# Provably Learning Nash Policies in Constrained Markov Potential Games

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Multi-agent reinforcement learning (MARL) addresses sequential decision-making problems with multiple agents, where each agent optimizes its own objective. In many real-world instances, the agents may not only want to optimize their objectives, but also ensure safe behavior. For example, in traffic routing, each car (agent) aims to reach its destination quickly (objective) while avoiding collisions (safety). Constrained Markov Games (CMGs) are a natural formalism for safe MARL problems, though generally intractable. In this work, we introduce and study *Constrained Markov Potential Games* (CMPGs), an important class of CMGs. We first show that a Nash policy for CMPGs can be found via constrained optimization. One tempting approach is to solve it by Lagrangian-based primal-dual methods. As we show, in contrast to the single-agent setting, however, CMPGs do not satisfy strong duality, rendering such approaches inapplicable and potentially unsafe. To solve the CMPG problem, we propose our algorithm **C**oordinate-**A**scent for **CMPG**s (CA-CMPG), which provably converges to a Nash policy in tabular, finite-horizon CMPGs. Furthermore, we provide the first sample complexity bounds for learning Nash policies in unknown CMPGs, and, which under additional assumptions, guarantee safe exploration.

## 1 Introduction

Multi-Agent Reinforcement Learning (MARL) addresses sequential decision-making problems with *multiple agents*, where the decisions of individual agents may also affect others. In this work, we focus on a rich and fundamental class of MARL problems, known as *Markov Potential Games*, (MPGs, Leonardos et al., 2022). Important applications, such as traffic routing (Altman et al., 2006) or wireless communication (Yamamoto, 2015), can be modeled as MPGs. The main characteristic of an MPG is the existence of an underlying *potential function*, which captures the agents' incentives to deviate between different policies. MPGs can model both fully cooperative scenarios[1] and scenarios, in which the agents have individual objectives, as long as such a potential function exists.

In many real-world applications, however, the standard MPG framework fails to incorporate additional requirements like *safety*. For instance, in traffic routing, we do want to find the fastest route to the individual destinations, while ensuring that the vehicles drive safely and do not collide. In this work, we introduce the framework of *Constrained Markov Potential Games* (CMPGs) to study safety in the context of MPGs. We incorporate safety using *coupled* constraints on the policies of the agents. Coupled constraints are relevant because they allow us to model requirements like collision avoidance. Our objective is to find a Nash policy (Nash et al., 1950; Altman and Shwartz, 2000), i.e., a set of policies such that no agent has the incentive to deviate unilaterally within the constrained set of policies. Prior work on algorithms for (unconstrained) MPGs, in which each agent improves its

[1] In fully cooperative scenarios, the agents have one common objective.

Submitted to 16th European Workshop on Reinforcement Learning (EWRL 2023). Do not distribute.

own objective *independently*, cannot be applied to the constrained setting, as the agents may need to *coordinate* to satisfy the constraints. A more detailed discussion of prior work is provided in Section 2. We study tabular CMPGs in the finite-horizon setting and summarize our contributions here:

1. First, we show that a Nash policy can in principle be recovered by solving a constrained optimization problem, which, however, becomes intractable as the number of agents increases (Section 4).
2. Given tractable algorithms for unconstrained MPGs (cf. Leonardos et al., 2022; Fox et al., 2022), a tempting approach would be to utilize Lagrangian duality to reduce the constrained problem to an unconstrained one (Diddigi et al., 2019; Parnika et al., 2021). Unfortunately, we show that strong duality does not hold for our problem (Section 4), rendering such approaches *sub-optimal* and *unsafe*. This is in sharp contrast to the single-agent setting, for which strong duality does hold (Paternain et al., 2019).
3. Instead of solving the constrained optimization problem, we propose to directly search for a Nash policy. We present our algorithm – **C**oordinate-**A**scent for **CMPG**s (CA-CMPG) – which provably converges to an $\varepsilon$-Nash policy, assuming that the agents have full knowledge of the CMPG (Section 5).
4. Finally, we prove a sample complexity bound for our algorithm CA-CMPG, when the agents do not know the CMPG beforehand (Section 6). With access to a generative model (Section 6.1), the agents converge to an $\varepsilon$-Nash policy with $\widetilde{\mathcal{O}}\left(\frac{H^8}{\varepsilon^3\zeta^2}\right)$ samples, where $\zeta$ is the Slater constant of the CMPG and $H$ is the horizon. On the other hand, if the agents do not have access to a generative model, but still want to ensure safe exploration, we obtain a sample complexity bound of $\widetilde{\mathcal{O}}\left(\frac{H^{10}}{\varepsilon^5 c^2}\right)$ (Section 6.2), where $c \in (0, \zeta]$ is a quantity related to the constraint set of the CMPG.

## 2 Related Work

**Markov Potential Games:** MPGs have become popular in recent years and have been studied for the tabular setting (Leonardos et al., 2022; Zhang et al., 2022, 2021b; Chen et al., 2022; Mao et al., 2022; Maheshwari et al., 2022; Fox et al., 2022) and for state-action spaces with function approximation (Ding et al., 2022; Cui et al., 2023). For the tabular setting with *known* rewards and transitions, Leonardos et al. (2022) prove that independent policy gradient (IPG) converges to an $\varepsilon$-Nash policy in $O(1/\varepsilon^2)$ iterations. If rewards and transitions are *unknown*, Mao et al. (2022) prove that IPG with access to a stochastic gradient oracle converges to an $\varepsilon$-Nash policy with a sample complexity of $\mathcal{O}\left(1/\varepsilon^{4.5}\right)$. In these IPG algorithms, the agents improve their own objectives *independently*. It is challenging to apply these algorithms with coupled constraints, as the agents may need to coordinate to satisfy those constraints, at least during the learning process. Song et al. (2021) present a different approach for tabular MPGs with unknown rewards and transitions, in which the agents *coordinate* to compute an $\varepsilon$-Nash policy with a sample complexity of $\widetilde{\mathcal{O}}(1/\varepsilon^3)$. While their algorithm is for unconstrained MPGs, we show in our work, that this type of approach can be extended to the constrained setting. Maheshwari et al. (2022) present a different approach with asymptotic convergence to a Nash policy, whereas we target finite-time convergence. Note that MPGs are only one way to model MARL problems, and for a more comprehensive overview on MARL, we refer the reader to the surveys by Yang and Wang (2021) and Zhang et al. (2021a).

**Constrained Markov Decision Processes:** A common approach to constrained *single-agent* RL are *Constrained Markov Decision Processes* (CMDPs, Altman, 1999). CMDPs are widely studied, and a comprehensive survey is given by García et al. (2015). Below, we focus on aspects relevant to our work. In CMDPs, the agent optimizes a reward function subject to constraints. Lagrangian duality is a common approach for constrained optimization and Paternain et al. (2019) proved that CMDPs possess the *strong duality property*, giving theoretical justification for the use of Lagrangian dual approaches.

**Constrained Markov Games:** One of the common approaches to constrained multi-agent RL are *Constrained Markov Games* (CMGs, Altman and Shwartz, 2000). CMGs restrict the policies of the agents, which can be used to model safety objectives. Note that CMPGs are one class of CMGs. In cooperative CMPGs[2], where the agents have one common reward function, the CMPG objective very much resembles the CMDP formulation. Furthermore, Diddigi et al. (2019) and Parnika et al. (2021) demonstrate good experimental results for cooperative CMPGs with Lagrangian dual approaches,

[2]Note that cooperative games are a strict subclass of CMPGs, as CMPGs are able to model non-cooperative settings too.

but provide no theoretical guarantees. We prove in our work, however, that strong duality does not hold in general for CMPGs (cf. Section 4), rendering Lagrangian dual approaches inapplicable in those cases. Furthermore, we demonstrate that the dual might even return unsafe solutions.

# 3 Background and Problem Definition

**Notation:** For any $n \in \mathbb{N}$, we use the short-hand notation $[n]$ to refer to the set of integers $\{1, ..., n\}$. For any finite set $X$, we denote by $\Delta_X$ the probability simplex over $X$, i.e., $\Delta_X = \{v \in [0,1]^{|X|} | \sum_{x \in X} v(x) = 1\}$.

## 3.1 Markov Potential Games

An $n$-agent *Markov Potential Game* (MPG) is a tuple $\mathcal{G} = (\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^n, H, \{\mathcal{P}_h\}_{h=1}^H, \{\{r_{i,h}\}_{h=1}^H\}_{i=1}^n, \mu)$, where $\mathcal{S}$ is the state space, $\mathcal{A}_i$ is agent $i$'s action space. We denote by $\mathcal{A} \triangleq \times_{i=1}^n \mathcal{A}_i$ the joint action space, $H \in \mathbb{N}_{>0}$ the horizon. $\mathcal{P}_h : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$ is the environment's transition function at time $h \in [H]$ and $\mathcal{P}_h(s'|s,a)$ denotes the probability of moving to state $s'$ from state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ at step $h \in [H]$, $r_{i,h} : \mathcal{S} \times \mathcal{A} \to [0,1]$ is agent $i$'s reward function at step $h \in [H]$ and $\mu \in \Delta_{\mathcal{S}}$ denotes the initial state distribution. We assume $\mathcal{S}$ and $\mathcal{A}$ to be finite.

**Policies:** For every agent $i \in [n]$, we define its policy space as $\Pi^i \triangleq \big\{\{\pi_{i,h}\}_{h=1}^H \mid \pi_{i,h} : \mathcal{S} \to \Delta_{\mathcal{A}_i}, \forall h \in [H]\big\}$. If agent $i$ follows a policy $\pi \in \Pi^i$, it means that at step $h \in [H]$ and state $s \in \mathcal{S}$, the agent samples its next action from $\pi_h(\cdot|s)$. We denote by $\Pi \triangleq \big\{\boldsymbol{\pi} = (\pi_1, ..., \pi_n) \,|\pi_i \in \Pi^i, \forall i \in [n]\big\}$ the set of *joint* policies. For any policy $\boldsymbol{\pi} \in \Pi$ and agent $i \in [n]$, we denote by $\boldsymbol{\pi}_{-i}$ the policy of the *other* $n-1$ agents.

**Value Function:** For any policy $\boldsymbol{\pi} \in \Pi$ and agent $i \in [n]$, the value function $V^{r_i}(\boldsymbol{\pi})$ measures the expected, cumulative reward of agent $i$, and is defined as follows:

$$V^{r_i}(\boldsymbol{\pi}) \triangleq \mathop{\mathbb{E}}_{\substack{s \sim \mu, \\ a_h \sim \boldsymbol{\pi}_h(\cdot|s_h), \\ s_{h+1} \sim \mathcal{P}_h(\cdot|s_h, a_h)}} \big[\sum_{h=1}^H r_{i,h}(s_h, a_h) | s_0 = s\big]. \tag{1}$$

**Potential Function:** An MPG possesses an underlying potential function $\Phi : \Pi \to \mathbb{R}$ such that:

$$V^{r_i}(\pi_i, \boldsymbol{\pi}_{-i}) - V^{r_i}(\pi_i', \boldsymbol{\pi}_{-i}) = \Phi(\pi_i, \boldsymbol{\pi}_{-i}) - \Phi(\pi_i', \boldsymbol{\pi}_{-i}) \qquad \forall \pi_i' \in \Pi^i, \forall \boldsymbol{\pi} \in \Pi, \forall i \in [n]. \tag{2}$$

This is an adaptation of the potential function defined in Leonardos et al. (2022) to the finite-horizon setting. Instead of defining a per-state potential function, we directly consider the potential function with respect to the initial distribution $\mu$.

**Remark:** Note that the potential function is a property of the MPG and is typically not known to the agents. In a cooperative game, the agents have one shared reward function $r$ such that $r_i \equiv r, \forall i \in [n]$. In this case, the potential function is simply the value function of the agents, i.e., $\Phi = V^r$. Note, however, that cooperative games are a *strict* subset of MPGs, and MPGs have the ability to express non-cooperative scenarios, such as traffic congestion. In Section 7, we describe different instances in detail.

## 3.2 Constrained Markov Potential Games

An $n$-agent *Constrained Markov Potential Game* (CMPG) is an MPG $\mathcal{G} = (\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^n, H, \{\mathcal{P}_h\}_{h=1}^H, \{\{r_{i,h}\}_{h=1}^H\}_{i=1}^n, \mu)$ with constraints $\{(\{c_{j,h}\}_{h=1}^H, \alpha_j)\}_{j=1}^k$, where $c_{j,h} : \mathcal{S} \times \mathcal{A} \to [0,1]$ denotes the $j$-th cost function at step $h \in [H]$ and $\alpha_j \in [0, H]$ is the constraint threshold.[3]

**Feasible Policies:** We call a policy $\boldsymbol{\pi} \in \Pi$ *feasible*, if it satisfies the following constraints:

$$V_\mu^{c_j}(\boldsymbol{\pi}) \triangleq \mathop{\mathbb{E}}_{\substack{s \sim \mu, \\ a_h \sim \boldsymbol{\pi}_h(\cdot|s_h), \\ s_{h+1} \sim \mathcal{P}_h(\cdot|s_h, a_h)}} \big[\sum_{h=1}^H c_{j,h}(s_h, a_h) \Big| s_0 = s\big] \leq \alpha_j, \quad \forall j \in [k].$$

[3]Even though we define our problem in the finite-horizon setting, our results can be easily extended to the discounted, infinite-horizon setting.

In the rest of the paper, we use $\Pi_C$ to refer to the set of *feasible* policies. For every agent $i$ and policy $\boldsymbol{\pi}_{-i}$ of the other $n-1$ agents, we define $\Pi_C^i(\boldsymbol{\pi}_{-i}) \triangleq \{\pi_i \in \Pi^i | (\pi_i, \boldsymbol{\pi}_{-i}) \in \Pi_C\}$. We refer to this type of constraints as *coupled* constraints, as the values of the constraints depend on the *joint* actions of the agents. If we wish to model an intersection in a traffic scenario, an important constraint to incorporate would be collision avoidance. To decide whether a certain set of actions causes a collision or not, we need to take the actions of *all* agents at the intersection into account.

In a CMPG, each agent $i$ aims to maximize its own value function $V^{r_i}$. Since the rewards and transitions depend on the *joint* policy, it may not be possible to find a policy that is globally optimal for all value functions simultaneously. Instead, the agents typically need to settle for an equilibrium policy, at which no agent has an incentive to deviate unilaterally. Many different types of equilibria exist in the literature, such as the Nash equilibrium (Nash et al., 1950), correlated equilibrium (Aumann, 1987) or Stackelberg equilibrium (Breton et al., 1988). In this work, our goal is to obtain a *Nash equilibrium policy* (Nash et al., 1950; Altman and Shwartz, 2000) in a CMPG. We define a relaxed notion in the following paragraph.

**$\varepsilon$-Nash Equilibrium Policy:** For any $\varepsilon \geq 0$, a policy $\boldsymbol{\pi}^* = (\pi_1^*, ..., \pi_n^*) \in \Pi_C$ is a *$\varepsilon$-Nash equilibrium policy*, if it is the $\varepsilon$-best-response policy for each agent, i.e.,[4]:

$$\max_{\pi_i \in \Pi_C^i(\boldsymbol{\pi}_{-i}^*)} V^{r_i}(\pi_i, \boldsymbol{\pi}_{-i}^*) - V^{r_i}(\boldsymbol{\pi}^*) \leq \varepsilon, \qquad \forall i \in [n]. \tag{3}$$

We call $\boldsymbol{\pi}^*$ a *Nash equilibrium policy*, if Eq. (3) holds with $\varepsilon = 0$. In the rest of the paper, we refer to the Nash equilibrium policy as *Nash policy*.

### 3.3 Constrained Markov Decision Processes

A *Constrained Markov Decision Process* (CMDP) is a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, \{\mathcal{P}_h\}_{h=1}^H, \{r_h\}_{h=1}^H, \mu, \{(\{c_{j,h}\}_{h=1}^H, \alpha_j)\}_{j=1}^k)$. In a CMDP, there is a *single* agent. However, the individual elements in $\mathcal{M}$ carry the same meaning as in CMPGs. Furthermore, the policy sets $\Pi, \Pi_C$ and the value functions $V^r : \Pi \to \mathbb{R}$ (reward), $V^{c_j} : \Pi \to \mathbb{R}, j \in [k]$ (costs) are defined in the same way as for CMPGs. In a CMDP, the agent aims to find a policy $\pi^*$, that satisfies:

$$\pi^* \in \arg\max_{\pi \in \Pi_C} V^r(\pi). \tag{4}$$

In the following section, we prove that a Nash policy in a CMPG can be found by maximizing the potential function with respect to the given constraints, similar to Eq. (4). We will show that Lagrangian duality, a common approach for constrained optimization, will not work in general for CMPGs.

## 4 Duality for Constrained Markov Potential Games?

For an MPG with potential function $\Phi$, a globally optimal policy $\boldsymbol{\pi}^* \in \arg\max_{\boldsymbol{\pi} \in \Pi} \Phi(\boldsymbol{\pi})$ is also a Nash policy (Leonardos et al., 2022). We show in Proposition 1 that this property generalizes to CMPGs. We defer the proofs for all theoretical results in this section to Appendix A.

**Proposition 1.** *Define the following constrained optimization problem:*

$$\boldsymbol{\pi}^* \in \arg\max_{\boldsymbol{\pi} \in \Pi_C} \Phi(\boldsymbol{\pi}). \tag{5}$$

*Then, $\boldsymbol{\pi}^*$ is a Nash policy for a CMPG with potential function $\Phi$.*

Solving Eq. (5) directly is not trivial; even if the agents know the rewards and transitions, the potential function is usually not known. Moreover, the fact that we have *coupled* constraints makes solving Eq. (5) directly intractable. Nevertheless, a common approach for solving constrained optimization problems is *Lagrangian duality*, which, in our case, turns the CMPG into an (unconstrained) MPG with modified rewards (Proposition 2). This would enable the use of scalable algorithms that have been developed for unconstrained MPGs (Leonardos et al., 2022). Furthermore, in previous works (Liu et al., 2021a; Diddigi et al., 2019), Lagrangian duality was used for cooperative CMPGs and showed promising experimental results. This makes Lagrangian duality a tempting approach for

[4]This is an extension of the *generalized Nash equilibrium* (Facchinei and Kanzow, 2010) to CMPGs.

CMPGs. For this, we define the *Lagrangian* $\mathcal{L} : \Pi \times \mathbb{R}_+^k \to \mathbb{R}$ and the primal[5] and dual problems for Eq. (5) as follows:

$$\mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\lambda}) \triangleq \Phi(\boldsymbol{\pi}) + \sum_{j=1}^{k} \lambda_j \left(\alpha_j - V^{c_j}(\boldsymbol{\pi})\right) \tag{Lagrangian}$$

$$P^* = \max_{\boldsymbol{\pi} \in \Pi} \min_{\boldsymbol{\lambda} \in \mathbb{R}_+^k} \mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\lambda}) \tag{Primal}$$

$$D^* = \min_{\boldsymbol{\lambda} \in \mathbb{R}_+^k} \max_{\boldsymbol{\pi} \in \Pi} \mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\lambda}). \tag{Dual}$$

As a first step, in Proposition 2, we prove that the dual problem does indeed correspond to an (unconstrained) MPG.

**Proposition 2.** *For any $\boldsymbol{\lambda} \in \mathbb{R}_+^k$, $\mathcal{L}(\cdot, \boldsymbol{\lambda})$ is a potential function for an MPG with reward functions $\tilde{r}_{i,h} \triangleq r_{i,h} - \sum_{j=1}^{k} \lambda_j c_{j,h}, \forall i \in [n], \forall h \in [H]$.*

Then, weak duality guarantees that $D^* \geq P^*$ holds. Unfortunately, in the following proposition, however, we show that *strong duality*, i.e., $D^* = P^*$, *does not hold* in general for CMPGs.

**Proposition 3.** *There exists a CMPG, for which strong duality does not hold, i.e., for which $P^* \neq D^*$.*

To give an intuition on Proposition 3, consider a cooperative CMPG with $\Phi \equiv V^r$, i.e., the potential function is equal to the shared value function $V^r$. Note that, in this case, the primal problem very much resembles the CMDP objective (Eq. (5)) and it is tempting to solve the CMPG as a CMDP with a large action space $\mathcal{A} = \times_{i=1}^{n} \mathcal{A}_i$. Recall also, that strong duality does indeed hold for CMDPs (Paternain et al., 2019) and CMDPs can be solved via primal-dual algorithms. By solving this large CMDP, we obtain a solution $\boldsymbol{\pi}^*$ that specifies distributions over the *joint* action space $\mathcal{A}$. To obtain a solution for the original CMPG, however, we require a policy that can be factored into a set of independent policies $\{\pi_i^*\}_{i \in [n]}$ such that $\boldsymbol{\pi}_h^*(a|s) = \prod_{i=1}^{n} \pi_{i,h}^*(a_i|s), \forall (s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$. We show in Appendix A that unfortunately, this property is not always guaranteed, implying that already for a simple class of CMPGs, strong duality may not always hold.

## 5 Solving Constrained Markov Potential Games

In this section, we propose an efficient algorithm to compute Nash policies in CMPGs[6]. Similar to the work on unconstrained MPGs by Song et al. (2021), in our algorithm **C**oordinate-**A**scent for **CMPG**s (CA-CMPG), agents take turns to solve a *Constrained Markov Decision Process* (CMDP), i.e., a single-agent reinforcement learning problem, in every iteration. To do this, the agents need to coordinate, such that, when one agent is solving the CMDP, the others provide a stationary environment to that agent by keeping their policies fixed. There are some technical challenges compared to the unconstrained MPG setting. The main difference is that in the CMPG setting, to ensure the convergence to a Nash policy, we need also to ensure that the intermediate policies remain *feasible* (see remark at the end of this section). Our algorithm CA-CMPG is described in Algorithm 1.

We assume for now that the agents know their own reward functions, the cost functions as well as the transition model. As a starting point for CA-CMPG, the agents require access to a feasible, initial policy, which we state in the following assumption:

**Assumption 1.** *Given a CMPG, the agents have access to a feasible policy $\boldsymbol{\pi}^S \in \Pi_C$.*

This type of assumption is common for safe exploration in CMDPs (Bura et al., 2022; Liu et al., 2021b). We discuss in Appendix B, why we require it for our setting. While, in general, it may be computationally hard to compute a feasible $\boldsymbol{\pi}^S$ in the multi-agent setting, we provide examples in Appendix B, for which it is easy to compute $\boldsymbol{\pi}^S$. In CA-CMPG, the agents start with the feasible policy $\boldsymbol{\pi}^S$. In every iteration, the agents take turns to maximize their own value function. While one agent is maximizing its value function, the other agents keep their policy fixed (Line 4); therefore, that agent is essentially solving a CMDP. We defer the exact description of the CMDP that agent $i$ faces in iteration $t$ to Appendix B. Let us recall the CMDP objective from Eq. (4). In practice, we can

[5]Note that the primal is equivalent to Eq. (5).

[6]Note that we may not find a Nash policy that solves Eq. (5) though.

**Algorithm 1** CA-CMPG (Known Transitions)

**Require:** $\varepsilon > 0$ (approximation error), $\boldsymbol{\pi}^S \in \Pi_C$ (feasible policy), $T$ (number of iterations)

$\boldsymbol{\pi}^0 \leftarrow \boldsymbol{\pi}^S$
**for** $t = 1, ..., T$ **do**
  **for** agent $i = 1, ..., n$ **do**
    Agent $i$ computes $\hat{\pi}_i^t$ such that Eq. (6) is satisfied.
    $\varepsilon_i^t \leftarrow V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) - V^{r_i}(\boldsymbol{\pi}^{t-1})$.
  **if** $\max_{i\in[n]} \varepsilon_i^t > \varepsilon/2$ **then**
    Set $\boldsymbol{\pi}^t = (\hat{\pi}_j^t, \boldsymbol{\pi}_{-j}^{t-1})$, where $j = \arg\max_{i\in[n]} \varepsilon_i^t$, break ties arbitrarily.
  **else**
    **break**

only solve Eq. (4) *approximately*. Given $\varepsilon > 0$, we assume that in every iteration $t$, agent $i \in [n]$ can efficiently compute a policy $\hat{\pi}_i^t \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{t-1})$ such that it satisfies the following conditions[7]:

$$\max_{\pi\in\Pi_C^i(\boldsymbol{\pi}_{-i}^{t-1})} V^{r_i}(\pi, \boldsymbol{\pi}_{-i}^{t-1}) - V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) \leq \varepsilon/2. \tag{6}$$

Due to the potential property (Eq. (2)), if agent $i \in [n]$ improves its own value function, it implicitly also improves the potential function. To prove that the potential function can be increased only a finite number of times, implying termination of CA-CMPG, we require the potential function to be bounded.

**Lemma 1.** *Fix an arbitrary base policy $\boldsymbol{\pi}^B \in \Pi$. Then, for every $\boldsymbol{\pi} \in \Pi$, the potential function can be bounded as: $\Phi(\boldsymbol{\pi}) \leq nH + \Phi(\boldsymbol{\pi}^B)$.*

We defer the proofs to all theoretical results in this section to Appendix B. CA-CMPG terminates when the agents cannot deviate unilaterally and improve their value function by more than $\varepsilon$, i.e., when they reach an $\varepsilon$-Nash policy. We state this result in the following theorem:

**Theorem 1.** *Suppose that Assumption 1 holds. Then, given $\varepsilon > 0$, if we invoke CA-CMPG with $T = \frac{2nH}{\varepsilon}$, it converges to an $\varepsilon$-Nash policy.*

**Remark:** What if we relax the feasibility requirement in Eq. (6) and allow the CMDP solver to return an *$\varepsilon$-feasible* policy $\boldsymbol{\pi}$ such that $V^{c_j}(\boldsymbol{\pi}) \leq \alpha_j + \varepsilon, \forall j \in [k]$, for an $\varepsilon > 0$? In that case, the intermediate policies might not be feasible and CA-CMPG may get stuck in an infeasible policy, which is not a Nash policy.

## 6 Learning in Unknown Constrained Markov Potential Games

In this section, we assume that the agents do not know the transition model beforehand. For simplicity, we assume that they do know the rewards and costs[8]. Our objective is to establish a *sample complexity* bound for learning in CMPGs. Concretely, we want to construct an algorithm, such that, given any $\varepsilon > 0, \delta \in (0, 1)$, the algorithm returns an $\varepsilon$-Nash policy with probability at least $1 - \delta$, using at most $\mathcal{F}(\varepsilon, \delta)$ *samples* from the transition model $\mathcal{P}$. Before we proceed, we define an important quantity related to the constraint set, which also contributes to the final sample complexity.

**Definition 1** (Slater constant)**.** *Given a feasible CMPG $\mathcal{G}$, we define its Slater constant $\zeta$ as follows:*

$$\zeta \triangleq \min_{j\in[k]} \min_{i\in[n]} \min_{\boldsymbol{\pi}_{-i}\in\Pi\setminus\Pi^i} \max_{\pi\in\Pi^i} \{\alpha_j - V^{c_j}(\pi, \boldsymbol{\pi}_{-i})\}.$$

*We call $\mathcal{G}$* strictly *feasible if and only if $\zeta > 0$.*

In the rest of this section, we assume that the agents face an unknown, strictly feasible CMPG with Slater constant $\zeta > 0$. Next, we discuss which parts of CA-CMPG need to be adapted for this setting.

[7]This can be achieved using state-of-the-art primal-dual methods, such as the work by Ding et al. (2020); Paternain et al. (2019).

[8]In general, learning the transitions is harder than learning rewards and costs. Concretely, this also means that learning rewards and costs will not add any dominating terms to the overall sample complexity (see Vaswani et al. (2022)).

1. In every iteration $t$, each agent $i \in [n]$ needs to solve the CMDP described in Section 5 (Line 4). To solve this CMDP, we assume access to a *sample-efficient* CMDP solver, which has the following guarantees: Given $\varepsilon > 0, \delta \in (0,1)$, the solver uses at most $\mathcal{F}_C\left(|\mathcal{S}|, |\mathcal{A}_i|, H, \zeta, \delta, \frac{\varepsilon}{4}\right)$ samples and returns a policy $\hat{\pi}_i^t \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{t-1})$ such that it satisfies the following, with probability at least $1-\delta$:

$$\max_{\pi \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{t-1})} V^{r_i}(\pi, \boldsymbol{\pi}_{-i}^{t-1}) - V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) \leq \varepsilon/4, \tag{7}$$

   Compared to the setting with known transitions, we have a stricter bound on the approximation error of $\varepsilon/4$ here. We discuss in Appendix C, why we require this.

2. To compute $\varepsilon_i^t$ in step $t$, agent $i$ needs to estimate the value functions $V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})$ and $V^{r_i}(\boldsymbol{\pi}^{t-1})$. For the former, the agents execute the policy $(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})$ for $M > 0$ episodes[9] and agent $i$ estimates $\hat{V}^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})$ with the average of the observed, cumulative rewards. For the latter, similarly, the agents execute $\boldsymbol{\pi}^{t-1}$ for $M$ episodes, but these observations can be used to estimate $V^{r_1}(\boldsymbol{\pi}^{t-1}), ..., V^{r_n}(\boldsymbol{\pi}^{t-1})$ simultaneously[10].

The resulting algorithm **C**oordinate-**A**scent for **CMPGs** with **E**xploration (CA-CMPG-E) is described in Algorithm 2 (cf. Appendix C).

**Theorem 2.** *Given a strictly feasible CMPG $\mathcal{G}$ with Slater constant $\zeta > 0$, suppose that the agents have access to an initial feasible policy (cf. Assumption 1). Furthermore, assume that the agents have access to a sample-efficient CMDP solver (Eq.* (7)*). Then, for any $\varepsilon > 0$, $\delta \in (0,1)$, CA-CMPG-E invoked with $M = \frac{32H^2}{\varepsilon^2}\log\left(\frac{32n^2H}{\varepsilon\delta}\right)$ and $T = \frac{4nH}{\varepsilon}$ returns an $\varepsilon$-Nash policy with probability at least $1-\delta$, using the following number of samples:*

$$\mathcal{F}(\varepsilon, \delta) \triangleq \sum_{t=1}^{T}\sum_{i=1}^{n} \mathcal{F}_C\left(|S|, |\mathcal{A}_i|, H, \zeta, \frac{\varepsilon\delta}{8n^2H}, \frac{\varepsilon}{4}\right) + \frac{256n^2H^4}{\varepsilon^3}\log\left(\frac{32n^2H}{\varepsilon\delta}\right).$$

In the next two sub-sections, we will instantiate CA-CMPG-E with two different state-of-the-art CMDP solvers and state the resulting sample complexity bounds. Both algorithms are designed for CMDPs with a *single* constraint. Due to this, we set $k = 1$ and denote our cost function by $\{c_h\}_{h=1}^H$ and refer to the constraint parameter as $\alpha$. Note that this is due to a limitation of the existing CMDP algorithms and not of CA-CMPG-E.

### 6.1 Generative model

In this section, we assume that the agents have access to a *generative model*, i.e., they can directly obtain samples from the transition model $\mathcal{P}_h(\cdot|s,a)$, for any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ and any $h \in [H]$. Similar to previous results in CMDPs (Vaswani et al., 2022) we propose a novel algorithm for finite-horizon CMDPs and describe it in Algorithm 3 (cf. Appendix D). Lemma 2 (cf. Appendix D) establishes the sample complexity for Algorithm 3.

**Corollary 1.** *Given a strictly feasible CMPG $\mathcal{G}$, assume that its Slater constant $\zeta > 0$ is known. Furthermore, assume that the agents invoke Algorithm 3 with $\varepsilon' = \frac{\varepsilon}{4}$, $\delta' = \mathcal{O}\left(\frac{\varepsilon\delta}{n^2H}\right)$ and parameters set as in Lemma 2 to solve Eq.* (7)*. Then, for any $\varepsilon > 0, \delta \in (0,1)$, CA-CMPG-E invoked with $M = \mathcal{O}\left(\frac{H^2}{\varepsilon^2}\log\left(\frac{nH}{\varepsilon\delta}\right)\right)$ and $T = \frac{4nH}{\varepsilon}$, returns an $\varepsilon$-Nash policy with probability at least $1-\delta$ with an overall sample complexity of:*

$$\mathcal{F}(\varepsilon, \delta) \leq \widetilde{\mathcal{O}}\left(\frac{n|\mathcal{S}|H^8\log\left(\frac{1}{\varepsilon\delta}\right)\sum_{i=1}^n|\mathcal{A}_i|}{\varepsilon^3\zeta^2} + \frac{n^2H^4\log\left(\frac{1}{\varepsilon\delta}\right)}{\varepsilon^3}\right).$$

**Remark:** Compared to the result for *unconstrained* MPGs (Song et al., 2021, Theorem 7), our Corollary 1 has an additional dependence on $\frac{1}{\zeta^2}$ and a worse dependence on the horizon $H$. These are due to the fact that our CMDP solver must always return a *feasible* policy. Finally, the sample complexity result in Song et al. (2021) explicitly depends on $\Phi_{max} \triangleq \max_{\boldsymbol{\pi}\in\Pi}\Phi(\boldsymbol{\pi})$, whereas we substituted $\Phi_{\max} \leq nH$ (Lemma 1).

[9] Each episode is a sequence of $H$ steps. At the beginning of each episode, the initial state is freshly sampled from $\mu$.

[10] This holds because we assumed that the reward functions are known.

### 6.2 Safe exploration without a generative model

We now consider the more challenging setting where the agents do not have access to a generative model, but can only explore by executing policies and observing the transitions. Moreover, during the learning process, we want to ensure that the agents explore *safely*. Existing algorithms with safe exploration (Bura et al., 2022; Liu et al., 2021b) have guarantees on the *regret*, but no sample complexity guarantees. To address this, we derive a sample complexity bound for the algorithm by Bura et al. (2022) (Algorithm 4 in Appendix E) in Lemma 14. To apply this CMDP solver in CA-CMPG-E, we need to ensure that in every iteration, the agents have access to a *strictly* feasible policy. We state a stronger condition in the following assumption.

**Assumption 2.** *There exists $c \in (0, \zeta]$ s.t. for any agent $i \in [n]$ and policy $\boldsymbol{\pi}_{-i} \in \Pi_C \setminus \Pi^i$ of the other agents, the agent can obtain a strictly feasible policy $\pi \in \Pi^i$ s.t. $V^c(\pi, \boldsymbol{\pi}_{-i}) \leq \alpha - c$.*

This is a stronger assumption than in Section 6.1, as we additionally require access to a strictly feasible policy for every CMDP that is solved in CA-CMPG-E.

**Corollary 2.** *Suppose that Assumption 2 holds. Given $\varepsilon > 0, \delta \in (0, 1)$, assume that we invoke CA-CMPG-E with $M = \mathcal{O}\left(\frac{H^2}{\varepsilon^2} \log\left(\frac{nH}{\varepsilon\delta}\right)\right)$ and $T = \frac{4nH}{\varepsilon}$. Furthermore, assume that we use Algorithm 4 as CMDP solver with $\varepsilon' = \frac{\varepsilon}{4}$, $\delta' = \mathcal{O}\left(\frac{\varepsilon\delta}{n^2 H}\right)$ and parameters set as in Lemma 14. Then, CA-CMPG-E returns an $\varepsilon$-Nash policy with probability at least $1 - \delta$ with an overall sample complexity of:*

$$\mathcal{F}(\varepsilon, \delta) \leq \widetilde{\mathcal{O}}\left(\frac{n|\mathcal{S}|^2 H^{10} \log\left(\frac{1}{\varepsilon\delta}\right) \sum_{i=1}^{n} |\mathcal{A}_i|}{\varepsilon^5 c^2} + \frac{n^2 H^4 \log\left(\frac{1}{\varepsilon\delta}\right)}{\varepsilon^3}\right).$$

Note that to satisfy Assumption 2, any $c \in (0, \zeta]$ is a valid choice. A large $c$ yields a better sample complexity for Corollary 2, but restricts the set of strictly feasible policies for the CMDP solver. A smaller $c$ increases the sample complexity, but gives more flexibility, as it allows for a larger set of strictly feasible policies. Comparing the two corollaries, we observe that safe exploration without a generative model leads to a worse dependence of $|S|$, $H$ and $\varepsilon$.

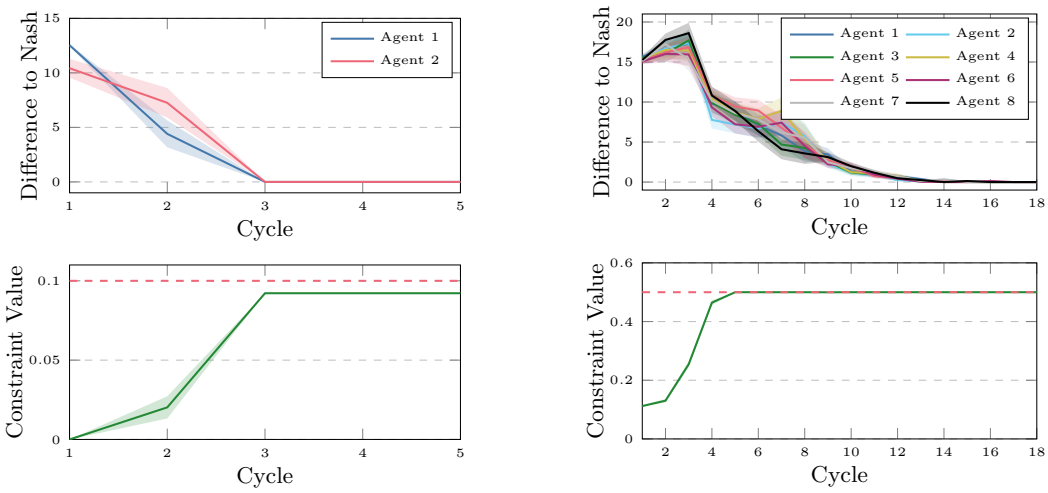
## 7 Experiments

**Grid world:** We consider a cooperative CMPG with two agents, in which the agents navigate in a 4x4 grid world (cf. Appendix F). Each cell in the grid represents a state and in every state, each agent can choose to move *up*, *right*, *down* or *left*. State transitions are deterministic and if an agent selects an action that would make it leave the grid, it remains in the current state. Fig. 4a illustrates the rewards that an agent can obtain in the individual states. Both agents start from the bottom left state and their goal is to reach the target state, which is the state with a reward of 10. To model this as a cooperative game, we set the agents' joint objective to be the sum of their individual rewards. Whenever the agents are on the same state, excluding the start and target states, they *collide* and incur a cost of 1. The agents must keep the expected cost below a pre-defined threshold $\alpha \in [0, 1]$.

We evaluate our algorithm CA-CMPG with known transitions and use a primal-dual algorithm as CMDP solver. We set the horizon to $H = 6$ and use a threshold of $\alpha = 0.1$. Fig. 1a (top row) displays the reward differences between the current policy and the new policy for both agents and after every cycle of the algorithm, averaged over 20 runs. One cycle corresponds to one full iteration of Algorithm 2, i.e. all agents solving their CMDPs. When the reward differences reach zero for both agents, this implies that the agents have converged to a Nash policy. The bottom row tracks the cost over the cycles of Algorithm 2. The agents start from a strictly feasible policy with a cost of 0, and converge to a policy with a cost close to $\alpha$. We provide more details about the resulting Nash policies in Appendix F. Note also, that for this experiment, solving the dual problem may lead to an infeasible policy (cf. Appendix F).

**Congestion game:** We consider a finite-horizon version of the setup described in Leonardos et al. (2022) in which every state is a congestion game[11]. The game consists of two states $\mathcal{S} = \{\texttt{safe}, \texttt{unsafe}\}$, $N$ agents and action space $\mathcal{A} = \{A, B, C, D\}$ for every agent. Each action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$ has a weight $w_a^s > 0$ associated with it. In the safe state, an agent that

[11]Note that every congestion game is also a potential game and vice versa (Monderer and Shapley, 1996); however, the setup considered here may not necessarily be MPGs as pointed out in Leonardos et al. (2022).




(a) Grid world

(b) Congestion game

Figure 1: These plots illustrate the results of the grid world (Fig. 1a) and congestion game (Fig. 1b) experiments. One cycle on the x-axis corresponds to one full iteration of Algorithm 2, i.e. all $N$ agents solving their CMDPs. The top row displays, for each agent, an average of their reward difference between the current and new policy. When the difference reaches zero, they have converged to a Nash policy. The bottom row tracks the averaged cost over the cycles of Algorithm 2 (green, solid line). The agents start from a strictly feasible policy and converge to a cost close to $\alpha$ (red, dashed line). In all plots, we additionally also plot the standard error.

selects action $a \in \mathcal{A}$, receives a reward of $k_a \cdot w_a^{\texttt{safe}}$, where $k_a$ denotes the number of agents that selected action $a$. In the unsafe state, the reward structure is similar, however, we subtract an offset $c \geq 0$, resulting in a reward of $k_a \cdot w_a^{\texttt{unsafe}} - c$. In both states $s \in \mathcal{S}$, the weights follow the order $w_A^s < w_B^s < w_C^s < w_D^s$. Thus, in both states, the agents prefer to take the action that is chosen by most agents. Furthermore, for every action $a \in \mathcal{A}$, $k_a \cdot w_a^{\texttt{safe}} \gg k_a \cdot w_a^{\texttt{unsafe}} - c$ s.t. the agents prefer to stay in the safe state. In the safe state, if more than $N/2$ agents choose the same action, the system transitions to the unsafe state. To get back to the safe state from the unsafe state, the agents must equally distribute themselves among the four actions (cf. Appendix F).

We evaluate our algorithm CA-CMPG with $N = 8$ agents and a horizon of $H = 2$. Furthermore, we assume that the transitions are known and use a linear program to solve the CMDPs (Altman, 1999). For the initial state, we set $\mu(\texttt{safe}) = \mu(\texttt{unsafe}) = 0.5$. At step $h = 1$, in the unsafe state, if more than $N/2$ agents select the same action, the agents incur a cost of 1. Their goal is to keep the cost below a threshold $\alpha = 0.5$. Fig. 1b (top row) displays, as before, the reward differences between the current policy and the new policy, for each agents and averaged over 50 runs. When this difference reaches zero, this implies that the agents have converged to a Nash policy. The bottom plots track the cost over the cycles of Algorithm 2. The agents start from a strictly feasible policy with a cost of 0, and converge to a value close to $\alpha$. We provide more details about the resulting Nash policies in Appendix F.

## 8 Conclusion

In this paper, we proved that strong duality does not hold always hold in CMPGs. An interesting future question could be to understand under which conditions primal-dual methods may work for CMPGs. To tackle CMPGs, we presented our algorithm CA-CMPG, which provably converges to an $\varepsilon$-Nash policy.[12] Furthermore, we established the first sample complexity bound for learning in CMPGs. In CA-CMPG, exploration happens only within the CMDP sub-routines. It would be interesting to understand whether the sample complexity bound for the generative model setting (Section 6.1) can be made tighter if we move the exploration outside the CMDP sub-routines.

[12]Note that while this paper is written in the finite-horizon setting, our algorithm CA-CMPG can be adapted to the discounted, infinite-horizon setting by using an appropriate CMDP solver as a sub-routine.

# References


Eitan Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

Eitan Altman and Adam Shwartz. Constrained markov games: Nash equilibria. In *Advances in dynamic games and applications*, pages 213–221. Springer, 2000.

Eitan Altman, Thomas Boulogne, Rachid El-Azouzi, Tania Jiménez, and Laura Wynter. A survey on networking games in telecommunications. *Computers & Operations Research*, 33(2):286–311, 2006.

Robert J Aumann. Correlated equilibrium as an expression of bayesian rationality. *Econometrica: Journal of the Econometric Society*, pages 1–18, 1987.

Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena scientific Belmont, MA, 1995.

Vivek S. Borkar. A convex analytic approach to markov decision processes. *Probability Theory and Related Fields*, 1988.

Michele Breton, Abderrahmane Alj, and Alain Haurie. Sequential stackelberg equilibria in two-person games. *Journal of Optimization Theory and Applications*, 59(1):71–97, 1988.

Archana Bura, Aria Hasanzadezonuzy, Dileep Kalathil, Srinivas Shakkottai, and Jean-Francois Chamberland. DOPE: Doubly Optimistic and Pessimistic Exploration for Safe Reinforcement Learning. October 2022. URL `https://openreview.net/forum?id=U4BUMoVTrB2`.

Dingyang Chen, Qi Zhang, and Thinh T. Doan. Convergence and Price of Anarchy Guarantees of the Softmax Policy Gradient in Markov Potential Games. June 2022. URL `https://openreview.net/forum?id=pe2ZGTUxVvJ`.

Qiwen Cui, Kaiqing Zhang, and Simon Du. Breaking the curse of multiagents in a large state space: Rl in markov games with independent linear function approximation. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 2651–2652. PMLR, 2023.

Raghuram Bharadwaj Diddigi, Sai Koti Reddy Danda, Prabuchandran K. J., and Shalabh Bhatnagar. Actor-critic algorithms for constrained multi-agent reinforcement learning. *CoRR*, abs/1905.02907, 2019. URL `http://arxiv.org/abs/1905.02907`.

Dongsheng Ding, Kaiqing Zhang, Tamer Basar, and Mihailo R Jovanovic. Natural policy gradient primal-dual method for constrained markov decision processes. In *NeurIPS*, 2020.

Dongsheng Ding, Chen-Yu Wei, Kaiqing Zhang, and Mihailo Jovanovic. Independent policy gradient for large-scale Markov potential games: Sharper rates, function approximation, and game-agnostic convergence. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5166–5220. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/ding22b.html`.

Yonathan Efroni, Shie Mannor, and Matteo Pirotta. Exploration-exploitation in constrained mdps. *arXiv preprint arXiv:2003.02189*, 2020.

Francisco Facchinei and Christian Kanzow. Generalized nash equilibrium problems. *Annals of Operations Research*, 175(1):177–211, 2010.

Roy Fox, Stephen M Mcaleer, Will Overman, and Ioannis Panageas. Independent natural policy gradient always converges in markov potential games. In *International Conference on Artificial Intelligence and Statistics*, pages 4414–4425. PMLR, 2022.

Javier García, Fern, and O Fernández. A Comprehensive Survey on Safe Reinforcement Learning. *Journal of Machine Learning Research*, 16(42):1437–1480, 2015. ISSN 1533-7928. URL `http://jmlr.org/papers/v16/garcia15a.html`.

Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? *arXiv preprint arXiv:1807.03765*, 2018.

Stefanos Leonardos, Will Overman, Ioannis Panageas, and Georgios Piliouras. Global convergence of multi-agent policy gradient in markov potential games. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=gfwON7rAm4`.

Chenyi Liu, Nan Geng, Vaneet Aggarwal, Tian Lan, Yuan Yang, and Mingwei Xu. Cmix: Deep multi-agent reinforcement learning with peak and average constraints. In *Proceedings of the 2021 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2021), Virtual Conference*, pages 13–17, 2021a.

Tao Liu, Ruida Zhou, Dileep Kalathil, PR Kumar, and Chao Tian. Learning policies with zero or bounded constraint violation for constrained mdps. *arXiv preprint arXiv:2106.02684*, 2021b.

Chinmay Maheshwari, Manxi Wu, Druv Pai, and Shankar Sastry. Independent and Decentralized Learning in Markov Potential Games. 2022. URL `http://arxiv.org/abs/2205.14590`.

Weichao Mao, Lin Yang, Kaiqing Zhang, and Tamer Basar. On Improving Model-Free Algorithms for Decentralized Multi-Agent Reinforcement Learning. In *Proceedings of the 39th International Conference on Machine Learning*, pages 15007–15049. PMLR, June 2022. URL `https://proceedings.mlr.press/v162/mao22a.html`. ISSN: 2640-3498.

Dov Monderer and Lloyd S Shapley. Potential games. *Games and economic behavior*, 14(1):124–143, 1996.

John F Nash et al. Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49, 1950.

P. Parnika, Raghuram Bharadwaj Diddigi, Sai Koti Reddy Danda, and Shalabh Bhatnagar. Attention actor-critic algorithm for multi-agent constrained co-operative reinforcement learning. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '21, page 1616–1618, Richland, SC, 2021. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450383073.

Santiago Paternain, Luiz FO Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. *arXiv preprint arXiv:1910.13393*, 2019.

Ziang Song, Song Mei, and Yu Bai. When Can We Learn General-Sum Markov Games with a Large Number of Players Sample-Efficiently? 2021. URL `http://arxiv.org/abs/2110.04184`.

Sharan Vaswani, Lin Yang, and Csaba Szepesvari. Near-Optimal Sample Complexity Bounds for Constrained MDPs. October 2022. URL `https://openreview.net/forum?id=ZJ7Lrtd12x_`.

Koji Yamamoto. A comprehensive survey of potential game approaches to wireless networks. *IEICE Transactions on Communications*, E98.B(9):1804–1823, 2015. doi: 10.1587/transcom.E98.B.1804.

Yaodong Yang and Jun Wang. An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective, March 2021. URL `http://arxiv.org/abs/2011.00583`. arXiv:2011.00583 [cs].

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pages 321–384, 2021a.

Runyu Zhang, Zhaolin Ren, and Na Li. Gradient play in stochastic games: stationary points, convergence, and sample complexity, December 2021b. URL `http://arxiv.org/abs/2106.00198`. arXiv:2106.00198 [cs, math].

Runyu Zhang, Jincheng Mei, Bo Dai, Dale Schuurmans, and Na Li. On the Global Convergence Rates of Decentralized Softmax Gradient Play in Markov Potential Games. October 2022. URL `https://openreview.net/forum?id=X1oVDZIABwF`.

# Part I

# Supplementary Material

## Table of Contents


**A Duality for Constrained Markov Potential Games? (Section 4)** **12**

**B Solving Constrained Markov Potential Games (Section 5)** **15**

**C Learning in Unknown Constrained Markov Potential Games (Section 6)** **17**

**D Learning in Unknown Constrained Markov Potential Games: Generative Model (Section 6.1)** **20**

D.1 Convert mixture to a single policy . . . . . 22

D.2 Proofs of auxiliary lemmas . . . . . 23

**E Learning in Unknown Constrained Markov Potential Games - Safe Exploration Without a Generative Model (Section 6.2)** **28**

E.1 Proofs of auxiliary lemmas . . . . . 30

**F Experiments** **31**


## A Duality for Constrained Markov Potential Games? (Section 4)

This section provides the theoretical proofs for Section 4. We start by proving that a Nash policy can be found via constrained optimization in the following proposition.

**Proposition 1.** *Define the following constrained optimization problem:*

$$\boldsymbol{\pi}^* \in \arg\max_{\boldsymbol{\pi}\in\Pi_C} \Phi(\boldsymbol{\pi}). \tag{5}$$

*Then, $\boldsymbol{\pi}^*$ is a Nash policy for a CMPG with potential function $\Phi$.*

*Proof.* We prove this by contradiction. Suppose that $\boldsymbol{\pi}^*$ is not a Nash policy. Therefore, there is an agent $i \in [n]$, for which there exists a policy $\hat{\pi}_i \in \arg\max_{\pi\in\Pi_C^i(\boldsymbol{\pi}^*_{-i})} V^{r_i}\left(\pi, \boldsymbol{\pi}^*_{-i}\right)$ such that $V^{r_i}\left(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}\right) > V^{r_i}\left(\boldsymbol{\pi}^*\right)$. Thus, agent $i$ can *strictly increase* its value function by deviating to $\hat{\pi}_i$. This implies that we can also increase the potential function, i.e.

$$\begin{aligned}
&\Phi(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}) - \Phi(\boldsymbol{\pi}^*) = V^{r_i}(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}) - V^{r_i}(\boldsymbol{\pi}^*) > 0 \quad \text{(By Eq. (2).)}\\
\Rightarrow\ &\Phi(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}) > \Phi(\boldsymbol{\pi}^*).
\end{aligned}$$

Since $(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}) \in \Pi_C$ and $\Phi(\hat{\pi}_i, \boldsymbol{\pi}^*_{-i}) > \Phi(\boldsymbol{\pi}^*)$, this contradicts our assumption that $\boldsymbol{\pi}^*$ is a solution to Eq. (5). Therefore, $\boldsymbol{\pi}^*$ must be a Nash policy. □

As stated in Section 4, we make use of the constrained optimization formulation in Proposition 1 to define the Lagrangian primal and dual problems in Eq. (Primal) and Eq. (Dual), respectively. We are interested in solving the dual problem, as it corresponds to a modified, *unconstrained* MPG, which we prove in the following lemma.

**Proposition 2.** *For any $\boldsymbol{\lambda} \in \mathbb{R}^k_+$, $\mathcal{L}(\cdot, \boldsymbol{\lambda})$ is a potential function for an MPG with reward functions $\tilde{r}_{i,h} \triangleq r_{i,h} - \sum_{j=1}^k \lambda_j c_{j,h}, \forall i \in [n], \forall h \in [H]$.*

*Proof.* Consider an arbitrary $\boldsymbol{\lambda} \in \mathbb{R}^k_+$. Then, we can write the new value function, for any agent $i \in [n]$, as follows:

$$V^{\tilde{r}_i}(\boldsymbol{\pi}) = \mathbb{E}\left[\sum_{h=1}^H \tilde{r}_{i,h}(s_h, a_h)|s_0 = s\right],$$

where the expectation is taken with respect to $\mu, \boldsymbol{\pi}$ and $\mathcal{P}$. Next, we plug in the definition for $\widetilde{r}_i$ and obtain:

$$\begin{aligned} V^{\tilde{r}_i} &= \mathbb{E}\left[\sum_{h=1}^{H}\left(r_{i,h}(s_h,a_h)+\sum_{j=1}^{k}\lambda_j c_{j,h}(s_h,a_h)\right)\Big| s_0=s\right] \\ &= \mathbb{E}\left[\sum_{h=1}^{H} r_{i,h}(s_h,a_h)\Big| s_0=s\right]+\sum_{j=1}^{k}\lambda_j\mathbb{E}\left[\sum_{h=1}^{H} c_{j,h}(s_h,a_h)\Big| s_0=s\right] \qquad (8) \\ &= V^{r_i}(\boldsymbol{\pi})+\sum_{j=1}^{k}\lambda_j V^{c_j}(\boldsymbol{\pi}), \end{aligned}$$

where Eq. (8) is due to linearity of expectation. Next, we show that $\mathcal{L}(\cdot,\boldsymbol{\lambda})$ is indeed a potential function for the value functions $\left\{V^{\widetilde{r}_i}\right\}_{i\in[n]}$. For any $\boldsymbol{\pi}\in\Pi$ and $\pi_i\in\Pi^i$, we evaluate the difference in $\mathcal{L}(\cdot,\boldsymbol{\lambda})$ between $\boldsymbol{\pi}$ and $(\pi_i,\boldsymbol{\pi}_{-i})$:

$$\begin{aligned} \mathcal{L}\left((\pi_i,\boldsymbol{\pi}_{-i}),\boldsymbol{\lambda}\right)-\mathcal{L}\left(\boldsymbol{\pi},\boldsymbol{\lambda}\right) &= \left(\Phi(\pi_i,\boldsymbol{\pi}_{-i})+\sum_{j=1}^{k}\lambda_j\left(V^{c_j}\left(\pi_i,\boldsymbol{\pi}_{-i}\right)-\alpha_j\right)\right)- \\ &\qquad \left(\Phi(\boldsymbol{\pi})+\sum_{j=1}^{k}\lambda_j\left(V^{c_j}(\boldsymbol{\pi})-\alpha_j\right)\right) \qquad \text{(By Eq. (Lagrangian).)} \\ &= \Phi(\pi_i,\boldsymbol{\pi}_{-i})-\Phi(\boldsymbol{\pi})+\sum_{j=1}^{k}\lambda_j\left(V^{c_j}(\pi_i,\boldsymbol{\pi}_{-i})-V^{c_j}(\boldsymbol{\pi})\right) \\ &= V^{r_i}(\pi_i,\boldsymbol{\pi}_{-i})-V^{r_i}(\boldsymbol{\pi})+ \qquad \text{(By Eq. (2).)} \\ &\qquad \sum_{j=1}^{k}\lambda_j\left(V^{c_j}(\pi_i,\boldsymbol{\pi}_{-i})-V^{c_j}(\boldsymbol{\pi})\right) \\ &= V^{\tilde{r}_i}(\pi_i,\boldsymbol{\pi}_{-i})-V^{\tilde{r}_i}(\boldsymbol{\pi}). \end{aligned}$$

The last equality implies that $\mathcal{L}(\cdot,\boldsymbol{\lambda})$ satisfies Eq. (2) for the new value functions. We conclude that $\mathcal{L}(\cdot,\boldsymbol{\lambda})$ is indeed potential function for the modified rewards. □

While Proposition 2 proves that the dual problem reduces to an unconstrained MPG, which can be solved using existing techniques (Leonardos et al., 2022), we prove in Proposition 3, that, unfortunately, strong duality does not always hold for CMPGs. Moreover, we show that the resulting policy is not only sub-optimal but also does not respect the constraints.

**Proposition 3.** *There exists a CMPG, for which strong duality does not hold, i.e., for which $P^*\neq D^*$.*

*Proof.* We prove this using a counter-example. Consider the following two-agent CMPG with $|\mathcal{S}|=1$, $\mathcal{A}_1=\mathcal{A}_2=\{1,2\}$, reward functions $r=r_1=r_2$, constraint function $c$ and threshold $\alpha=1/2$. The rewards and constraints are specified via the matrices

$$A=\begin{bmatrix}3 & 2\\ 2 & 4\end{bmatrix},\quad B=\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix},$$

where $r(i,j)=A(i,j)$ and $c(i,j)=B(i,j),\forall i,j\in\{1,2\}$. This is a *cooperative* CMPG with potential function $\Phi(\boldsymbol{\pi})=\pi_1^T A\pi_2$. The optimization formulation (Eq. (5)) corresponding to this CMPG is:

$$\max_{\pi_1,\pi_2\in\Delta_2}\pi_1^T A\pi_2,\ \text{subject to: } \pi_1^T B\pi_2\leq 1/2. \qquad (9)$$

**Primal problem:** First, we solve the primal problem, which is defined as follows:

$$P^*=\max_{\pi_1,\pi_2\in\Delta_2}\min_{\lambda\in\mathbb{R}_+}\left(\pi_1^T A\pi_2+\lambda\left(\frac{1}{2}-\pi_1^T B\pi_2\right)\right)$$

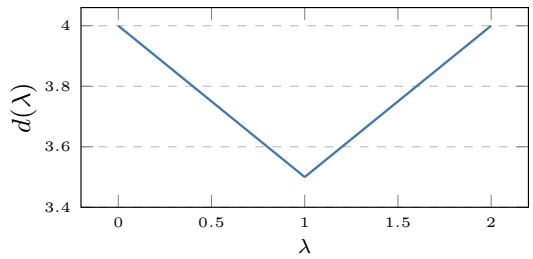



Figure 2: This figure displays the dual function $d(\lambda)$ for the CMPG in Eq. (9), evaluated at 1000 equidistant locations $\lambda \in [0, 2]$.

The policies $\pi_1 = \pi_2 = \left[1 - \sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}\right]$ solve the primal problem with a reward of $P^* \approx 3.09$. One can easily verify that these three policies are also Nash policies.

**Dual problem:** Next, we solve the dual problem, which is defined as follows:

$$D^* = \min_{\lambda \in \mathbb{R}_+} \underbrace{\max_{\pi_1, \pi_2 \in \Delta_2} \left( \pi_1^T A \pi_2 + \lambda \left( \frac{1}{2} - \pi_1^T B \pi_2 \right) \right)}_{=: d(\lambda)},$$

where $d(\lambda)$ is the *dual function*. Fig. 2 visualizes $d(\lambda)$ for $\lambda \in [0, 2]$. Since the dual function is always convex, it is sufficient to focus only on this interval. From Fig. 2, we can see that $d(\lambda)$ reaches its minimum at $\lambda_D^* = 1$ with $D^* = d(\lambda_D^*) = 3.5$. Note that this is strictly larger than the primal solution $P^* = 3$, and therefore, strong duality does not hold here. Next, we list two policies that are solutions to the dual problem, i.e., policies $\pi_D^*$ that satisfy $\pi_D^* \in \arg\max_\pi \left\{ \pi_1^T A \pi_2 + \lambda_D^* \left( \frac{1}{2} - \pi_1^T B \pi_2 \right) \right\}$:

1. $\pi_1 = [1, 0], \pi_2 = [1, 0]$
2. $\pi_1 = [0, 1], \pi_2 = [0, 1]$

The first policy satisfies the constraints and is indeed a Nash policy, with a reward of $3 < P^*$. The second policy, however, does not satisfy the constraints. Solving the dual problem does therefore not necessarily guarantee a feasible policy. □

We have proved in Proposition 3 that strong duality does not always hold in CMPGs. Furthermore, we showed that solving the dual may not lead to a feasible policy. What if a CMPG satisfies the strong duality property? Next, we modify the CMPG in Eq. (9) such that strong duality holds. We will demonstrate that even in that case, the dual problem might not return a feasible Nash policy. Consider a modified version of the CMPG in Eq. (9) with $A$ defined as follows:

$$A = \begin{bmatrix} 3 & 3 \\ 3 & 4 \end{bmatrix}. \tag{10}$$

**Primal problem:** First, we recall the primal problem, which is defined as follows:

$$P^* = \max_{\pi_1, \pi_2 \in \Delta_2} \min_{\lambda \in \mathbb{R}_+} \left( \pi_1^T A \pi_2 + \lambda \left( \frac{1}{2} - \pi_1^T B \pi_2 \right) \right)$$

It is easy to see that $P^* = 3.5$ here. The following policies return an expected reward of $P^*$:

1. $\pi_1 = \left[\frac{1}{2}, \frac{1}{2}\right], \pi_2 = [0, 1]$
2. $\pi_1 = [0, 1], \pi_2 = \left[\frac{1}{2}, \frac{1}{2}\right]$

**Dual problem:** Next, we recall the dual problem, which is defined as follows:

$$D^* = \min_{\lambda \in \mathbb{R}_+} \underbrace{\max_{\pi_1, \pi_2 \in \Delta_2} \left( \pi_1^T A \pi_2 + \lambda \left( \frac{1}{2} - \pi_1^T B \pi_2 \right) \right)}_{=: d(\lambda)},$$

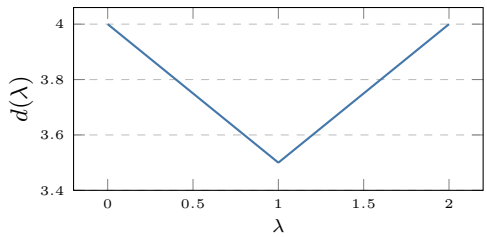



Figure 3: This figure displays the dual function $d(\lambda)$ for the CMPG in Eq. (10), evaluated at 1000 equidistant locations $\lambda \in [0, 2]$.

Fig. 3 visualizes the dual function in the interval $\lambda \in [0, 2]$. Using the same reasoning as before, it is easy to see that the optimal dual variable is $\lambda_D^* = 1$, resulting in the dual value $D^* = 3.5$. Note that in this CMPG, strong duality holds, i.e., $P^* = D^*$. Next, we list three policies that are solutions to the dual problem, i.e., policies that satisfy $\pi_D^* \in \arg\max_\pi \left\{\pi_1^T A \pi_2 + \lambda_D^* \left(\frac{1}{2} - \pi_1^T B \pi_2\right)\right\}$:

1. $\pi_1 = [1, 0], \pi_2 = [1, 0]$
2. $\pi_1 = [0, 1], \pi_2 = [0, 1]$
3. $\pi_1 = \left[\frac{1}{2}, \frac{1}{2}\right], \pi_2 = \left[\frac{1}{2}, \frac{1}{2}\right]$

The first policy is feasible and is indeed a Nash policy. As established before, the second policy does not satisfy the constraints. The third policy is feasible, *but it is not a Nash policy*. For example, agent 1 can improve the expected reward by switching to policy $\pi_1 = [0, 1]$, and vice versa. Therefore, even if strong duality hold, the dual problem may not return a Nash policy.

# B Solving Constrained Markov Potential Games (Section 5)

This section provides the theoretical proofs for Section 5. We start by discussing the initialization for CA-CMPG. As stated in Assumption 1, we require access to an initial, feasible policy. This type of assumption is common in the single-agent CMDP setting (Bura et al., 2022; Liu et al., 2021b). While finding such a policy in the *multi-agent* setting may be computationally expensive, we provide two examples here and explain, how such a policy can be computed by the agents. In both cases, we assume that the CMPG is feasible, i.e., $\Pi_C \neq \emptyset$.

**Example 1** (Single Constraint)**.** *Consider the problem* $\min_{\boldsymbol{\pi} \in \Pi} V^{c_1}(\boldsymbol{\pi})$*. Since the constraint set is feasible, we must have that* $\min_{\boldsymbol{\pi} \in \Pi} V^{c_1}(\boldsymbol{\pi}) \leq \alpha_1$*. Note that this is an unconstrained Markov decision process (MDP) with state space* $\mathcal{S}$ *and action space* $\mathcal{A}$*. It is well-known that MDPs always possess at least one* deterministic*, optimal policy, which can be computed using dynamic programming techniques. Thus, we compute a deterministic policy* $\boldsymbol{\pi}^C \in \arg\min_{\boldsymbol{\pi} \in \Pi} V^{c_1}(\boldsymbol{\pi})$*, s.t. for every state* $s \in \mathcal{S}$ *and step* $h \in [H]$*, there is exactly one action* $a = (a_1, ..., a_n) \in \mathcal{A}$*, for which* $\boldsymbol{\pi}_h^C(a|s) = 1$ *and* $\boldsymbol{\pi}_h^C(a'|s) = 0, \forall a' \neq a$*. Then, for every agent* $i \in [n]$*, we set* $\pi_{i,h}^C(a_i|s) = 1$ *and* $\pi_{i,h}^C(a_i'|s) = 0$*, for all* $a_i' \neq a_i$*. It is easy to verify that* $\boldsymbol{\pi}^C = \prod_{i=1}^n \pi_i^C$*.*

**Example 2** (Independent Transitions and Composite Constraints)**.** *Consider a CMPG with per-agent state spaces* $\mathcal{S}_1, ..., \mathcal{S}_n$ *and transition models* $\mathcal{P}_1, ..., \mathcal{P}_n$*, where* $\mathcal{P}_{j,h}(s'|s, a)$ *is the probability that agent* $j$ *transitions to state* $s' \in \mathcal{S}_j$ *from state-action pair* $(s, a) \in \mathcal{S}_j \times \mathcal{A}_j$ *at step* $h \in [H]$*. We denote by* $\mathcal{S} \triangleq \times_{i=1}^n \mathcal{S}$ *the joint state space and define* $\mathcal{P}_h(s'|s, a) \triangleq \prod_{i=1}^n \mathcal{P}_h^i(s_i'|s_i, a_i)$ *as the joint probability of transitioning to state* $s' \in \mathcal{S}$ *from state-action pair* $(s, a) \in \mathcal{S} \times \mathcal{A}$ *at step* $h \in [H]$*. Furthermore, assume that for each* $j \in [k]$*, the constraint function* $c_j$ *can be written as* $c_{j,h}(s, a) \triangleq \sum_{i=1}^n c_{j,h}^i(s_i, a_i)$*. Due to this, the cumulative constraints can be written as* $V^{c_j}(\boldsymbol{\pi}) = \sum_{i=1}^n V^{c_j^i}(\pi_i)$*,* $\forall j \in [k]$*. To find a feasible policy, each agent* $i \in [n]$ *computes* $\pi_i \in \left\{\pi \in \Pi^i \middle| V^{c_j^i}(\pi) \leq c_i^*, \forall j \in [k]\right\}$*, where* $c_i^* \triangleq \min_{c \in \mathbb{R}} \left\{\exists \pi \in \Pi^i \middle| V^{c_j^i}(\pi) \leq c, \forall j \in [k]\right\}$*. Assuming that the constraint set is feasible, it is easy to see that* $\boldsymbol{\pi}^S = (\pi_1^S, ..., \pi_n^S)$ *must be feasible.*

We are now ready to move to the next part of CA-CMPG, in which each agent solves a single-agent CMDP. As stated in Section 5, in CA-CMPG, while one agent is maximizing its own value function,

the others keep their policy fixed. By doing this, the others provide a stationary environment and the agent essentially faces a CMDP. We now provide an exact description of said CMDP. In iteration $t$, agent $i \in [n]$ faces the CMDP $\mathcal{M} = \left(\mathcal{S}, \mathcal{A}_i, H, \left\{\widetilde{\mathcal{P}}_h\right\}_{h=1}^{H}, \{\widetilde{r}_h\}_{h=1}^{H}, \mu, \left\{\left(\{\widetilde{c}_{j,h}\}_{h=1}^{H}, \alpha_j\right)\right\}_{j=1}^{k}\right)$, where the reward function $\widetilde{r}$, cost functions $\{\widetilde{c}_j\}_{j\in[n]}$ and transition model $\widetilde{\mathcal{P}}$ are defined according to Eq. (11), Eq. (12) and Eq. (13):

$$\widetilde{r}_h(s, a_i) \triangleq \sum_{a_{-i}\in\mathcal{A}\setminus\mathcal{A}_i} r_{i,h}(s, (a_i, a_{-i})) \cdot \boldsymbol{\pi}_{-i,h}^{t-1}(a_{-i}|s), \tag{11}$$

$$\widetilde{c}_{j,h}(s, a_i) \triangleq \sum_{a_{-i}\in\mathcal{A}\setminus\mathcal{A}_i} c_{j,h}(s, (a_i, a_{-i})) \cdot \boldsymbol{\pi}_{-i,h}^{t-1}(a_{-i}|s), \tag{12}$$

$$\widetilde{\mathcal{P}}_h(s'|s, a_i) \triangleq \sum_{a_{-i}\in\mathcal{A}\setminus\mathcal{A}_i} \mathcal{P}_h(s'|s, (a_i, a_{-i})) \cdot \boldsymbol{\pi}_{-i,h}^{t-1}(a_{-i}|s), \tag{13}$$

for all $(s, a_i, s', h) \in \mathcal{S} \times \mathcal{A}_i \times \mathcal{S} \times [H]$.

We are almost ready to prove the convergence of CA-CMPG (Theorem 1). One important requirement for proving convergence is that the potential function is bounded, which we prove in the following lemma.

**Lemma 1.** *Fix an arbitrary base policy $\boldsymbol{\pi}^B \in \Pi$. Then, for every $\boldsymbol{\pi} \in \Pi$, the potential function can be bounded as: $\Phi(\boldsymbol{\pi}) \leq nH + \Phi(\boldsymbol{\pi}^B)$.*

*Proof.* For any $\boldsymbol{\pi} \in \Pi$, we define a new sequence of policies $\left\{\widetilde{\boldsymbol{\pi}}^k\right\}_{k=0,\dots,n}$ as follows:

$$\widetilde{\boldsymbol{\pi}}^k \triangleq \begin{bmatrix} \pi_1^B \\ \vdots \\ \pi_k^B \\ \pi_{k+1} \\ \vdots \\ \pi_n \end{bmatrix}, \quad k = 0, ..., n. \tag{14}$$

It is easy to see that $\widetilde{\boldsymbol{\pi}}^0 \equiv \boldsymbol{\pi}$ and $\widetilde{\boldsymbol{\pi}}^n \equiv \boldsymbol{\pi}^B$. For any agent $i \in [n]$, the policies $\widetilde{\boldsymbol{\pi}}^{i-1}$ and $\widetilde{\boldsymbol{\pi}}^i$ differ only in agent i. By applying the potential property (Eq. (2)), we obtain:

$$\Phi(\widetilde{\boldsymbol{\pi}}^{i-1}) - \Phi(\widetilde{\boldsymbol{\pi}}^i) = V^{r_i}(\widetilde{\boldsymbol{\pi}}^{i-1}) - V^{r_i}(\widetilde{\boldsymbol{\pi}}^i).$$

Summing over all $i \in [n]$ and rearranging the terms, we get:

$$\begin{aligned}
\Phi(\boldsymbol{\pi}) &= \sum_{i=1}^{n} \left[V^{r_i}\left(\widetilde{\boldsymbol{\pi}}^{i-1}\right) - V^{r_i}\left(\widetilde{\boldsymbol{\pi}}^{i}\right)\right] + \Phi(\boldsymbol{\pi}^B) \\
&\leq \sum_{i=1}^{n} \underbrace{\left|V^{r_i}\left(\widetilde{\boldsymbol{\pi}}^{i-1}\right) - V^{r_i}\left(\widetilde{\boldsymbol{\pi}}^{i}\right)\right|}_{\leq H} + \Phi(\boldsymbol{\pi}^B) \\
&\leq nH + \Phi(\boldsymbol{\pi}^B) \qquad \text{(Since } r_{i,h} \in [0,1], \forall i \in [n], \forall h \in [H]\text{).)}
\end{aligned}$$

□

We are now ready to prove the convergence of Algorithm 1. For completeness, we repeat the statement of Theorem 1 here.

**Theorem 1.** *Suppose that Assumption 1 holds. Then, given $\varepsilon > 0$, if we invoke CA-CMPG with $T = \frac{2nH}{\varepsilon}$, it converges to an $\varepsilon$-Nash policy.*

*Proof.* Consider time step $t$ and assume that the algorithm has not converged yet. This implies that there is an agent $i \in [n]$ s.t. $V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) - V^{r_i}(\boldsymbol{\pi}^{t-1}) > \varepsilon/2$; therefore, $\Phi(\boldsymbol{\pi}^t) - \Phi(\boldsymbol{\pi}^{t-1}) > \varepsilon/2$ holds due to the potential property (Eq. (2)).

**Algorithm 2** CA-CMPG-E (Unknown Transitions)

**Require:** $\varepsilon > 0$ (approximation error), $\delta \in (0,1)$ (confidence), $\boldsymbol{\pi}^S \in \Pi_C$ (feasible policy), $T$ (number of iterations), $M > 0$ (number of samples per policy)

$\boldsymbol{\pi}^0 \leftarrow \boldsymbol{\pi}^S$
**for** $t = 1, ..., T$ **do**
  Execute policy $\boldsymbol{\pi}^{t-1}$ for $M$ episodes and estimate $\hat{V}^{r_1}(\boldsymbol{\pi}^{t-1}), ..., \hat{V}^{r_n}(\boldsymbol{\pi}^{t-1})$.
  **for** agent $i = 1, ..., n$ **do**
    Agent $i$ computes $\hat{\pi}_i^t$ such that Eq. (7) is satisfied.
    Execute policy $(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})$ for $M$ episodes and estimate $\hat{V}^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})$.
    $\varepsilon_i^t \leftarrow \hat{V}^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) - \hat{V}^{r_i}(\boldsymbol{\pi}^{t-1})$.
  **if** $\max_{i\in[n]} \varepsilon_i^t > \varepsilon/2$ **then**
    Set $\boldsymbol{\pi}^t = (\hat{\pi}_j^t, \boldsymbol{\pi}_{-j}^{t-1})$, where $j = \arg\max_{i\in[n]} \varepsilon_i^t$, break ties arbitrarily.
  **else**
    **break**

By applying Lemma 11 with the initial policy $\boldsymbol{\pi}^0$, we know that $\Phi(\boldsymbol{\pi}) \leq nH + \Phi(\boldsymbol{\pi}^0)$ holds, for all $\boldsymbol{\pi} \in \Pi$. Since, in every iteration, the potential function increases by at least $\varepsilon/2$, we must have $V^{r_i}(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}\left(\boldsymbol{\pi}^{T-1}\right) \leq \varepsilon/2, \forall i \in [n]$ after $T$ iterations.

Combining this with Eq. (6), we obtain the following guarantee for every agent $i \in [n]$:

$$\max_{\pi \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{T-1})} V^{r_i}(\pi, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}) \leq \varepsilon.$$

Finally, recall that we start from a feasible policy $\boldsymbol{\pi}^0 \in \Pi_C$ and solving Eq. (6) ensures that $\boldsymbol{\pi}^t \in \Pi_C$, $\forall t \leq T$. Therefore, $\boldsymbol{\pi}^{T-1}$ is feasible and is indeed an $\varepsilon$-Nash policy.

□

# C Learning in Unknown Constrained Markov Potential Games (Section 6)

This section provides the theoretical proofs for Section 6. As discussed in Section 6, in the learning setting, we need to modify CA-CMPG to make it work in the learning setting. The resulting algorithm CA-CMPG-E is described in Algorithm 2. We prove the sample complexity bound for CA-CMPG-E in the following theorem.

**Theorem 2.** *Given a strictly feasible CMPG $\mathcal{G}$ with Slater constant $\zeta > 0$, suppose that the agents have access to an initial feasible policy (cf. Assumption 1). Furthermore, assume that the agents have access to a sample-efficient CMDP solver (Eq. (7)). Then, for any $\varepsilon > 0$, $\delta \in (0,1)$, CA-CMPG-E invoked with $M = \frac{32H^2}{\varepsilon^2}\log\left(\frac{32n^2H}{\varepsilon\delta}\right)$ and $T = \frac{4nH}{\varepsilon}$ returns an $\varepsilon$-Nash policy with probability at least $1-\delta$, using the following number of samples:*

$$\mathcal{F}(\varepsilon, \delta) \triangleq \sum_{t=1}^{T}\sum_{i=1}^{n} \mathcal{F}_C\left(|S|, |\mathcal{A}_i|, H, \zeta, \frac{\varepsilon\delta}{8n^2H}, \frac{\varepsilon}{4}\right) + \frac{256n^2H^4}{\varepsilon^3}\log\left(\frac{32n^2H}{\varepsilon\delta}\right).$$

*Proof.* First, we define the following events:

1. $\mathcal{G}_{CMDP} = \{\forall i \in [n], \forall t \in [T] : \hat{\pi}_i^t \text{ satisfies Eq. (7).}\}$
2. $\hat{\mathcal{G}}_{estimate} = \left\{\forall i \in [n], \forall t \in [T] : \left|V^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1}) - \hat{V}^{r_i}(\hat{\pi}_i^t, \boldsymbol{\pi}_{-i}^{t-1})\right| \leq \varepsilon/8\right\}$
3. $\mathcal{G}_{estimate} = \left\{\forall i \in [n], \forall t \in [T] : \left|V^{r_i}(\boldsymbol{\pi}^{t-1}) - \hat{V}^{r_i}(\boldsymbol{\pi}^{t-1})\right| \leq \varepsilon/8\right\}$

Then, we define the "good event" as $\mathcal{G} \triangleq \mathcal{G}_{CMDP} \cap \hat{\mathcal{G}}_{estimate} \cap \mathcal{G}_{estimate}$. In the rest of the proof, we will use the notation $\mathcal{E}^C$ to refer to the complement of an event $\mathcal{E}$.

**1. For any $t \leq T$, if the algorithm did not terminate, the potential function is *strictly* increased:** Assume that $\mathcal{G}$ holds. For any $t \leq T$ before termination, there exists an agent $j \in [n]$, for which $\varepsilon_j^t > \varepsilon/2$ holds. This implies the following increase in the potential function:

$$\begin{aligned}
\Phi(\hat{\pi}_j^t, \boldsymbol{\pi}_{-j}^{t-1}) - \Phi(\boldsymbol{\pi}^{t-1}) &= V^{r_j}(\hat{\pi}_j^t, \boldsymbol{\pi}_{-j}^{t-1}) - V^{r_j}(\boldsymbol{\pi}^{t-1}) && \text{(By Eq. (2).)} \\
&\geq \left(\hat{V}^{r_j}(\hat{\pi}_j^t, \boldsymbol{\pi}_{-j}^{t-1}) - \frac{\varepsilon}{8}\right) - \left(\hat{V}^{r_j}(\boldsymbol{\pi}^{t-1}) + \frac{\varepsilon}{8}\right) \\
& && \text{(By } \mathcal{G}_{estimate}, \hat{\mathcal{G}}_{estimate}.\text{)} \\
&= \varepsilon_j^t - \frac{\varepsilon}{4} \\
&> \frac{\varepsilon}{4} && \text{(Since } \varepsilon_j^t > \varepsilon/2.\text{)}
\end{aligned}$$

**2. $\boldsymbol{\pi}^T$ is an $\varepsilon$-Nash policy:** Assume that $\mathcal{G}$ holds. We apply Lemma 1 with the base policy $\boldsymbol{\pi}^0$ to establish that $\Phi(\boldsymbol{\pi}^T) - \Phi(\boldsymbol{\pi}^0) \leq nH$. Since $\Phi$ is increased by *at least* $\varepsilon/4$ in every iteration $t \leq T$, the condition $\max_{i \in [n]} \varepsilon_i^T \leq \varepsilon/2$ must hold at time $T$. With this, we can bound the differences in the *true* value functions as follows, for every agent $i \in [n]$:

$$\begin{aligned}
V^{r_i}(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}(\boldsymbol{\pi}^{T-1}) &\leq \left(\hat{V}^{r_i}\left(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}\right) + \varepsilon/8\right) - \left(\hat{V}^{r_i}\left(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}\right) - \varepsilon/8\right) \\
& && \text{(By } G_{estimate}, \hat{\mathcal{G}}_{estimate}.\text{)} \\
&= \varepsilon_i^T + \frac{\varepsilon}{4} \\
&\leq \frac{3}{4}\varepsilon.
\end{aligned}$$

Combining this with $\mathcal{G}_{CMDP}$, we obtain:

$$\begin{aligned}
\max_{\pi \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{T-1})} V^{r_i}(\pi, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}(\boldsymbol{\pi}^{T-1}) = & \max_{\pi \in \Pi_C^i(\boldsymbol{\pi}_{-i}^{T-1})} V^{r_i}(\pi, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}) + \\
& V^{r_i}(\hat{\pi}_i^T, \boldsymbol{\pi}_{-i}^{T-1}) - V^{r_i}(\boldsymbol{\pi}^{T-1}) \\
\leq & \ \varepsilon.
\end{aligned}$$

Finally, observe that $\boldsymbol{\pi}^T \in \Pi_C$ and therefore, $\boldsymbol{\pi}^T$ is an $\varepsilon$-Nash policy. Until now, we focused on the convergence of CA-CMPG-E, *assuming* that the event $\mathcal{G}$ holds. In the following paragraphs, we bound the number of samples that are required for $\mathcal{G}$ to hold with high probability.

**3. Number of samples to obtain $Pr\left[\mathcal{G}_{CMDP}\right] \geq 1 - \frac{\delta}{2}$:** Define $\delta' \triangleq \frac{\varepsilon\delta}{8n^2H} \in (0,1)$. Suppose that in iteration $t \leq T$, agent $i \in [n]$ uses a CMDP solver with a sample complexity of $\mathcal{F}_C\left(|\mathcal{S}|, |\mathcal{A}_i|, H, \zeta_i^t, \delta', \frac{\varepsilon}{4}\right)$ s.t. $\hat{\pi}_i^t$ satisfies Eq. (7) with probability at least $1 - \delta'$. We take a union bound over all iterations $t \in \{1, ..., T\}$ and all agents $i \in [n]$, and obtain the following bound on $Pr\left[\mathcal{G}_{CMDP}^C\right]$:

$$\begin{aligned}
Pr\left[\mathcal{G}_{CMDP}^C\right] &\leq \sum_{i=1}^{n}\sum_{t=1}^{T} Pr\left[\hat{\pi}_i^t \text{ does not satisfy Eq. (7)}\right] && \text{(Union bound.)} \\
&\leq nT\delta' \\
&= \frac{\delta}{2}. \\
\Rightarrow Pr\left[\mathcal{G}_{CMDP}\right] &= 1 - Pr\left[\mathcal{G}_{CMDP}^C\right] \geq 1 - \frac{\delta}{2}.
\end{aligned}$$

Finally, to obtain $Pr\left[\mathcal{G}_{CMDP}\right] \geq 1 - \frac{\delta}{2}$, we require the following number of samples:

$$\sum_{t=1}^{T}\sum_{i=1}^{n} \mathcal{F}_C\left(|S|, |\mathcal{A}_i|, H, \zeta_i^t, \frac{\varepsilon\delta}{8n^2H}, \frac{\varepsilon}{4}\right).$$

**4. Number of samples to obtain $Pr\left[\mathcal{G}_{estimate} \cap \hat{\mathcal{G}}_{estimate}\right] \geq 1 - \frac{\delta}{2}$:** Consider an arbitrary policy $\boldsymbol{\pi} \in \Pi$ and suppose that the agents execute $\boldsymbol{\pi}$ for $\mathcal{M} > 0$ episodes Then, each agent $i \in [n]$

estimates $\hat{V}^{r_i}(\boldsymbol{\pi})$ with the averaged, cumulative reward from those $M$ episodes. Since the episodes are *independent*[13] from each other, and the cumulative reward per episode is bounded in the range $[0, H]$, we can apply Hoeffding's inequality to bound the estimation error for agent $i$ as follows:

$$Pr\left[\left|\hat{V}^{r_i}(\boldsymbol{\pi}) - V^{r_i}(\boldsymbol{\pi})\right| \geq \frac{\varepsilon}{8}\right] \leq 2\exp\left(\frac{-2M\left(\varepsilon/8\right)^2}{H^2}\right) \qquad \text{(By Hoeffding's inequality.)}$$
$$= 2\exp\left(\frac{-M\varepsilon^2}{32H^2}\right).$$

The agents need to estimate $2nT$ value functions in total for $\mathcal{G}_{estimate}$ and $\hat{\mathcal{G}}_{estimate}$. By setting $M = \frac{32H^2}{\varepsilon^2}\log\left(\frac{32n^2H}{\varepsilon\delta}\right)$, we ensure that $Pr\left[\left|\hat{V}^{r_i}(\boldsymbol{\pi}) - V^{r_i}(\boldsymbol{\pi})\right| \geq \frac{\varepsilon}{8}\right] \leq \frac{\varepsilon\delta}{16n^2H}$ holds, for all policies $\boldsymbol{\pi}$ that are required for $\mathcal{G}_{estimate}$ and $\hat{\mathcal{G}}_{estimate}$. We are now ready to bound $Pr\left[\mathcal{G}^C_{estimate} \cup \hat{\mathcal{G}}^C_{estimate}\right]$:

$$\begin{aligned}
Pr\left[\mathcal{G}^C_{estimate} \cup \hat{\mathcal{G}}^C_{estimate}\right] &\leq Pr\left[\mathcal{G}^C_{estimate}\right] + Pr\left[\hat{\mathcal{G}}^C_{estimate}\right] && \text{(Union bound.)}\\
&\leq \sum_{t=1}^{T}\sum_{i=1}^{n} Pr\left[\left|\hat{V}^{r_i}(\boldsymbol{\pi}^{t-1}) - V^{r_i}(\boldsymbol{\pi}^{t-1})\right| \geq \frac{\varepsilon}{8}\right] + \\
&\qquad \sum_{t=1}^{T}\sum_{i=1}^{n} Pr\left[\left|\hat{V}^{r_i}(\hat{\pi}^t_i, \boldsymbol{\pi}^{t-1}_{-i}) - V^{r_i}(\hat{\pi}^t_i, \boldsymbol{\pi}^{t-1}_{-i})\right| \geq \frac{\varepsilon}{8}\right] \\
&&& \text{(Union bound.)}\\
&\leq \frac{8n^2H}{\varepsilon}\cdot\frac{\varepsilon\delta}{16n^2H}\\
&\leq \frac{\delta}{2}.\\
\Rightarrow Pr\left[\mathcal{G}_{estimate} \cap \hat{\mathcal{G}}_{estimate}\right] &= 1 - Pr\left[\mathcal{G}^C_{estimate} \cup \hat{\mathcal{G}}^C_{estimate}\right] \geq 1 - \frac{\delta}{2}.
\end{aligned}$$

To obtain $Pr\left[\mathcal{G}_{estimate} \cap \hat{\mathcal{G}}_{estimate}\right] \geq 1 - \frac{\delta}{2}$, considering that each episode requires $H$ samples, the number of samples required is at most:

$$2nTMH \leq \frac{256n^2H^4}{\varepsilon^3}\log\left(\frac{32n^2H}{\varepsilon\delta}\right).$$

**5. Conclusion:** Combining 3. and 4., we obtain $Pr[\mathcal{G}] \geq 1 - \delta$ with the following number of samples:

$$\sum_{t=1}^{T}\sum_{i=1}^{n}\mathcal{F}_C\left(|S|, |\mathcal{A}_i|, H, \zeta^t_i, \frac{\varepsilon\delta}{8n^2H}, \frac{\varepsilon}{4}\right) + \frac{256n^2H^4}{\varepsilon^3}\log\left(\frac{32n^2H}{\varepsilon\delta}\right).$$

□

In Theorem 2, we proved the sample complexity bound for CA-CMPG-E, however, the actual number of samples will depend on the CMDP solver that is used within CA-CMPG-E. In the following two sections, we will instantiate CA-CMPG-E with two different state-of-the-art CMDP solvers and state the resulting sample complexity bounds. Recall that both algorithms are designed for CMDPs with a *single* constraint. Due to this, we set $k = 1$ and denote our cost function by $\{c_h\}_{h=1}^{H}$ and refer to the constraint parameter as $\alpha$. Note that this is due to a limitation of the existing CMDP algorithms and not of CA-CMPG-E.

[13] After every episode, the initial state for the next episode is freshly sampled from $\mu$.

**Algorithm 3** CMDPs with generative model

**Require:** $\mathcal{S}$ (state space), $\mathcal{A}$ (action space), $H$ (horizon), $\{r_h\}_{h=1}^H$ (reward function), $\{c_h\}_{h=1}^H$ (constraint function), $\zeta > 0$ (Slater constant), $N$ (number of samples), $\alpha'$ (constraint threshold), $U$ (projection upper bound), $\lambda_0 = 0$ (initialization).

1: For each state-action $(s, a)$ pair and step $h \in [H]$, collect $N$ samples from $P_h(\cdot|s, a)$ and form the empirical transition model $\hat{\mathcal{P}}_h(\cdot|s, a)$.

2: Form the empirical CMDP $\hat{\mathcal{M}} = \left(\mathcal{S}, \mathcal{A}, H, \left\{\hat{\mathcal{P}}_h\right\}_{h=1}^H, \mu, \{r_h\}_{h=1}^H, \{c_h\}_{h=1}^H, \alpha'\right)$.

3: **for** $t = 0, ..., T-1$ **do**

4: Update the policy: $\hat{\pi}_t \in \arg\max_{\pi\in\Pi} \hat{V}^{r-\lambda_t c}(\pi)$

5: Update the dual-variables: $\lambda_{t+1} = \mathbb{P}_{[0,U]}\left[\lambda_t - \eta\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right)\right]$

6: Convert $\{\hat{\pi}_t\}_{t=0}^{T-1}$ into a single policy $\bar{\pi}$ s.t. $\hat{V}^l(\bar{\pi}) = \frac{1}{T}\sum_{t=1}^T \hat{V}^l(\hat{\pi}_t)$ for $l = r, c$ (see Appendix D.1).

# D Learning in Unknown Constrained Markov Potential Games: Generative Model (Section 6.1)

In this section, similar to Vaswani et al. (2022), we present a novel algorithm for finite-horizon CMDPs, assuming access to a generative model. While the algorithm by Vaswani et al. (2022) returns a *mixture* policy[14], we require an *actual* policy. To address this issue, we make use of *occupancy measures*, which we discuss this in detail in Appendix D.1. Our CMDP algorithm is described in Algorithm 3. We start by proving the sample complexity bound for Algorithm 3. Before we start, we first introduce the following definitions, which are used in the proofs in this section:

$$\pi^* \in \arg\max_{\pi\in\Pi_C} V^r(\pi), \text{ subject to: } V^c(\pi) \leq \alpha \qquad \text{(Optimal policy)}$$

$$\hat{\pi}^* \in \arg\max_{\pi\in\Pi} \hat{V}^r(\pi), \text{ subject to: } \hat{V}^c(\pi) \leq \alpha' \qquad \text{(Optimal empirical policy)}$$

$$\lambda^* \text{ such that: } \hat{V}^r(\hat{\pi}^*) = \max_{\pi\in\Pi}\left[\hat{V}^r(\pi) + \lambda^*\left(\alpha' - \hat{V}^c(\pi)\right)\right],$$

(Optimal dual variable, (Paternain et al., 2019).)

where the value function $\hat{V}^l$ is defined with respect to the empirical transition model $\hat{\mathcal{P}}$ (Line 1), for $l = r, c$. We are now ready to state the sample complexity bound for Algorithm 3 in the following lemma.

**Lemma 2.** *Consider a CMDP* $\mathcal{M} = \left(\mathcal{S}, \mathcal{A}, H, \{r_h\}_{h\in[H]}, \{\mathcal{P}_h\}_{h\in[H]}, \{c_h\}_{h\in[H]}, \alpha\right)$. *Assume that the CMDP is* strictly *feasible, i.e.,* $\exists\pi \in \Pi_C$ *s.t.* $V^c(\pi) < \alpha$. *Furthermore, assume that the Slater constant* $\zeta \triangleq \max_{\pi\in\Pi}\{\alpha - V^c(\pi)\}$, $\zeta > 0$, *is known. Then, for any* $\varepsilon' > 0$ *and* $\delta' \in (0, 1)$, *Algorithm 3 with parameters* $N = \widetilde{\mathcal{O}}\left(\frac{H^6 \log\left(\frac{1}{\delta'}\right)}{\varepsilon'^2\zeta^2}\right)$, $\alpha' = \alpha - \frac{\varepsilon'\zeta}{16H}$, $U = \frac{8H}{\zeta}$, $T = \mathcal{O}\left(\frac{H^6}{\zeta^4\varepsilon'^2}\right)$ *and* $\eta = \frac{U}{\sqrt{T}H}$ *returns a policy* $\bar{\pi} \in \Pi_C$ *s.t.* $\max_{\pi\in\Pi_C} V^r(\pi) - V^r(\bar{\pi}) \leq \varepsilon'$, *with probability at least* $1 - \delta'$. *Thus, the sample complexity of Algorithm 3 is* $\mathcal{F}_C\left(|\mathcal{S}|, |\mathcal{A}_i|, H, \zeta, \delta', \varepsilon'\right) = |\mathcal{S}||\mathcal{A}|HN = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|H^7 \log\left(\frac{1}{\delta'}\right)}{\varepsilon'^2\zeta^2}\right)$.

*Proof.* We prove the lemma for an arbitrary $\Delta > 0$ and constraint threshold $\alpha' \triangleq \alpha - \Delta$. We denote by $\varepsilon_{opt}$ the error from the primal-dual algorithm (Line 3-Line 5). As shown in Lemma 7, the policies

[14] A mixture policy is a probability distribution over a set of policies.

$\{\hat{\pi}_t\}_{t=0}^{T-1}$ satisfy:

$$\hat{V}^r(\hat{\pi}^*) - \frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^r(\hat{\pi}_t) \leq \varepsilon_{opt},$$
$$\frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^c(\hat{\pi}_t) \leq \alpha' + \varepsilon_{opt}.$$

For now, we restrict $\varepsilon_{opt} < \Delta$ and later determine the right choice of $\varepsilon_{opt}$ and $\Delta$. Next, we analyze the reward sub-optimality and constraint violation guarantees.

**Constraint Violation:** Assume that $\left|V^c(\bar{\pi}) - \hat{V}^c(\bar{\pi})\right| \leq \Delta - \varepsilon_{opt}$ holds. We will later determine how many samples are required in Line 1 to satisfy this. Then, it is easy to verify that $\bar{\pi}$ does indeed satisfy the constraints in the original CMDP $\mathcal{M}$, i.e.

$$\begin{aligned} V^c(\bar{\pi}) &\leq \hat{V}^c(\bar{\pi}) + \Delta - \varepsilon_{opt} \\ &\leq (\alpha' + \varepsilon_{opt}) - \Delta + \varepsilon_{opt} && \text{(By Lemma 7.)} \\ &= \alpha && \text{(Since } \alpha' = \alpha + \Delta.\text{)} \end{aligned}$$

**Reward Sub-Optimality:** First, we define the following *relaxed* objective with respect to the empirical CMDP $\hat{\mathcal{M}}$:

$$\widetilde{\pi}^* \in \arg\max_{\pi\in\Pi} \hat{V}^r(\pi), \text{ subject to: } \hat{V}^c(\pi) \leq \alpha + \Delta. \tag{15}$$

Next, we decompose the reward sub-optimality, by adding and subtracting common terms, as follows:

$$\begin{aligned} V^r(\pi^*) - V^r(\bar{\pi}) &= \left[V^r(\pi^*) - \hat{V}^r(\pi^*)\right] + \left[\hat{V}^r(\pi^*) - \hat{V}^r(\hat{\pi}^*)\right] + \underbrace{\left[\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\bar{\pi})\right]}_{\leq\varepsilon_{opt}, \text{ Lemma 7}} + \\ &\quad \left[\hat{V}^r(\bar{\pi}) - V^r(\bar{\pi})\right] \\ &\leq \underbrace{\left[V^r(\pi^*) - \hat{V}^r(\pi^*)\right]}_{\text{Concentration error}} + \underbrace{\left[\hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*)\right]}_{\text{Sensitivity error}} + \varepsilon_{opt} + \underbrace{\left[\hat{V}^r(\bar{\pi}) - V^r(\bar{\pi})\right]}_{\text{Concentration error}}. \end{aligned} \tag{16}$$

To bound the sensitivity error, first assume that $\left|V^c(\pi^*) - \hat{V}^c(\pi^*)\right| \leq \Delta$ holds. Then, $\pi^*$ is feasible for Eq. (15):

$$\hat{V}^c(\pi^*) \leq V^c(\pi^*) + \Delta \leq \alpha + \Delta.$$

Since $\widetilde{\pi}^*$ is optimal for Eq. (15), we know that $\hat{V}^r(\pi^*) \leq \hat{V}^r(\widetilde{\pi}^*)$ must hold. Then, we apply Lemma 10 to bound the sensitivity error as:

$$\hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*) = \left|\hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*)\right| \leq 2\Delta\lambda^*.$$

To further bound $\lambda^*$, we define $\pi_c^* \in \arg\min_{\pi\in\Pi} V^c(\pi)$ and assume that $\left|V^c(\pi_c^*) - \hat{V}^c(\pi_c^*)\right| \leq \frac{\zeta}{2} - \Delta$ holds. Then, we can apply Lemma 11 and obtain the bound $\lambda^* \leq \frac{2H}{\zeta}$. Plugging this into Eq. (16), we obtain:

$$V^r(\pi^*) - V^r(\bar{\pi}) \leq \left[V^r(\pi^*) - \hat{V}^r(\pi^*)\right] + \frac{4\Delta H}{\zeta} + \varepsilon_{opt} + \left[\hat{V}^r(\bar{\pi}) - V^r(\bar{\pi})\right].$$

Next, we discuss how to set $\Delta$ and $\varepsilon_{opt}$. To achieve $V^r(\pi^*) - V^r(\bar{\pi}) \leq \varepsilon$ in the end, we set $\Delta = \frac{\varepsilon\zeta}{16H} < \frac{\zeta}{2}$ and $\varepsilon_{opt} = \frac{\Delta}{5} < \frac{\varepsilon}{4}$[15]. This simplifies our reward sub-optimality to the following expression:

$$V^r(\pi^*) - V^r(\bar{\pi}) \leq \frac{\varepsilon}{2} + \left[V^r(\pi^*) - \hat{V}^r(\pi^*)\right] + \left[\hat{V}^r(\bar{\pi}) - V^r(\bar{\pi})\right].$$

[15] Here, we make a trade-off between sample and computational complexity. We only require $\varepsilon_{opt} < \Delta$, but selecting a large $\varepsilon_{opt}$ increases the sample complexity, due to the concentration bounds, whereas a smaller $\varepsilon_{opt}$ increases the computational complexity in the primal-dual algorithm.

Based on our choice of $\varepsilon_{opt}$, we set $U = \frac{8H}{\zeta}$ (Lemma 7). Note that Lemma 11 guarantees that $U > \lambda^*$ then. Furthermore, we set $T = \frac{U^2 H^2}{\varepsilon_{opt}^2}\left(1 + \frac{1}{(U-\lambda^*)^2}\right) = \mathcal{O}\left(\frac{H^6}{\zeta^4 \varepsilon^2}\right)$ (by Lemma 7).

To obtain $V^r(\pi^*) - V^r(\bar{\pi}) \leq \varepsilon$, we require the following concentration bounds to hold overall:

$$\left|V^c(\pi^*) - \hat{V}^c(\pi^*)\right| \leq \Delta, \quad \left|V^r(\pi^*) - \hat{V}^r(\pi^*)\right| \leq \frac{\varepsilon}{4}, \quad \left|V^c(\pi_c^*) - \hat{V}^c(\pi_c^*)\right| \leq 7\Delta,$$
$$\left|V^c(\bar{\pi}) - \hat{V}^c(\bar{\pi})\right| \leq \Delta - \varepsilon_{opt}, \quad \left|V^r(\bar{\pi}) - \hat{V}^r(\bar{\pi})\right| \leq \frac{\varepsilon}{4}. \tag{17}$$

For the third inequality in Eq. (17), note that $7\Delta \leq \frac{\zeta}{2} - \Delta$.

**Sample complexity:** Applying Lemma 12 to each inequality in Eq. (17), we have that with $N = \mathcal{O}\left(\frac{H^6 \log\left(\frac{|\mathcal{S}||\mathcal{A}|H}{\delta}\right)}{\varepsilon^2 \zeta^2}\right) = \widetilde{\mathcal{O}}\left(\frac{H^6 \log\left(\frac{1}{\delta}\right)}{\varepsilon^2 \zeta^2}\right)$, Eq. (17) holds with probability at least $1 - \delta$. Thus, the sample complexity of Algorithm 3 is $F_C(|\mathcal{S}|, |\mathcal{A}|, H, \zeta, \delta, \varepsilon) = |S||A|HN = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|H^7 \log\left(\frac{1}{\delta}\right)}{\varepsilon^2 \zeta^2}\right)$.

□

In the following section, we discuss how to construct a policy $\bar{\pi}$ s.t. $\hat{V}^l(\bar{\pi}) = \frac{1}{T}\sum_{t=0}^{T-1} \hat{V}^l(\hat{\pi}_t)$, for $l = r, c$.

### D.1 Convert mixture to a single policy

Value functions are typically not linear in the policy, i.e., $\frac{1}{T}\sum_{t=0}^{T-1} \hat{V}^l(\hat{\pi}_t)$ is, in general, not equal to $\hat{V}^l\left(\frac{1}{T}\sum_{t=0}^{T-1} \hat{\pi}_t\right)$, for $l = r, c$. State-action occupancy measures (Borkar, 1988), however, allow us to reformulate the value function in a *linear* way. For brevity, we will refer to state-action occupancy measures as "occupancy measures" in the rest of this section.

**Definition 2** (Occupancy Measure)**.** *For every policy $\pi$, we define its* occupancy measure $\{\hat{\rho}_h^\pi\}_{h=1}^H$ *with respect to the transition model $\left\{\hat{\mathcal{P}}_h\right\}_{h=1}^H$ as follows:*

$$\hat{\rho}_h^\pi(s,a) \triangleq \sum_{(s,a)} Pr^\pi(s_h = s, a_h = a), \quad \forall (s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H].$$

*Alternatively, the occupancy measure can be expressed* recursively *as follows:*

$$\hat{\rho}_1^\pi(s,a) = \mu(s) \cdot \pi_1(a|s),$$
$$\hat{\rho}_h^\pi(s,a) = \sum_{(s',a')} \hat{\rho}_{h-1}^\pi(s',a') \cdot \hat{\mathcal{P}}_h(s|s',a') \cdot \pi_h(a|s), \forall h \in [H] \setminus \{1\},$$

for all state-action pairs $(s,a) \in \mathcal{S} \times \mathcal{A}$. The following lemma establishes how we can construct a policy $\bar{\pi}$ from the primal-dual policies $\{\hat{\pi}_t\}_{t=0}^{T-1}$ s.t. $\hat{V}^l(\bar{\pi}) = \frac{1}{T}\sum_{t=0}^{T-1} \hat{V}^l(\hat{\pi}_t)$, for $l = r, c$.

**Lemma 3.** *Given policies $\{\hat{\pi}_t\}_{t=0}^{T-1}$, consider the* averaged *occupancy measure $\bar{\rho} \triangleq \frac{1}{T}\sum_{t=0}^{T-1} \hat{\rho}^{\hat{\pi}_t}$. Define the policy $\bar{\pi}$ as follows:*

$$\bar{\pi}_h(a|s) \triangleq \begin{cases} \frac{\bar{\rho}_h(s,a)}{\sum_{a'} \bar{\rho}_h(s,a')} & \sum_{a'} \bar{\rho}_h(s,a') > 0 \\ \frac{1}{|\mathcal{A}|} & \textit{otherwise}, \end{cases} \tag{18}$$

*$\forall h \in [H]$ and $\forall (s,a) \in \mathcal{S} \times \mathcal{A}$. Then, $\bar{\pi}$ satisfies the following property:*

$$\hat{V}^l(\bar{\pi}) = \frac{1}{T}\sum_{t=0}^{T-1} \hat{V}^l(\hat{\pi}_t), . \qquad \text{(for } l = r, c.)$$

*Proof.* Since the set of occupation measures is convex (Lemma 5), $\bar{\rho}$ is a valid occupation measure, i.e., there exists a policy $\pi$ s.t. $\bar{\rho} \equiv \hat{\rho}^\pi$. Then, Lemma 6 guarantees that the policy $\bar{\pi}$ constructed in

Eq. (18) indeed satisfies $\hat{\rho}^{\bar{\pi}} \equiv \bar{\rho}$. We are now ready to prove the statement of this lemma:

$$\begin{aligned}
\hat{V}^l(\bar{\pi}) &= \sum_{h=1}^{H} \sum_{(s,a)} \hat{\rho}_h^{\bar{\pi}}(s,a) \cdot l_h(s,a) && \text{(By Lemma 4.)} \\
&= \sum_{h=1}^{H} \sum_{(s,a)} \left( \frac{1}{T} \sum_{t=0}^{T-1} \hat{\rho}_h^{\hat{\pi}_t}(s,a) \right) \cdot l_h(s,a) \\
&= \frac{1}{T} \left( \sum_{t=0}^{T-1} \underbrace{\sum_{h=1}^{H} \sum_{(s,a)} \hat{\rho}_h^{\hat{\pi}_t}(s,a) \cdot l_h(s,a)}_{=\hat{V}^l(\hat{\pi}_t),\ \text{Lemma 4}} \right) \\
&= \frac{1}{T} \sum_{t=0}^{T-1} \hat{V}^l(\hat{\pi}_t).
\end{aligned}$$

□

### D.1.1 Standard lemmas for occupancy measures

**Note:** The lemmas in this section are well-known in the literature (Efroni et al., 2020). We state them here for completeness.

**Lemma 4.** *Consider a policy $\pi \in \Pi$ and its respective occupancy measure $\{\hat{\rho}_h^{\pi}\}_{h\in[H]}$ (Definition 2). Then, for any $l = r, c$, the value function $\hat{V}^l(\pi)$ can be expressed in terms of the occupancy measure as follows:*

$$\hat{V}^l(\pi) = \sum_{h=1}^{H} \sum_{(s,a)} \hat{\rho}_h^{\pi}(s,a) \cdot l_h(s,a).$$

*Proof.* See Efroni et al. (2020, section 2.1). □

**Lemma 5** (Convexity of Occupancy Measures)**.** *The set of occupancy measures $\mathcal{D} \triangleq \left\{ \{\hat{\rho}_h^{\pi}\}_{h\in[H]} \,|\pi \in \Pi \right\}$ is convex.*

*Proof.* See Efroni et al. (2020, section 2.3). □

**Lemma 6.** *Given a valid occupation measure $\{\hat{\rho}_h\}_{h\in[H]} \in \mathcal{D}$, the following policy $\pi$ induces $\{\hat{\rho}_h\}_{h\in[H]}$:*

$$\pi_h(a|s) = \begin{cases} \frac{\rho_h(s,a)}{\sum_{a'} \rho_h(s,a')} & \sum_{a'} \rho_h(s,a') > 0 \\ \frac{1}{|\mathcal{A}|} & \textit{otherwise}, \end{cases} \tag{19}$$

*$\forall h \in [H]$ and $\forall (s,a) \in \mathcal{S} \times \mathcal{A}$.*

*Proof.* See Efroni et al. (2020, section 2.3). □

## D.2 Proofs of auxiliary lemmas

**Lemma 7** (Guarantees for the primal-dual algorithm)**.** *For any $\varepsilon_{opt} > 0$, with $U > \lambda^*$, $T = \frac{U^2 H^2}{\varepsilon_{opt}^2}\left(1 + \frac{1}{(U-\lambda^*)^2}\right)$, $\eta = \frac{U}{\sqrt{T}H}$, the primal-dual algorithm (Line 3 - Line 5) produces policies*

$\{\hat{\pi}_t\}_{t=0}^{T-1}$, *which satisfy:*

$$\hat{V}^r(\hat{\pi}^*) - \frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^r(\hat{\pi}_t) \leq \varepsilon_{opt}, \tag{20}$$

$$\frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^c(\hat{\pi}_t) \leq \alpha' + \varepsilon_{opt}. \tag{21}$$

*Proof.* In iteration $t \in \{0, ..., T-1\}$, we compute $\hat{\pi}_t \in \arg\max_{\pi\in\Pi} \hat{V}^{r-\lambda_t c}(\pi)$.[16]. Since $\hat{\pi}_t$ is optimal at time $t$, the following inequality holds for every $\pi \in \Pi$:

$$\hat{V}^r(\hat{\pi}_t) + \lambda_t\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) \geq \hat{V}^r(\pi) + \lambda_t\left(\alpha' - \hat{V}^c(\pi)\right)$$

Setting $\pi = \hat{\pi}^*$, and using the fact that $\hat{\pi}^*$ is feasible with respect to $\hat{\mathcal{M}}$ i.e., $\hat{V}^c(\hat{\pi}^*) \leq \alpha'$, we obtain:

$$\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\hat{\pi}_t) \leq \lambda_t\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right).$$

Fixing $\lambda \in [0, U]$, subtracting $\lambda\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right)$, summing over all $t = 0, ..., T-1$, and dividing by $T$, we obtain:

$$\begin{aligned}\hat{V}^r(\hat{\pi}^*) - \frac{1}{T}\sum_{t=1}^{T}\hat{V}^r(\hat{\pi}_t) - \lambda\left(\alpha' - \frac{1}{T}\sum_{t=1}^{T}\hat{V}^c(\hat{\pi}_t)\right) &\leq \frac{1}{T}\sum_{t=0}^{T-1}(\lambda_t - \lambda)\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) \\ &\leq \frac{UH}{\sqrt{T}} \quad \text{(By Lemma 8, setting } \eta = \frac{U}{\sqrt{T}H}.\text{)}\end{aligned} \tag{22}$$

We will now bound the reward sub-optimality (Eq. (20)) and constraint violation (Eq. (21)) separately.

**Reward Sub-Optimality:** Since Eq. (22) holds for any $\lambda \in [0, U]$, we can set $\lambda = 0$ to obtain:

$$\hat{V}^r(\hat{\pi}^*) - \frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^r(\hat{\pi}_t) \leq \frac{UH}{\sqrt{T}}. \tag{23}$$

**Constraint Violation:** If $\frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^c(\hat{\pi}_t) \leq \alpha'$, then Eq. (21) holds trivially. Consider the case $\frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^c(\hat{\pi}_t) > \alpha'$. Recall that $\bar{\pi}$ is constructed in a way that it satisfies $\hat{V}^l(\bar{\pi}) = \frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^l(\hat{\pi}_t)$ for $l = r, c$ (Line 6). Plugging this into Eq. (22) with $\lambda = U \in [0, U]$, we obtain:

$$\begin{aligned}\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\bar{\pi}) + U\underbrace{\left(\hat{V}^c(\bar{\pi}) - \alpha'\right)}_{\geq 0} &\leq \frac{UH}{\sqrt{T}} \\ \Rightarrow \hat{V}^c(\bar{\pi}) - \alpha' &\leq \frac{UH}{\sqrt{T}(U-\lambda^*)}. \qquad \text{(By Lemma 9.)} \\ \Rightarrow \frac{1}{T}\sum_{t=0}^{T-1}\hat{V}^c(\hat{\pi}_t) - \alpha' &\leq \frac{UH}{\sqrt{T}(U-\lambda^*)}.\end{aligned} \tag{24}$$

Finally, we set $T = \frac{U^2H^2}{\varepsilon_{opt}^2}\left(1 + \frac{1}{(U-\lambda^*)^2}\right)$ s.t. Eq. (23) and Eq. (24) are both bounded by $\varepsilon_{opt}$. This completes the proof of this lemma. □

**Lemma 8** (Dual regret)**.** *For any $\lambda \in [0, U]$, the dual regret $R^d(\lambda, T) \triangleq \sum_{t=0}^{T-1}(\lambda_t - \lambda)\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right)$ can be bounded as follows:*

$$R^d(\boldsymbol{\lambda}, T) \leq U\sqrt{T}H,$$

*by setting $\eta = \frac{U}{\sqrt{T}H}$ in Algorithm 3.*

[16]This is an *unconstrained* Markov decision process (MDP) and can be solved *exactly* using dynamic programming (Bertsekas, 1995)

*Proof.* For any $t \in \{0, ..., T-1\}$, consider the term $|\lambda_{t+1} - \lambda|^2$:

$$\begin{aligned}
|\lambda_{t+1} - \lambda|^2 &= \left|\mathbb{P}_{[0,U]}\left[\lambda_t - \eta\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right)\right] - \lambda\right|^2 \\
&\leq \left|\lambda_t - \eta\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) - \lambda\right|^2 \qquad \text{(The projection } \mathbb{P}_{[0,U]} \text{ is non-expansive.)} \\
&= |\lambda_t - \lambda|^2 - 2\eta\left(\lambda_t - \lambda\right)\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) + \eta^2 \underbrace{\left|\alpha' - \hat{V}^c(\hat{\pi}_t)\right|^2}_{\leq H^2} \\
&\leq |\lambda_t - \lambda|^2 - 2\eta\left(\lambda_t - \lambda\right)\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) + \eta^2 H^2.
\end{aligned}$$

Rearranging the terms, dividing by $2\eta$ and summing over $t = 0, ..., T-1$, we obtain a bound on the dual regret:

$$\begin{aligned}
R^d(\lambda, T) &= \sum_{t=0}^{T-1}\left(\lambda_t - \lambda\right)\left(\alpha' - \hat{V}^c(\hat{\pi}_t)\right) \\
&\leq \frac{1}{2\eta}\sum_{t=0}^{T-1}\left(|\lambda_t - \lambda|^2 - |\lambda_{t+1} - \lambda|^2\right) + \frac{\eta T H^2}{2} \\
&= \frac{\overbrace{|\lambda_0 - \lambda|^2}^{\leq U^2} - \overbrace{|\lambda_T - \lambda|^2}^{\geq 0}}{2\eta} + \frac{\eta T H^2}{2} \\
&\leq U\sqrt{T}H. \qquad \text{(Setting } \eta = \tfrac{U}{\sqrt{T}H}\text{.)}
\end{aligned}$$

□

**Lemma 9.** *For any $C > \lambda^*$ and any policy $\widetilde{\pi}$ s.t. $\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\widetilde{\pi}) + C\left[\hat{V}^c(\widetilde{\pi}) - \alpha'\right]_+ \leq \beta$, we have:*

$$\left[\hat{V}^c(\widetilde{\pi}) - \alpha'\right]_+ \leq \frac{\beta}{C - \lambda^*}.$$

*Proof.* We define the function $\nu(\tau) \triangleq \max_{\pi\in\Pi}\left\{\hat{V}^r(\pi)\middle|\hat{V}^c(\pi) \leq \alpha' - \tau\right\}$ for any $\tau \in \mathbb{R}$. Strong duality for CMDPs (Paternain et al., 2019) gives us the following inequality:

$$\nu(0) = \hat{V}^r(\hat{\pi}^*) = \max_{\pi\in\Pi}\left[\hat{V}^r(\pi) + \lambda^*\left(\alpha' - \hat{V}^c(\pi)\right)\right].$$

Next, consider an arbitrary $\tau$ and any policy $\pi'$ s.t. $\hat{V}^c(\pi') \leq \alpha' - \tau$. Subtracting $\tau\lambda^*$ from the inequality above, we obtain:

$$\begin{aligned}
\nu(0) - \tau\lambda^* &= \max_{\pi\in\Pi}\left[\hat{V}^r(\pi) + \lambda^*\left(\alpha' - \hat{V}^c(\pi)\right)\right] - \tau\lambda^* \\
&\geq \hat{V}^r(\pi') + \lambda^*\left(\alpha' - \hat{V}^c(\pi')\right) - \tau\lambda^* \\
&= \hat{V}^r(\pi') + \lambda^*\underbrace{\left(\alpha' - \tau - \hat{V}^c(\pi')\right)}_{\geq 0} \\
&\geq \hat{V}^r(\pi').
\end{aligned}$$

Since this inequality holds for any $\pi'$ with $\hat{V}^c(\pi') \leq \alpha' - \tau$, it also holds for $\pi'^* \in \arg\max_{\pi\in\Pi}\left\{\hat{V}^r(\pi)\middle|\hat{V}^c(\pi) \leq \alpha' - \tau\right\}$. Plugging this into the inequality above, we obtain:

$$\begin{aligned}
\nu(0) - \tau\lambda^* &\geq \underbrace{\max_{\pi\in\Pi}\left\{\hat{V}^r(\pi)\middle|\hat{V}^c(\pi) \leq \alpha' - \tau\right\}}_{=\nu(\tau)} \\
\Rightarrow \tau\lambda^* &\leq \nu(0) - \nu(\tau).
\end{aligned}$$

Now, we select $\widetilde{\tau} = -\left(\hat{V}^c(\widetilde{\pi}) - \alpha'\right)$ and bound the following expression:

$$\begin{aligned}
(C - \lambda^*)\,|\widetilde{\tau}| &= \widetilde{\tau}\lambda^* + C\,|\widetilde{\tau}| \\
&\leq \nu(0) - \nu(\widetilde{\tau}) + C\,|\widetilde{\tau}| \\
&= \underbrace{\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\widetilde{\pi}) + C\,|\widetilde{\tau}|}_{=\beta} + \underbrace{\hat{V}^r(\widetilde{\pi}) - \nu(\widetilde{\tau})}_{\leq 0} \\
&\qquad\qquad \text{(Addition and subtraction of common terms.)}
\end{aligned}$$

$$\Rightarrow |\widetilde{\tau}| = \left[\hat{V}^c(\widetilde{\pi}) - \alpha'\right]_+ \leq \frac{\beta}{C - \lambda^*}.$$

□

**Lemma 10** (Sensitivity Error)**.** *Let $\Delta > 0$ and define $\hat{\pi}^*, \widetilde{\pi}^*$ as follows:*

$$\begin{aligned}
\hat{\pi}^* &\in \arg\max_{\pi\in\Pi} \left[\hat{V}^r(\pi),\ \textit{subject to: } \hat{V}^c(\pi) \leq \alpha - \Delta\right] \\
\widetilde{\pi}^* &\in \arg\max_{\pi\in\Pi} \left[\hat{V}^r(\pi),\ \textit{subject to: } \hat{V}^c(\pi) \leq \alpha + \Delta\right]
\end{aligned} \tag{25}$$

*Then, the sensitivity error $\left|\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\widetilde{\pi}^*)\right|$ can be bounded as follows:*

$$\left|\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\widetilde{\pi}^*)\right| \leq 2\Delta\lambda^*,$$

*where $\lambda^*$ is the optimal dual variable for Eq.* (25)*, i.e.:*

$$\hat{V}^r(\hat{\pi}^*) = \max_{\pi\in\Pi} \left[\hat{V}^r(\pi) + \lambda^* \left(\alpha - \Delta - \hat{V}^c(\pi)\right)\right],$$

*which holds due to the strong duality property for CMDPs (Paternain et al., 2019).*

*Proof.* We start with the strong duality property for $\hat{\pi}^*$:

$$\begin{aligned}
\hat{V}^r(\hat{\pi}^*) &= \max_{\pi\in\Pi} \left[\hat{V}^r(\pi) + \lambda^* \left(\alpha - \Delta - \hat{V}^c(\pi)\right)\right] \\
&\geq \hat{V}^r(\widetilde{\pi}^*) + \lambda^* \left(\alpha - \Delta - \underbrace{\hat{V}^c(\widetilde{\pi}^*)}_{\leq \alpha+\Delta}\right) \\
&\geq \hat{V}^r(\widetilde{\pi}^*) - 2\Delta\lambda^* \\
&\Rightarrow \hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*) \leq 2\Delta\lambda^*.
\end{aligned}$$

Note that the constraint set considered for $\widetilde{\pi}^*$ is larger than the one for $\hat{\pi}^*$. Therefore, $\hat{V}^r(\widetilde{\pi}^*) \geq \hat{V}^r(\hat{\pi}^*)$ holds and this implies:

$$\left|\hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*)\right| = \hat{V}^r(\widetilde{\pi}^*) - \hat{V}^r(\hat{\pi}^*) \leq 2\Delta\lambda^*. \tag{26}$$

□

**Lemma 11** (Bound on the dual variable)**.** *Define $\pi_c^* \triangleq \arg\min_{\pi\in\Pi} V^c(\pi)$ and $\zeta \triangleq \max_{\pi\in\Pi}\{\alpha - V^c(\pi)\}$. Let $\alpha' = \alpha - \Delta$, for $\Delta \in \left(0, \frac{\zeta}{2}\right)$ and assume that $\left|\hat{V}^c(\pi_c^*) - V^c(\pi_c^*)\right| \leq \frac{\zeta}{2} - \Delta$ holds. Furthermore, let $\hat{\pi}^* \in \arg\max_{\pi\in\Pi}\left[\hat{V}^r(\pi)\ \textit{subject to: } \hat{V}^c(\pi) \leq \alpha'\right]$ denote the optimal policy to the empirical CMDP, and let $\lambda^*$ denote the corresponding optimal dual variable, i.e.:*

$$\hat{V}^r(\hat{\pi}^*) = \max_{\pi\in\Pi} \left[\hat{V}^r(\pi) + \lambda^* \left(\alpha' - \hat{V}^c(\pi)\right)\right]$$

*Then, the dual variable $\lambda^*$ can be bounded as follows:*

$$\lambda^* \leq \frac{2H}{\zeta}.$$

*Proof.* First, we define the policy $\hat{\pi}_c^* \in \arg\min_{\pi\in\Pi} \hat{V}^c(\pi)$. Next, we use the strong duality property (Paternain et al., 2019) for $\hat{\pi}^*$:

$$
\begin{aligned}
\hat{V}^r(\hat{\pi}^*) &= \max_{\pi\in\Pi} \left[\hat{V}^r(\pi) + \lambda^* \left(\alpha' - \hat{V}^c(\pi)\right)\right] \\
&\geq \hat{V}^r(\hat{\pi}_c^*) + \lambda^* \left(\alpha' - \hat{V}^c(\hat{\pi}_c^*)\right) \\
&= \hat{V}^r(\hat{\pi}_c^*) + \lambda^* \left(\underbrace{\alpha - V^c(\pi_c^*)}_{=\zeta} - \Delta + V^c(\pi_c^*) - \hat{V}^c(\hat{\pi}_c^*)\right) \\
&= \hat{V}^r(\hat{\pi}_c^*) + \lambda^* \left(\zeta - \Delta + \underbrace{\hat{V}^c(\pi_c^*) - \hat{V}^c(\hat{\pi}_c^*)}_{\geq 0} + \underbrace{V^c(\pi_c^*) - \hat{V}^c(\pi_c^*)}_{\geq -\left|V^c(\pi_c^*) - \hat{V}^c(\pi_c^*)\right|}\right) \\
&\geq \hat{V}^r(\hat{\pi}_c^*) + \lambda^* \left(\zeta - \Delta - \underbrace{\left|\hat{V}^c(\pi_c^*) - V^c(\pi_c^*)\right|}_{\leq \frac{\zeta}{2} - \Delta}\right) \\
&\geq \hat{V}^r(\hat{\pi}_c^*) + \frac{\lambda^*\zeta}{2} \\
\Rightarrow \lambda^* &\leq \frac{2\left(\hat{V}^r(\hat{\pi}^*) - \hat{V}^r(\hat{\pi}_c^*)\right)}{\zeta} \leq \frac{2H}{\zeta}.
\end{aligned}
$$

□

**Lemma 12.** *Given $\varepsilon \in (0, H], \delta > 0$, for each $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$, obtain $N = \frac{\log\left(\frac{2|\mathcal{S}|^2|\mathcal{A}|H}{\delta}\right)H^4}{\varepsilon^2}$ independent samples from $\mathcal{P}_h(\cdot|s, a)$ and form the estimate $\hat{\mathcal{P}}_h(\cdot|s, a)$. Then, the following concentration bound holds for all policies $\pi$ and all $l \in \{r, c\}$ uniformly with probability at least $1 - \delta$:*

$$
\left|V^l(\pi) - \hat{V}^l(\pi)\right| \leq \varepsilon.
$$

*Proof.* We start with an arbitrary $\beta > 0$ and assume that the difference between $\{\mathcal{P}_h\}_{h\in[H]}$ and $\left\{\hat{\mathcal{P}}_h\right\}_{h\in[H]}$ is bounded by $\beta$, i.e.

$$
\max_{(s,a,s',h)\in\mathcal{S}\times\mathcal{A}\times\mathcal{S}\times[H]} \left|\mathcal{P}_h(s'|s, a) - \hat{\mathcal{P}}_h(s'|s, a)\right| \leq \beta. \tag{27}
$$

For any policy $\pi \in \Pi$ and $l \in \{r, c\}$, by Lemma 13, the difference $V^l(\pi) - \hat{V}^l(\pi)$ can be written as $V^l(\pi) - \hat{V}^l(\pi) = \mathbb{E}_{\substack{s\sim\mu, \\ a_h\sim\pi_h(\cdot|s_h), \\ s_{h+1}\sim\hat{\mathcal{P}}_h(\cdot|s_h,a_h)}} \left[\sum_{h=1}^{H} \left(\mathcal{P}_h - \hat{\mathcal{P}}_h\right)(s_{h+1}|s_h, a_h)\, V_{h+1}^l(s_{h+1}; \pi)|s_0 = s\right]$,

where $V^l_{h+1}(\cdot;\pi)$ is defined according to Eq. (29). Next, we bound $\left|V^l(\pi) - \hat{V}^l(\pi)\right|$ as follows:

$$\begin{aligned}
\left|V^l(\pi) - \hat{V}^l(\pi)\right| &= \left| \mathop{\mathbb{E}}_{\substack{s\sim\mu,\\ a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\hat{\mathcal{P}}_h(\cdot|s_h,a_h)}} \left[\sum_{h=1}^{H} \left(\mathcal{P}_h - \hat{\mathcal{P}}_h\right)(s_{h+1}|s_h,a_h)\, V^l_{h+1}(s_{h+1};\pi)|s_0 = s\right]\right| \\
&\leq \mathop{\mathbb{E}}_{\substack{s\sim\mu,\\ a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\mathcal{P}_h(\cdot|s_h,a_h)}} \left[\left|\sum_{h=1}^{H} \left(\mathcal{P}_h - \hat{\mathcal{P}}_h\right)(s_{h+1}|s_h,a_h)\cdot V^l_{h+1}(s_{h+1};\pi)|s_0 = s\right|\right] \\
&\qquad\qquad \text{(Triangle inequality.)} \\
&\leq \mathop{\mathbb{E}}_{\substack{s\sim\mu,\\ a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\mathcal{P}_h(\cdot|s_h,a_h)}} \left[\sum_{h=1}^{H} \underbrace{\left|\left(\mathcal{P}_h - \hat{\mathcal{P}}_h\right)(s_{h+1}|s_h,a_h)\right|}_{\leq\beta}\cdot\underbrace{\left|V^l_{h+1}(s_{h+1};\pi)\right|}_{\leq H-h}|s_0 = s\right] \\
&\qquad\qquad \text{(Triangle inequality.)} \\
&\leq \beta H^2.
\end{aligned}$$

To ensure that $\left|V^l(\pi) - \hat{V}^l(\pi)\right| \leq \varepsilon$, we set $\beta = \frac{\varepsilon}{H^2}$. Now, consider an arbitrary $(s,a,h)$ and assume that we obtain $N > 0$ *independent* samples from $\mathcal{P}_h(\cdot|s,a)$. Applying Hoeffding's inequality, we obtain the following bound on the estimation error:

$$Pr\left[\left|\mathcal{P}_h(s'|s,a) - \hat{\mathcal{P}}_h(s'|s,a)\right| > \beta\right] \leq 2\exp\left(-2N\beta^2\right). \tag{28}$$

Setting $N = \frac{\log\left(\frac{2|\mathcal{S}|^2|\mathcal{A}|H}{\delta}\right)H^4}{\varepsilon^2}$, we obtain that $Pr\left[\left|\mathcal{P}_h(s'|s,a) - \hat{\mathcal{P}}_h(s'|s,a)\right| > \beta\right] \leq \frac{\delta}{|\mathcal{S}|^2|\mathcal{A}|H}$. Taking the union bound over all $(s,a,s',h)$, we obtain Eq. (27) with probability at least $1-\delta$. □

**Lemma 13** (Value difference lemma)**.** *For any policy $\pi \in \Pi$ and $l \in \{r,c\}$, the value difference $V^l(\pi) - \hat{V}^l(\pi)$ can be expressed as follows:*

$$\begin{aligned}
V^l(\pi) - \hat{V}^l(\pi) &= \mathop{\mathbb{E}}_{\substack{s\sim\mu,\\ a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\hat{\mathcal{P}}_h(\cdot|s_h,a_h)}} \left[\sum_{h=1}^{H} \left(\mathcal{P}_h - \hat{\mathcal{P}}_h\right)(s_{h+1}|s_h,a_h)\, V^l_{h+1}(\pi;s_{h+1})|s_0 = s\right] \\
&= \mathop{\mathbb{E}}_{\substack{s\sim\mu,\\ a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\mathcal{P}_h(\cdot|s_h,a_h)}} \left[\sum_{h=1}^{H} \left(\hat{\mathcal{P}}_h - \mathcal{P}_h\right)(s_{h+1}|s_h,a_h)\, \hat{V}^l_{h+1}(\pi;s_{h+1})|s_0 = s\right],
\end{aligned}$$

*where the per-step and per-state value functions are defined as follows:*

$$\begin{aligned}
V^l_h(\pi;s) &\triangleq \mathop{\mathbb{E}}_{\substack{a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\mathcal{P}_h(\cdot|s_h,a_h)}} \left[\sum_{h'=h}^{H} l_h(s_h,a_h)|s_0 = s\right], \forall s\in\mathcal{S}, \forall h\in[H], \qquad (29)\\
\hat{V}^l_h(\pi;s) &\triangleq \mathop{\mathbb{E}}_{\substack{a_h\sim\pi_h(\cdot|s_h),\\ s_{h+1}\sim\hat{\mathcal{P}}_h(\cdot|s_h,a_h)}} \left[\sum_{h'=h}^{H} l_h(s_h,a_h)|s_0 = s\right], \forall s\in\mathcal{S}, \forall h\in[H].
\end{aligned}$$

*Proof.* See Efroni et al. (2020, Lemma 35). □

# E Learning in Unknown Constrained Markov Potential Games - Safe Exploration Without a Generative Model (Section 6.2)

In this section, we consider the setting, where the agents want to explore safely, but they do not have access to a generative model anymore. Existing algorithms with safe exploration (Bura et al., 2022;

**Algorithm 4** CMDPs with safe exploration

**Require:** $T$ (total number of iterations), $\delta \in (0,1)$ (confidence), $(\pi^S, \alpha_S)$ (strictly feasible policy and its constraint value), $M > 0$ (episodes per policy)

1: Execute DOPE$(\delta, \pi^S, \alpha_S, \alpha, T)$ and obtain policies $\pi_1, ..., \pi_T$.
2: **for** $t = 1, ..., T$ **do**
3: Execute policy $\pi_t$ for $M$ episodes and form the estimate $\hat{V}^r(\pi_t)$.
4: Return policy $\hat{\pi} \in \arg\max_{\pi \in \{\pi_t\}_{t\in[T]}} \hat{V}^r(\pi)$.

Liu et al., 2021b) have guarantees on the *regret*, but no sample complexity guarantees. First, we define a no-regret algorithm with safe exploration guarantees as follows:

**Definition 3** (No-regret algorithm with safe exploration)**.** *Consider a fixed number of $T > 0$ rounds and an algorithm $\mathfrak{A}$, which selects a policy $\pi_t \in \Pi$ in every round $t \in [T]$. $\mathfrak{A}$'s regret after $T$ rounds is defined as follows:*

$$R(T) \triangleq \sum_{t=1}^{T} \max_{\pi \in \Pi_C} V^r(\pi) - V^r(\pi_t).$$

*We call $\mathfrak{A}$ is called a no-regret algorithm with safe exploration, if $R(T) \in o(T)$ and if $\pi_t \in \Pi_C, \forall t \in [T]$. Examples of such algorithms are Liu et al. (2021b) and Bura et al. (2022).*

Next, we discuss how we can use the notion of regret to derive a sample complexity bound. For *unconstrained* MDPs, sub-linear regret bounds can be converted to a sample complexity bound by applying the well-known *online-to-batch* conversion trick (Jin et al., 2018). Applying the same trick to the DOPE algorithm Bura et al. (2022), we derive a sample-efficient algorithm in Algorithm 4 and prove its sample complexity in the following lemma.

**Lemma 14.** *Consider a CMDP $\mathcal{M} = \left(\mathcal{S}, \mathcal{A}, H, \{r_h\}_{h\in[H]}, \{\mathcal{P}_h\}_{h\in[H]}, \{c_h\}_{h\in[H]}, \alpha\right)$. Assume that a strictly feasible policy $\pi^S$ and its constraint value $\alpha_S \triangleq V^c(\pi^S)$ are known s.t. $\alpha_S < \alpha$. Then, for any $\varepsilon' \in (0, H]$ and $\delta' \in (0,1)$, Algorithm 4 with $T = \mathcal{O}\left(\frac{|\mathcal{S}|^2 H^6 |\mathcal{A}|}{(\alpha-\alpha_S)^2 \varepsilon'^2}\right)$ and $M = \widetilde{\mathcal{O}}\left(\frac{H^2}{\varepsilon'^2} \log\left(\frac{1}{\varepsilon'\delta'}\right)\right)$ returns a policy $\hat{\pi} \in \Pi_C$ s.t. $\max_{\pi\in\Pi_C} V^r(\pi) - V^r(\hat{\pi}) \leq \varepsilon$ holds, with probability at least $1-\delta'$. This results in a sample complexity of $\mathcal{F}_C\left(|\mathcal{S}|, |\mathcal{A}|, H, \alpha - \alpha_S, \delta', \varepsilon'\right) = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|^2|\mathcal{A}|H^9}{(\alpha-\alpha_S)^2\varepsilon'^4} \log\left(\frac{1}{\varepsilon'\delta'}\right)\right)$.*

*Proof.* First, recall that Lemma 15 provides the following regret guarantees with probability at least $1 - 5\delta$:

$$\pi_t \in \Pi_C, \forall t \in [T], \tag{30}$$

$$R(T) \leq \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|H^3}{(\alpha - \alpha^S)}\sqrt{|\mathcal{A}|T}\right). \tag{31}$$

**Constraint violation:** Since $\hat{\pi}$ is selected from $\{\pi_t\}_{t\in[T]}$ (Line 4) and Eq. (30) holds, the policy $\hat{\pi}$ is a feasible policy too.

**Reward sub-optimality:** Applying Lemma 16 with $C = \frac{|\mathcal{S}|H^3\sqrt{|\mathcal{A}|}}{(\alpha-\alpha^S)}$, we obtain that with $T = \mathcal{O}\left(\frac{|\mathcal{S}|^2 H^6 |\mathcal{A}|}{(\alpha-\alpha^S)^2\varepsilon^2}\right)$ and $M = \widetilde{\mathcal{O}}\left(\frac{H^2}{\varepsilon^2}\log\left(\frac{1}{\varepsilon\delta}\right)\right)$, the returned policy $\hat{\pi}$ satisfies $\max_{\pi\in\Pi_C} V^r(\pi) - V^r(\hat{\pi}) \leq \varepsilon$, with probability at least $1-\delta$. Taking a union bound over this, Eq. (30) and Eq. (31), the following guarantees hold with probability at least $1 - 6\delta$:

$$\max_{\pi\in\Pi_C} V^r(\pi) - V^r(\hat{\pi}) \leq \varepsilon,$$
$$V^c(\hat{\pi}) \leq \alpha.$$

To compute the final sample complexity, note that DOPE internally uses an initial phase of $T_0$ episodes, during which the agent plays the initial policy $\pi^S$. We do not discuss the details here, but need

to account for those $T_0$ episodes in the sample complexity. The resulting sample complexity is as follows:

$$\begin{aligned}
\mathcal{F}_C\left(|\mathcal{S}|,|\mathcal{A}|,H,\zeta,\delta,\varepsilon\right) &= \left(\underbrace{T+T_0}_{\text{Line 1}}+\underbrace{TM}_{\text{Line 4}}\right)H \\
&= \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|^2|\mathcal{A}|H^9}{\left(\alpha-\alpha^S\right)^2\varepsilon^4}\log\left(\frac{1}{\varepsilon\delta}\right)\right)+\widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|^2|\mathcal{A}|H^5}{\left(\alpha-\alpha^S\right)^2}\right) \\
&\qquad\qquad (T_0=\widetilde{\mathcal{O}}\left(\tfrac{|\mathcal{S}|^2|\mathcal{A}|H^4}{(\alpha-\alpha^S)^2}\right) \text{ by Lemma 15.}) \\
&= \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|^2|\mathcal{A}|H^9}{\left(\alpha-\alpha^S\right)^2\varepsilon^4}\log\left(\frac{1}{\varepsilon\delta}\right)\right).
\end{aligned}$$

□

**Remark:** Note that we used a specific definition of no-regret algorithms in Definition 3. Other notions of no-regret algorithms for CMDPs exist in the literature (Efroni et al., 2020), but they usually do not require feasibility of the iterates (safe exploration). The trick in Lemma 16 assumes that at least one of the iterates is *both* feasible and $\varepsilon$-optimal. This may not be guaranteed by no-regret algorithms without the safe exploration guarantee.

### E.1 Proofs of auxiliary lemmas

**Lemma 15** (Theorem 3 from Bura et al. (2022))**.** *Consider a CMDP* $\mathcal{M} = \left(\mathcal{S},\mathcal{A},H,\{r_h\}_{h\in[H]},\{\mathcal{P}_h\}_{h\in[H]},\{c_h\}_{h\in[H]},\alpha\right)$*. Fix any* $\delta\in(0,1)$*. Then,* DOPE *invoked with* $T_0=\widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|^2|\mathcal{A}|H^4}{(\alpha-\alpha^S)^2}\right)$ *generates a sequence of policies* $\{\pi_t\}_{t\in[T]}$ *s.t.* $\pi_t\in\Pi_C,\forall t\in[T]$ *and the sequence has the following regret:*

$$R(T)=\sum_{t=1}^{T}\max_{\pi\in\Pi_C}V^r(\pi)-V^r(\pi_t)\le\widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|H^3}{(\alpha-\alpha^S)}\sqrt{|\mathcal{A}|T}\right),$$

*with probability at least* $1-5\delta$*.*

**Lemma 16** (Regret to Sample Complexity)**.** *Consider a no-regret CMDP algorithm* $\mathfrak{A}$ *(see Definition 3) with regret bound* $R(T)=\sum_{t=1}^{T}\max_{\pi\in\Pi_C}V^r(\pi)-V^r(\pi_t)\le C\sqrt{T}$*, where* $C$ *is a constant that does not depend on* $T$*, and* $\pi_t\in\Pi_C,\forall t\in[T]$*. Given* $\varepsilon\in(0,H],\delta\in(0,1)$*, run algorithm* $\mathfrak{A}$ *with* $T=\frac{4C^2}{\varepsilon^2}$ *episodes. Next, execute each policy* $\pi\in\{\pi_t\}_{t\in[T]}$ *for* $M=\frac{16H^2}{\varepsilon^2}\log\left(\frac{2C}{\varepsilon\delta}\right)$ *episodes and estimate the value functions* $\left\{\hat{V}^r(\pi_t)\right\}_{t\in[T]}$*. Then, the policy* $\hat{\pi}=\arg\max_{\pi\in\{\pi_t\}_{t\in[T]}}\left\{\hat{V}^r(\pi)\right\}$ *satisfies the following guarantees:*

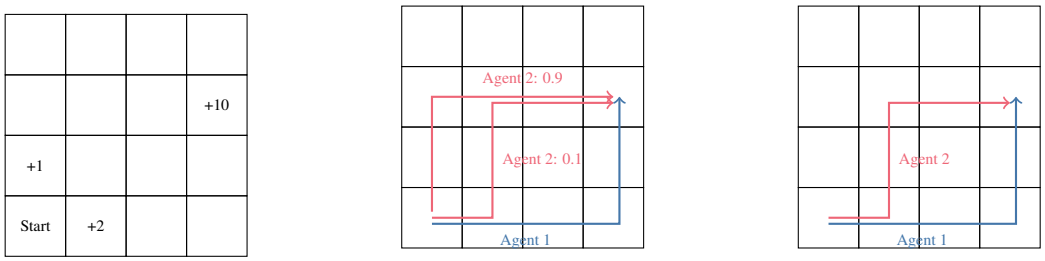
$$V^r(\pi^*)-V^r(\hat{\pi})\le\varepsilon,$$

*with probability at least* $1-\delta$*.*

*Proof.* For any $t\in[T]$, we can bound the estimation error of the value function for $\pi_t$ as follows:

$$\begin{aligned}
Pr\left[\left|V^r(\pi_t)-\hat{V}^r(\pi_t)\right|\ge\frac{\varepsilon}{4}\right] &\le 2\exp\left(\frac{-2N\left(\frac{\varepsilon}{4}\right)^2}{H^2}\right) && \text{(Hoeffding's inequality.)} \\
&= 2\exp\left(\frac{-N\varepsilon^2}{8H^2}\right) \\
&= \frac{\delta}{T} && \text{(Setting } N=\tfrac{16H^2}{\varepsilon^2}\log\left(\tfrac{2C}{\varepsilon\delta}\right)\text{.)}
\end{aligned}$$

Taking the union bound over all $t\in[T]$, we obtain $Pr\left[\exists t\in[T]:\left|V^r(\pi_t)-\hat{V}^r(\pi_t)\right|\ge\frac{\varepsilon}{4}\right]\le\delta$.




(a) (b) (c)

Figure 4: Grid world experiment: Fig. 4a illustrates the state space that the agents navigate in. Both agents start from the bottom left state and their goal is to maximize the sum of their individual rewards. The numbers on the states indicate the rewards associated with those states. The choice of parameters for our evaluation is described in Section 7. Fig. 4b displays the policies with their corresponding probabilities returned by CA-CMPG. If the agents were to solve the dual problem directly (Section 4), they might obtain the policy illustrated in Fig. 4c, which is not feasible.

Next, we note that the average regret can be bounded as $\frac{R(T)}{T} \leq \frac{C}{\sqrt{T}} = \frac{\varepsilon}{2}$ with $T = \frac{4C^2}{\varepsilon^2}$. Thus, there exists $t^* \in \arg\min_{t\in[T]} \{V^r(\pi^*) - V^r(\pi_t)\}$ s.t. $V^r(\pi^*) - V^r(\pi_{t^*}) \leq \frac{R(T)}{T} \leq \frac{\varepsilon}{2}$ holds. We do a case analysis on $V^r(\pi_{t^*}) - V^r(\pi) \geq 0$:

**Case 1:** $V^r(\pi_{t^*}) - V^r(\pi) \leq \frac{\varepsilon}{2}$. Then, we can bound $V^r(\pi^*) - V^r(\pi)$ as follows:

$$\begin{aligned} V^r(\pi^*) - V^r(\pi) &= [V^r(\pi^*) - V^r(\pi_{t^*})] + [V^r(\pi_{t^*}) - V^r(\pi)] \\ &\qquad \text{(Addition/subtraction of the common term.)} \\ &\leq \varepsilon. \end{aligned}$$

**Case 2:** $V^r(\pi_{t^*}) - V^r(\pi) > \frac{\varepsilon}{2}$. In this case, note that the following property must hold for the estimated value functions:

$$\begin{aligned} \hat{V}^r(\pi_{t^*}) &\geq V^r(\pi_{t^*}) - \frac{\varepsilon}{4} \\ &> V^r(\pi) + \frac{\varepsilon}{4} \\ &\geq \hat{V}^r(\pi). \end{aligned}$$

Thus, in this case, $\pi$ cannot be the maximizer of $\left\{\hat{V}^r(\pi_t)\right\}_{t\in[T]}$.

Overall number of episodes required for this technique: $TN = \frac{64C^2H^2}{\varepsilon^4} \log\left(\frac{2C}{\varepsilon\delta}\right)$. □

# F Experiments

This section contains further details to the experiments in Section 7.

**Grid world:** Fig. 4a illustrates the 4x4 grid world (state space) that the agents navigate in. Both agents start from the bottom left state and their goal is to maximize the sum of their individual rewards. The numbers on the states indicate the rewards associated with those states. We evaluate our algorithm Algorithm 2 with horizon $H = 6$ and constraint threshold $\alpha = 0.1$. The resulting policies with the corresponding probabilities are shown in Fig. 4b. With this, the agents collide once with probability 0.1, thus, satisfying the constraint of the experiment. On the other hand, if we solve the Lagrangian dual problem directly (Section 4), one of the returned policies is illustrated in Fig. 4c. In this case, they always have one collision, which does not satisfy the constraint of the experiment.

**Congestion game:**

For every action $a \in \mathcal{A}$, we denote by $k_a$ the number of agents that select $a$ in the current step. Figure 5 visualizes the state transitions in every step, where $k^* \triangleq \max_{a\in\mathcal{A}} k_a$ denotes the maximum number of agents that have selected the same action. The choice of parameters and constraint function is

detailed in Section 7. Fig. 6a and Fig. 6b plot the resulting distributions over the actions for steps $h = 1$ and $h = 2$, respectively. We observe that in step $h = 1$, if the agents start from the safe state, they select their actions s.t. in step $h = 2$, the system remains in the safe state. At step $h = 2$, the agents maximize their rewards by selecting action D, irrespective of the state that the system is in. At step $h = 1$, without the constraints, all agents would prefer to choose action D at the unsafe state. With the choice of our constraints, as we can observe in Fig. 6a, the agents distribute themselves equally amongst actions C and D.

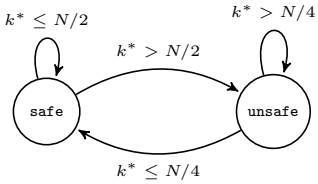
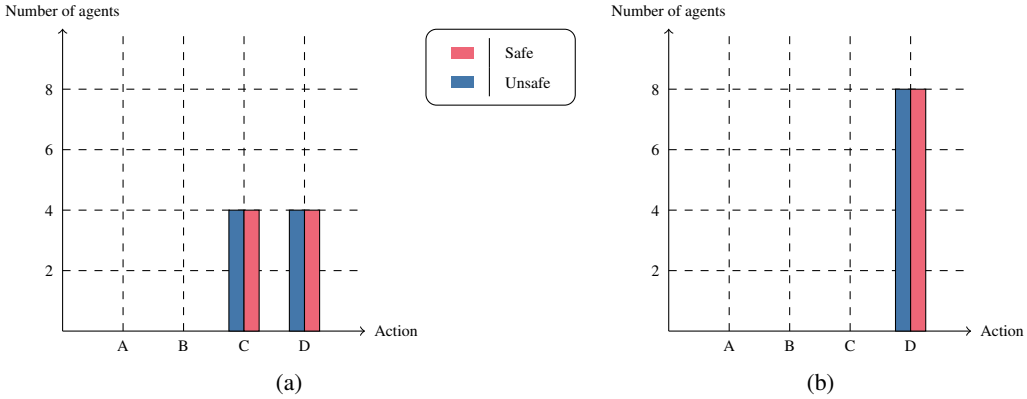



Figure 5: Congestion game experiment: For every action $a \in \mathcal{A}$, we denote by $k_a$ the number of agents that select $a$ in the current step. This figure visualizes the state transitions in every step, where $k^* \triangleq \max_{a \in \mathcal{A}} k_a$ denotes the maximum number of agents that have selected the same action.




(a) (b)

Figure 6: Congestion game experiment: The choice of parameters and constraint function is detailed in Section 7. Fig. 6a and Fig. 6b plot the resulting distributions over the actions for steps $h = 1$ and $h = 2$, respectively.