OFFLINE EQUILIBRIUM COMPUTATION IN CONVEX MARKOV GAMES

Anonymous authors
Paper under double-blind review

ABSTRACT

Convex Markov games extend Markov games to utilities beyond expected returns, capturing essential notions such as fairness, safety, and imitation. This expressiveness stems from evaluating the global game status through concave functions of state–action occupancies. However, this non-linearity makes policy updates dependent on the current occupancy, formalized as a shadow reward. Consequently, computing equilibria of convex Markov games requires tracking the current occupancy at every update, through fresh environment interactions or an exact transition model, making the process computationally expensive. We ask whether such equilibria can instead be approximated purely from logged multi-agent play. We propose cMG-DICE, which combines the shadow reward formulation with stationary distribution correction estimation (DICE) so that each agent's best response reduces to a single minimization over logged transitions. We demonstrate the efficacy of this framework by approximating equilibria under diverse utilities and initial joint policies on tabular games and the Clean Up environment from a single dataset, bypassing the need to restart costly online optimization from scratch for each new equilibrium.

1 INTRODUCTION

Convex Markov games (cMGs) (Gemp et al., 2025) broaden the scope of multi-agent reinforcement learning by defining agent utilities as concave functions of state–action occupancies. Because the state–action occupancy characterizes the long-term visitation distribution over states and actions, it encapsulates the global status of the game. Within this view, expected returns are linear in the occupancy, and concave utilities extend them to objectives such as equitable returns across agents, safety, imitation, and behavioral diversity. This non-linearity is also what makes the optimization expensive: the reward that guides a policy update is the gradient of the utility at the current occupancy, known as the *shadow reward* (Zhang et al., 2022), and it changes with every update. Existing approaches track the current occupancy in two ways, and both are costly: policy-space methods re-evaluate the shadow reward through on-policy interaction (Zhang et al., 2020; Barakat et al., 2026; Kalogiannis et al., 2025), whereas occupancy-space methods require a known transition model to project updates onto valid distributions (Gemp et al., 2025).

In multi-agent settings, this cost is multiplied as the update is iterated across agents: on the interaction side, the joint policy of all agents has to be rolled out anew after every update, and on the model side, the transition model must cover a joint state–action space that grows with the number of agents. In addition, since the shadow reward depends on the current occupancy, data collected under earlier policies cannot be used for the next update. The cost also recurs across runs: whenever one asks for a different Nash equilibrium of the same utility, or for the equilibrium of a different utility, existing approaches have to start learning from scratch, and cannot use the logged multi-agent play that is often already available from prior evaluations. This motivates the central question of our work: *can equilibria of a convex Markov game be approximated purely from pre-collected data, without further interaction with the environment or reliance on a transition model?*

To answer this question, we introduce cMG-DICE, short for offline convex Markov games via stationary distribution correction estimation. cMG-DICE solves a game whose utilities are regularized toward the distribution of the dataset, so that its equilibria approximate the Nash equilibria of the target game. By applying Fenchel duality to the concave utility and Lagrangian duality to the occu-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

pancy constraint, cMG-DICE turns each agent's best response in this game into a single minimization over logged transitions, in which the shadow reward is estimated off-policy and the policy is optimized under it. This removes the on-policy rollouts and the model-based occupancy projections that existing approaches require. It also removes the alternation between re-evaluating the shadow reward and updating the policy within each agent's best response.

To validate our approach, we identify which factors decide the equilibrium that cMG-DICE reaches, bound how far it can be from that of the target game, and sweep these factors on a fixed dataset, on tabular games and on the seven-agent Clean Up domain (Guo et al., 2025). Since the distribution ratio learned by cMG-DICE evaluates arbitrary rewards (Nachum et al., 2019a), we also check the learned shadow rewards and estimate the returns of the reached equilibrium before deployment.

Our contributions are threefold: (i) cMG-DICE, which approximates the Nash equilibria of a convex Markov game from logged data alone; (ii) an analysis of what decides the equilibrium it reaches and a bound, in terms of the regularization strength, on how far that can be from the target game; (iii) experiments on tabular games and Clean Up showing that one dataset yields different equilibria of the target game as the utility and the initial joint policy change.

2 PRELIMINARIES

Convex Markov games and state–action occupancy. An n-agent Markov game is defined by the tuple $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, P, \{r_i\}_{i\in\mathcal{N}}, p_0, \gamma\rangle$, where $\mathcal{N} = \{1, \dots, n\}$ is the set of agents, $\mathcal{S}$ is the state space, $\mathcal{A} = \times_i \mathcal{A}_i$ is the joint action space with joint actions $a = (a_1, \dots, a_n)$, $P(s'|s, a)$ is the transition kernel, $r_i(s, a)$ is agent i's reward function, p_0 is the initial state distribution, and $\gamma \in (0, 1)$ is the discount factor. We focus on decentralized joint policies $\pi(a|s) = \prod_i \pi_i(a_i|s)$, denoting by π_{-i} and a_{-i} the policies and actions of all agents other than i.

The state–action occupancy measure of a joint policy π, $d^\pi(s, a) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s, a_t = a)$, is the discounted distribution of the states and joint actions that π visits over an infinite horizon. It satisfies the Bellman flow constraint:

$$\sum_{a\in\mathcal{A}} d(s', a) = (1-\gamma)p_0(s') + \gamma \sum_{s,a} P(s'|s, a)d(s, a) \qquad \forall s' \in \mathcal{S}. \tag{1}$$

The constraint balances the occupancy entering and leaving each state. The non-negative measures that satisfy it form the flow polytope Ω, and every $d \in \Omega$ is the occupancy of the joint policy $\pi(a|s) = d(s, a)/\sum_{a'} d(s, a')$. Throughout, functions on $\mathcal{S} \times \mathcal{A}$ are treated as vectors, $\langle\cdot, \cdot\rangle$ is the inner product, and products of such functions are pointwise.

Many objectives are functionals of the occupancy; the expected return $J_i(\pi) = \langle r_i, d^\pi\rangle$ of a standard Markov game is the linear one. A convex Markov game (cMG) (Gemp et al., 2025) lets agent i's utility be any concave function $u_i(d^\pi)$ of the occupancy; examples include the following:

$$\begin{aligned} &\text{Expected Return: } \langle r, d^\pi\rangle && \text{Safety: } -\textstyle\sum_{(s,a)\in\mathcal{U}} d^\pi(s, a) \\ &\text{Entropy: } \mathcal{H}(d^\pi) && \text{Imitation: } -D_{\mathrm{KL}}(d^\pi \,\|\, d^E) \\ &\text{Welfare: } F\big(J(d^\pi)\big) \end{aligned} \tag{2}$$

All of these are concave in the occupancy and can be combined additively. The safety objective penalizes visits to a set $\mathcal{U}$ of unsafe state–action pairs, $\mathcal{H}$ is the entropy, and d^E is the occupancy of an expert to imitate. In the welfare objective, $J(d) = (\langle r_1, d\rangle, \dots, \langle r_n, d\rangle)$ is the vector of per-agent expected returns and F is concave and non-decreasing; since J is affine in d, $F(J(d))$ is concave in d. F aggregates the per-agent returns as a social welfare function does, and a strictly concave F, having steeper gradients at lower returns, weights gains to the agents that are behind more heavily.

Nash equilibrium in cMGs. The solution concept of a cMG is the Nash equilibrium, a joint policy π^* at which no agent can improve its utility by deviating alone. Checking this condition for agent i is a best-response problem with the opponents held fixed:

$$\max_{d_i \in \Omega_i(\pi_{-i})} u_i(d_i \otimes \pi_{-i}), \tag{3}$$

where $d_i(s, a_i) = \sum_{a_{-i}} d(s, a)$ is agent i's marginal occupancy, the state–action occupancy of the MDP $\bar{M}_i^{\pi_{-i}}$ to which the game reduces for agent i given π_{-i}; $\Omega_i(\pi_{-i}) = \{d_i \geq 0 : d_i \otimes \pi_{-i} \in$

$\Omega\}$ is the reduced flow polytope; and $(d_i \otimes \pi_{-i})(s,a) = d_i(s,a_i)\pi_{-i}(a_{-i}|s)$ recovers the joint occupancy. To measure how close a joint policy π is to a Nash equilibrium, we use its *exploitability*, the largest utility gain any agent can obtain by deviating to its best response; it is zero exactly at a Nash equilibrium:

$$\text{Exploit}(\pi) = \max_{i \in \mathcal{N}} \left[\max_{d_i \in \Omega_i(\pi_{-i})} u_i(d_i \otimes \pi_{-i}) - u_i(d^\pi) \right]. \tag{4}$$

DICE-based offline RL and AlberDICE. Offline RL targets settings in which further interaction with the environment is costly or unsafe, and optimizes a policy from a fixed dataset $\mathcal{D}$ of logged transitions alone (Levine et al., 2020), whose empirical state–action distribution we denote by d^D. One line of work casts offline RL as a convex program over the state–action occupancy. DICE-based offline RL (Lee et al., 2021), built on stationary distribution correction estimation, maximizes an objective of the occupancy regularized toward d^D under the Bellman flow constraint (1), and turns this constrained problem into its Lagrangian dual, which is minimized over samples from d^D. The same construction has been carried out for particular concave objectives in the single-agent setting (Kim et al., 2022b;a; 2024; 2025; Lee et al., 2025).

DICE-based offline RL has also been extended to multi-agent settings: AlberDICE (Matsunaga et al., 2023) considers a cooperative game, in which all agents share one linear objective $\langle r, d\rangle$, and each agent computes its best response (3) in turn. Its regularization is placed on the joint occupancy $d_i \otimes \pi_{-i}$ rather than on agent i's own marginal: with the opponents fixed, agent i solves

$$\max_{d_i \in \Omega_i(\pi_{-i})} \langle r,\ d_i \otimes \pi_{-i} \rangle - \alpha\, D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D), \tag{5}$$

so that a deviation is judged together with the opponents' play, and no agent proposes a joint action the data does not support, even though the agents update one at a time. The derivations of these methods, and how they relate to cMG-DICE, are given in Appendix B.

3 From On-Policy Shadow Rewards to an Offline Best Response

In this section, we identify the challenges of computing a Nash equilibrium of a convex Markov game, which arise from the shadow reward, and derive an offline alternative. Section 3.1 shows that the shadow reward, the gradient of each agent's utility with respect to the joint occupancy, moves with the occupancy it is evaluated at, so every update requires on-policy interaction with the environment or a transition model. Section 3.2 proposes a convex Markov game regularized toward the data, in which Fenchel and Lagrangian duality turn an agent's best response into a single minimization over logged transitions; this extends DICE-based offline MARL from a common linear reward to agents with different concave objectives.

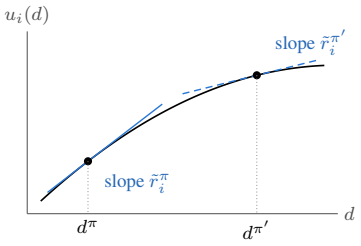

Figure 1: The shadow reward $\tilde{r}_i^\pi = \nabla u_i(d^\pi)$ is the slope of u_i at the current occupancy, and changes as the policy moves from π to π'.

3.1 Why Concavity Makes Equilibrium Computation Expensive

Shadow reward. What separates a convex Markov game from a standard Markov game is that the utilities are concave in the state–action occupancy rather than linear in it, and the distinction is captured by the *shadow reward* (Zhang et al., 2020; 2022). Given a joint policy π and the joint occupancy d^π it induces, the shadow reward of agent i is given by differentiating its utility at that occupancy,

$$\tilde{r}_i^\pi \ := \ \left[\nabla_d\, u_i(d)\right]_{d=d^\pi}. \tag{6}$$

It is termed a reward because the gradient of the expected return $u_i(d) = \langle r_i, d\rangle$ is the reward r_i itself; for a concave utility, the gradient plays the same role at the current joint policy. The difference is that it is valid only at d^π and has to be re-evaluated at every update (Figure 1). For example, the shadow reward of the Nash social welfare of the returns, $u_i(d) = \sum_j \log J_j(d)$ for

all i, is $\tilde{r}^\pi = \sum_j r_j / J_j(d^\pi)$: each agent's reward is weighted by the inverse of that agent's current return, and the weights shift as the policy is updated.

Rather than evaluating the shadow reward at the current occupancy, one can treat it as a variable to be optimized (Zahavy et al., 2021). Applying the Fenchel conjugate turns agent i's best response (3) into the saddle problem

$$\max_{d_i \in \Omega_i(\pi_{-i})} u_i(d_i \otimes \pi_{-i}) \;=\; \max_{d_i \in \Omega_i(\pi_{-i})} \min_{\tilde{r}_i} \Big[\langle \tilde{r}_i,\, d_i \otimes \pi_{-i} \rangle - u_i^*(\tilde{r}_i) \Big], \tag{7}$$

where $u_i^*(\tilde{r}_i) := \min_{d \geq 0} \big[\langle \tilde{r}_i, d \rangle - u_i(d) \big]$ is the concave conjugate of u_i. For a fixed $\tilde{r}_i$, the outer problem is an ordinary return maximization, and for a fixed occupancy, the inner minimizer is the shadow reward at that occupancy. Either way, the shadow reward has to be obtained at the current occupancy, and the cost depends on the domain in which the update is taken.

Policy and occupancy domains. In the policy domain, the shadow reward is estimated from trajectories of the current joint policy before a policy-gradient step is taken (Zhang et al., 2020; Zahavy et al., 2021; Barakat et al., 2026; Kalogiannis et al., 2025); since the step moves the occupancy, the next step needs new trajectories, and data collected under earlier policies cannot be used. In the occupancy domain, the shadow reward is computed directly from the current occupancy, $\tilde{r}^\pi = \sum_j r_j / J_j(d^\pi) = \sum_j r_j / \langle r_j, d^\pi \rangle$ in the example above, and the occupancy is updated along it. The updated occupancy, however, generally leaves the reduced flow polytope $\Omega_i(\pi_{-i})$, and returning to it means projecting the iterate back onto the flow constraint (1), which requires the transition kernel (Gemp et al., 2025).

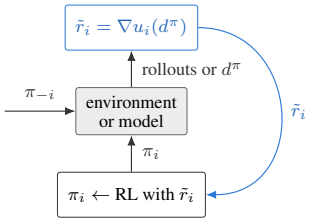

Figure 2: Best response of agent i with π_{-i} fixed: each policy update moves the occupancy, so the shadow reward has to be evaluated again.

How a game compounds the cost. The shadow reward is already costly in a convex MDP, the single-agent counterpart of agent i's best response (Zhang et al., 2020; Zahavy et al., 2021). However, we focus on how a convex Markov game compounds this cost in two ways. Within a run, the best responses are iterated over the agents until the joint policy converges, and each of them re-evaluates a shadow reward that depends on the joint occupancy, whose rollouts or transition model span the joint action space of all agents. Across runs, a convex Markov game generally admits several Nash equilibria, and which one is reached depends on the initial joint policy and on the utility; whenever either changes, the computation starts over, since the logged play of earlier runs was collected under other joint policies and a shadow reward evaluated on-policy cannot use it. Section 3.2 proposes an offline alternative that estimates the shadow reward from logged play.

3.2 cMG-DICE: Offline Approximation of cMG Equilibria

Regularized convex Markov game. To enable offline optimization, cMG-DICE solves a regularized convex Markov game in place of the target game: every agent's utility carries one common penalty on the deviation of the joint state–action occupancy from the data distribution, $u_i^\alpha := u_i - \alpha\, D_{\mathrm{KL}}(\cdot \,\|\, d^D)$, and the best response of agent i becomes

$$\max_{d_i \geq 0} \; u_i(d_i \otimes \pi_{-i}) - \alpha\, D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D) \quad \text{s.t.} \quad (\mathcal{B} d_i)(s) = 0 \;\; \forall s \in \mathcal{S}. \tag{8}$$

The operator $\mathcal{B}$ collects the Bellman flow constraint (1) of the reduced MDP $\bar{M}_i^{\pi_{-i}}$, whose kernel is $\bar{P}(s'|s,a_i) = \sum_{a_{-i}} \pi_{-i}(a_{-i}|s) P(s'|s,a)$, into the residual

$$(\mathcal{B} d_i)(s') := (1-\gamma) p_0(s') + \gamma \sum_{s,a_i} \bar{P}(s'|s,a_i)\, d_i(s,a_i) - \sum_{a_i} d_i(s',a_i), \tag{9}$$

so that the feasible set is $\Omega_i(\pi_{-i})$. The regularized problem is also a convex Markov game, since $D_{\mathrm{KL}}(\cdot \,\|\, d^D)$ is a convex function of the state–action occupancy. The cost of the regularization is paid in target-game exploitability, as an equilibrium of the regularized problem only approximates one of the target game.

Proposition 1 (the price of data regularization)**.** *Write* $\textit{Exploit}_\alpha(\pi)$ *for the exploitability (4) under the regularized utilities* u_i^α*, and let* π^α *satisfy* $\textit{Exploit}_\alpha(\pi^\alpha) \le \epsilon$*. Let* $d_i^\star \in \arg\max_{d_i \in \Omega_i(\pi_{-i}^\alpha)} u_i(d_i \otimes \pi_{-i}^\alpha)$ *be agent* i*'s best response in the target game. Then*

$$\textit{Exploit}(\pi^\alpha) \;\le\; \epsilon + \alpha \max_i \Big[D_{\mathrm{KL}}\big(d_i^\star \otimes \pi_{-i}^\alpha \,\|\, d^D\big) - D_{\mathrm{KL}}\big(d^{\pi^\alpha} \,\|\, d^D\big) \Big],$$

so the price of the regularization is α *times the excess data divergence of the best response. If the data covers every state–action pair, the bracket is at most* $C := \log\big(1/\min_{s,a} d^D(s,a)\big)$*, and* $\textit{Exploit}(\pi^\alpha) \le \epsilon + \alpha C$ *(Appendix C).*

Decreasing α lowers this bound, but empirically the exploitability itself does not decrease monotonically (Section 4.1).

Shadow reward meets DICE. To solve the regularized best response (8), we reuse the saddle form (7) with one change: instead of applying the Fenchel conjugate to the entire objective, we apply it to u_i alone and leave the divergence in the primal. Exchanging the order of optimization under Sion's minimax theorem then turns (8) into a min–max problem whose inner maximization is a regularized return maximization under the shadow reward $\tilde{r}_i$:

$$\begin{aligned} &\max_{d_i \in \Omega_i(\pi_{-i})} \min_{\tilde{r}_i} \Big[\langle \tilde{r}_i, d_i \otimes \pi_{-i} \rangle - u_i^*(\tilde{r}_i) - \alpha\, D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D) \Big] \\ &\quad = \min_{\tilde{r}_i} \Big\{ - u_i^*(\tilde{r}_i) + \max_{d_i \in \Omega_i(\pi_{-i})} \big[\langle \tilde{r}_i, d_i \otimes \pi_{-i} \rangle - \alpha\, D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D) \big] \Big\}. \end{aligned} \tag{10}$$

Although the conjugate is taken of u_i, the shadow reward of the regularized game is the gradient of u_i^α, $\tilde{r}_i - \alpha\big(\log\big((d_i \otimes \pi_{-i})/d^D\big) + 1\big)$; the second term penalizes each joint state–action pair by its log-ratio to the data, which keeps the policy where the data support it and, in a cMG, where the shadow reward, itself estimated from the data, is accurate.

Shadow reward and policy from one minimization. With the shadow reward $\tilde{r}_i$ fixed, the inner maximization of (10) is the problem AlberDICE (Matsunaga et al., 2023) solves for a common reward, and we follow its derivation. Attaching the Lagrange multiplier $\nu \in \mathbb{R}^{\mathcal{S}}$ of the Bellman flow constraint and exchanging the max and min by strong duality, splitting the joint divergence as $D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D) = D_{\mathrm{KL}}(d_i \,\|\, d_i^D) + \mathbb{E}_{d_i}[\mathrm{KL}_{-i}]$ with $\mathrm{KL}_{-i}(s,a_i) = D_{\mathrm{KL}}(\pi_{-i}(\cdot|s) \,\|\, \pi_{-i}^D(\cdot|s,a_i))$, and collecting the coefficient of each $d_i(s,a_i)$ leaves

$$\begin{aligned} &\min_{\nu} \max_{d_i \ge 0} \; (1-\gamma)\, \mathbb{E}_{p_0}[\nu] + \sum_{s,a_i} d_i(s,a_i)\, e_{\nu,\tilde{r}_i}(s,a_i) - \alpha\, D_{\mathrm{KL}}(d_i \,\|\, d_i^D), \\ &e_{\nu,\tilde{r}_i}(s,a_i) = \mathbb{E}_{(a_{-i},s') \sim D|s,a_i} \Big[\xi_i(s,a)\big(\tilde{r}_i(s,a) + \gamma \nu(s')\big) \Big] - \alpha\, \mathrm{KL}_{-i}(s,a_i) - \nu(s), \end{aligned} \tag{11}$$

where $\xi_i(s,a) = \pi_{-i}(a_{-i}|s)/\pi_{-i}^D(a_{-i}|s,a_i)$ moves the logged opponents to the current ones. Apart from the divergence, the objective is linear in d_i, so the inner maximization has the closed-form solution $d_i^* = w\, d_i^D$ with $w = \exp(e_{\nu,\tilde{r}_i}/\alpha - 1)$. Substituting it eliminates d_i, and the offline best response of agent i becomes

$$\min_{\tilde{r}_i,\, \nu} \; -u_i^*(\tilde{r}_i) + (1-\gamma)\, \mathbb{E}_{p_0}[\nu] + \alpha\, \mathbb{E}_{(s,a_i) \sim d_i^D} \Big[\exp\Big(\tfrac{e_{\nu,\tilde{r}_i}(s,a_i)}{\alpha} - 1 \Big) \Big]. \tag{12}$$

The shadow reward is what makes this reduction possible for a concave utility: the closed-form maximizer w, on which DICE rests, exists only when the objective is linear in the occupancy apart from the divergence, and conjugating u_i alone restores this linearity, since u_i enters (12) only through $\langle \tilde{r}_i, \cdot \rangle$ and $-u_i^*(\tilde{r}_i)$. DICE methods for particular concave objectives, such as imitation (Kim et al., 2022b), state entropy (Lee et al., 2025), and a welfare of the returns (Kim et al., 2025), each derive a reduction tailored to the form of $u(d)$; the shadow reward gives them a common explanation and extends the reduction to arbitrary concave utilities and to a game. For a general concave utility, $\tilde{r}_i$ is a function of the state and the joint action and has to be represented by a function approximator over $\mathcal{S} \times \mathcal{A}$, whereas particular utilities admit a far smaller parameterization or none. For the welfare objective in (2), $u_i(d) = F_i(J(d))$, since J is affine in d, the shadow reward stays in the span of the base rewards, $\tilde{r}_i = \sum_j \mu_{ij} r_j$, so agent i trains only the n weights $\mu_{i1}, \ldots, \mu_{in}$. At the minimizer

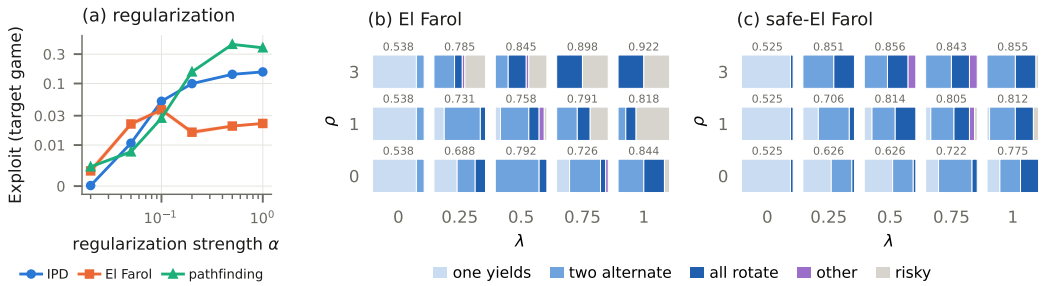

Figure 3: cMG-DICE on tabular games from 10^4 transitions of uniform play. **(a)** Target-game exploitability against α (mean and standard error over five seeds; $\rho = 0$, and $\lambda = 0$ except $\lambda = 0.75$ for El Farol). **(b)** El Farol and **(c)** safe-El Farol at $\alpha = 0.02$: share of runs reaching each type of equilibrium, with the ratio $\min_j J_j / \max_j J_j$ averaged over twenty seeds above each bar.

of (12), these weights equal $\nabla F_i(J)$ at the returns of the joint occupancy $(w\, d_i^D) \otimes \pi_{-i}$ that the solution itself induces, rather than at the returns $J^{\mathcal{D}}$ of the data, which Section 4.4 validates.

The policy that induces the optimal occupancy $w\, d_i^D$ is then extracted by weighted behavior cloning, $\pi_i(a_i|s) \propto w(s, a_i)\, d_i^D(s, a_i)$. Iterating this best response over the agents until convergence gives cMG-DICE (convex Markov games via stationary distribution correction estimation; Algorithm 1). The derivation and training details are given in Appendix B.

4 Nash Equilibria of Convex Markov Games with cMG-DICE

We study cMG-DICE through four questions, combining theoretical analysis with experiments on tabular games from Gemp et al. (2025): the iterated prisoner's dilemma, El Farol, and pathfinding.

Q1. How well does the offline policy of cMG-DICE approximate a target-game Nash equilibrium?
Q2. How does the concave utility move the equilibrium that cMG-DICE reaches?
Q3. Which of the equilibria a utility admits does the initial joint policy select?
Q4. Does cMG-DICE track the shadow rewards, and can the reached returns be predicted offline?

Offline cMG setting. In each game, the dataset consists of 10^4 transitions of a uniformly random joint policy, collected without prior knowledge of the game, and cMG-DICE uses only these samples, without a transition model or online interaction. To see how the utility moves the equilibrium cMG-DICE reaches, we use a family of target utilities $u_i(d) = F_i(J(d))$ with two interpretable parameters, the weighted power mean of the nonnegative returns offset by a small $\varepsilon > 0$,

$$F_i(J) = \Big(\sum_j \bar{\lambda}_{ij} \, (J_j + \varepsilon)^{1-\rho} \Big)^{\frac{1}{1-\rho}}, \qquad \bar{\lambda}_{ij} = \frac{\lambda_{ij}}{\sum_l \lambda_{il}}, \qquad \lambda_{ii} = 1, \quad \lambda_{ij} = \lambda \in [0, 1]. \tag{13}$$

Here, λ_{ij} measures how much agent i weighs agent j's return, and ρ how averse agent i is to unequal returns, from the weighted mean of the returns at $\rho = 0$ to their minimum as $\rho \to \infty$. The family contains objectives commonly used in mixed-motive and fair MARL, such as reward exchange (Willis et al., 2025) and welfare-based fairness (Siddique et al., 2020; Zimmer et al., 2021). We sweep $\lambda \in \{0, 0.25, 0.5, 0.75, 1\}$, $\rho \in \{0, 1, 3\}$ and the regularization strength $\alpha \in \{0.02, 0.05, 0.1, 0.2, 0.5, 1\}$ on five seeds per game, and on twenty for El Farol at $\alpha = 0.02$. We evaluate the returned joint policy by its exploitability in the target game, Exploit(π), computed exactly with the known transition kernel; other details are given in Appendix F.

4.1 Approximating the Target-Game Equilibrium

Proposition 1 bounds the target-game exploitability by $\epsilon + \alpha C$, where ϵ is the regularized exploitability that the alternating best responses leave, and how small ϵ can be depends on the target utility. At $\lambda = 1$ every agent maximizes the same utility, which we denote $\bar{F}(J)$, so the regularized game is an exact potential game with potential $\Phi_\alpha := \bar{F}(J) - \alpha D_{\mathrm{KL}}(\cdot \,\|\, d^D)$; for $\lambda < 1$ an improvement of u_i can lower u_j, and the following proposition bounds the resulting ϵ.

Proposition 2 (near-potential structure). *For every λ, $u_i^\alpha = \Phi_\alpha + (F_i - \bar{F})$, and the regularized game is an ϵ_λ-potential game with $\epsilon_\lambda \le (1-\lambda)\, K_{\rho,\varepsilon,n}$ for a constant $K_{\rho,\varepsilon,n}$ depending only on (ρ, ε, n). For any $\delta > 0$, alternating best responses that move only on gains above $\epsilon_\lambda + \delta$ stop in finitely many steps with Exploit$_\alpha(\pi) \le \epsilon_\lambda + \delta$; at $\lambda = 1$, their limit points are Nash equilibria of the regularized game (Appendix C).*

Combined with Proposition 1, Exploit$(\pi^\alpha) \le \epsilon_\lambda + \delta + \alpha C$ under full coverage. Figure 3a reports the exploitability of the returned joint policy. At large α the joint policy stays close to the uniform play of the dataset, and the agents remain exploitable. As α decreases, the exploitability falls in all three games, to at most 0.005 at $\alpha = 0.02$. The decrease is not monotone: on El Farol the exploitability peaks at $\alpha = 0.1$, where the joint policy has moved away from the uniform play of the data but has not yet reached another equilibrium of the target game.

4.2 How the Utility Moves the Equilibrium

We demonstrate how the utility moves the equilibrium that cMG-DICE reaches by varying λ and ρ on El Farol, with the result in Figure 3b. We fix the small $\alpha = 0.02$, so every reached equilibrium is close to one of the target game, with target-game exploitability at most 0.014. We first describe the game and its equilibria.

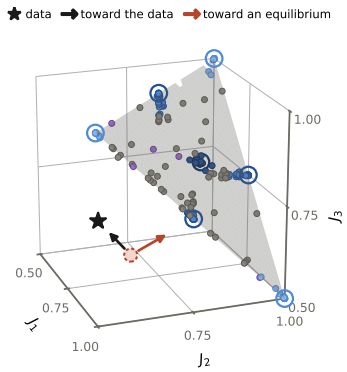

Figure 4: Equilibria reached on El Farol (colors as in Figure 3). A joint policy under optimization (dashed) is pulled toward the data by the regularization and toward an equilibrium by the utility.

In El Farol, three agents decide each round whether to go to a bar or stay home, and going pays only if at most two agents go, so one agent has to stay home in every round. With the state carrying the previous round's actions, cMG-DICE converges to five types of equilibria (Figure 4). Three are conventions that settle who stays home: one agent always yields, two alternate, or all three rotate, at the vertices, edge midpoints and center of the triangle, away from the uniform play of the data. Purple points mix the conventions and lie on the triangle between them, and gray points make a fixed stochastic decision every round.

At $\lambda = 0$ each agent values only its own return, and one agent yields in 17 of 20 seeds, regardless of ρ. As λ grows, the reached equilibria move from the vertices of the triangle toward its edge midpoints and center, since each agent's utility includes the others' returns. A larger ρ moves them further toward the center, to rotation or to stochastic profiles, which are fair in expectation as well; at $\rho = 3$ the mean ratio $\min_j J_j / \max_j J_j$ rises from 0.54 at $\lambda = 0$ to 0.92 at $\lambda = 1$.

Adding safety to the utility. However, stochastic profiles are risky even when they are close to equilibria of the target game: since the agents do not coordinate, some rounds leave two agents at home and occasionally all three go to the bar. A safety penalty in the utility removes them, and cMG-DICE reaches the equilibria of the shifted target from the same datasets, whereas existing approaches would collect new rollouts.

We add to every agent's utility the penalty $-c\,\langle r_{\text{waste}}, d\rangle$ with $c = 0.2$, where $r_{\text{waste}}(s,a) = 1$ if the joint action a leaves at least two agents at home, and call the resulting game safe-El Farol. The penalty acts on joint actions rather than on any single agent's choice, and cMG-DICE accommodates it without modification, since each best response already optimizes over the joint occupancy $d_i \otimes \pi_{-i}$. A stochastic decision now risks the penalty in every round, so the reached equilibria leave the stochastic profiles for the conventions: risky profiles drop from 58 to 4 of 300 runs (Figure 3c), with target-game exploitability below 0.01 in the penalized game.

4.3 How the Initial Joint Policy Selects the Equilibrium

Which of these equilibria cMG-DICE reaches depends on the initial joint policy. In Figure 3b the initial policy is behavior cloning of uniform data, far from every equilibrium, and the convention reached varies with the dataset draw. A start near an equilibrium leaves little of this variation: from a policy in which all three rotate, cMG-DICE stays at rotation under every utility tested, and from

one in which one agent always yields, it stays there at $(\lambda, \rho) = (0, 0)$ and moves to two alternating under the other three utilities tested (Appendix E). Offline, the initial policy must be one the dataset can represent, such as behavior cloning on the data or a subset of it, and from each start cMG-DICE approximates an equilibrium of the target game, with target-game exploitability below 0.012 at $\alpha = 0.02$.

4.4 Tracking the Shadow Rewards and Predicting the Equilibrium

We validate the two estimates on which cMG-DICE rests. First, cMG-DICE optimizes the offline policy under a shadow reward that it estimates off-policy, so the estimated shadow reward should match the shadow reward of the returned joint policy in the environment. For the welfare utilities, we compare the learned weights μ_{ij} with ∇F_i evaluated at the true returns of the returned joint policy. Second, if the learned ratio w is correct, it can be used to evaluate any reward recorded in the data, as in DualDICE (Nachum et al., 2019a): $\hat{J}_j = \mathbb{E}_{d^{\mathcal{D}}}[w\, \xi_i\, r_j]$ predicts the returns of the equilibrium cMG-DICE reaches, which we compare with its true returns. Table 1 reports both over the grid of each game: μ agrees with ∇F to 0.048, and $\hat{J}$ with the true returns to 0.163 in the worst cell and 0.015 in the median, whereas the average returns in the data, $J^{\mathcal{D}}$, differ from them by up to 0.962. The errors grow as α decreases and the joint policy moves further from the data. Given the utility and an initial joint policy, the return profile of the equilibrium that cMG-DICE reaches can thus be approximated from the logged data before deployment.

Table 1: Off-policy evaluation over the 90 cells and five seeds of each game, on returns normalized to $(0, 1]$: worst case, with the median in brackets (μ for $\rho > 0$).

game	cells	KL	$\|J^{\mathcal{D}} - J\|_\infty$	$\|\hat{J} - J\|_\infty$	$\|\mu - \nabla F\|_\infty$
iterated PD	90	2.72	0.179	0.024 (0.001)	0.040 (<0.001)
El Farol	90	4.09	0.375	0.053 (0.004)	0.048 (0.001)
pathfinding	90	3.64	0.962	0.163 (0.015)	0.005 (<0.001)

5 cMG-DICE on a Larger Domain: Clean Up

Section 4 used games small enough for a tabular representation, with data from a uniform policy that covers every state and joint action. Clean Up, a seven-agent sequential social dilemma, removes both: cMG-DICE uses function approximation, and the data, logged from training runs, cover only part of the state–joint-action space, so the bound of Proposition 1 no longer applies. We ask two further questions.

Q5. Does cMG-DICE reach a joint policy less exploitable in the target game than the logged behavior, including at targets absent from the data?
Q6. Does the learned shadow reward suffice on its own, or must it be learned together with the regularization of the joint state–action occupancy?

Environment and data. In Clean Up (Guo et al., 2025), apples, the only source of reward, regrow only while a river is kept clean, and cleaning earns nothing, so each agent prefers that the others clean. We build two offline datasets from online training runs. Dataset A contains behavior trained for a single utility, (13) at $(\lambda, \rho) = (1, 0)$; Dataset B mixes four utilities, $\lambda \in \{0.5, 1\}$ and $\rho \in \{0, 1\}$. Each run contributes 512 trajectories of 1000 steps from each of four policies, a uniform random policy and three checkpoints of training (Appendix F). We evaluate two scenarios: a cooperative one, $\lambda = 1$ with $\rho \in \{0, 0.5, 1, 2\}$ on both datasets, and a general-sum one, $(\lambda, \rho) \in \{0.75, 0.25\} \times \{0, 1\}$ on Dataset B. Nine of the twelve targets match no utility the logged behavior was trained for, so cMG-DICE has to find their equilibria rather than imitate one from the data.

Baselines. Existing offline MARL methods optimize a fixed reward and have no mechanism to estimate a shadow reward, so we compare cMG-DICE with baselines that receive the shadow reward fixed in advance, evaluated at the data, $\mu_i^{\mathcal{D}} = \nabla F_i(J^{\mathcal{D}})$, or at the solution of cMG-DICE, μ_i^*, the weights it converges to (Section 4.4). (i) BC reproduces the data and initializes every other method.

Table 2: Exploitability in the target game on Clean Up: mean and standard error over three seeds, and one run for BC; best per row in bold. At $\rho = 0$, $\mu_i^* = \mu_i^{\mathcal{D}}$ and (iii) coincides with (ii).

data	(λ, ρ)	(i) BC	(ii) Indep-$\mu^{\mathcal{D}}$	(iii) Indep-μ^*	(iv) IQL-$\mu^{\mathcal{D}}$	(v) cMG-DICE
Dataset A	$(1, 0)$	0.121	0.103 $\pm$0.003	= (ii)	0.164 $\pm$0.007	**0.071** $\pm$0.003
	$(1, 0.5)$	0.116	0.089 $\pm$0.004	0.085 $\pm$0.002	0.099 $\pm$0.022	**0.069** $\pm$0.002
	$(1, 1)$	0.114	0.097 $\pm$0.004	0.094 $\pm$0.005	0.120 $\pm$0.004	**0.073** $\pm$0.001
	$(1, 2)$	0.131	**0.091** $\pm$0.002	0.104 $\pm$0.002	0.122 $\pm$0.005	0.098 $\pm$0.005
Dataset B	$(1, 0)$	0.131	0.132 $\pm$0.008	= (ii)	0.200 $\pm$0.006	**0.091** $\pm$0.006
	$(1, 0.5)$	0.123	0.116 $\pm$0.002	0.119 $\pm$0.006	0.155 $\pm$0.019	**0.079** $\pm$0.005
	$(1, 1)$	0.115	0.125 $\pm$0.004	0.128 $\pm$0.003	0.164 $\pm$0.005	**0.077** $\pm$0.001
	$(1, 2)$	0.125	0.127 $\pm$0.004	0.130 $\pm$0.009	0.180 $\pm$0.020	**0.085** $\pm$0.004
	$(0.75, 0)$	0.150	0.122 $\pm$0.012	= (ii)	0.146 $\pm$0.001	**0.089** $\pm$0.005
	$(0.75, 1)$	0.128	0.136 $\pm$0.006	0.124 $\pm$0.002	0.157 $\pm$0.010	**0.084** $\pm$0.005
	$(0.25, 0)$	0.294	0.227 $\pm$0.005	= (ii)	**0.154** $\pm$0.009	0.214 $\pm$0.006
	$(0.25, 1)$	0.210	0.211 $\pm$0.002	0.209 $\pm$0.013	**0.132** $\pm$0.014	0.183 $\pm$0.002

(ii) Indep-$\mu^{\mathcal{D}}$ and (iii) Indep-μ^* solve, for each agent separately,

$$\max_{d_i} \langle \bar{r}_i, d_i \rangle - \alpha \, D_{\mathrm{KL}}(d_i \,\|\, d_i^D), \qquad \bar{r}_i(s, a_i) = \mathbb{E}_{a_{-i} \sim \mathcal{D}|s, a_i} \Big[\textstyle\sum_j \mu_{ij} r_j(s, a) \Big], \tag{14}$$

under the Bellman flow constraint of the agent's own marginal, with $\mu_i = \mu_i^{\mathcal{D}}$ in (ii) and $\mu_i = \mu_i^*$ in (iii). The co-players act as in the data, and the divergence is placed on the agent's own marginal, whereas (8) places it on the joint occupancy $d_i \otimes \pi_{-i}$. (iv) IQL-$\mu^{\mathcal{D}}$ (Kostrikov et al., 2022), an offline method outside the DICE family, receives $\mu_i^{\mathcal{D}}$. (v) cMG-DICE learns μ_i jointly with the regularization of the joint occupancy, and at $(\lambda, \rho) = (1, 0)$ reduces to AlberDICE (Matsunaga et al., 2023). Exploitability is estimated with a PPO best response per agent against the frozen co-players at 2×10^7 environment steps.

Results. For Q5, cMG-DICE is less exploitable than BC in every cell of Table 2, including the nine targets absent from the data, by 32% on average, and has the lowest mean in nine of twelve cells. The reduction shrinks when the target's equilibrium requires behavior that the data rarely contain: on Dataset A, whose behavior was trained only at $\rho = 0$, it falls from 41% at $\rho = 0$ to 25% at $\rho = 2$, where Indep-$\mu^{\mathcal{D}}$ is less exploitable than cMG-DICE, while on Dataset B, which also contains behavior trained at $\rho = 1$, it stays between 30% and 36%. At $\lambda = 0.25$, IQL-$\mu^{\mathcal{D}}$ is less exploitable than cMG-DICE: its agents collect about a sixth of the apples, leaving the cooperative behavior of Dataset B, whereas cMG-DICE and Indep-$\mu^{\mathcal{D}}$ are regularized toward it. At $\lambda = 1$, cMG-DICE also reaches a higher common utility than BC in every cell (Appendix G).

For Q6, handing the shadow reward of the reached equilibrium, μ_i^*, to the independent learner changes little: the means of Indep-$\mu^{\mathcal{D}}$ and Indep-μ^* differ by at most 0.013 over the eight cells with $\rho > 0$. IQL-$\mu^{\mathcal{D}}$, which optimizes the same fixed reward with a different offline method, is more exploitable than BC in seven of its twelve cells. The weights μ_i^* capture only the first term of the shadow reward of the regularized game (Section 3.2); the second term, which penalizes the joint occupancy $d_i \otimes \pi_{-i}$ with the current co-players, is absent from (ii)–(iv), which regularize only the agent's own distribution. The shadow reward therefore has to be learned with the regularization of the joint occupancy.

6 Conclusion

We asked whether the Nash equilibria of a convex Markov game can be approximated from logged play. cMG-DICE turns each agent's best response into a single minimization over logged transitions, in which the shadow reward is estimated off-policy. From one dataset, it reaches different equilibria of the target game as the utility, including a safety penalty, and the initial joint policy change, and approximates their shadow rewards and returns before deployment. On Clean Up, it lowers the exploitability of the data, and the shadow reward has to be learned together with the regularization of the joint occupancy rather than handed to an independent learner. Our work has two limitations. First, cMG-DICE can reach only equilibria whose behavior appears in the data, and its gains shrink when the target's equilibrium requires behavior that the data rarely contain. Second, the equilibrium it reaches depends on the initial joint policy, and how to choose a good initial joint policy is left open.

AI USE STATEMENT

We used generative AI tools for manuscript editing, code development, literature search, drawing figures / tables and occasional checks of mathematical derivations written by the authors. The theoretical formulation and the interpretation of experimental results were developed by the authors. All AI-assisted work was reviewed and verified by the authors, and we take responsibility for the final content of this work.

ETHICS STATEMENT

This work studies offline equilibrium computation in simulated environments, the tabular games of Gemp et al. (2025) and Clean Up from SocialJax (Guo et al., 2025), with datasets generated in them. Our experiments do not involve human subjects or personally identifiable information, and we do not identify specific ethical concerns arising from them. The fairness utilities we study are one normative choice among many, and applying offline multi-agent methods to real-world systems may raise safety and fairness considerations that these benchmarks do not capture.

REPRODUCIBILITY STATEMENT

We provide the details needed to reproduce both the theoretical and empirical results of this work. Section 3.2 derives cMG-DICE, and Appendix B gives the full derivation and Algorithm 1; Appendix C proves Propositions 1 and 2 under the stated assumptions. Sections 4 and 5 describe the experimental settings and baselines, Appendix F reports all hyperparameters, which were fixed before evaluation, and Appendices E and G report the full results. Code for reproducing the experiments is provided as supplementary material.

REFERENCES

Anas Barakat, Ioannis Panageas, and Antonios Varvitsiotis. Convex Markov games and beyond: General utility markov games. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2026. arXiv:2602.12181.

The Viet Bui, Thanh Hong Nguyen, and Tien Mai. ComaDICE: Offline cooperative multi-agent reinforcement learning with stationary distribution shift regularization. *arXiv preprint arXiv:2410.01954*, 2024.

Claude Formanek, Marcel Hedman, Kale-ab Abebe Tessera, Chris Holmes, and Jonathan P. Shock. Molten Pot: Evaluations & datasets for social offline reinforcement learning. In *ICML 2026 Workshop on Decision-Making from Offline Datasets to Online Adaptation: Black-Box Optimization to Reinforcement Learning*, 2026.

Juan Claude Formanek, Callum Rhys Tilbury, Louise Beyers, Jonathan Shock, and Arnu Pretorius. Dispelling the mirage of progress in offline MARL through standardised baselines and evaluation. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Ian Gemp, Andreas Haupt, Luke Marris, Siqi Liu, and Georgios Piliouras. Convex Markov games: A framework for creativity, imitation, fairness, and safety in multiagent learning. In *International Conference on Machine Learning (ICML)*, 2025. arXiv:2410.16600.

Zihao Guo, Richard Willis, Shuqing Shi, Tristan Tomilin, Joel Z. Leibo, and Yali Du. SocialJax: An evaluation suite for multi-agent reinforcement learning in sequential social dilemmas. In *arXiv preprint arXiv:2503.14576*, 2025.

Edward Hughes, Joel Z. Leibo, Matthew Phillips, Karl Tuyls, Edgar Dueñez-Guzman, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin McKee, Raphael Koster, Heather Roff, and Thore Graepel. Inequity aversion improves cooperation in intertemporal social dilemmas. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Fivos Kalogiannis, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Ian Gemp, and Georgios Piliouras. Solving zero-sum convex Markov games. *arXiv preprint arXiv:2506.16120*, 2025.

Geon-Hyeong Kim, Jongmin Lee, Youngsoo Jang, Hongseok Yang, and Kee-Eung Kim. LobsDICE: Offline learning from observation via stationary distribution correction estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022a.

Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. DemoDICE: Offline imitation learning with supplementary imperfect demonstrations. In *International Conference on Learning Representations (ICLR)*, 2022b.

Woojun Kim and Katia Sycara. Fair cooperation in mixed-motive games via conflict-aware gradient adjustment. In *arXiv preprint arXiv:2508.17696*, 2025.

Woosung Kim, Hayeong Lee, Jongmin Lee, and Byung-Jun Lee. ROIDICE: Offline return on investment maximization for efficient decision making. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Woosung Kim, Jinho Lee, Jongmin Lee, and Byung-Jun Lee. FairDICE: Fairness-driven offline multi-objective reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.08062.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations*, 2022.

Jongmin Lee, Wonseok Jeon, Byung-Jun Lee, Joelle Pineau, and Kee-Eung Kim. OptiDICE: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning (ICML)*, 2021.

Jongmin Lee, Meiqi Sun, and Pieter Abbeel. SEMDICE: Off-policy state entropy maximization via stationary distribution correction estimation. In *International Conference on Learning Representations*, 2025.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Andrei Lupu and Doina Precup. Gifting in multi-agent reinforcement learning. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2020.

Daiki E. Matsunaga, Jongmin Lee, Jaeseok Yoon, Stefanos Leonardos, Pieter Abbeel, and Kee-Eung Kim. AlberDICE: Addressing out-of-distribution joint actions in offline multi-agent RL via alternating stationary distribution correction estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. DualDICE: Behavior-agnostic estimation of discounted stationary distribution corrections. In *Advances in Neural Information Processing Systems*, 2019a.

Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans. AlgaeDICE: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019b.

Ling Pan, Longbo Huang, Tengyu Ma, and Huazhe Xu. Plan better amid conservatism: Offline multi-agent reinforcement learning with actor rectification. In *International Conference on Machine Learning (ICML)*, 2022.

Umer Siddique, Paul Weng, and Matthieu Zimmer. Learning fair policies in multi-objective (deep) reinforcement learning with average and discounted rewards. In *International Conference on Machine Learning (ICML)*, 2020.

Umer Siddique, Peilang Li, and Yongcan Cao. Fair multi-agent reinforcement learning for traffic signal control. *ACM Journal on Autonomous Transportation Systems*, 3(2), 2025.

Richard Willis, Yali Du, Joel Z. Leibo, and Michael Luck. Quantifying the self-interest level of Markov social dilemmas. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2025. arXiv:2501.16138.

Yiqin Yang, Xiaoteng Ma, Chenghao Li, Zewu Zheng, Qiyuan Zhang, Gao Huang, Jun Yang, and Qianchuan Zhao. Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Tom Zahavy, Brendan O'Donoghue, Guillaume Desjardins, and Satinder Singh. Reward is enough for convex MDPs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Junyu Zhang, Alec Koppel, Amrit Singh Bedi, Csaba Szepesvári, and Mengdi Wang. Variational policy gradient method for reinforcement learning with general utilities. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Junyu Zhang, Amrit Singh Bedi, Mengdi Wang, and Alec Koppel. Multi-agent reinforcement learning with general utilities via decentralized shadow reward actor-critic. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

Baiting Zhu, Meihua Dang, and Aditya Grover. Scaling pareto-efficient decision making via offline multi-objective RL. In *International Conference on Learning Representations (ICLR)*, 2023.

Matthieu Zimmer, Claire Glanois, Umer Siddique, and Paul Weng. Learning fair policies in decentralized cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2021.

A Related Work

Convex MDPs and convex Markov games. A convex MDP maximizes a concave function of the state–action occupancy rather than an expected return. Zhang et al. (2020) derive the policy gradient of such utilities, whose first factor is the gradient of the utility at the current occupancy, and Zhang et al. (2022) call this gradient the shadow reward. Zahavy et al. (2021) write a convex MDP as a game between a policy player and a cost player that chooses the reward through the Fenchel conjugate, and solve it online by no-regret dynamics. Gemp et al. (2025) extend the setting to several agents, prove that pure Nash equilibria exist in convex Markov games, and compute them by minimizing a differentiable upper bound on exploitability with access to the transition kernel. Kalogiannis et al. (2025) give convergence guarantees for independent policy gradient in two-player zero-sum convex Markov games, and Barakat et al. (2026) study policy-gradient methods for general-utility Markov games, with sample-complexity results for the potential case. All of these methods evaluate the shadow reward on-policy, from rollouts of the current policy or from its occupancy computed with the kernel; cMG-DICE evaluates it from a fixed dataset.

DICE-based offline RL. DICE methods optimize the occupancy directly, under the Bellman flow constraint, and correct the data distribution by a learned ratio (Nachum et al., 2019b; Lee et al., 2021). With a divergence to the data as regularizer, the Lagrangian dual becomes an expectation over logged transitions and the ratio has a closed form. The same construction has been carried out for particular concave objectives in the single-agent setting: imitation from demonstrations (Kim et al., 2022b;a), return-on-investment (Kim et al., 2024), nonlinear welfare of several returns (Kim et al., 2025), and further objectives (Lee et al., 2025). In the multi-agent setting, AlberDICE (Matsunaga et al., 2023) computes per-agent best responses in turn for a common reward, with the divergence placed on the joint occupancy, and ComaDICE (Bui et al., 2024) combines the stationary-distribution regularization with value decomposition for cooperative tasks. These methods assume either a single agent or a common linear reward; cMG-DICE allows each agent a different concave utility.

Offline multi-agent RL. Offline MARL methods optimize a fixed reward, most often a common reward in cooperative tasks (Yang et al., 2021; Pan et al., 2022; Formanek et al., 2024). Single-agent offline methods such as implicit Q-learning (IQL) (Kostrikov et al., 2022) can also be run independently for each agent on a shared dataset. Run this way, each agent treats the others as part of the environment and optimizes a fixed reward; baselines (ii)–(iv) of Section 5 are of this kind.

Mixed-motive MARL and fairness. In sequential social dilemmas, agents are often given a modified reward that includes the returns of others: reward exchange (Willis et al., 2025), gifting (Lupu & Precup, 2020), and inequity aversion (Hughes et al., 2018). Welfare-based fairness optimizes a

concave social welfare function of the per-agent returns, online (Siddique et al., 2020; Zimmer et al., 2021; Siddique et al., 2025; Kim & Sycara, 2025) or offline for a single policy (Kim et al., 2025; Zhu et al., 2023). The utility family (13) contains reward exchange as its linear members ($\rho = 0$) and the power-mean welfare functions as its common members ($\lambda = 1$); a general member gives each agent its own concave utility, so the result is a game rather than a single welfare maximization.

B Derivation of cMG-DICE

Algorithm 1 cMG-DICE

Require: dataset D; utilities $\{u_i\}$; regularization α; initial joint policy $\pi^{(0)}$
1: fit the data model π^D (for ξ_i and KL_{-i}); $\pi \leftarrow \pi^{(0)}$
2: **repeat**
3: **for** $i = 1, \dots, n$ **do**
4: $(\tilde{r}_i, \nu) \leftarrow \arg\min$ of the dual (12) with π_{-i} held fixed
5: $w \leftarrow \exp(e_{\nu,\tilde{r}_i}/\alpha - 1)$; $\pi_i \leftarrow$ weighted behavior cloning of a_i on w over D
6: **end for**
7: **until** convergence
8: **return** π

This appendix derives the offline best response (12) step by step. Section B.1 recalls single-agent DICE, Section B.2 adds the fixed opponents as in AlberDICE, Section B.3 adds the conjugate of the utility, Section B.4 works out the welfare utilities (13), Section B.5 describes how the dual is estimated from samples, and Section B.6 states which utilities admit the construction.

B.1 Single-agent DICE

With a single agent and reward r, the regularized problem is

$$\max_{d \geq 0} \langle r, d \rangle - \alpha\, D_{\mathrm{KL}}(d \,\|\, d^D) \quad \text{s.t.} \quad \sum_a d(s', a) = (1-\gamma) p_0(s') + \gamma \sum_{s,a} P(s'|s,a) d(s,a) \;\; \forall s'. \tag{15}$$

Attaching a multiplier $\nu(s')$ to each flow constraint and collecting the coefficient of each $d(s,a)$ gives the Lagrangian

$$(1-\gamma)\langle p_0, \nu \rangle + \sum_{s,a} d(s,a) \Big[e_\nu(s,a) - \alpha \log \tfrac{d(s,a)}{d^D(s,a)} \Big],$$

with $e_\nu(s,a) = r(s,a) + \gamma \sum_{s'} P(s'|s,a)\nu(s') - \nu(s)$. The problem is concave with linear constraints, so strong duality holds whenever a feasible occupancy is absolutely continuous with respect to d^D. The maximization over d separates over (s,a): $\max_{t \geq 0} t\,[e - \alpha \log(t/q)]$ with $q = d^D(s,a)$ is attained at $t = q \exp(e/\alpha - 1)$ with value $\alpha q \exp(e/\alpha - 1)$. Hence $d^* = w\, d^D$ with $w = \exp(e_\nu/\alpha - 1)$, and ν minimizes

$$(1-\gamma)\, \mathbb{E}_{p_0}[\nu] + \alpha\, \mathbb{E}_{(s,a)\sim d^D} \big[\exp\big(e_\nu(s,a)/\alpha - 1\big) \big]. \tag{16}$$

Every term is an expectation over logged data: e_ν is estimated from a logged transition (s, a, s'), and $\mathbb{E}_{p_0}$ from the logged episode starts. The residual e_ν plays the role of an advantage, and ν is the multiplier of the flow constraint, not a value function of the data policy. This is the structure of OptiDICE (Lee et al., 2021) with the KL divergence as the f-divergence. The objectives of Kim et al. (2022b;a; 2024; 2025); Lee et al. (2025) replace $\langle r, d \rangle$ by other concave functions of the occupancy; each of them is, in the terms of Section B.3, a choice of the conjugate u^* for which the dual stays an expectation over the data.

B.2 The opponents: AlberDICE

Fix agent i and the opponents π_{-i}. Throughout, $d_i^D(s, a_i) = \sum_{a_{-i}} d^D(s,a)$ is the data marginal of agent i and $\pi_{-i}^D(a_{-i}|s, a_i) = d^D(s,a)/d_i^D(s,a_i)$ is the logged opponents' behavior conditioned on agent i's action.

The reduced MDP. Averaging over the opponents gives the reward and kernel of $\bar{M}_i^{\pi_{-i}}$,

$$\bar{r}(s,a_i)=\sum_{a_{-i}}\pi_{-i}(a_{-i}|s)\,r(s,a),\qquad \bar{P}(s'|s,a_i)=\sum_{a_{-i}}\pi_{-i}(a_{-i}|s)\,P(s'|s,a),$$

so that $\langle r, d_i\otimes\pi_{-i}\rangle=\langle\bar{r},d_i\rangle$, and $d_i\otimes\pi_{-i}$ satisfies (1) if and only if $(\mathcal{B}d_i)(s)=0$ for all s, with $\mathcal{B}$ the residual (9).

The divergence splits. Writing $d^D(s,a)=d_i^D(s,a_i)\,\pi_{-i}^D(a_{-i}|s,a_i)$,

$$D_{\mathrm{KL}}(d_i\otimes\pi_{-i}\,\|\,d^D)=\sum_{s,a_i}d_i(s,a_i)\Big[\log\tfrac{d_i(s,a_i)}{d_i^D(s,a_i)}+\mathrm{KL}_{-i}(s,a_i)\Big],\tag{17}$$

with $\mathrm{KL}_{-i}(s,a_i)=\sum_{a_{-i}}\pi_{-i}(a_{-i}|s)\log\big(\pi_{-i}(a_{-i}|s)/\pi_{-i}^D(a_{-i}|s,a_i)\big)$. The first term is a divergence of agent i's own occupancy from its data marginal. The second is a cost per (s,a_i) that does not depend on d_i; it is large when the current opponents choose joint actions that the data does not contain together with a_i, and it is the out-of-distribution penalty of AlberDICE.

The change of measure. The reduced reward and kernel are expectations under the current opponents, while the logged opponents' actions follow π_{-i}^D. With $\xi_i(s,a)=\pi_{-i}(a_{-i}|s)/\pi_{-i}^D(a_{-i}|s,a_i)$,

$$\bar{r}(s,a_i)+\gamma\sum_{s'}\bar{P}(s'|s,a_i)\nu(s')=\mathbb{E}_{a_{-i}\sim\pi_{-i}^D(\cdot|s,a_i),\;s'\sim P(\cdot|s,a)}\Big[\xi_i(s,a)\big(r(s,a)+\gamma\nu(s')\big)\Big],\tag{18}$$

an expectation over the logged transitions that share (s,a_i). Repeating the steps of Section B.1 on d_i gives the per-agent dual of AlberDICE: (16) with d^D replaced by d_i^D and

$$e_\nu(s,a_i)=\mathbb{E}_{(a_{-i},s')\sim D|s,a_i}\big[\xi_i(s,a)\big(r(s,a)+\gamma\nu(s')\big)\big]-\alpha\,\mathrm{KL}_{-i}(s,a_i)-\nu(s).$$

Since the regularizer is on the joint occupancy, the opponents enter the dual only through ξ_i and KL_{-i}, and both depend on the logged opponents only through $\pi_{-i}^D(\cdot|s,a_i)$.

B.3 The utility: conjugate and exchange

For a concave utility u_i the linear term $\langle r,\cdot\rangle$ above is replaced by u_i, and the closed form of Section B.1 is lost: the maximization over d_i no longer separates over (s,a_i). Conjugating u_i alone restores it. By concavity and upper semicontinuity, $u_i(d)=\min_{\tilde{r}}[\langle\tilde{r},d\rangle-u_i^*(\tilde{r})]$, so the regularized best response (8) is $\max_{d_i}\min_{\tilde{r}_i}\Psi(d_i,\tilde{r}_i)$ with

$$\Psi(d_i,\tilde{r}_i)=\langle\tilde{r}_i,d_i\otimes\pi_{-i}\rangle-u_i^*(\tilde{r}_i)-\alpha\,D_{\mathrm{KL}}(d_i\otimes\pi_{-i}\,\|\,d^D).$$

The exchange. For each $\tilde{r}_i$, Ψ is concave in d_i (linear minus a convex divergence); for each d_i it is convex in $\tilde{r}_i$ (affine minus the concave u_i^*). The set $\Omega_i(\pi_{-i})$ is convex and compact, so Sion's minimax theorem exchanges the two optimizations without compactness in $\tilde{r}_i$, which gives (10). For a fixed $\tilde{r}_i$ the inner problem is the AlberDICE program of Section B.2 with reward $\tilde{r}_i$. Applying its derivation and joining the two minimizations gives (12), a single minimization over $(\tilde{r}_i,\nu)$. The objective is jointly convex: $-u_i^*$ is convex, $e_{\nu,\tilde{r}_i}$ is affine in $(\tilde{r}_i,\nu)$, and the exponential is convex and increasing.

Relation to the convex-MDP game. Zahavy et al. (2021) make the same exchange for a convex MDP, with $\tilde{r}$ chosen by a cost player and the occupancy by a policy player, and solve the resulting game by no-regret dynamics, reporting the average of the policy iterates. The averaging is needed because their inner problem is a linear MDP: its best responses are extreme points and its value is not strictly concave in the occupancy. With the divergence term the inner problem is strictly concave in d_i, its maximizer is unique and given by $w=\exp(e/\alpha-1)$, and the outer function of $\tilde{r}_i$ is convex. The saddle point is therefore found by minimizing (12) directly, and the last iterate is the solution. The two constructions also differ in where the occupancy comes from: the cost player of Zahavy et al. (2021) reads the occupancy of rollouts, while in (12) the occupancy is $w\,d_i^D$, available from the data.

Relation to FairDICE. For a utility of the returns, $u_i(d) = F_i(J(d))$, FairDICE (Kim et al., 2025) introduces slack variables $k_j = \langle r_j, d\rangle$ in a single-agent DICE program and dualizes them with multipliers μ_j. The resulting dual is (12) with $\tilde{r} = \sum_j \mu_j r_j$ (Section B.4): the slack multipliers are the shadow reward restricted to the span of the rewards. Writing the construction through the conjugate of u_i states it for any concave utility (Section B.6), and the fixed opponents enter it through ξ_i and KL_{-i} as in AlberDICE.

B.4 Welfare utilities

For $u_i(d) = F_i(J(d))$ with F_i in (13), the conjugate is computed through slack variables. With the opponents fixed, agent i solves

$$\max_{d_i \in \Omega_i(\pi_{-i}),\, k} F_i(k) - \alpha\, D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D) \quad \text{s.t.} \quad k_j = \langle r_j, d_i \otimes \pi_{-i}\rangle \;\; \forall j, \tag{19}$$

a concave program. Attaching μ_j to the slack constraints, the part of the Lagrangian that involves k is $\sup_k [F_i(k) - \langle \mu, k\rangle]$.

The conjugate is a constraint. F_i is positively homogeneous of degree one in $x = k + \varepsilon\mathbf{1}$, so it is the support function of the compact convex set $M_i = \{\mu \geq 0 : \langle \mu, x\rangle \geq F_i(x - \varepsilon\mathbf{1}) \;\forall x \geq 0\}$, and

$$\sup_k \big[F_i(k) - \langle \mu, k\rangle\big] = \begin{cases} \varepsilon \sum_j \mu_j, & \mu \in M_i, \\ +\infty, & \text{otherwise.} \end{cases}$$

By the reverse Hölder inequality, with $p = 1 - \rho$ and $q = (\rho - 1)/\rho$ its conjugate exponent,

$$M_i = \{\mu \geq 0 : h_i(\mu) \geq 1\}, \qquad h_i(\mu) = \Big(\sum_j \bar{\lambda}_{ij}\, (\mu_j / \bar{\lambda}_{ij})^q \Big)^{1/q},$$

the $\bar{\lambda}_i$-weighted power mean of order q of $\mu_j/\bar{\lambda}_{ij}$. At $\rho = 1$ it is the weighted geometric mean, as $\rho \to \infty$ it tends to $\sum_j \mu_j$, and at $\rho = 0$ it is $\min_j \mu_j/\bar{\lambda}_{ij}$, so that $\mu \geq \bar{\lambda}_i$ and the linear case $\tilde{r} = \sum_j \bar{\lambda}_{ij} r_j$ is recovered.

The dual. The dual objective is increasing in each μ_j, so the constraint binds, $h_i(\mu) = 1$, and the optimality condition of the slacks is $\mu = \nabla F_i(k)$. Euler's identity for homogeneous functions then gives $\langle \mu, k + \varepsilon\mathbf{1}\rangle = F_i(k)$: the weights are in the units of return for every ρ. The remaining steps are those of Sections B.1 and B.2 with reward $\sum_j \mu_j r_j$, which give

$$\min_{\nu,\, \mu:\, h_i(\mu)=1} \varepsilon \sum_j \mu_j + (1-\gamma)\, \mathbb{E}_{p_0}[\nu] + \alpha\, \mathbb{E}_{(s,a_i)\sim d_i^D} \Big[\exp\Big(\tfrac{e_{\nu,\mu}(s,a_i)}{\alpha} - 1 \Big) \Big], \tag{20}$$

with $e_{\nu,\mu} = e_{\nu,\tilde{r}_i}$ at $\tilde{r}_i = \sum_j \mu_j r_j$. This is (12) with $-u_i^*(\tilde{r}_i) = \varepsilon \sum_j \mu_j$ on the constraint set. Agent i trains the n numbers $\mu_{i1}, \ldots, \mu_{in}$ in place of a function on $\mathcal{S} \times \mathcal{A}$; in the implementation the constraint is met by the rescaling $\mu = \bar{\mu}/h_i(\bar{\mu})$ with $\log \bar{\mu}$ unconstrained, and at $\rho = 0$, where $\nabla F_i = \bar{\lambda}_i$ is constant, μ is fixed.

B.5 Estimating the dual from samples

Each best response in Algorithm 1 minimizes (12) over logged transitions. Five points of the implementation decide whether the minimizer is the one derived above; the settings used in the experiments are in Appendix F.

The out-of-distribution penalty. $\mathrm{KL}_{-i}(s, a_i)$ is the one term in (12) that is not an expectation over logged transitions. It is a function of the two policies at (s, a_i) and has to be evaluated over $a_{-i} \sim \pi_{-i}$, exactly in the tabular case and by sampling otherwise, and not at the logged a_{-i}. A logged joint action always has $\pi^D_{-i} > 0$, so evaluating at the logged action would never register a joint action that the data does not contain, which is where the penalty is meant to act (Matsunaga et al., 2023).

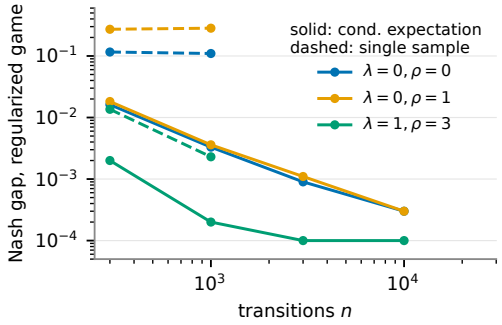

Figure 5: The two estimators of the dual on the iterated prisoner's dilemma: exploitability in the regularized game (against the exact data occupancy) against the number of logged transitions, three seeds, with the dual minimized exactly as a convex program. The conditional-expectation estimator converges to the exact equilibria; the single-sample estimator does not improve with n.

Two estimators. The conditional expectation in $e_{\nu,\tilde{r}_i}$ sits inside the exponential of (12). Replacing it by a single logged transition, $\hat{e} = \xi_i(\tilde{r}_i + \gamma\nu(s')) - \alpha\mathrm{KL}_{-i} - \nu(s)$, and averaging $\exp(\hat{e}/\alpha - 1)$ over the data gives by Jensen's inequality an upper bound on the objective, which is tight only when $\hat{e}$ is deterministic given (s, a_i), that is, when both the transitions and the opponents are deterministic. This single-sample form corresponds to AlberDICE's Corollary B.2. In a game the opponents' stochasticity alone makes it biased. The conditional-expectation estimator first averages $\hat{e}$ over the transitions that share (s, a_i), weighted by ξ_i, and then exponentiates; in the tabular case this is a group mean, and with function approximation it is a regression of $\hat{e}$ on (s, a_i). Figure 5 compares the two on the iterated prisoner's dilemma: the second converges to the exact equilibria as the number of transitions grows, while the first stays biased.

The level of ν. Shifting ν by a constant b changes e by $-(1-\gamma)b$ and the objective by $(1-\gamma)b\,(1-\mathbb{E}_{d_i^D}[w])$, so at the optimum $\mathbb{E}_{d_i^D}[w] = 1$: the level of ν enforces the normalization of the occupancy. At small α this level is determined only through an exponential of $1/\alpha$, and a small error in it scales the occupancy by a large factor. We therefore minimize the equivalent log-sum-exp form, in which the normalization is eliminated in closed form,

$$(1-\gamma)\,\mathbb{E}_{p_0}[\nu] + \alpha \log \mathbb{E}_{(s,a_i)\sim d_i^D}\left[\exp\left(e_{\nu,\tilde{r}_i}(s,a_i)/\alpha\right)\right] - u_i^*(\tilde{r}_i), \qquad w = \frac{\exp(e/\alpha)}{\mathbb{E}_{d_i^D}[\exp(e/\alpha)]},$$

so that $\mathbb{E}_{d_i^D}[w] = 1$ exactly. Any ridge on ν is placed on $\nu - \bar{\nu}$, since a ridge on the level would compete with the normalization and bias μ.

Policy extraction. Since $d_i = w\, d_i^D$, the tabular policy is $\pi_i(a_i|s) \propto w(s,a_i)\, d_i^D(s,a_i)$. With function approximation the policy maximizes $\mathbb{E}_{(s,a_i)\sim d_i^D}[w(s,a_i)\log \pi_i(a_i|o_i)]$ over a policy class on the agent's observation o_i (weighted behavior cloning), which projects d_i onto the policies the agent can execute. The ratio w is defined on the global state, and the projection averages it over the states that the observation does not distinguish.

The data model. ξ_i, KL_{-i} and d_i^D involve the logged opponents only through $\pi_{-i}^D(a_{-i}|s,a_i)$. When the data is generated by a single joint policy of independent agents, as in the tabular experiments, it equals $\prod_{j\neq i}\pi_j^D(a_j|s)$, and each factor is estimated by counts. When the data pools several policies, as the Clean Up datasets pool runs and checkpoints, the logged joint action is not a product, and we fit an autoregressive model of it as in AlberDICE (Matsunaga et al., 2023). Since π_{-i}^D is conditioned on the logged a_i, $\mathbb{E}[\xi_i \mid s, a_i] = 1$ under the data, so the extraction weight is w alone.

B.6 Which utilities admit the construction

The dual (12) needs u_i^*, or at least a way to update $\tilde{r}_i$ from samples. Three classes of utilities, covering the examples of (2), have the conjugate in closed form (cf. Table 1 of Zahavy et al., 2021).

(i) Utilities of finitely many linear functionals, $u_i(d) = F_i(\langle r_1, d\rangle, \ldots, \langle r_n, d\rangle)$: the conjugate is finite only on the span of the r_j, where it is the conjugate of F_i, as worked out in Section B.4. (ii) Pointwise separable utilities, $u_i(d) = \sum_{s,a} \phi(d(s,a))$ with ϕ concave: the conjugate separates over (s,a); the entropy is of this form. (iii) Norms and distances, $u_i(d) = -\|d - d_{\mathrm{ref}}\|$: the conjugate is $\langle \tilde{r}, d_{\mathrm{ref}}\rangle$ on the dual unit ball.

The closed form is not necessary. The minimum over $\tilde{r}_i$ in (10) is attained at $\tilde{r}_i = \nabla u_i(d)$, so $\tilde{r}_i$ can instead be set to the supergradient at the current occupancy (Zahavy et al., 2021). Offline the occupancy is known only as $w\, d^D$, so the requirement is that $\nabla u_i(w\, d^D)$ can be computed from samples of d^D and w. This holds when the supergradient depends on d through finitely many expectations, as in class (i), where $\nabla u_i = \sum_j \partial_j F_i(J)\, r_j$ and each $J_j = \mathbb{E}_{d^D}[w\, \xi_i\, r_j]$ is an off-policy estimate, and when it is a function of w alone, as for the divergence from the data, $\nabla[-D_{\mathrm{KL}}(d \,\|\, d^D)] = -\log w - 1$. For the entropy, $\nabla\mathcal{H}(d) = -\log w - \log d^D - 1$, and for a divergence from a reference other than the data, the supergradient also needs the density $d^D(s,a)$; in a large state space it can be learned from the data, as in the offline state-entropy maximization of Lee et al. (2025). In the tabular case, where d^D is a table of counts, it is available directly.

C Proofs

Proposition 1. Since $\mathrm{Exploit}_\alpha(\pi^\alpha) \le \epsilon$, the definition of the regularized exploitability at the deviation $d_i^\star$ gives $u_i^\alpha(d_i^\star \otimes \pi_{-i}^\alpha) - u_i^\alpha(d^{\pi^\alpha}) \le \epsilon$ for every agent i. Expanding $u_i^\alpha = u_i - \alpha D_{\mathrm{KL}}(\cdot \,\|\, d^D)$ and rearranging,

$$u_i(d_i^\star \otimes \pi_{-i}^\alpha) - u_i(d^{\pi^\alpha}) \le \epsilon + \alpha\big[D_{\mathrm{KL}}(d_i^\star \otimes \pi_{-i}^\alpha \,\|\, d^D) - D_{\mathrm{KL}}(d^{\pi^\alpha} \,\|\, d^D)\big].$$

The left-hand side is agent i's term in (4), so taking the maximum over i gives the bound. If $d^D(s,a) > 0$ everywhere, then for every occupancy d, $D_{\mathrm{KL}}(d \,\|\, d^D) = \sum_{s,a} d(s,a) \log d(s,a) - \sum_{s,a} d(s,a) \log d^D(s,a) \le 0 + \log\big(1/\min_{s,a} d^D(s,a)\big) = C$, and $D_{\mathrm{KL}} \ge 0$, so the bracket is at most C. If d^D does not cover the best response, the bracket is infinite and the bound is vacuous; the regularized game then restricts the deviations to the support of the data. □

Proposition 2. Write $\bar{F}$ for the common utility at $\lambda = 1$, the power mean (13) with equal weights $1/n$. Since the regularizer is the same for every agent, $u_i^\alpha = \Phi_\alpha + (F_i - \bar{F})$. For a unilateral deviation of agent i from π to $\pi' = (\pi_i', \pi_{-i})$,

$$u_i^\alpha(\pi') - u_i^\alpha(\pi) - \big[\Phi_\alpha(\pi') - \Phi_\alpha(\pi)\big] = \big[F_i - \bar{F}\big](J(\pi')) - \big[F_i - \bar{F}\big](J(\pi)),$$

whose absolute value is at most $\epsilon_\lambda := 2\sup_J |F_i(J) - \bar{F}(J)|$, which is the ϵ_λ-potential property. The regularizer cancels, so ϵ_λ does not depend on α.

For the bound on ϵ_λ, write $F(x;\bar{\lambda}) = (\sum_j \bar{\lambda}_j x_j^p)^{1/p}$ with $p = 1-\rho$ and $x = J + \varepsilon\mathbf{1} \in [\varepsilon, 1+\varepsilon]^n$ for returns in $[0,1]$. On this box F is Lipschitz in $\bar{\lambda}$ for the ℓ_1 norm with a constant $L_{\rho,\varepsilon}$: at $\rho = 1$, $\partial F/\partial\bar{\lambda}_j = F \log x_j$ and $L_{1,\varepsilon} = (1+\varepsilon)\log\frac{1+\varepsilon}{\varepsilon}$; for $\rho \ne 1$, $\partial F/\partial\bar{\lambda}_j = p^{-1}F^{1-p}x_j^p$ is bounded on the box. With $\bar{\lambda}_{ii} = 1/(1+(n-1)\lambda)$, $\bar{\lambda}_{ij} = \lambda/(1+(n-1)\lambda)$ and equal weights $1/n$,

$$\|\bar{\lambda}_i - \tfrac{1}{n}\mathbf{1}\|_1 = \frac{2(n-1)(1-\lambda)}{n\,(1+(n-1)\lambda)} \le \frac{2(n-1)(1-\lambda)}{n},$$

hence $\epsilon_\lambda \le 4L_{\rho,\varepsilon}(n-1)(1-\lambda)/n =: (1-\lambda)K_{\rho,\varepsilon,n}$.

For the convergence statement, consider exact best responses, and let an agent move only when its gain g exceeds $\epsilon_\lambda + \delta$. Each move then increases Φ_α by at least $g - \epsilon_\lambda > \delta$, and Φ_α is bounded above on the flow polytope, so the alternation stops after finitely many moves, at a joint policy where no agent gains more than $\epsilon_\lambda + \delta$. At $\lambda = 1$, $\epsilon_\lambda = 0$ and Φ_α is nondecreasing along the alternation; with $\alpha > 0$ each best response is unique, so the alternating map is continuous and every limit point is a Nash equilibrium of the regularized game. □

The shadow reward as a linearization. The multipliers of (19) give the shadow reward a second reading, which is used in the discussion of the utility family (Appendix D).

Lemma 1. *Let (d_i^*, k^*) solve (19) and $\mu_i^* = \nabla F_i(k^*)$. Then d_i^* also solves the linear problem $\max_{d_i \in \Omega_i(\pi_{-i})} \langle \sum_j \mu_{ij}^* r_j,\, d_i \otimes \pi_{-i} \rangle - \alpha D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \,\|\, d^D)$. Hence every equilibrium of the regularized game is also an equilibrium of the regularized game with linear rewards $\sum_j \mu_{ij}^* r_j$, where the weights are the marginal utilities at that equilibrium.*

Proof. Let $G(d_i) = F_i(k(d_i)) - \alpha D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \| d^D)$ with $k_j(d_i) = \langle r_j, d_i \otimes \pi_{-i} \rangle$ affine in d_i, and let $L(d_i) = \sum_j \mu_{ij}^* k_j(d_i) - \alpha D_{\mathrm{KL}}(d_i \otimes \pi_{-i} \| d^D)$. Both are concave on the convex set $\Omega_i(\pi_{-i})$, and by the chain rule they have the same gradient at d_i^*. A point maximizes a concave function over a convex set if and only if the first-order condition $\langle \nabla(\cdot)(d_i^*), d_i - d_i^* \rangle \le 0$ holds for all feasible d_i; the condition holds for G by optimality and is the same condition for L. Where d_i^* has zero entries the same argument applies with supergradients. □

The ratio $\mu_{ij}^*/\mu_{ii}^* = \lambda\big((J_i^* + \varepsilon)/(J_j^* + \varepsilon)\big)^\rho$ is the weight that agent i places on agent j's reward at the equilibrium. In the terms of mixed-motive MARL, the linear game of the lemma is reward exchange (Willis et al., 2025) with weights that differ across agents and are determined together with the equilibrium, larger toward the agents whose returns are low. The lemma does not give a way to compute the equilibrium, since the weights are known only once it is reached.

The chain rule for the shadow reward. The gradient of agent i's utility with respect to its policy parameters θ_i factors as (Zhang et al., 2020)

$$\nabla_{\theta_i} u_i(d^\pi) = \big\langle \nabla_d u_i(d^\pi),\, \nabla_{\theta_i} d^\pi \big\rangle,$$

so with the first factor, the shadow reward (6), held fixed, the gradient is the policy gradient of the MDP $\bar{M}_i^{\pi_{-i}}$ with reward $\tilde{r}_i^\pi$. The identity holds when u_i is differentiable at d^π, each policy has a smooth, strictly positive probability mass function or density on a parameter-independent support, P and p_0 do not depend on θ_i, the other agents' parameters are held fixed, and differentiation can be exchanged with the expectations and the discounted sums.

D THE UTILITY FAMILY

Parameters. In (13), λ is the weight agent i places on another agent's return when the two returns are equal, and ρ determines how that weight changes when the returns differ. The marginal rate at which agent i trades its own return for agent j's is

$$\frac{\partial F_i/\partial J_j}{\partial F_i/\partial J_i} = \lambda\Big(\frac{J_i + \varepsilon}{J_j + \varepsilon}\Big)^\rho.$$

At $\rho = 0$ it is the constant λ, the rate of reward exchange. For $\rho > 0$ it grows as agent j falls behind, and it exceeds one, so that agent i weighs j's return above its own, once $(J_i + \varepsilon)/(J_j + \varepsilon) > \lambda^{-1/\rho}$; at $\lambda = 0.5$ this threshold ratio is 2 at $\rho = 1$ and 1.26 at $\rho = 3$. At $\lambda = 0$ the utility is a monotone function of the agent's own return for every ρ, so all values of ρ give the same best responses and the same equilibria. We treat (λ, ρ) as the specification of the game and report results on a grid of them; α is the only parameter of the method.

Units and the offset. Returns are normalized per agent by the largest return that agent can achieve, so that $J_j \in [0, 1]$, and the offset ε is stated in the same units ($\varepsilon = 0.02$ throughout). Because F_i is positively homogeneous of degree one in $J + \varepsilon\mathbf{1}$, a common rescaling of the returns rescales F_i by the same factor, and the weights satisfy $\langle \mu_i, J + \varepsilon\mathbf{1} \rangle = F_i(J)$ (Appendix B.4). F_i therefore has the units of return for every ρ, and a given α sets the same trade-off between utility and divergence across the family. The additive isoelastic form $\sum_j \lambda_{ij}(J_j + \varepsilon)^{1-\rho}/(1-\rho)$ has the same unregularized best responses, but its marginal utilities scale as $(J + \varepsilon)^{-\rho}$, so a fixed α would mean a different regularization at each ρ; this is why the power mean is used. The offset bounds the marginal utility by $\varepsilon^{-\rho}$ and is needed for $\rho \ge 1$ whenever a return can be zero.

Initial states. The utilities are functions of the occupancy from p_0, and unlike in a Markov game with linear returns the equilibrium depends on p_0. With linear returns the occupancy is linear in p_0, so a joint policy that is a best response from every start is a best response from any mixture of starts. With concave utilities the weights μ_{ij} are marginal utilities of the returns averaged over p_0,

so changing p_0 changes the weights and the behavior in every state (cf. Zimmer et al., 2021). We take p_0 as part of the game; offline it is estimated from the episode starts in the data, and the fairness it defines is over the distribution of starts.

The regularization strength. α trades the bound of Proposition 1 against the reliability of the estimates: a smaller α allows equilibria farther from the data, whose returns and shadow rewards are estimated from fewer effective samples. The divergence of the solution from the data, $\mathbb{E}_{d_i^{\mathcal{D}}}[w \log w] + \mathbb{E}_{d_i^{\mathcal{D}}}[w\, \mathrm{KL}_{-i}]$, is available from the same samples, so α can be chosen offline as the smallest value whose estimated divergence stays within a stated budget. In the experiments we report the whole range of α in the tabular games and fix α in advance on Clean Up.

E Additional Tabular Results

E.1 Games

All three games follow Gemp et al. (2025), with $\gamma = 0.99$, a uniform initial state distribution, and returns normalized per agent as in Appendix D.

Iterated prisoner's dilemma. Two agents play the prisoner's dilemma repeatedly, and the state is the previous joint action ($|\mathcal{S}| = 4$, $|\mathcal{A}| = 4$). The normalized stage payoffs are $(2/3, 2/3)$ for mutual cooperation, $(0, 1)$ and $(1, 0)$ when one agent defects, and $(1/3, 1/3)$ for mutual defection. At $\rho = 0$ agent i's stage utility is its payoff plus λ times the other's, so defection is dominant in every round for $\lambda < 0.5$, cooperation is dominant for $\lambda > 0.5$, and at $\lambda = 0.5$ the agent is indifferent. The parameter λ therefore changes which profiles are equilibria.

El Farol. Three agents decide each round whether to go to a bar or stay home, and the state is the previous joint action ($|\mathcal{S}| = 8$, $|\mathcal{A}| = 8$). Staying home pays 1, going to the bar pays 2 if at most two agents go and 0 if all three go; normalized by the largest achievable return, 2, these are 0.5, 1 and 0. Every round, at least one agent has to stay home for the others to be paid. Three ways of deciding who stays home are equilibria: one agent always stays home (O, sorted returns $(0.5, 1, 1)$), two agents take turns while the third always goes (T, $(0.75, 0.75, 1)$), and all three rotate (R, $(0.83, 0.83, 0.83)$). All three have the same total return of 2.5, and we verified with the exact best response that each is a Nash equilibrium of the target game under all fifteen utilities of the grid $\lambda \in \{0, 0.25, 0.5, 0.75, 1\}$, $\rho \in \{0, 1, 3\}$. On El Farol the utility therefore does not change the set of these equilibria; it changes how the alternation moves between them.

Pathfinding. Two agents move on a 3×3 grid whose top and bottom middle cells are walls, each from a bottom corner to the goal in the opposite top corner, so their paths cross in the middle row. Each agent stays or moves in one of four directions ($|\mathcal{A}_i| = 5$); when both try to enter the same cell, a chance move of the OpenSpiel implementation decides which one moves, and we fold it into P. An agent is paid once, on its first arrival at its goal, every other step costs 0.01 before normalization, and once both have arrived the game restarts from the initial state, as in Gemp et al. (2025); enumerating the reachable positions and arrival flags gives $|\mathcal{S}| = 145$ and $|\mathcal{A}| = 25$.

E.2 Exploitability over the full grid

Tables 3–5 report the target-game exploitability of the joint policy returned by cMG-DICE at every cell of the grid. At the smallest α the mean is at most 0.008 in every cell of the three games, and no dataset draw exceeds 0.010. At large α the exploitability on the prisoner's dilemma and El Farol is highest at $\lambda = 0$, where the equilibrium is farthest from the uniform play of the data. On pathfinding it is high at every utility, up to 0.43 at $\alpha = 0.5$, since the uniform play of the data rarely reaches a goal and a joint policy near it leaves each agent a large gain from moving to its goal. On El Farol, in every row with $\lambda > 0$ the exploitability is higher at $\alpha = 0.1$ than at $\alpha = 0.2$: at $\alpha = 0.1$ the joint policy is moving from near the data toward one of the equilibria described above (Section 4.1).

Table 3: Target-game exploitability reached by cMG-DICE on the iterated prisoner's dilemma, mean and standard error over five dataset draws. At $\lambda = 0$ every ρ gives the same utility up to a monotone transformation, so the three rows coincide.

ρ	λ	$\alpha = 0.02$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.5$	$\alpha = 1$
0	0	0.000 ±0.000	0.011 ±0.000	0.051 ±0.000	0.100 ±0.000	0.141 ±0.000	0.154 ±0.001
	0.25	0.004 ±0.000	0.027 ±0.000	0.045 ±0.000	0.056 ±0.001	0.064 ±0.000	0.066 ±0.001
	0.5	0.005 ±0.002	0.005 ±0.001	0.003 ±0.001	0.003 ±0.000	0.003 ±0.000	0.003 ±0.000
	0.75	0.008 ±0.000	0.027 ±0.000	0.037 ±0.000	0.042 ±0.001	0.045 ±0.001	0.047 ±0.001
	1	0.002 ±0.000	0.026 ±0.000	0.051 ±0.000	0.066 ±0.000	0.077 ±0.001	0.080 ±0.000
1	0	0.000 ±0.000	0.011 ±0.000	0.051 ±0.000	0.100 ±0.000	0.141 ±0.000	0.154 ±0.001
	0.25	0.004 ±0.000	0.018 ±0.000	0.021 ±0.001	0.022 ±0.001	0.024 ±0.001	0.024 ±0.000
	0.5	0.001 ±0.000	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000
	0.75	0.007 ±0.000	0.013 ±0.001	0.011 ±0.000	0.011 ±0.001	0.011 ±0.001	0.011 ±0.000
	1	0.002 ±0.000	0.022 ±0.000	0.033 ±0.000	0.034 ±0.001	0.033 ±0.001	0.032 ±0.001
3	0	0.000 ±0.000	0.011 ±0.000	0.051 ±0.000	0.100 ±0.000	0.141 ±0.000	0.154 ±0.001
	0.25	0.004 ±0.000	0.007 ±0.000	0.008 ±0.000	0.008 ±0.000	0.009 ±0.000	0.008 ±0.000
	0.5	0.001 ±0.001	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000	0.000 ±0.000
	0.75	0.004 ±0.000	0.004 ±0.000	0.004 ±0.000	0.004 ±0.000	0.004 ±0.000	0.004 ±0.000
	1	0.002 ±0.000	0.013 ±0.000	0.012 ±0.000	0.011 ±0.000	0.011 ±0.000	0.011 ±0.000

Table 4: Target-game exploitability reached by cMG-DICE on El Farol, mean and standard error over five dataset draws. At $\lambda = 0$ the three rows coincide up to solver noise.

ρ	λ	$\alpha = 0.02$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.5$	$\alpha = 1$
0	0	0.000 ±0.000	0.006 ±0.002	0.044 ±0.001	0.071 ±0.004	0.092 ±0.001	0.106 ±0.001
	0.25	0.001 ±0.000	0.016 ±0.003	0.041 ±0.006	0.030 ±0.001	0.035 ±0.000	0.038 ±0.000
	0.5	0.003 ±0.001	0.022 ±0.004	0.045 ±0.008	0.008 ±0.001	0.005 ±0.001	0.005 ±0.000
	0.75	0.004 ±0.001	0.022 ±0.003	0.037 ±0.003	0.016 ±0.002	0.020 ±0.001	0.022 ±0.001
	1	0.002 ±0.001	0.023 ±0.003	0.035 ±0.003	0.026 ±0.001	0.033 ±0.001	0.037 ±0.001
1	0	0.000 ±0.000	0.008 ±0.002	0.044 ±0.001	0.071 ±0.004	0.092 ±0.001	0.106 ±0.001
	0.25	0.003 ±0.002	0.019 ±0.004	0.039 ±0.005	0.020 ±0.001	0.024 ±0.001	0.027 ±0.000
	0.5	0.001 ±0.000	0.023 ±0.004	0.034 ±0.002	0.008 ±0.001	0.005 ±0.001	0.005 ±0.001
	0.75	0.005 ±0.002	0.021 ±0.002	0.033 ±0.002	0.011 ±0.001	0.011 ±0.000	0.012 ±0.000
	1	0.004 ±0.001	0.015 ±0.002	0.028 ±0.001	0.016 ±0.001	0.023 ±0.001	0.026 ±0.001
3	0	0.000 ±0.000	0.008 ±0.002	0.044 ±0.001	0.071 ±0.004	0.092 ±0.001	0.106 ±0.001
	0.25	0.002 ±0.001	0.015 ±0.002	0.033 ±0.002	0.013 ±0.001	0.013 ±0.000	0.014 ±0.001
	0.5	0.002 ±0.001	0.015 ±0.002	0.029 ±0.001	0.008 ±0.001	0.005 ±0.001	0.005 ±0.001
	0.75	0.002 ±0.001	0.013 ±0.001	0.031 ±0.001	0.009 ±0.001	0.007 ±0.000	0.008 ±0.000
	1	0.002 ±0.001	0.012 ±0.001	0.025 ±0.001	0.011 ±0.001	0.012 ±0.000	0.014 ±0.000

E.3 Which equilibrium is reached

Prisoner's dilemma: λ changes the equilibrium. At $\alpha = 0.02$, cMG-DICE reaches mutual defection, returns $(0.33, 0.33)$, at $\lambda \in \{0, 0.25\}$ and returns within 0.03 of mutual cooperation, $(0.67, 0.67)$, at $\lambda \in \{0.75, 1\}$, on all five dataset draws and at every ρ. At the boundary $\lambda = 0.5$ the draws differ at $\rho = 0$ and $\rho = 3$, and all five cooperate at $\rho = 1$. Since efficiency and equality coincide in this game, ρ has no further effect on the equilibrium.

El Farol: five types of equilibria. We classify the joint policy reached at $\alpha = 0.02$ by its sorted returns. It is one of the three conventions, one yields (O), two alternate (T) or all rotate (R), if its sorted returns are within 0.06 of $(0.5, 1, 1)$, $(0.75, 0.75, 1)$ or $(0.83, 0.83, 0.83)$, respectively. Otherwise it is *other* if its total return is at least 2.495, so that almost no round is wasted, and *risky* if not: its agents decide stochastically every round, so that some rounds leave two agents at home or send all three to the bar. Table 6 gives the count of each type over twenty dataset draws in every cell of Figure 3b,c. Since the three conventions are equilibria of every utility in the grid and have the same total return, the utility does not single out one of them; it changes how often each is reached, and the dataset draw decides among them. At $\lambda = 0$ the utility does not depend on ρ, and one agent yields in 17 of 20 draws. As λ and ρ grow, T and R replace O, and at $\rho = 3$ with $\lambda \geq 0.75$ half of the draws are risky. On safe-El Farol, the risky profiles drop from 58 to 4 of 300 runs, and the draws that were risky at $\rho = 3$ reach T or R instead.

Table 5: Target-game exploitability reached by cMG-DICE on pathfinding, mean and standard error over five dataset draws.

ρ	λ	$\alpha = 0.02$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.5$	$\alpha = 1$
0	0	0.005 ±0.000	0.008 ±0.001	0.027 ±0.002	0.154 ±0.008	0.432 ±0.012	0.380 ±0.022
	0.25	0.004 ±0.000	0.006 ±0.000	0.027 ±0.002	0.153 ±0.009	0.425 ±0.012	0.367 ±0.023
	0.5	0.002 ±0.000	0.005 ±0.001	0.025 ±0.002	0.154 ±0.009	0.420 ±0.012	0.356 ±0.023
	0.75	0.001 ±0.000	0.004 ±0.001	0.024 ±0.002	0.154 ±0.009	0.416 ±0.012	0.349 ±0.024
	1	0.001 ±0.000	0.004 ±0.001	0.022 ±0.002	0.154 ±0.009	0.414 ±0.013	0.342 ±0.024
1	0	0.005 ±0.000	0.008 ±0.001	0.027 ±0.002	0.154 ±0.008	0.432 ±0.012	0.380 ±0.022
	0.25	0.003 ±0.000	0.006 ±0.000	0.027 ±0.002	0.153 ±0.009	0.425 ±0.012	0.366 ±0.022
	0.5	0.002 ±0.000	0.004 ±0.001	0.025 ±0.002	0.154 ±0.009	0.420 ±0.012	0.355 ±0.024
	0.75	0.002 ±0.000	0.004 ±0.001	0.023 ±0.002	0.154 ±0.009	0.416 ±0.013	0.348 ±0.024
	1	0.001 ±0.000	0.004 ±0.001	0.022 ±0.002	0.153 ±0.009	0.414 ±0.012	0.343 ±0.024
3	0	0.005 ±0.000	0.008 ±0.001	0.027 ±0.002	0.154 ±0.008	0.432 ±0.012	0.380 ±0.022
	0.25	0.003 ±0.000	0.007 ±0.000	0.027 ±0.002	0.154 ±0.009	0.425 ±0.012	0.366 ±0.023
	0.5	0.002 ±0.000	0.005 ±0.001	0.024 ±0.002	0.154 ±0.009	0.420 ±0.012	0.355 ±0.023
	0.75	0.001 ±0.000	0.004 ±0.001	0.023 ±0.002	0.153 ±0.009	0.416 ±0.012	0.347 ±0.024
	1	0.001 ±0.000	0.004 ±0.001	0.022 ±0.002	0.153 ±0.009	0.414 ±0.013	0.342 ±0.024

Table 6: The equilibria cMG-DICE reaches on El Farol and safe-El Farol at $\alpha = 0.02$: number of the twenty dataset draws in each cell that reach each type (O: one yields, T: two alternate, R: all rotate, oth.: other, risky), and the ratio $\min_j J_j / \max_j J_j$ (mean and standard error). Every run has target-game exploitability below 0.015 on El Farol and below 0.01 on safe-El Farol, measured in the penalized game.

		El Farol						safe-El Farol					
ρ	λ	O	T	R	oth.	risky	ratio	O	T	R	oth.	risky	ratio
0	0	17	3	0	0	0	0.54 ±0.02	19	0	1	0	0	0.52 ±0.02
	0.25	9	7	4	0	0	0.69 ±0.04	11	8	1	0	0	0.63 ±0.03
	0.5	0	17	3	0	0	0.79 ±0.02	11	8	1	0	0	0.63 ±0.03
	0.75	5	12	2	1	0	0.73 ±0.03	5	13	2	0	0	0.72 ±0.03
	1	0	10	8	0	2	0.84 ±0.03	5	8	7	0	0	0.77 ±0.04
1	0	17	3	0	0	0	0.54 ±0.02	19	0	1	0	0	0.52 ±0.02
	0.25	4	14	2	0	0	0.73 ±0.03	6	12	2	0	0	0.71 ±0.03
	0.5	2	11	4	2	1	0.76 ±0.03	3	9	8	0	0	0.81 ±0.04
	0.75	0	8	5	0	7	0.79 ±0.03	2	9	6	2	1	0.80 ±0.03
	1	0	3	4	0	13	0.82 ±0.02	1	10	7	0	2	0.81 ±0.03
3	0	17	3	0	0	0	0.54 ±0.02	19	0	1	0	0	0.52 ±0.02
	0.25	0	8	3	1	8	0.78 ±0.03	0	12	8	0	0	0.85 ±0.03
	0.5	0	5	7	1	7	0.84 ±0.03	0	9	8	3	0	0.86 ±0.03
	0.75	0	0	10	0	10	0.90 ±0.02	0	10	7	3	0	0.84 ±0.02
	1	0	0	10	0	10	0.92 ±0.02	0	11	8	0	1	0.85 ±0.03
total		71	104	62	5	58		101	119	68	8	4	

Pathfinding: the utility does not change the equilibrium. At $\alpha = 0.02$ both agents reach their goals under every utility of the grid, with normalized returns between 0.988 and 0.995 on every dataset draw, so the return profile of the equilibrium reached is the same for every (λ, ρ).

The initial joint policy. Table 7 lists the equilibria reached from four initial joint policies under four utilities, on one dataset draw. The four initial joint policies are behavior cloning of the data; a one-yields policy in which agent 1 stays home and the others go; a random policy, one Dirichlet draw per state and agent; and the all-rotate policy, in which the agent that stays home is the one after the agent that stayed home in the previous round. The one-yields and all-rotate policies are mixed with the uniform policy at weight 0.1, so that they have full support. At $\alpha = 0.02$ the all-rotate start reaches R under every utility, while the other starts reach O, T, or a profile between them, depending on the utility. At $\alpha = 0.1$ the four starts end at similar profiles, with target-game exploitability between 0.021 and 0.044.

Table 7: El Farol, one dataset draw: sorted returns of the equilibrium reached from four initial joint policies, with its target-game exploitability in small type.

(λ, ρ)	α	BC	one yields	random	all rotate
$(0, 0)$	0.02	(0.50, 1.00, 1.00) 0.000	(0.50, 1.00, 1.00) 0.000	(0.75, 0.75, 1.00) 0.000	(0.83, 0.83, 0.83) 0.001
	0.1	(0.67, 0.68, 0.90) 0.036	(0.65, 0.68, 0.88) 0.043	(0.65, 0.68, 0.89) 0.044	(0.71, 0.76, 0.77) 0.037
$(0.25, 3)$	0.02	(0.75, 0.75, 1.00) 0.000	(0.75, 0.75, 1.00) 0.000	(0.75, 0.75, 1.00) 0.000	(0.83, 0.83, 0.83) 0.001
	0.1	(0.73, 0.77, 0.78) 0.030	(0.73, 0.78, 0.78) 0.030	(0.70, 0.72, 0.83) 0.040	(0.73, 0.78, 0.78) 0.030
$(1, 0)$	0.02	(0.75, 0.77, 0.96) 0.005	(0.75, 0.75, 0.99) 0.002	(0.59, 0.92, 0.94) 0.012	(0.83, 0.83, 0.83) 0.001
	0.1	(0.71, 0.72, 0.72) 0.042	(0.70, 0.72, 0.73) 0.030	(0.71, 0.72, 0.73) 0.037	(0.71, 0.73, 0.73) 0.029
$(1, 3)$	0.02	(0.75, 0.80, 0.88) 0.006	(0.74, 0.76, 0.97) 0.006	(0.77, 0.78, 0.89) 0.003	(0.83, 0.83, 0.83) 0.001
	0.1	(0.72, 0.72, 0.72) 0.024	(0.72, 0.72, 0.73) 0.029	(0.72, 0.72, 0.72) 0.021	(0.71, 0.72, 0.73) 0.029

E.4 Numerical checks of the propositions

Proposition 1. With uniform behavior data and a uniform initial distribution, the exact data occupancy is uniform, so $C = \log 16$ on the prisoner's dilemma and $C = \log 64$ on El Farol. Over all 900 runs of the grid (two games, 90 cells, five dataset draws), the target-game exploitability is below $\text{Exploit}_\alpha(\pi) + \alpha C$ in every run, where Exploit_α is computed exactly against the exact data occupancy. The bound is loose: the largest ratio of the exploitability to the bound is 0.199 on the prisoner's dilemma and 0.167 on El Farol, since the best response rarely moves the divergence from the data by the full C.

Proposition 2. Table 8 runs exact alternating best response and records, at every step, the difference between the change in the deviating agent's regularized utility and the change in the potential Φ_α. The largest difference is below ϵ_λ in every case, by a factor of 4 to 38, and it decreases as λ increases.

Table 8: Proposition 2 along exact alternating best response ($\alpha = 0.05$, uniform data): the largest $|\Delta u_i^\alpha - \Delta\Phi_\alpha|$ over all best-response steps, against $\epsilon_\lambda = 2 \sup_J |F_i(J) - \bar{F}(J)|$.

game	λ	ρ	steps	error	ϵ_λ
IPD	0	1	4	0.2342	1.0000
IPD	0	3	4	0.2342	1.0000
IPD	0.25	1	10	0.0735	0.4807
IPD	0.25	3	16	0.0453	0.2650
IPD	0.75	1	10	0.0122	0.1074
IPD	0.75	3	18	0.0069	0.0563
El Farol	0	1	15	0.1424	1.3333
El Farol	0	3	15	0.1424	1.3333
El Farol	0.25	1	39	0.0422	0.6422
El Farol	0.25	3	57	0.0640	0.3532
El Farol	0.75	1	48	0.0038	0.1431
El Farol	0.75	3	12	0.0025	0.0750

F Experimental Details

F.1 Tabular games

Data. Each dataset contains 10^4 transitions (s, a, r, s') of the uniformly random joint policy. Episodes terminate with probability $1 - \gamma$ after every step and restart from p_0, so that the logged pairs (s, a) are independent samples from the discounted occupancy of the behavior policy. The initial distribution p_0 is uniform and given. The data model is the product of per-agent behavior policies estimated by counts, and the logged-opponent probabilities are floored at 10^{-7}. A seed is one dataset draw; five draws are used for every cell of the grid, and twenty for El Farol and safe-El Farol at $\alpha = 0.02$ (Figure 3b,c and Table 6).

Algorithm 2 Exploitability evaluation (not part of cMG-DICE)

Require: joint policy π returned by cMG-DICE; target utilities $\{u_i\}$; access to the environment
for $i = 1, \dots, n$ **do**
  fix the opponents π_{-i}
  if the game is tabular **then**
    solve $\max_{d_i \in \Omega_i(\pi_{-i})} u_i(d_i \otimes \pi_{-i})$ exactly as a convex program with the known kernel
  else
    train a best response π_i' online against π_{-i} to maximize u_i
  end if
  $g_i \leftarrow u_i(\text{best response}) - u_i(d^\pi)$
end for
return $\text{Exploit}(\pi) = \max_i g_i$

Optimization. The dual (12) is minimized in its log-sum-exp form (Section B.5) with tabular ν and $\log \bar{\mu}$, a ridge $10^{-6}\|\nu - \bar{\nu}\|^2$, the conditional-expectation estimator (group means over (s, a_i), weighted by ξ_i), and KL_{-i} computed exactly over $a_{-i} \sim \pi_{-i}$. The policy is a table of logits trained by weighted behavior cloning toward $w\, d_i^D$. Each iteration takes one Adam step on ν, one on $\log \bar{\mu}$ and one on the policy logits of each agent in turn, with step sizes 0.05, 0.005 and 0.05; after 3×10^4 iterations the step sizes decay exponentially to a tenth of their value by the last iteration. The run stops after at least 4×10^4 iterations once the range of every agent's return over the last five checks, taken every 5000 iterations, is below 10^{-3}, or after 1.5×10^5 iterations. The returns used by this stopping rule are computed exactly with the model; they decide only when to stop and enter no update, and the estimate $\hat{J}$ of Section 4.4 could be used in their place. Probabilities below 10^{-9} are set to zero in the returned policy. The initial joint policy is behavior cloning of the data unless stated otherwise.

Evaluation. Target-game exploitability is computed by Algorithm 2: each agent's best response at $\alpha = 0$, without any restriction to the support of the data, is solved as a convex program over the reduced flow polytope with the known kernel (cvxpy with MOSEK, and CLARABEL and SCS as fallbacks). The regularized exploitability in Appendix E.4 is computed in the same way against the exact data occupancy. The equilibria reached on El Farol and safe-El Farol are classified by their sorted returns as in Appendix E.

F.2 Clean Up

Environment and datasets. We use the Clean Up environment of SocialJax (Guo et al., 2025) with seven agents, episodes of 1000 steps, and the original map with fixed spawn points.

Behavior policies. The behavior policies are trained online with IPPO, one network shared by the seven agents, for 3×10^8 environment steps per run and three seeds per utility. Each agent optimizes the shadow reward $\sum_j \mu_{ij} r_j$ with $\mu_i = \nabla F_i$ evaluated at an exponential moving average (weight 0.5) of the returns, refreshed every ten PPO updates. The actor reads the agent's egocentric observation and the critic the global state, each through a convolutional trunk (32 filters of 5×5, two layers of 32 filters of 3×3, and a dense layer of 64 units, with ReLU); the utility weights and a one-hot agent index are appended to the features, followed by a dense layer of 64 units and the output. PPO uses Adam with learning rate 5×10^{-4} decayed linearly, 32 parallel environments, rollouts of 1000 steps, 32 minibatches, two epochs, clipping 0.2, entropy coefficient 0.01, value coefficient 0.5, GAE $\lambda = 0.95$, $\gamma = 0.99$, and gradient-norm clipping 0.5.

Datasets. A dataset is a union of components of 512 trajectories of 1000 steps. Each run contributes one component from each of three checkpoints, at 2×10^7, 8×10^7 and 3×10^8 environment steps, and components from a uniform random policy make up a quarter of each dataset, so that the data are not dominated by near-expert behavior. Dataset A contains the runs at $(\lambda, \rho) = (1, 0)$ (12 components, 6144 trajectories); Dataset B contains the runs at $\lambda \in \{0.5, 1\}$ and $\rho \in \{0, 1\}$ (48 components, 24576 trajectories). Each step stores the global state, the joint action, the per-agent

rewards, and episode boundaries; observations are regenerated from the state by the environment's observation function.

Settings shared by the DICE methods. Methods (ii), (iii) and (v) of Section 5 share one implementation and one set of settings, fixed before evaluation: $\alpha = 0.3$, $\gamma = 0.99$, the ratio w clipped at 10, 2×10^4 gradient updates with batches of 1024 transitions, the policy initialized by behavior cloning of the data, a conservative penalty on the e-network of weight 0.1 in the style of CQL, a proximal term of weight 0.01 on ν, and returns scaled by 0.5 with offset $\varepsilon = 0.1$ inside F. The networks for ν, e and the actor use the convolutional trunk above followed by a dense layer of 64 units; ν has a scalar output without bias, e one output per action of agent i, and the actor one logit per action. Adam is used with learning rate 3×10^{-4} for ν, e and the actor and 10^{-2} for $\log \mu_i$, and gradients are clipped at global norm 10. Each update takes, for each agent in turn, one dual step, with two regression steps on e, and one actor step, and the updated policy is used by the next agent. Behavior cloning, which initializes every offline method, and the autoregressive model of the logged joint action are trained for 10^5 steps with learning rate 3×10^{-4} and batches of 1024 and 256. A seed changes the network initialization, the minibatch sampling and the evaluation rollouts of an offline run; the datasets and the behavior policies are fixed.

Method-specific settings. (ii) Indep-$\mu^{\mathcal{D}}$ sets $\xi_i = 1$ and $\mathrm{KL}_{-i} = 0$, gives ν and e the agent's own observation, and fixes $\mu_i = \nabla F_i(J^{\mathcal{D}})$ at the returns of the data. (iii) Indep-μ^* is (ii) with μ_i fixed at the value that cMG-DICE reaches on the same dataset and target. At $\rho = 0$, ∇F_i does not depend on the returns, so (iii) is the same run as (ii). (iv) IQL-$\mu^{\mathcal{D}}$ (Kostrikov et al., 2022) learns Q and V on the agent's own observation by expectile regression and extracts the policy by advantage-weighted regression, with the reward of (ii). We use the settings of Formanek et al. (2026) for the Clean Up substrate of Melting Pot, expectile 0.7, inverse temperature 5.0, learning rate 3×10^{-4}, $\gamma = 0.99$, soft target updates at rate 0.005 and 2×10^4 updates, without tuning, and the same initialization. Unlike Formanek et al. (2026), we use feedforward networks rather than a recurrent one over the agent's history, so that IQL shares the encoder of the other methods; they infer unknown partners, whereas here all seven agents are optimized. (v) cMG-DICE learns μ_i, and uses ξ_i and KL_{-i} under an autoregressive model of the logged joint action, with ν and e on the global state, the seven observations stacked along the channel axis. Methods (ii)–(iv) are independent learners, which treat the co-players as part of the environment; as in independent offline MARL (Formanek et al., 2024), their networks read the agent's own observation. cMG-DICE estimates the ratio of the joint occupancy $d_i \otimes \pi_{-i}$ to the data, which depends on the co-players' actions, so its ν and e read the global state. In every method the policy reads only the agent's own observation, so execution is decentralized.

Evaluation. For each agent, a best response is trained by PPO against the other six agents, which are frozen, and the exploitability is the largest gain over the seven agents in the target utility. The best response uses 32 environments, rollouts of full episodes, two epochs of 8 minibatches, clipping 0.2, no entropy bonus, learning rates 2.5×10^{-4} for the actor and 5×10^{-4} for the critic, Monte Carlo return targets, and 60 critic updates with the actor frozen before training; the deviating agent's weights μ_i are refreshed every five updates from an exponential moving average (weight 0.5) of its returns. The utilities are evaluated without discounting on 512 episodes with common random numbers, so that each gain is a paired difference. The main text reports a best-response budget of 2×10^7 environment steps; Table 10 reports the other budgets.

G Additional Clean Up Results

Utility and exploitability. Table 9 extends Table 2 with the utility averaged over the agents, $\bar{F} = \frac{1}{n}\sum_i F_i$, which is the common utility at $\lambda = 1$, and the target $\lambda = 0$. cMG-DICE reaches a higher $\bar{F}$ than BC in every row and than Indep-$\mu^{\mathcal{D}}$ in every row except Dataset A at $\rho = 2$.

The shadow reward alone. Replacing the weights $\mu_i^{\mathcal{D}}$ of (ii) by the weights μ_i^* that cMG-DICE converges to changes little: over the eight cells with $\rho > 0$, the mean $\bar{F}$ moves by at most 0.010 and the mean exploitability by at most 0.013. On the same cells cMG-DICE has $\bar{F}$ higher than both by 0.01 to 0.10 and exploitability lower by 0.02 to 0.05, except at Dataset A, $\rho = 2$, where (ii)

has the highest $\bar{F}$ and the lowest exploitability. Given the shadow reward of the equilibrium, the independent learner still does not reach that equilibrium; the difference comes from solving for the occupancy together with the shadow reward.

IQL-$\mu^{\mathcal{D}}$. IQL-$\mu^{\mathcal{D}}$ raises $\bar{F}$ above BC by 0.07 to 0.17 in every cell, and in seven of the twelve its exploitability is higher than that of BC, the policy it starts from (e.g. Dataset B, $(1, 0)$: 0.131 to 0.200). cMG-DICE has lower exploitability than IQL-$\mu^{\mathcal{D}}$ in ten cells and higher $\bar{F}$ in nine. The two exceptions in exploitability are at $\lambda = 0.25$, where IQL-$\mu^{\mathcal{D}}$ reaches 0.154 and 0.132 against 0.214 and 0.183, while its agents collect about 330 apples per episode against more than 1900 for cMG-DICE. Maximizing each agent's utility without accounting for the co-players raises the utility and can move the joint policy away from an equilibrium.

Table 9: Clean Up: average utility $\bar{F}$ (higher is better) and target-game exploitability (lower is better), shown as $\bar{F}$ / exploitability, mean and standard error over three seeds, at a best-response budget of 2×10^7 steps.

data	(λ, ρ)	(i) BC	(ii) Indep-$\mu^{\mathcal{D}}$	(iii) Indep-μ^{*}	(iv) IQL-$\mu^{\mathcal{D}}$	(v) cMG-DICE
Dataset A	$(1, 0)$	0.305 / 0.121	0.401±0.004 / 0.103±0.003	= (ii)	0.376±0.006 / 0.164±0.007	0.426±0.018 / 0.071±0.003
	$(1, 0.5)$	0.298 / 0.116	0.392±0.008 / 0.089±0.004	0.394±0.009 / 0.085±0.002	0.405±0.023 / 0.099±0.022	0.402±0.010 / 0.069±0.002
	$(1, 1)$	0.290 / 0.114	0.383±0.009 / 0.097±0.004	0.380±0.013 / 0.094±0.005	0.360±0.002 / 0.120±0.004	0.395±0.014 / 0.073±0.001
	$(1, 2)$	0.273 / 0.131	0.378±0.006 / 0.091±0.002	0.369±0.008 / 0.104±0.002	0.363±0.003 / 0.122±0.005	0.355±0.006 / 0.098±0.005
Dataset B	$(1, 0)$	0.368 / 0.131	0.474±0.010 / 0.132±0.008	= (ii)	0.474±0.006 / 0.200±0.006	0.523±0.004 / 0.091±0.006
	$(1, 0.5)$	0.365 / 0.123	0.465±0.005 / 0.116±0.002	0.467±0.009 / 0.119±0.006	0.485±0.017 / 0.155±0.019	0.521±0.007 / 0.079±0.005
	$(1, 1)$	0.363 / 0.115	0.460±0.000 / 0.125±0.004	0.452±0.005 / 0.128±0.003	0.458±0.008 / 0.164±0.005	0.518±0.009 / 0.077±0.001
	$(1, 2)$	0.357 / 0.125	0.452±0.007 / 0.127±0.004	0.454±0.005 / 0.130±0.009	0.426±0.009 / 0.180±0.020	0.505±0.009 / 0.085±0.004
	$(0.75, 0)$	0.369 / 0.150	0.465±0.011 / 0.122±0.012	= (ii)	0.489±0.003 / 0.146±0.001	0.518±0.014 / 0.089±0.005
	$(0.75, 1)$	0.363 / 0.128	0.452±0.007 / 0.136±0.006	0.457±0.009 / 0.124±0.002	0.463±0.009 / 0.157±0.010	0.531±0.006 / 0.084±0.005
	$(0.25, 0)$	0.369 / 0.294	0.440±0.010 / 0.227±0.005	= (ii)	0.537±0.004 / 0.154±0.009	0.556±0.007 / 0.214±0.006
	$(0.25, 1)$	0.363 / 0.210	0.425±0.009 / 0.211±0.002	0.434±0.010 / 0.209±0.013	0.537±0.003 / 0.132±0.014	0.535±0.009 / 0.183±0.002
	$(0, 0)$	0.368 / 0.744	0.393±0.009 / 0.599±0.032	= (ii)	–	0.513±0.004 / 0.718±0.020

A target the data cannot represent ($\lambda = 0$). At $\lambda = 0$ each agent maximizes its own return, and the game is the commons dilemma, whose equilibrium is that no agent cleans the river. Online training at $\lambda = 0$ reaches it, with exploitability between 0.005 and 0.024 over three seeds and 618 apples per episode (Table 11). Every offline method remains on a cooperative profile and is exploitable by a free rider, with exploitability between 0.60 and 0.74 (Table 9). No behavior policy in the data behaves as a free rider, so this equilibrium is not represented in the data; this can be read off the dataset before any method is run.

Best-response budget. Table 10 reports the exploitability as the budget of the learned best response increases from 10^6 to 10^8 environment steps. A learned best response gives a lower bound on the exploitability; on every run shown, cMG-DICE stays below BC at every budget, and on the two Dataset B runs it also stays below Indep-$\mu^{\mathcal{D}}$.

Table 10: Clean Up: exploitability against the budget of the learned best response (environment steps), one seed per run.

method	data	(λ, ρ)	10^6	5×10^6	10^7	2×10^7	5×10^7	10^8
cMG-DICE	B	$(1, 1)$	.050	.071	.076	.086	.101	.102
	A	$(1, 0)$	.033	.044	.048	.055	.069	.072
	A	$(1, 2)$	.059	.095	.109	.112	.118	.116
	B	$(0.75, 1)$	.051	.072	.076	.085	.088	.090
BC	B	$(1, 1)$	.081	.113	.122	.128	.141	.139
	A	$(1, 0)$	.079	.091	.111	.124	.125	.129
	A	$(1, 2)$	.072	.100	.120	.130	.127	.126
	B	$(0.75, 1)$	.093	.115	.125	.137	.140	.142
Indep-$\mu^{\mathcal{D}}$	B	$(1, 1)$	.085	.105	.120	.113	.132	.137
	B	$(0.75, 1)$	.094	.116	.118	.123	.131	.143
	A	$(1, 2)$	.054	.079	.091	.101	.104	.109

Online reference. Table 11 reports the joint policies trained online for the behavior data, evaluated at their own utility under the same best-response evaluation. Their exploitability, between 0.005 and

0.061, indicates the level that the learned best response reaches on policies trained to equilibrium with environment access.

Table 11: Clean Up, online training with environment access: exploitability of the final policy for each of three seeds, average utility $\bar{F}$, apples collected per episode, and the apples of the agent that collects the fewest.

(λ, ρ)	exploitability (3 seeds)	F	apples	fewest
$(1, 0)$	.034 / .037 / .045	.516	2002	12
$(1, 1)$	.009 / .023 / .024	.678	2295	217
$(1, 2)$	.031 / .022 / .049	.570	2036	190
$(0.75, 0)$	.035 / .031 / .061	.585	2202	43
$(0.75, 1)$	.020 / .017 / .011	.758	2698	158
$(0.75, 2)$	.028 / .010 / .034	.686	2498	196
$(0.25, 0)$	.012 / .015 / .024	.674	1913	147
$(0.25, 1)$	.007 / .010 / .007	.648	1912	136
$(0, 0)$	.005 / .011 / .024	.304	618	40