the lens of multi-agent Reinforcement Learning (MARL). Consider the Cleanup domain, depicted in Figure 1. Agents get reward from eating apples that only grow if a nearby river is unpolluted. In a pro-social solution to Cleanup, agents need to work together: one cleans while the other eats apples. However, self-interested agents cannot sustain this solution. Cleaning has no direct benefit, so selfish agents focus exclusively on eating apples. A social dilemma ensues.

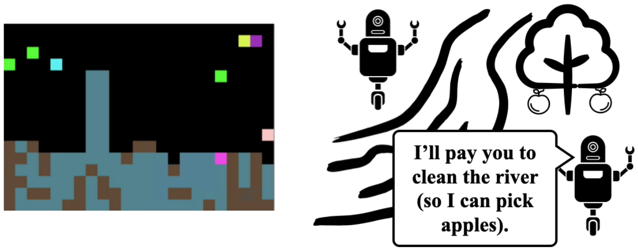


Figure 1: We evaluate our method in the Cleanup domain [11]. Left: A screenshot of the environment. The different agents correspond to the pink, yellow, and purple tiles. Agents get reward for eating apples (green) but apples will only grow if the river (blue) is clean of pollution (brown). Agents can clean up pollution, but aren't directly rewarded for cleaning. This creates a *social dilemma* where no agents clean because they don't expect to benefit from cleaning directly. Right: An illustration of the solution that our contracting augmentation facilitates. In the Cleanup domain, one agent commits to "pay" the other to clean the river. As a result, the agents are able to coordinate on policies that maximize the total reward across both agents.

Prior work has considered modifying MARL domains with the goal of mitigating such social dilemmas. One idea is to allow agents to transfer some of their reward to others in exchange for helpful actions, such as cleaning the river: gifting [19]. Other approaches allow agents to make commitments that they will take particular actions in the future [10]. Both approaches have limitations: gifting cannot change the Nash equilibria of a game, and hence cannot change the fundamental incentive structure of the game [39, Proposition 1]. Moreover, (binding) contracts in the sense of Hughes et al. [10] are only ever enacted when all agents are made better off in the original reward of the game. No agent will ever consent to a binding contract cleaning trash in Cleanup. Further, since binding contracts are action-level contracts, the system designer would have to manually encode a "clean trash" policy, instead of relying on a reward signal to *incentivise* them to clean.

This article studies contracts as zero-sum modifications of the environment reward. More specifically, contracts transfer rewards

[*] We thank Asher Wolitzky and David Parkes, as well as an audience at Stanford for helpful comments.
[†] Equal Contribution

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom. © 2023 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

	C	D
C	−1,−1	−3,0
D	0, −3	−2, −2

(a) Prisoner's Dilemma

	C	D
C	−1,−1	−1.5,−1.5
D	−1.5, −1.5	−2, −2

(b) After Contract

Figure 2: (a) Prisoner's Dilemma (b) Prisoner's Dilemma after signing a contract in which a defector transfers 1.5 reward to a cooperator. With this contract in force, cooperating becomes a dominant action for both players.

between agents depending on states and actions. Contracts are proposed by agents, and can be vetoed by any single agent—participation is voluntary. Even upon acceptance, agents may choose any action, only their rewards are changed by the contract.

In the Cleanup domain, an agent could propose to pay an amount r_{clean} of reward for each polluted river square that is cleaned. The proposing agent is 'charged' a reward penalty of $-r_{clean}$. The proposing agent has an incentive to propose this contract because the expected reward from eating apples will be larger than the expected payment to others for growing them. Similarly, the other agent prefers this contract to the competitive outcome, where no apples grow.

To make this concrete, consider the classic Prisoner's Dilemma [36].The tables in Figure 2 show the payoffs for the unmodified game (Figure 2a) and the modified incentives (Figure 2b) under the following contract:

> *Any agent who defects is fined* 1.5 *units of reward by the other agent.*

If both defect, both pay, and the payments cancel. In this modified game, cooperation is dominant, and hence (C, C) is the only Nash equilibrium.

Would the agents agree to this contract if proposed? If it is rejected, they subsequently play the game in Figure 2a which has a unique Nash equilibrium of (D, D), yielding a reward of -2 for both agents. However, if it is accepted, they subsequently play the game in Figure 2b which has a Nash equilibrium of (C, C), yielding a reward of -1. Thus, agents want to accept the contract and subsequently play the socially optimal outcome. Hence, a possibility to commit to a state-action dependent reward transfer, to *sign a formal contract* for short, mitigates a social dilemma, even among selfish agents.

Contributions. We provide three main contributions.

1. (1) We formalize *Formal Contracting* as a generic augmentation of Markov games (i.e., non-cooperative MARL);
2. (2) We prove that this augmentation makes socially optimal behavior an SPE, and that every SPE of the augmented game is socially optimal;
3. (3) We provide a multi-objective training procedure (MOCA), which performs close to, or better than, a joint controller after a fixed number of time periods in complex dynamic domains such as Cleanup and Harvest [11]. Using a state-of-the-art deep reinforcement learning algorithms without MOCA also yields results close to optimal in several domains.

Outline. We provide preliminaries and define our augmentation in section 2. In section 3, we provide our main theoretical result showing that formal contracting mitigates social dilemmas in all fully observed Markov games, and outline its proof. We describe our evaluation methodology and introduce MOCA in section 4. Experimental results are in section 5. We outline related work in section 6. In section 7, we discuss the real-world application and enforcement of contracts, fairness concerns, and avenues for future work. Appendices contain proofs, additional statements and experiments, a formal definition of a more general contracting augmentation, and hyperparameter settings for our experiments.

2 FORMAL CONTRACTING

2.1 Definitions

Full-Information Markov Games. We define an N-agent (complete-information) Markov game as a 6-tuple, $M = \langle S, s_0, \mathbf{A}, T, \mathbf{R}, \gamma \rangle$, where

- S is a state space;
- $s_0 \in S$ is the initial state;
- $\mathbf{A} = A_1 \times A_2 \times \cdots \times A_n$ is the space of action profiles $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ for n agents;
- $T \colon S \times \mathbf{A} \to \Delta(S)$ is a transition function;
- $\mathbf{R} \colon S \times \mathbf{A} \to [-R_{\max}, R_{\max}]^n$ is a (bounded) reward function mapping state-action profiles to reward vectors for the n agents; and
- $\gamma \in [0, 1)$ is a discount factor.

Agents choose policies $\pi_i \colon S \to \Delta(A_i)$, $i = 1, 2, \ldots, n$. We write $\boldsymbol{\pi} := (\pi_1, \pi_2, \ldots, \pi_N)$ to denote a *policy profile.* For a policy profile $\boldsymbol{\pi}$, we denote by $V_i^{\boldsymbol{\pi}}(s_0) := \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t R_i(s_t, \mathbf{a}_t)]$ the *value* to agent $i \in [n] := \{1, 2, \ldots, n\}$. In the value expression, the expectation is with respect to the generating process $s_t \sim T(s_{t-1}, \mathbf{a}_{t-1})$ and $a_{t,i} \sim \pi_i(s_t)$, $t = 1, 2, \ldots$. A subscript $_{-i}$ denotes a partial profile of policies or actions or policies excluding agent i, e.g., $\boldsymbol{\pi}_{-i} := (\pi_1, \pi_2, \ldots, \pi_{i-1}, \pi_{i+1}, \ldots, \pi_n)$.

Optimal Policy Profiles. Denote the welfare of a policy profile by $W^{\boldsymbol{\pi}}(s_0) := \sum_{i=1}^{n} V_i^{\boldsymbol{\pi}}(s_0)$. We refer to a policy profile that maximizes welfare as *jointly optimal*: $\boldsymbol{\pi}^* \in \arg\max_{\boldsymbol{\pi}} W^{\boldsymbol{\pi}}(s_0)$. A policy profile $\boldsymbol{\pi}$ is *Pareto-optimal* if there is no policy profile $\boldsymbol{\pi}'$ such that $V_i^{\boldsymbol{\pi}}(s_0) \leq V_i^{\boldsymbol{\pi}'}(s_0)$ for all $i = 1, 2, \ldots, n$, with a strict inequality for at least one agent. Intuitively, in such profiles, there are no "win-wins": no agent can attain higher reward without at least one other agent losing reward.

Stable Policy Profiles and Equilibria. In social dilemmas, social and individual incentives diverge. In our game-theoretic analysis, we use an equilibrium notion to capture outcomes of selfish incentives. One potential solution concept is *Nash equilibrium.* A policy profile is Nash equilibrium if unilateral deviation is suboptimal for all agents. Formally, a policy profile $\boldsymbol{\pi}$ is a Nash equilibrium if for any agent $i \in [n]$ and any policy $\pi_i' \colon S \to \Delta(A_i)$, $V_i^{\boldsymbol{\pi}}(s_0) \geq V_i^{(\pi_i', \boldsymbol{\pi}_{-i})}(s_0)$.

While this solution concept is common, it has its drawbacks. For example, in Cleanup, a policy profile in which agents never clean under some contract even if it is in their best interest, and another agent not proposing it, might be a Nash equilibrium, as no agent would benefit unilaterally from changing their behavior.

The “threat” of one of the agents to not clean, however, is non-credible, as, when the contract is enforced, they would rather clean. Such *non-credible threats* are well-known in Game Theory [25, Section 5.5], and can be expected from RL agents if they approximate sequentially rational agents.

To avoid non-credible threats, we model selfish incentives in MARL with *subgame-perfect equilibria* (SPE). Subgame perfection requires that for any state s, there cannot be a profitable deviation to another policy, for any agent. This is stronger than a Nash equilibrium, which only requires this to hold at the initial state s_0.

Definition 2.1 (Subgame-Perfect Equilibrium). A policy profile $\boldsymbol{\pi}$ is a *subgame-perfect Nash equilibrium* or *subgame-perfect* if for all states $s \in S$, agents $i = 1, 2, \ldots, n$ and policies $\pi'_i : S \longrightarrow \Delta(A_i)$,

$$V_i^{\boldsymbol{\pi}}(s) \geq V_i^{(\pi'_i, \boldsymbol{\pi}_{-i})}(s).$$

The Economics literature also often refers to SPE in Markov games as Markov Perfect Equilibria (MPE) [20].

Our game-theoretic analysis shows that a sufficiently rich contracting augmentation of Markov games forces socially optimal behavior in SPE and our experiments show that such equilibria are learned in decoupled MARL. The intuition behind our result is that all SPE for a contracting-augmented game are welfare maximizing if the contract space is rich enough to penalize all deviations from some welfare-maximizing policy profile. We say that the state space S is *sufficient to detect deviations from a policy profile* $\boldsymbol{\pi}$, or *has detectable deviations from* $\boldsymbol{\pi}$ if for any distinct agents $i \neq j$ and any state $s \in S$, the $N + 1$ sets

$$\operatorname{supp} T(s, \boldsymbol{\pi}(s)) \text{ and } \operatorname{supp} T(s, (a'_i, \boldsymbol{\pi}_{-i}(s)))$$

are mutually disjoint. Here, $\operatorname{supp} T(s, \mathbf{a})$ denotes the support of the transition function.

2.2 The Contracting Augmentation

Before we define our contracting augmentation, we first define *contracts*. They are state-action-dependent reward transfers, in addition to an *acceptance transfer*.

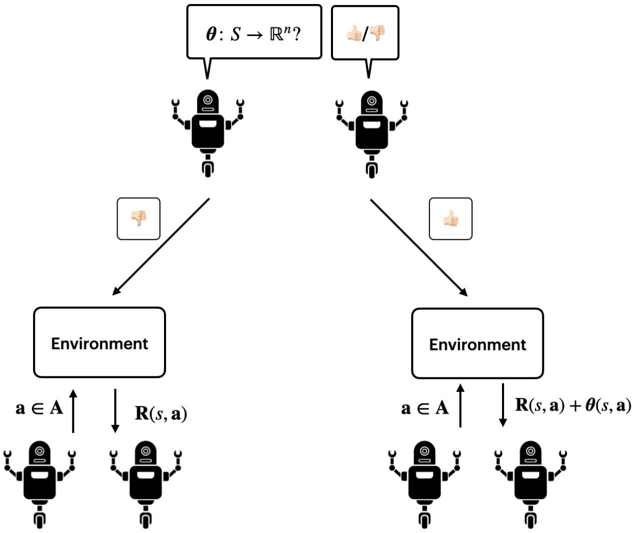
Definition 2.2 (Contract). A contract is a function $\boldsymbol{\theta} : (S \times A) \cup \{\mathrm{acc}\} \longrightarrow \mathbb{R}^N$ whose range consists of zero-sum vectors, i.e.

$$\sum_{i=1}^{N} \theta_i = 0$$

for any $(\theta_1, \theta_2, \ldots, \theta_n) \in \operatorname{range}(\theta)$. We denote a generic set of contracts by Θ.

A contract will be added to the reward vector that agents get, influencing the incentives in social dilemmas. The central definition of this article is the *contract augmentation*.

Definition 2.3 (Θ-Augmented Game). Let $M = \langle S, s_0, \mathbf{A}, T, \mathbf{R}, \gamma \rangle$ be a full-information Markov game and Θ be a set of contracts. The i-proposing, Θ-*augmented game* is $M^{\Theta} = \langle S', (i, \mathbf{0}), \mathbf{A}', T', \mathbf{R}', \gamma \rangle$, with the following components.



$\boldsymbol{\theta} : S \rightarrow \mathbb{R}^n$?
👍/👎
👎
👍
Environment
Environment
$\mathbf{a} \in \mathbf{A}$
$\mathbf{R}(s, \mathbf{a})$
$\mathbf{a} \in \mathbf{A}$
$\mathbf{R}(s, \mathbf{a}) + \boldsymbol{\theta}(s, \mathbf{a})$

Figure 3: The Contracting Augmentation. Top: Agents can propose *contracts*, state dependent, zero-sum, additive augmentations to their reward functions. Agents can accept or decline contracts. Left: In case of declination, the interaction between agents happens as before. Right: In case of acceptance of the contract, the reward of the agents is altered according to the rules of the contract.

States. The augmented state space is

$$S' = ([n] \cup S) \times (\{\mathbf{0}\} \cup \Theta).$$

States have the following meanings:

- $(i, \mathbf{0})$: Agent i has the opportunity to propose a contract $\boldsymbol{\theta} \in \Theta$;
- $(i, \boldsymbol{\theta})$: $\boldsymbol{\theta} \in \Theta$ awaits acceptance or rejection by all agents;
- $(s, \mathbf{0})$: The game is in state $s \in S$ with a null contract, $\mathbf{0}(s, a) = 0$, for all $s \in S, \mathbf{a} \in \mathbf{A}$, in force;
- $(s, \boldsymbol{\theta})$: The system is in state s with contract $\boldsymbol{\theta} \in \Theta$ in force.

Actions. The action spaces for the agents are

$$A'_i = A_i \cup \Theta \cup \{\mathrm{acc}\}$$

which corresponds to actions in the game (A_i), proposal actions (Θ) and the acceptance action $(\{\mathrm{acc}\})$.

Transitions. There are deterministic transitions, given by

$$T'((i, \mathbf{0}), (\boldsymbol{\theta}, \mathbf{a}_{-i})) = (i, \boldsymbol{\theta}), \qquad \text{for any } \mathbf{a}_{-i}$$

$$T'((i, \boldsymbol{\theta}), \mathbf{a}) = \begin{cases} (s_0, \boldsymbol{\theta}) & \text{if } \mathbf{a} = \mathbf{acc} \\ (s_0, \mathbf{0}) & \text{otherwise.} \end{cases}$$

for any contract $\boldsymbol{\theta} \in \Theta$ and any action profile $\mathbf{a} \in \mathbf{A}$. Here, we denoted $\mathbf{acc} := (\mathrm{acc}, \mathrm{acc}, \ldots, \mathrm{acc})$ the profile of unanimous acceptance of a contract.

Transitions in states $(s, \mathbf{0})$ and $(s, \boldsymbol{\theta})$ are as in the underlying game M,

$$T'((s, \boldsymbol{\theta}), \mathbf{a}) = T(s, \mathbf{a})$$

for any $s \in S$, $\boldsymbol{\theta} \in \Theta$ and $a \in A$.

Rewards.

$$\mathbf{R}'((s,\boldsymbol{\theta}),\mathbf{a}) = \mathbf{R}(s,\mathbf{a}) + \boldsymbol{\theta}(s,a),$$
$$\mathbf{R}'((i,\boldsymbol{\theta}),\mathbf{acc}) = \boldsymbol{\theta}(\mathrm{acc}).$$

for $\boldsymbol{\theta} \in \Theta$ and $s \in S$. All other rewards are zero. The first line means that depending on a state-action profile pair, reward is transferred between the agents. The second line refers to reward being transferred on signing a contract.

In the contracting augmentation, once enforced, the rewards of agents are directly changed. Note that agents maximize their reward as modified under the contract, so there is no concept of "breaking" a contract. The incentives that align agents' behavior with pro-social goals are encoded in the reward function.

3 GAME-THEORETIC ANALYSIS

The following is our main theoretical result: formal contracting with a sufficiently rich sets of contracts mitigates social dilemmas. Formally, we show that any subgame-perfect equilibrium of any fully-observed Markov game is jointly optimal.

Theorem 3.1. *Let $M = \langle S, s_0, \mathbf{A}, T, \mathbf{R}, \gamma\rangle$ be a full-information Markov game. For any sufficiently rich contracting space*

$$\Theta \supseteq \{(S \times \mathbf{A}) \cup \{acc\} \to [-R_{max}/(1-\gamma), R_{max}/(1-\gamma)]\},$$

all subgame-perfect equilibria $\boldsymbol{\pi}$ of M^{Θ} are jointly optimal and there is a jointly optimal policy profile π^* of M such that $\boldsymbol{\pi}(s,\boldsymbol{\theta}) = \boldsymbol{\pi}^*(s)$ for the contract $\boldsymbol{\theta}$ that agent i chooses in $\boldsymbol{\pi}$.

If there is a socially optimal policy profile π^* of M that has detectable deviators, there is a contract space Θ of dimension at most $|S|^2$ such that the above conclusion holds.

The theorem shows that, under the assumption of richness, social dilemmas are mitigated in equilibrium. Under detectability, a contract space of much small dimensionality (smaller by a factor of $|A_1| \times |A_2| \times \cdots \times |A_n|$ compared to the contract space needed in general games) is rich enough to mitigate dilemmas. For a full proof of the theorem, consult Appendix A.

Proof Sketch. Consider any subgame-perfect equilibrium $\boldsymbol{\pi}$ of M^{Θ}. We call the values $V_j^{\boldsymbol{\pi}}(s_0, \mathbf{0})$ the *non-acceptance value for agent i*, and $W^{\boldsymbol{\pi}^*}(s_0)$ the jointly optimal welfare in the game M. We prove this statement in four steps:

First, we show an upper bound on the value for the proposing agent, $V_i^{\boldsymbol{\pi}}(i,\boldsymbol{\theta})$. The proposing agent cannot get more value than the optimal welfare in M minus the aggregate non-acceptance value for agents $j \in [n] \setminus \{i\}$, since otherwise at least one agent will reject the proposed contract

$$V_i^{\boldsymbol{\pi}}(i,\boldsymbol{\theta}) \leq W^{\boldsymbol{\pi}^*}(s_0) - \sum_{j\in[n]\setminus\{i\}} V_j^{\boldsymbol{\pi}}(s_0,\mathbf{0}).$$

Next, show that there is a contract $\boldsymbol{\theta}^*$ such that this bound is attained,

$$V_i^{\boldsymbol{\pi}}(i,\boldsymbol{\theta}^*) = W^{\boldsymbol{\pi}^*}(s_0) - \sum_{j\in[n]\setminus\{i\}} V_j^{\boldsymbol{\pi}}(s_0,\mathbf{0}). \qquad (1)$$

This step involves two observations: First, if agents $j \in [n] \setminus \{i\}$ accept any contract $\boldsymbol{\theta}^*$ for which (1) holds, agent i will choose one such contract, as it yields the highest payoff among all contracts. One can observe that agents $j \in [n] \setminus \{i\}$ are indifferent between accepting and rejecting $\boldsymbol{\theta}^*$ (i.e. $V_j^{\boldsymbol{\pi}}(s_0,\boldsymbol{\theta}) = V_j^{\boldsymbol{\pi}}(s_0,\mathbf{0})$). Hence, there could be an equilibrium where all contracts $\boldsymbol{\theta}^*$ are rejected. This case, where (1) holds, requires more care, but we show in the full proof in Appendix A that such subgame-perfect equilibria do not exist, as agent i's best response is undefined in this case.

Moreover, the proposer needs to be able to infer which agents deviated from socially optimal play, in order to accurately punish deviation. Without any further assumption, this requires knowledge of the current state and the actions of *all* players. However, under detectability, the state reached after actions are taken by players is sufficient for deciding punishment, and therefore the contract can be represented in $|S|^2$ dimensions.

Finally, we observe that any contract $\boldsymbol{\theta}^*$ that is accepted and for which (1) holds, is played only in a subgame-perfect equilibrium that is jointly optimal. Hence, the subgame-perfect equilibrium $\boldsymbol{\pi}$ is jointly optimal. □

Fairness. One striking observation in the proof of Theorem 3.1 is that agents $j \in [n] \setminus \{i\}$ are indifferent between accepting the contract $\boldsymbol{\theta}^*$ and not accepting it. Hence, the contract leads to an improvement in welfare, but no agents but agent i gets any benefit from this improvement. In many decentralized learning tasks, this is not of concern, for example if a robot fleet needs to coordinate on locations. In others, this property is clearly unfair. We discuss ways to compensate this unfairness in section 7.

4 EXPERIMENTAL METHODOLOGY

We now evaluate the performance of the contracting augmentation. First, we introduce the baseline methods that we use to evaluate our approach. Then, we introduce our experimental domains. Finally, we provide details on MOCA, our training procedure for contracting.

4.1 Evaluation

We evaluate MOCA by comparing to the following baselines.

- Joint Training: a centralized algorithm with joint action space $\mathbf{A} = \times_{i=1}^{N} A_i$ chooses actions to maximize welfare;
- Separate Training: Agents selfishly maximize their reward;
- Gifting: Agents can "gift" [19] another agent at every timestep by directly transferring some of their reward;
- Vanilla Contracting: Run an off-the-shelf deep RL on the contract-augmented versions of the respective domains.

We train all domains with 2, 4, and 8 agents, using Proximal Policy Optimization [32] with continuous state and action spaces with Gaussian sampling in ray rllib's [18] implementation (hyperparameter choices can be found in Appendix C). In each domain, we train gifting agents with a lower bound of 0 and an upper bound on transfer value in contracts. This allows the same magnitude of transfers in gifting and contracts, for fair comparison. In one of the games, Emergency Merge, we reduced the gifting values to 10 per timestep, as this improved gifting's performance. On the Prisoner's Dilemma and the Public Goods game, we trained agents for 1M environment steps, and the complex dynamic games are trained for 10M environment steps.

4.2 Games

We test on several classes of games. We use Prisoner's Dilemma and a Public Goods game as static, simultaneous-move games, and Harvest, Cleanup, and Emergency Merge as dynamic domains.

Prisoner's Dilemma. First, we study the Prisoner's Dilemma, mentioned earlier. To scale this to multiple agents, we follow the following scheme for payoffs: in the n-agent game, if all agents cooperate, they each get reward n, and if all defect, they all get reward 1. However, if some defect and some cooperate, the ones that cooperate get reward 0 and the ones that defect get reward $n + 1$. Again, the socially optimal outcome is the one where all agents cooperate, but only Nash equilibrium is where all agents defect. We run an additional timestep after the matrix game is played for gifting actions to take place.

Public Goods. We study the following public goods game [13]. Agents choose an *investment* $a_i \in [0, 1]$, and get reward $R_i(\mathbf{a}) = \frac{1.2}{N} \sum_{j=1}^{N} a_j - a_i$, i.e. they are given their share of the public returns, the investment returning 20%, minus their own investment level. At social optimum, all agents choose $a_i = 1$ to get optimal joint reward. However, selfish agents are not incentivized to invest at this high level, as they would like to free-ride on the other agents' efforts.

Harvest. In Harvest, from Hughes et al. [11], agents move along a square grid to consume apples, gaining a unit of reward. Apples grow faster if more apples are close by, which leads to incentives to overconsume now, leading to an intertemporal dilemma. We choose engineered features to limit the amount of computational resources needed. In particular, agents receive their position and orientation, the coordinates and orientation of the closest other agent, the position of the nearest apple, the number of apples close to the agent, the number of total apples, and the number of apples eaten by each agent in the last timestep. We don't allow agents to use a punishment beam following Lupu and Precup [19]. The environment runs for 1,000 timesteps per episode.

Cleanup. In Cleanup, also from [11], agents similarly move along a square grid to consume apples and gain one unit of reward. Apples only spawn if a nearby river contains a number of waste objects lower than a threshold. Removing a waste object is a costless, but also rewardless, task. Apple-eating agents can free-ride on other agents, which leads to degraded performance. The observation space used for agents is simplified to limit computational requirements, and agents are passed their position and orientation, the position and orientation of the closest agent, the positions of the closest apple and waste object, and the number of current apples and waste objects. The environment runs for 1,000 timesteps per episode.

Emergency Merge. A set of $n - 1$ cars approaches a merge, an ambulance behind them, compare Figure 4. The ambulance incurs a penalty of 100 per timestep that it has not reached the end of a road segment past the merge. The cars in front also want to get to the end of the road segment, but incur a penalty of only 1 per timestep. They are limited to one-fourth of the velocity that the ambulance can go. We assume access to controllers preventing cars from colliding (stopping cars short of crashing into another car) and managing merging, and so the actions $a_i \in [-0.1, 0.1]$ only control the forward acceleration of each vehicle. A dilemma arises as cars prefer to drive to the merge fast, not internalizing the strong negative effect this has on the ambulance. The environment resets after 200 rounds or when cars crash, whichever is earlier. Note that here, due to the asymmetry of agent capabilities and rewards, attaining optimal social welfare cannot be done via Pareto improvement within the original game.

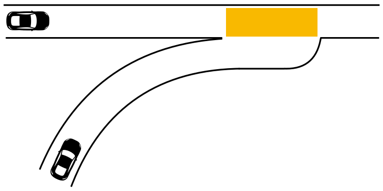


Figure 4: A depiction of the emergency merge domain.

4.3 Contract Spaces

We consider low-dimensional contract spaces for different domains.

- Prisoner's Dilemma. Contracts are parameterized by a transfer $\theta \in [0, n]$ for defecting, which is distributed to the other agents in equal proportions.
- Public Goods. Contracts are parameterized by a transfer $\theta \in [0, 1.2]$, agents transfer $\theta(1 - a_i)$, which is distributed to the other agents in equal proportions.
- Harvest. Contracts are parameterized by $\theta \in [0, 10]$. When an agent takes a consumption action of an apple in a low-density region, defined as an apple having less than 4 neighboring apples within a radius of 5, they transfer θ to the other agents, which is equally distributed to the other agents.
- Cleanup. Contracts are parameterized by $\theta \in [0, 0.2]$, which correspond to a payment per garbage piece cleaned, paid for evenly by the other agents.
- Emergency Merge. The ambulance can propose a per-unit subsidy of $\theta \in [0, 100]$ to the cars at the time of ambulance crossing. Each car is transferred θ times its distance behind the ambulance at time of merge by the ambulance. If a car is ahead of the ambulance at time of reward, it pays the ambulance θ times its distance ahead of the ambulance.

4.4 Training

The contracting augmentation yields a Markov game, for which one could directly train agents with deep reinforcement learning (we will call this *vanilla contracting*). However, as can be observed from Figure 5, and Figure 6, this implementation of contracting does not outperform joint training in problems with more complex dynamics, or higher-dimensional state and action spaces. To fix this, we propose an algorithm inspired by multi-objective reinforcement learning, compare [1], *Multi-Objective Contract Augmentation Learning* (MOCA). We present it in algorithm 1. MOCA consists of two phases: first, the algorithm draws random contracts (which, in the language of multi-objective reinforcement learning can be viewed as different "objectives"). This can be used to estimate $V_i^{\boldsymbol{\pi}}(s_0, \boldsymbol{\theta})$,

$i = 1, 2, \ldots, n$ for the initial state s_0 and any contract $\boldsymbol{\theta}$, i.e. the values for agents when contract $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ is in force. This allows it to learn estimates of the utility agents will get under a particular contract. Due to random sampling, these estimates are not biased by contract exploration, which may be an issue when using deep reinforcement learning directly.

In a second phase, we freeze play following $(s_0, \boldsymbol{\theta})$ for any contract $\boldsymbol{\theta}$ and the policy at states $(i, \mathbf{0})$ and $(i, \boldsymbol{\theta})$. We do so by choosing a contract repeatedly from the policy $\pi_i(i, \mathbf{0})$, and use as a proxy for acceptance the expected probability of acceptance, $\prod_{j=1}^{n} \pi_j(i, \boldsymbol{\theta})$. In order to help exploration of the contract space in this stage, we sample $\boldsymbol{\nu}$ agents from the space of non-proposing agents, and only use these agent's accept-reject probabilities in determining contract acceptance. Here, the introduced $\boldsymbol{\nu}$ becomes a tunable hyperparameter, for which $\boldsymbol{\nu} = 2$ obtained strong performance across all domains, which we report in section 5. We update the weights for the actions of all agents at $\pi_i(i, \mathbf{0})$ and $\pi_j(i, \boldsymbol{\theta})$, for $j = 1, 2, \ldots, n$. Finally, the algorithm returns the so-obtained policy profile.

Algorithm 1: Multi-Objective Contract-Augmentation Learning (MOCA)

```
Data: Contract Space $\boldsymbol{\Theta}$ including the null contract $\mathbf{0}$,
      Markov Game $M$, probability distribution $P(\boldsymbol{\Theta})$
Result: Policy Profile $\boldsymbol{\pi}$
$\boldsymbol{\pi} \leftarrow$ initialize_policies();
for $t = 1$ to $\frac{9}{10}$ num_episodes do
    $\boldsymbol{\theta} \sim P(\boldsymbol{\Theta})$;
    train_subgame_episode($\boldsymbol{\pi}(s_0, \boldsymbol{\theta})$)
Freeze $\pi|_{S \times \Theta}$;
for $i = 1$ to $\frac{1}{10}$ num_episodes do
    $\boldsymbol{\theta} \sim \pi_i(i, \mathbf{0})$;
    if rand() $< \prod_{j=1}^{n} \pi_j(i, \theta)$ then contract $\leftarrow \boldsymbol{\theta}$;
    else contract $\leftarrow \mathbf{0}$;
    R $\leftarrow$ sample_episode_reward($\boldsymbol{\pi}$, contract);
    train_with_rewards($\boldsymbol{\pi}$, R);
return $\boldsymbol{\pi}$;
```

We evaluate the performance of the final trained algorithm on rollouts. The choice of length of the two periods (e.g. the $\frac{9}{10}$th for the first phase) is arbitrary.

5 RESULTS

We first present a sample of our experiments with our baselines, which motivate the need for MOCA (algorithm 1), in Figure 5. Then, we discuss overall trends from all conducted experiments, Figure 6.

Vanilla Contracting. Consider first Figure 5. we observe that, in Prisoner's Dilemma and Cleanup, the baseline implementation of contracting is sufficient to achieve optimal or near-optimal performance, as can be seen by contracting either matching or surpassing the social welfare of training all agents jointly, and vastly surpassing the welfare of both gifting and separate training (both of which converge to socially suboptimal Nash equilibrium welfare). However, in more complex domains, such as Cleanup, this ceases to be the case. One potential reason for this is that, in these domains, learning the best responses to contracts becomes much more challenging, and so estimates of value for given contracts are less reliable early in training. Therefore, the proposing agent may benefit from additional exploration of the space of contracts, the main feature of MOCA. As seen in Figure 6, MOCA again attains higher social welfare than joint training, separate training, and gifting. However, since intermediate levels of reward are not directly comparable with the baselines (since contracts are randomly sampled in the first stage of training, and are not run for the same number of timesteps in the second stage), MOCA is omitted from Figure 5. For this, results are presented in Figure 6 with bar plots summarizing welfare at the end of training, for all evaluated methods.

MOCA. Now, we take a closer look at the full results in Figure 6. In the simpler domains (left two columns), MOCA, like vanilla contracting, attained social welfare is vastly higher than for separately trained agents and agents trained with gifting. In Prisoner's Dilemma, contracting reaches joint optimality for 2, 4, and 8 agents. A smaller action space (and hence easier exploration) is a potential reason reason for why contracting can perform *even better* than joint training, since the action space for joint training grows exponentially in the number of agents. In Public Goods, especially for higher number of agents, joint training interestingly outperforms MOCA, but not vanilla contracting. One possible reason for this is that, uniquely in our suite of environments, learning best responses to each contract is challenging, while the socially optimal policy is itself trivial to execute. Therefore, early in training, it is likely that socially optimal play is learned as a response to some of the contracts, particularly for those θ which are near-optimal. Therefore, biasing contract exploration early on is good for performance. In complex environments, since the socially optimal contracts are harder to execute, early biasing of contract exploration is unlikely to be well-informed, and so converging onto a poor contract proposal algorithm is likely in vanilla contracting at scale.

In the more complex domains (right three columns of Figure 6), MOCA in almost all cases attains at least the level of social welfare as joint training, and often far exceeds it. The exceptions to this trend are 2-agent Harvest and 2-agent Cleanup - in both cases, joint training and separate training exhibit strong performance (the former performing comparably to MOCA in both cases, the latter only for Harvest). The reason for this is that the two-agent settings for this problem are not strong social dilemmas, since the grid is wide enough that agents will not directly interact. Notably, this trend even applies in cases where the vanilla contracting fails (particularly in Harvest and Cleanup), motivating MOCA. In the merge domain, MOCA, contracting with a standard PPO training, and joint training, all perform similarly, given the fact that this is a substantially lower-dimensional, and simpler in terms of best-response policies, than Harvest or Cleanup.

6 RELATED WORK

We review related work in Computer Science and Economics.

Social Dilemmas. Our work intends to mitigate social dilemmas in games. In addition to classical static social dilemmas such as Prisoner's Dilemma (compare [36]), a public goods game (compare [13]) and Stag Hunt [28], we also consider more complex social

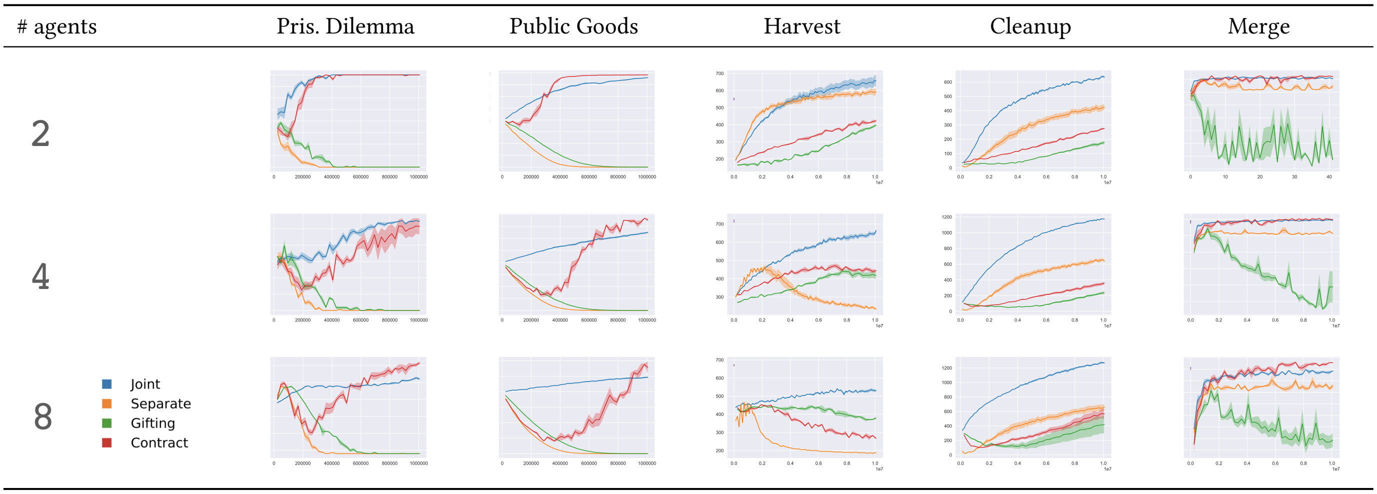


Figure 5: Welfare throughout training the benchmark algorithms (including a *vanilla* implementation of contracting using off-the-shelf deep reinforcement learning algorithms). In simple static domains, contracting achieves welfare that is close to joint optimality, but more complex domains (i.e. Harvest and Cleanup), biased policy exploration due to the difficulty of learning best-responses has a greater effect. Therefore, vanilla contracting suffers in performance. For each figure, the *x*-axis plots number of environment steps (e.g. all agents taking an action is one step), and error is one standard deviation over 5 independent runs.

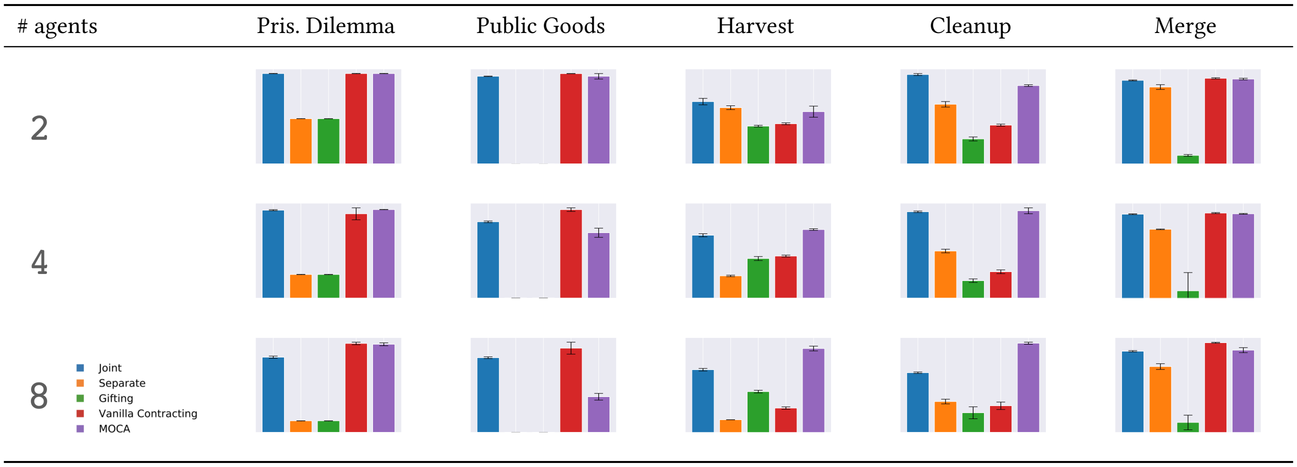


Figure 6: Experimental results including MOCA. Every plot is the mean social reward per episode at the end of training (1M plays of the static domains, 5M timesteps for the dynamic domains) for each of the 5 algorithmic setups tested. Cells vary across number of agents present (2, 4, 8), environments (Prisoner's Dilemma, Public Goods, Harvest, Cleanup, Emergency Merge) with each cell comparing different algorithms (joint, gifting, separate, vanilla contracting, MOCA). Error bars denote one standard deviation over five independent runs. For simpler games in the left two columns, MOCA attains higher social reward than both gifting and separate training. However, it fails to match joint training in Public Goods, since this is a domain with simple environment dynamics where learning to respond to contracts is difficult, so early best responses are more likely to be informative, and biasing towards these early on is likely to help performance. For all of the more complex domains in the right 3 columns, contracting leads to higher social reward than gifting and separate training, and always at least matches that of joint training, except for 2-agent Harvest and Cleanup, where sufficient resources are available to make these very mild social dilemmas, as the higher welfare resulting from separate training compared to joint training shows. In Emergency Merge, both vanilla contracting and MOCA contracting significantly outperforms separate training. In several domains, contracting outperforms joint training, which is a result of the large action space of the joint problem.

dilemmas such as the Harvest and Cleanup domains of [11]. Cooperation and prosociality in complex domains are of keen interest to MARL researchers, with related challenges including dilemmas like Gathering and Wolfpack [17], the StarCraft challenge [30], or more recently with MARL results in Diplomacy [16].

Augmentations of Markov Games. We relate to the study of augmentations of Markov games to enhance pro-social behavior. Gifting [19, 39] is as augmentation expanding agents' action spaces to allow for reward transfers to other agents. [39] prove that gifting is unable to change the set of Nash equilibria of the underlying game. Our approach differs from gifting in that contracting forces commitment to a given modification of reward *before* taking an action in the original game, and that this commitment is binding for the length of time the contract is in force. This improves total welfare by allowing to support play that is not a Nash equilibrium in the original game, as we demonstrate theoretically and experimentally.

Commitment Mechanisms. A large class of prior work has considered different forms of agent commitments. [33] proposes the contract net protocol, which allocates tasks among agents and commits them to complete a particular task. [10] is a recent study in this line of work. The paper considers multi-agent zero-sum games in which agents may give other agents the option to commit to taking a particular action. As we discuss in the introduction, [10]'s contracts—which we will call *binding contracts* as opposed to our concept of *formal contracts*—might be insufficient to induce the desired joint behavior, as agents will only commit to actions which improve their own individual welfare over the basic game, leading to Pareto optima, which may not be jointly optimal. [27] considers the idea of commitments, without transfers, in evolutionary game theory. The paper [21] lets an auxiliary agent propose a Pareto-optimal equilibrium in a game. [31] proposes to allow agents to be able to decommit from a task and paying a side payment. Our approach can be seen as "soft commitment" in which agents always only incur a cost in terms of reward when taking different actions, but are not forced to take a particular action. [34] considers in 2-player games the proposal of commitment and side payments, and reaches social cooperation. We provide a general approach that only considers reward transfer without the need to commit to actions. The idea of negotiations between rounds to arrive to a commitment to an actions, was considered in [3]. Formal contracting does not have the dynamic structure of a negotiation and lets a proposing agent make a take-it-or-leave-it offer. Also related is a literature on the emergence and learning of social norms [14, 38] and conventions [15], which do not require an explicit consent by agents, in contrast to the present paper.

Stackelberg Learning. Another related paradigm to ours is Stackelberg Learning. In such models, typically, a special agent, the principal, optimizes incentives for other agents in a bi-level optimization problem. Stackelberg learning has received a lot of attention in strategic Machine Learning [8, 22, 41] and has been used to learn large scale mechanisms such as auctions [2, 5]. Also, [40]'s approach to learning to incentivize other learning agents may be seen as a Stackelberg Learning. In contrast to Stackelberg Learning, the focus of formal contracts is that no additional agents—Stackelberg leaders—are introduced into an environment, but proposing agents are part of the environment.

Organizational and Contract Economics. Formal contracts have been considered as an alternative to relational contracts in the fields of organizational Economics, see [7] and [6, 5.2.3]. The setting of an agent proposing state-dependent reward transfers has received considerable attention in contract economics, compare the literature following [23], and mechanism design, compare [12, 24, 37]. In our proof of Theorem 3.1, we use a *forcing contract*, compare [9].

7 DISCUSSION

We discuss that the assumption that a single agent proposes a contract is crucial for our results, and its fairness implications, in subsection 7.1. Finally, we discuss approaches to scaling formal contracting to more complex domains in subsection 7.2.

7.1 Limitations for Formal Contracting

One crucial assumption in our analysis is that a single agent proposes contracts. Game-theoretic analysis, given in Appendix A shows that if two or more agents may propose in a game, SPEs may be socially suboptimal. The intuition is that an agent i may choose a contract to affect the state distribution in a way that gives them a rejection reward when j proposes a contract, hence increasing their reward when contracting. Our game-theoretic analysis also showed that unfair outcomes might result from contracts. As hence proposal by different agents and joint optimal behavior are incompatible, system designers that would like to ensure fairness need to design contracts in a way that limit the number of transfers that can be made, potentially at the expense of welfare.

7.2 Scaling Formal Contracting

The clearest avenue for future work is in scaling contracts to more realistic domains. Here, we outline three ways to do that.

First, contracts in this work were hand-engineered with relevant internal logic, in order to make the transfers a useful signal. For this approach to scale, a complexity tradeoff must be managed: Contracts need flexible enough to extract enough relevant information to incentivize welfare-optimal play, while being simple enough for MARL agents to allow fast learning of which contracts to choose resp. accept. General techniques allowing to choose contracts would greatly improve the scalability of the method.

Manually managing this tradeoff is undesirable. In particular, not all domains might have social inefficiencies that are as transparent, or have features that make it hard for a system designer to design good contract spaces. Therefore, learning which aspects of a state are useful for contracting will allow us to scale the approach to more realistic scenarios while keeping contract space sizes low.

Even with a fixed contract space, sample efficiency may be improved. MOCA took a first step into improving contract learning, by decreasing the bias in estimated $V_i^{\boldsymbol{\pi}}(s_0, \boldsymbol{\theta})$ values. MOCE outperformed benchmarks in all, even complex, dynamic, environments that exhibited a social dilemma. Increasing sample efficiency will allow using formal contracting to mitigate social dilemmas in even more complex domains. One potential way to increase the sample efficiency of the first phase of MOCA is to leverage more of the literature on multi-task reinforcement learning methods [4, 26, 29, 35].

REFERENCES

[1] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. 2017. Hindsight experience replay. *Advances in neural information processing systems* 30 (2017).

[2] Michael Curry, Tuomas Sandholm, and John Dickerson. 2022. Differentiable Economics for Randomized Affine Maximizer Auctions. *arXiv preprint arXiv:2202.02872* (2022).

[3] Dave De Jonge and Dongmo Zhang. 2020. Strategic negotiations for extensive-form games. *Autonomous Agents and Multi-Agent Systems* 34, 1 (2020), 1–41.

[4] Coline Devin, Abhishek Gupta, Trevor Darrell, Pieter Abbeel, and Sergey Levine. 2017. Learning modular neural network policies for multi-task and multi-robot transfer. In *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2169–2176.

[5] Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C Parkes, and Sai S Ravindranath. 2021. Optimal auctions through deep learning. *Commun. ACM* 64, 8 (2021), 109–116.

[6] Robert Gibbons. 2005. Four formal (izable) theories of the firm? *Journal of Economic Behavior & Organization* 58, 2 (2005), 200–245.

[7] Ricard Gil and Giorgio Zanarone. 2017. Formal and informal contracting: Theory and evidence. *Annual Review of Law and Social Science* 13 (2017), 141–159.

[8] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. 2016. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*. 111–122.

[9] Bengt Holmström. 1979. Moral hazard and observability. *The Bell journal of economics* (1979), 74–91.

[10] Edward Hughes, Thomas W Anthony, Tom Eccles, Joel Z Leibo, David Balduzzi, and Yoram Bachrach. 2020. Learning to Resolve Alliance Dilemmas in Many-Player Zero-Sum Games. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 538–547.

[11] Edward Hughes, Joel Z Leibo, Matthew G Phillips, Karl Tuyls, Edgar A Duéñez-Guzmán, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin R McKee, Raphael Koster, et al. 2018. Inequity aversion improves cooperation in intertemporal social dilemmas. *arXiv preprint arXiv:1803.08884* (2018).

[12] Leonid Hurwicz. 1973. The design of mechanisms for resource allocation. *The American Economic Review* 63, 2 (1973), 1–30.

[13] Marco Janssen and TK Ahn. 2003. Adaptation vs. anticipation in public-good games. In *annual meeting of the American Political Science Association, Philadelphia, PA*.

[14] Raphael Köster, Dylan Hadfield-Menell, Richard Everett, Laura Weidinger, Gillian K Hadfield, and Joel Z Leibo. 2022. Spurious normativity enhances learning of compliance and enforcement behavior in artificial agents. *Proceedings of the National Academy of Sciences* 119, 3 (2022).

[15] Raphael Köster, Kevin R McKee, Richard Everett, Laura Weidinger, William S Isaac, Edward Hughes, Edgar A Duéñez-Guzmán, Thore Graepel, Matthew Botvinick, and Joel Z Leibo. 2020. Model-free conventions in multi-agent reinforcement learning with heterogeneous preferences. *arXiv preprint arXiv:2010.09054* (2020).

[16] János Kramár, Tom Eccles, Ian Gemp, Andrea Tacchetti, Kevin R McKee, Mateusz Malinowski, Thore Graepel, and Yoram Bachrach. 2022. Negotiation and honesty in artificial intelligence methods for the board game of Diplomacy. *Nature Communications* 13, 1 (2022), 7214.

[17] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-agent reinforcement learning in sequential social dilemmas. *arXiv preprint arXiv:1702.03037* (2017).

[18] Eric Liang, Richard Liaw, Robert Nishihara, Philipp Moritz, Roy Fox, Ken Goldberg, Joseph Gonzalez, Michael Jordan, and Ion Stoica. 2018. RLlib: Abstractions for distributed reinforcement learning. In *International Conference on Machine Learning*. PMLR, 3053–3062.

[19] Andrei Lupu and Doina Precup. 2020. Gifting in multi-agent reinforcement learning. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 789–797.

[20] Eric Maskin and Jean Tirole. 2001. Markov perfect equilibrium: I. Observable actions. *Journal of Economic Theory* 100, 2 (2001), 191–219.

[21] Stephen McAleer, John Lanier, Michael Dennis, Pierre Baldi, and Roy Fox. 2021. Improving Social Welfare While Preserving Autonomy via a Pareto Mediator. *arXiv preprint arXiv:2106.03927* (2021).

[22] Smitha Milli, John Miller, Anca D Dragan, and Moritz Hardt. 2019. The social cost of strategic classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 230–239.

[23] Michael Mussa and Sherwin Rosen. 1978. Monopoly and product quality. *Journal of Economic theory* 18, 2 (1978), 301–317.

[24] Roger B Myerson. 1981. Optimal auction design. *Mathematics of operations research* 6, 1 (1981), 58–73.

[25] Martin J Osborne and Ariel Rubinstein. 1994. *A course in game theory*. MIT press.

[26] Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. 2015. Actor-mimic: Deep multitask and transfer reinforcement learning. *arXiv preprint arXiv:1511.06342* (2015).

[27] Luís Moniz Pereira, Tom Lenaerts, et al. 2017. Evolution of commitment and level of participation in public goods games. *Autonomous Agents and Multi-Agent Systems* 31, 3 (2017), 561–583.

[28] Jean-Jacques Rousseau. 1985. *A discourse on inequality*. Penguin.

[29] Andrei A. Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. 2015. Policy distillation. *arXiv preprint arXiv:1511.06295* (2015).

[30] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. 2019. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043* (2019).

[31] Tuomas W Sandholm and Victor R Lesser. 1996. Advantages of a leveled commitment contracting protocol. In *AAAI/IAAI, Vol. 1*. Citeseer, 126–133.

[32] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[33] Reid G Smith. 1980. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on computers* 29, 12 (1980), 1104–1113.

[34] Eric Sodomka, Elizabeth Hilliard, Michael Littman, and Amy Greenwald. 2013. Coco-q: Learning in stochastic games with side payments. In *International Conference on Machine Learning*. PMLR, 1471–1479.

[35] Yee Teh, Victor Bapst, Wojciech M Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. 2017. Distral: Robust multitask reinforcement learning. *Advances in Neural Information Processing Dystems* 30 (2017).

[36] Albert W Tucker and Philip D Straffin Jr. 1983. The mathematics of Tucker: A sampler. *The Two-Year College Mathematics Journal* 14, 3 (1983), 228–232.

[37] William Vickrey. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance* 16, 1 (1961), 8–37.

[38] Eugene Vinitsky, Raphael Köster, John P Agapiou, Edgar Duéñez-Guzmán, Alexander Sasha Vezhnevets, and Joel Z Leibo. 2021. A learning agent that acquires social norms from public sanctions in decentralized multi-agent settings. *arXiv preprint arXiv:2106.09012* (2021).

[39] Woodrow Z Wang, Mark Beliaev, Erdem Bıyık, Daniel A Lazar, Ramtin Pedarsani, and Dorsa Sadigh. 2021. Emergent Prosociality in Multi-Agent Games Through Gifting. *arXiv preprint arXiv:2105.06593* (2021).

[40] Jiachen Yang, Ang Li, Mehrdad Farajtabar, Peter Sunehag, Edward Hughes, and Hongyuan Zha. 2020. Learning to incentivize other learning agents. *Advances in Neural Information Processing Systems* 33 (2020), 15208–15219.

[41] Tijana Zrnic, Eric Mazumdar, Shankar Sastry, and Michael Jordan. 2021. Who Leads and Who Follows in Strategic Classification? *Advances in Neural Information Processing Systems* 34 (2021), 15257–15269.

A OMITTED PROOF AND STATEMENTS

Proof of Theorem 3.1. For convenience, call the proposition agent $i = 1$. Consider any subgame-perfect equilibrium of M^{Θ}. First observe that any contract $\boldsymbol{\theta}$ that brings agent 1 more value than $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}$ is not accepted by at least one agent $j = 2, 3, \ldots, n$. Indeed, in the case of rejection of the contract, agent $j = 2, 3, \ldots, n$ gets value $V_i^{\boldsymbol{\pi}}$. Hence, of the total welfare of an optimal contract $\boldsymbol{\pi}^*$, agent 1 can capture at most $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0)$ without strictly incentivizing at least one agent to reject the contract.

Next note that any not jointly optimal contract that yields utility $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0)$ for agent 1 is rejected. By definition, $W^{\boldsymbol{\pi}'}(s_0) < W^{\boldsymbol{\pi}^*}(s_0)$. Hence, if agent 1 gets value

$$W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^{n} V_i^{\boldsymbol{\pi}}(s_0),$$

this means that under the contract at least one agent gets value strictly smaller than $V_i^{\boldsymbol{\pi}'}(s_0) < V_i^{\boldsymbol{\pi}}(s_0)$, also leading to rejection.

Next, show that utility $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0)$ for agent 1 is achievable. Consider a forcing contract which punishes any deviation from an efficient policy profile by $-R_{\max}/(1-\gamma)$. A signing transfer of $V_i^{\boldsymbol{\pi}}(s_0) - V_i^{\boldsymbol{\pi}^*}(s_0)$ from agents $i = 2, 3, \ldots, n$ leads to a total value for agent 1 of

$$V_1^{\boldsymbol{\pi}^*}(s_0) + \sum_{i=2}^{n} V_i^{\boldsymbol{\pi}^*}(s_0) - V_i^{\boldsymbol{\pi}}(s_0) = W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^{n} V_i^{\boldsymbol{\pi}}(s_0),$$

and makes all agents $i = 2, 3, \ldots, n$ indifferent between accepting and rejecting the contract, as they get value $V_i^{\boldsymbol{\pi}^*}(s_0) + V_i^{\boldsymbol{\pi}}(s_0) - V_i^{\boldsymbol{\pi}^*}(s_0) = V_i^{\boldsymbol{\pi}}(s_0)$, which is the value they would get without the contract. As there is no contract in which agent 1 can get more value, this shows that agents are acting selfishly optimally in states $(\mathbf{0}, \mathbf{0})$ and $(\mathbf{0}, \boldsymbol{\theta})$. By assumption, agents are acting optimally in state $(s, \mathbf{0})$, as they are following an SPE π of M. We assume any play in the irrelevant, because never reached in the course of play, states $(s, \boldsymbol{\theta})$, $\boldsymbol{\theta} \neq \boldsymbol{\theta}^*$. It remains to show that $\boldsymbol{\pi}^*$ is an SPE in states $(s, \boldsymbol{\theta}^*)$ for some $\boldsymbol{\theta}^*$ yielding values as above. Let $\boldsymbol{\theta}^*$ be such that if $\mathbf{a}_t = \boldsymbol{\pi}^*(s_t)$, no transfers are made. This means, that the values of accepting resp. proposing $\boldsymbol{\theta}^*$ are indeed as written above. If a single agent takes an action $\mathbf{a}'_{it} \neq \boldsymbol{\pi}_i(s_t)$, then the agent transfers a reward $R_{\max}/(1-\gamma)$, which is the maximal value they can get in the course of the game. It is a best response to play $\mathbf{a}_{it} = \boldsymbol{\pi}_t(s_t)$.

Lastly, assume that all socially optimal contracts $\boldsymbol{\theta}^*$ that give agent 2 value $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0)$ are rejected by at least one agent $i = 2, 3, \ldots, n$. We show that in this case, the best response for agent i is not well-defined by showing that there are contracts that give agent 1 value arbitrarily close to $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0)$, but this value is never reached. In particular, consider the contracts considered above in which agents only transfer $V_i^{\boldsymbol{\pi}}(s_0) - V_i^{\boldsymbol{\pi}^*}(s_0) - \frac{\varepsilon}{n-1}$ to the agent, which then receives value $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}(s_0) - \varepsilon$. Note that agents $i = 2, 3, \ldots, n$ receive value $V_i^{\boldsymbol{\pi}^*}(s_0) + V_i^{\boldsymbol{\pi}}(s_0) - V_i^{\boldsymbol{\pi}^*}(s_0) + \frac{\varepsilon}{n-1} = V_i^{\boldsymbol{\pi}} + \frac{\varepsilon}{n-1}$, strictly incentivizing them to accept this contract. Hence, agent 1 get a value arbitrarily close to $W^{\boldsymbol{\pi}^*}(s_0) - \sum_{i=2}^n V_i^{\boldsymbol{\pi}}$, but not exactly this value.

As outlined in the text, under detectable deviators, $\boldsymbol{\theta}(s, \mathbf{a})$ can be determined only depending on the current and next state, which is a space of dimension $|S|^2$. □

Next, we prove that the statements of Theorem 3.1 continue to hold if we have an agent i that repeatedly proposes contracts. For a Markov game M, we say that a state $s \in S$ is unreachable from state s_0 is there is no sequence of action profiles $a_1, a_2, \ldots, a_{t-1} \in \mathbf{A}$ such that $s_t = s$.

Proposition A.1. *Consider a general formulation in the sense of Appendix B such that there is an agent i such that $(\mathbf{0}, j)$, $j \neq i$ is unreachable from $(\mathbf{0}, i)$, and that $(s, \mathbf{0})$ is unreachable from $(s', \boldsymbol{\theta})$ for any $s, s' \in S$, $\boldsymbol{\theta} \in \Theta$. Then, the conclusion of Theorem 3.1 continue to hold.*

Proof. First, observe that the strategies in which agent i repeatedly proposes a contract $\boldsymbol{\theta}^*$ as in the proof of Theorem 3.1 is an SPE: At each proposal and acceptance state, the agent can assume that the value is as if the agent proposes an efficient contract in the next proposal state. As in previous proofs more informally, we invoke the single-deviation principle for SPE (comparee [25]) stating that to check an SPE, it is sufficient to only consider deviations by one agent at a time. First, observe that there is no reason for the proposer to propose another contract to the other agents, as the upper bound on value demostrated in Theorem 3.1 continues to hold. The forcing contracts still exist and incentivize all agents to follow $\boldsymbol{\pi}^*$. Finally, rejection of all contracts is again not possible as the proposing agent's best response would not be defined. Hence, the statement continues to hold for a repeatedly proposing agent.

The uniqueness of such an equilibrium follows equivalently, as an upper bound on the proposal agent is exactly attained at each proposal point. □

Finally, we show by means of an example that this statement breaks if multiple agents may propose a contract.

Example A.2. Consider a repeated Prisoner's Dilemma in which agent 1 may propose as long as (C, D) has been played. Then, agent 2 proposes thereafter. Consider Prisoner's Dilemma, Figure 2a. An SPE of this game involves proposal of a forcing contract on (C, D) with values after a signing bonus that leads to values $(-2+\varepsilon, -1-\varepsilon)$. All contracts that give agent 1 more value are rejected. It is a best response for agent 2 to accept, as long as $(-1-\varepsilon)/(1-\gamma)$, the value of accepting this contract, is larger than $-2 + 0 \times \gamma/(1-\gamma) = -2$. The best such contract is the one with the largest ε for which this inequality holds.

B CONTRACTING AUGMENTATIONS WITH CONDITIONAL PROPOSALS

In this section, we present the augmentation for several proposing agents. Let $M = \langle S, s_0, \mathbf{A}, T, \mathbf{R}, \gamma \rangle$ be an n-agent Markov game, Θ be a contract space as before. To generalize, we require one additional component: define the *contracting initiation dynamics* to be a function

$$P\colon \{(\boldsymbol{\Theta} \cup \{\mathbf{0}\}) \times S \times \mathbf{A}\} \cup \{\text{init}\} \to \Delta([n] \cup \{\mathbf{0}\} \cup \{\bullet\})$$

which determine whether or not a new contract phase begins or ends at a given state of M, and if one begins, which agent is proposing the contract. More concretely, sampling from $P(\text{init})$ at the start of an episode, or from $P(\boldsymbol{\theta}, s, a)$ at state-action $(s, \mathbf{a})$ under contract $\boldsymbol{\theta}$, either:

(1) Agent i is given the option to propose a new contract, and the game is frozen ($i \in [n]$);
(2) None of the agents proposes a contract, but the current contract stays in force, and the game continues ($\mathbf{0}$), or;
(3) The current contract becomes void, and no agent proposes a new contract, and again the game continues ($\bullet$)

We again assume that the contracting space Θ contains the null contract $\mathbf{0}(s, \mathbf{a}) \equiv 0$. Define the contract-augmented Markov game $M^{\boldsymbol{\Theta}} = \langle S', S_0, \mathbf{A}', T', \mathbf{R}', \gamma \rangle$ as

$$\begin{aligned}
S' &= (S \cup (S \times \{\mathbf{0}\})) \times (\boldsymbol{\Theta} \cup [N]) \\
A' &= A \cup \boldsymbol{\Theta} \cup \{\text{acc}\} \\
T'((s, \theta), a) &= \begin{cases} T(s, \mathbf{a}) & P(\boldsymbol{\theta}, s, \mathbf{a}) = \mathbf{0} \\ (s, \mathbf{0}) & P(\boldsymbol{\theta}, s, \mathbf{a}) = \bullet \\ (\mathbf{0}, i) & P(\boldsymbol{\theta}, s, \mathbf{a}) = i \end{cases} \\
T'(((s, \mathbf{0}), i), (\boldsymbol{\theta}, \mathbf{a}_{-i})) &= ((s, \mathbf{0}), \boldsymbol{\theta}) \\
T'(((s, \mathbf{0}), \boldsymbol{\theta}), \mathbf{a}) &= \begin{cases} (s, \theta) & \mathbf{a} = \mathbf{acc} \\ (s, \mathbf{0}) & \text{else.} \end{cases} \\
R((s, \theta), \mathbf{a}) &= R(s, \mathbf{a}) + \theta(s, \mathbf{a}).
\end{aligned}$$

with S_0 being state $((s_0, \mathbf{0}), i)$ (e.g. agent i proposing a contract at the start of an episode) with probability $\mathbb{P}[P(\text{init}) = i]$ and $(s_0, \mathbf{0})$ otherwise. Note that, in contrast to the definition of S' in subsection 2.2, we use tuples $(s, \mathbf{0})$ more generally to denote cases where state s is frozen in the game while agents are negotiating a contract (which can now happen during the game, and not just at the start). For example, to capture the contracting initiation dynamics of subsection 2.2 in this general model, we have $P(\text{init}) = 1$ almost surely, and $P(\cdot) = \mathbf{0}$ almost surely for any other state (i.e., contracting is initiated and proposal rights given to agent 1 with certainty at the start of an episode, and that contract is held in force for the remainder of the episode).

More complex augmentations can be contructed in a similar way. Examples of such augmentations are multiple contracts in force at the same time, majority of acceptance as opposed to unanimity, and contract overriding only if new contracts are accepted. Exploring such augmentations is an avenue for future work.

C HYPERPARAMETER SETTINGS

In this section, we describe the hyperparameters we experimented with in the process of developing the study of section 5, and emphasize the ones we ended up using in the final results in Figure 6.

Optimization Details. In all methods, we used a learning rate of $\alpha = 10^{-4}$. Experimentation with $\alpha = 10^{-2}, 10^{-6}, 10^{-8}, 10^{-10}$, as well as a linearly decaying learning schedule from $\alpha = 10^{-2}$ to 10^{-10}, none of which improved performance on any method, the smaller learning rates significantly degrading the sample efficiency of the methods involved.

Training happened using stochastic gradient descent, with 30 SGD updates per training iteration and a momentum of 0.99. For all methods second phase of MOCA, training batches consisted of 12000 sampled timesteps for the static dilemmas, and 120000 sampled timesteps for the dynamic games, all of them with minibatches of size 4092. Other minibatch values attempted for these methods were 128, 256, 1024 and 40000, all of which were found to degrade the performance, the first two very significantly. However, in training $\boldsymbol{\pi}(s, \boldsymbol{\theta})$, we restricted ourselves to 10% of timesteps used in training $\boldsymbol{\pi}(\mathbf{0}, \mathbf{0})$ (and the baseline algorithms), and therefore found that sampling 128 episodes per training iteration, and using a minibatch size of 128 (in all domains) was sufficient for strong performance.

Model Architectures. On all algorithms, we experimented with 32×32, 64×64, 256×256, $64 \times 64 \times 64$, $256 \times 256 \times 256$, and 1024×1024 MLP networks, and took the models of minimal complexity attaining maximal performance. For separate training, gifting, and contracting, we found 64×64 MLP networks to be this value, while for joint training, due to the added complexity of the joint policy, we found that expanding the network to a 256×256 MLP architecture yielded the highest performance.

Environment Horizon and Discount Factors. All domains have discount factor $\gamma = 0.99$. Moreover, the maximum horizon for the matrix domains was 2, for the Public Goods Dilemma was 100, for Emergency Merge was 200 (although episodes can terminate earlier by cars reaching the end early), and for both Harvest and Cleanup was 1000.

PPO-specific parameters. We used a standard KL-coefficient of 0.2, KL-target of 0.01, clip parameter of 0.3, value function coefficient of 1.0, entropy coefficient of 0.0, for all experiments in all domains.

Novel hyperparameters. As mentioned in the main text, we introduced a novel hyperparameter $\boldsymbol{\nu}$, governing the number of agents sampled in the accept-reject decision for contracts. In all experiments, we used $\boldsymbol{\nu} = 2$.

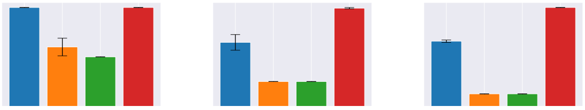
Other unmentioned hyperparameters are as found in the default settings in the RLLib implementation of PPO [18].

D ADDITIONAL EXPERIMENTS

In the Stag Hunt game (see Figure 8 for a game table for two players), we observe a similar pattern as in the other studied games, outperforming separate training and gifting in the 2, 4, 8 agent cases. Additionally, in particular for larger domains, contracting outperforms joint training, due to hardness of exploration of the joint action space. Gifting does not even outperform separate training in this domain.

To scale this game to higher number of players, in general all agents get payout equal to the number of agents if all cooperate, and payout equal to 1 if all defect. If agents both cooperate and defect, then the defecting agents get $n_{agents} - 1$ and cooperating agents get 0.

See Figure 7 for experimental details.



(a) 2 agents

(b) 4 agents

(c) 8 agents

Figure 7: Empirical Results for Stag Hunt. As in the case for Prisoner's Dilemma, contracting outperforms both separate and joint training, and always performs as least as well as joint training, significantly outperforming it for high agent counts. As in the Prisoner's Dilemma case in Figure 6, 5 trials were conducted, algorithms run for 1M timesteps, and the standard error across these trials is reported in the error bars.

$$\begin{array}{c|c|c|}
 & C & D \\
\hline
C & 2, 2 & 0, 1 \\
\hline
D & 1, 0 & 1, 1 \\
\hline
\end{array}$$

Figure 8: Stag Hunt