A. Other Games Satisfying EPW

In this section, we shall verify that several other games besides Pong and Skiing satisfy the EPW condition. In Appendix A.1 we will verify this for the Atari games Tennis and Journey Escape. In Appendix A.2 we will verify this for the RL gaming benchmark CoinRun (Cobbe et al., 2019), which is more complex than Atari.

A.1. Atari Games

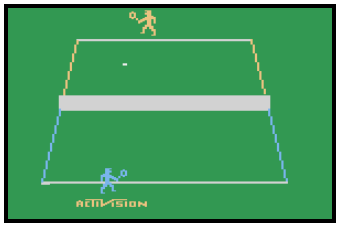

Figure 5. An image of the Atari Tennis game. The yellow player must move to hit the ball (the white dot) while playing against the opposing blue player.

Tennis. This game is very similar to Pong, and is depicted in Figure 5. Here the agent controls the tennis player depicted at the top of the screen, who must hit the ball and prevent it from crossing its boundary. It plays against a player which hits the ball back according to a pre-specified stochastic decision rule (which is not trained). The agent loses if the ball crosses its own boundary, and wins if it hits the ball past the opponent's boundary.

The first two conditions of Generic Games are easy to verify. Note that the states in Tennis are raw images, so $\mathcal{F}$ is defined by any state where the ball has crossed the agent's boundary since this corresponds to the agent losing. It is known that Atari can be solved using a neural network policy (Mnih et al., 2015), and this ensures that a policy class parameterized by neural networks is indeed complete.

To ensure that Tennis satisfies the third condition, we need to design an appropriate binary reward function. This is handled by redefining $\mathcal{F}$ to include any state $s \in \mathcal{S}_{H-1}$ where the ball has not crossed the opposing player's boundary. Then one can simply assign a reward of 1 to any state in $\mathcal{S}_{H-1} \setminus \mathcal{F}$, and 0 to all other states, as required by the Generic Game condition. Hence, playing optimally in this Generic Game framework ensures the ball has moved past the opponent's boundary, corresponding to winning the game.

We now verify that Tennis satisfies the EPW condition with a relatively small value of C. In Tennis, after the opposing player hits the ball, the agent must react to the trajectory of the ball and adjust its position accordingly to hit it. If it takes too long to react before it starts adjusting its position, then it will be unable to reach the ball in time. More formally, assume that at timestep t the paddle has not lost the game and the ball is moving towards its boundary. At timestep t, the ball may be too close to the boundary, and so the agent will not not have enough time to move its player fast enough in order to reach the ball in time. However, at timestep $t - C$ the ball is further away from the boundary, so the agent has enough time to move its player appropriately in order to react, reach the ball and hit it back. So at timestep $t - C$ the agent lies in a safe state in $\mathcal{S}^\star$, since it has enough time to adjust its player and hit the ball back, and hence play optimally. Notably, if we let C' be the number of timesteps it takes for the ball to traverse from one end of the screen to the other, then $C \leq C'$. Hence, when H is large and the agent needs to control its player for many rounds, then C is a constant independent of H.

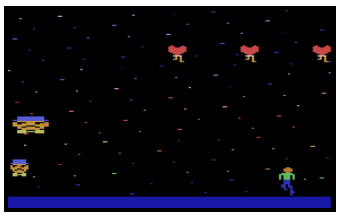

Figure 6. An image of the Atari Journey Escape game. The agent must avoid the enemy objects on screen to avoid receiving a penalty, and must collide with other friendly objects (not depicted) to increase the score.

Journey Escape. This game is similar to Skiing, and is depicted in Figure 7. In this game, there are friendly and enemy objects. These objects come sequentially, and the agent must dodge enemy objects and collide with friendly objects.

The first two conditions of Generic Games are easy to verify. Note that the states in Journey Escape are raw images, so $\mathcal{F}$ is defined by any state where the agent has collided with an enemy object or missed a friendly object. It is known that Atari can be solved using a neural network policy (Mnih et al., 2015), and this ensures that a policy class parameterized by neural networks is indeed complete.

To ensure that Journey Escape satisfies the third condition, we need to design an appropriate binary reward function. This is done by ensuring that $\mathcal{F}$ includes any state where the agent has collided with an enemy object or missed a friendly object, as described above. Then one can simply assign a reward of 1 to any state in $\mathcal{S}_{H-1} \setminus \mathcal{F}$, and 0 to all other states, as required by the Generic Game condition. Hence, playing optimally in this Generic Game framework ensures the agent has avoided all enemy objects while colliding with all friendly objects, corresponding to winning the game.

We now verify that Journey Escape satisfies the EPW condition with a relatively small value of C. As the objects come towards the agent, it must react appropriately to adjust its position depending on whether the oncoming object is friendly or enemy. Let us focus on the enemy object case, since the friendly object case is symmetric. Formally, assume that at timestep t an enemy object is moving towards the agent. At timestep t, the object may be too close to the agent, and so the agent will not not have enough time to move away fast enough and get away from the enemy object. However, at timestep $t - C$ the agent is further away from the enemy object, so the agent has enough time to move away appropriately in order to react. So at timestep $t - C$ the agent lies in a safe state in $\mathcal{S}^\star$, since it has enough time to adjust its position and hence play optimally. Notably, if we let C' be the number of timesteps it takes for the agent to traverse from one end of the screen to the other, then $C \leq C'$. Hence, when H is large and the agent needs to play for many rounds, then C is a constant independent of H.

A.2. CoinRun

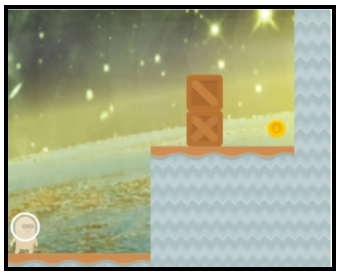

Figure 7. An image of the CoinRun game. The agent must move towards the coin while avoiding obstacles.

CoinRun is a recent RL gaming benchmark (Cobbe et al., 2019). In any CoinRun instance, an agent must move right and jump to avoid obstacles, which are sometimes randomly moving, until it arrives at a coin. It receives unit reward if it reaches the coin, and zero reward otherwise. If it collides with an obstacle then the game is over.

The first two conditions of Generic Games are easy to verify. Note that the states in CoinRun are raw images, so $\mathcal{F}$ is defined by any state where the agent has collided with an obstacle, as well as any state $s \in \mathcal{S}_{H-1}$ where the agent has not already reached the coin. It is known that CoinRun can be solved using a neural network policy (Cobbe et al., 2019), and this ensures that a policy class parameterized by neural networks is indeed complete.

Note that the third condition of Generic Games is automatically satisfied by the reward function we described above. This is because the game already has a binary reward function, with unit reward for reaching the coin and completing the game, and zero reward otherwise.

To verify the EPW condition, note that the agent must react to obstacles which move towards it. Formally, assume that at timestep t an obstacle is moving towards the agent. At timestep t, the obstacle may be too close to the agent, and so the agent will not not have enough time to move away fast enough and get away from the obstacle. However, at timestep $t - C$ the agent is further away from the obstacle, so the agent has enough time to move away appropriately in order to react. So at timestep $t - C$ the agent lies in a safe state in $\mathcal{S}^\star$, since it has enough time to adjust its position and hence play optimally. In this game, C is a small constant, since it only takes a few timesteps for the agent to have enough time to move away from an

oncoming obstacle.

B. Upper Bound Proof

In this section we will prove Theorem 1. First, we shall develop notation and state some helpful lemmas. We then present the proof of Theorem 1, and return to complete the proofs of the lemmas.

Recall that Algorithm 1 iteratively constructs a new $\overline{\theta}(t)$ after each timestep in its inner loop. So after t iterations of its inner loop, the algorithm has constructed $\overline{\theta}(t)$. For each $t \in \{0, 1 \dots H-2\}$ define the function $L_t : \Theta \to \mathbb{R}$ as follows:

$$L_t(\theta) = |\mathcal{A}|^{C+1} \cdot \mathbb{E}_{\tau \sim \pi(\overline{\theta}(t))} \left[\mathbb{I}_{s_{t+1+C} \in \mathcal{F}} \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta) \right]$$

Here, the expectation is over the sampling of trajectories τ from policy $\pi(\overline{\theta}(t))$, where $\tau = \left\{(s_h, a_h)_{h=0}^{H-1}\right\}$. So for each $t \in \{0, 1 \dots H-2\}$, the function L_t is associated with the quantity $\overline{\theta}(t)$ that has been constructed by Algorithm 1. Observe that the function $\widehat{L}_t$ defined in the inner loop of Algorithm 1 is the empirical version of L_t. Also recall the definition of $\widehat{\theta}_t$ from Eq. (2).

We now define the notation $\mathbb{P}_\theta(E \mid X)$ to denote the probability of event E occurring when using policy $\pi(\theta)$, conditioned on executing $\pi(\theta)$ from the state X. We now state a key lemma, which is essential to our proof of Theorem 1.

Lemma 1. *For any $(\mathcal{M}, \pi(\Theta))$ satisfying the EPW condition, and $\overline{\theta}(t)$ constructed by Algorithm 1 for any $t \in \{0, 1 \dots H-2\}$ when given sample size n, assume that $\mathbb{P}_{\overline{\theta}(t)}(s_t \in \mathcal{S}^\star \mid s_0) \geq 1 - \alpha$ for some $\alpha \in (0, 1)$. Then the event*

$$\mathbb{P}_{\overline{\theta}(t+1)}(s_{t+1} \notin \mathcal{S}^\star \mid s_0) \leq 2|\mathcal{A}|^{C+1} \sqrt{\frac{\log(2/\delta')}{n}} + \alpha$$

holds with probability at least $1 - \delta' \left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}} C\phi B\right)^k$.

Simply put, the lemma shows the following. Assume that up till timestep t, Algorithm 1 has computed a policy which arrives at a state in $\mathcal{S}^\star$ with high probability. Then at timestep $t+1$ it will compute a policy which arrives at a state in $\mathcal{S}^\star$ with only slightly worse probability. We shall return to prove this lemma in Appendix B.2. Let us now prove Theorem 1.

B.1. Proof Of Theorem 1

Note that by definition of a Generic Game, $V_{\mathcal{M}}^{s_0}(\pi(\theta^\star)) = 1$. Furthermore we have

$$\begin{aligned} V_{\mathcal{M}}^{s_0}(\pi(\overline{\theta})) &= \mathbb{P}_{\overline{\theta}}(s_{H-1} \in \mathcal{S}_{H-1} - \mathcal{F} \mid s_0) \\ &\geq \mathbb{P}_{\overline{\theta}}(s_{H-1} \in \mathcal{S}^\star \mid s_0) \end{aligned}$$

where the equality is by the definition of the binary reward function in Generic Games, and the inequality is since the MDP is partitioned into disjoint levels and additionally because $\mathcal{S}^\star \subseteq \mathcal{S} - \mathcal{F}$. It is hence to sufficient to show that Algorithm 1 returns $\overline{\theta} \equiv \overline{\theta}(H-1)$ satisfying

$$\mathbb{P}_{\overline{\theta}(H-1)}(s_{H-1} \in \mathcal{S}^\star \mid s_0) \geq 1 - \epsilon \tag{3}$$

with probability at least $1 - \delta$. We shall devote the remainder of the proof to this.

Let δ' be some real number in the interval $(0, 1)$, whose precise value we will specify later. For each $t \in [H]$, let us define $\mathcal{E}_t$ to be the event that Algorithm 1 constructs $\overline{\theta}(t)$ satisfying

$$\mathbb{P}_{\overline{\theta}(t)}(s_t \notin \mathcal{S}^\star \mid s_0) \leq t \cdot 2|\mathcal{A}|^{C+1} \sqrt{\frac{\log(2/\delta')}{n}}. \tag{4}$$

Let $\mathbb{P}_n$ denote the randomness of Algorithm 1, which manifests due to sampling of trajectories at each timestep of the inner loop of the algorithm. We claim that

$$\mathbb{P}_n(\cap_{j \leq t} \mathcal{E}_j) \geq 1 - t \cdot \delta' \left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}} C\phi B\right)^k \tag{5}$$

for each $t \in [H]$. We will prove this by strong induction, repeatedly using Lemma 1 and union bounding to obtain the desired estimate.

For the base case at timestep $t = 0$, notice we trivially have $\mathbb{P}_{\overline{\theta}(0)}(s_0 \in \mathcal{S}^\star \mid s_0) = 1$, since $s_0 \in \mathcal{S}^\star$ by definition of $\theta^\star$. This implies Eq. (4). In particular, we have $\mathbb{P}_n(\mathcal{E}_0) = 1$, verifying Eq. (5) when $t = 0$.

Now for the inductive step, assume that for some t we have $\mathbb{P}_n\left(\bigcap_{j\le t}\mathcal{E}_j\right) \ge 1 - t\cdot\delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. Then by conditioning on $\bigcap\limits_{j\le t}\mathcal{E}_j$ and applying Lemma 1, we obtain that the event

$$\begin{aligned}\mathbb{P}_{\overline{\theta}(t+1)}(s_{t+1} \in \mathcal{S}^\star \mid s_0) &\ge 1 - 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} - t\cdot 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} \\ &= 1 - (t+1)\cdot 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}}\end{aligned}$$

holds with probability at least $1 - \delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. Note that the above equation exactly matches Eq. (4), so conditioned on $\bigcap\limits_{j\le t}\mathcal{E}_j$ we have shown $\mathbb{P}_n(\mathcal{E}_{t+1}) \ge 1 - \delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. Applying a union bound, we have shown that

$$\mathbb{P}_n\left(\bigcap_{j\le t+1}\mathcal{E}_j\right) \ge 1 - (t+1)\cdot\delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k,$$

which thus verifies Eq. (5) and hence the inductive step. We have therefore shown that Algorithm 1 constructs $\overline{\theta}(H-1)$ satisfying

$$\mathbb{P}_{\overline{\theta}(H-1)}(s_{H-1} \in \mathcal{S}^\star \mid s_0) \ge 1 - 2H|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}}$$

with probability at least $1 - H\delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. To show that the above equation exactly matches Eq. (3), we need only check that our choice of n yields the desired values of ϵ and δ.

To obtain our choice of n, we first set $2H|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} = \epsilon$. This yields

$$n = \frac{4H^2|\mathcal{A}|^{2C+2}}{\epsilon^2}\log\left(\frac{2}{\delta'}\right) \iff \sqrt{\frac{n}{\log(2/\delta')}} = \frac{2H|\mathcal{A}|^{C+1}}{\epsilon}. \tag{6}$$

Next, we make the substitution

$$\delta = H\delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k,$$

which results in

$$\delta = H\delta'\left(1 + \frac{32H|\mathcal{A}|^{C+1}}{\epsilon}C\phi B\right)^k,$$

implying that

$$\delta' = \delta\left(H\left(1 + \frac{32H|\mathcal{A}|^{C+1}}{\epsilon}C\phi B\right)^k\right)^{-1}.$$

Substituting the above expression for δ' into the first equivalence of Eq. (6), we finally get

$$n = \frac{4H^2|\mathcal{A}|^{2C+2}}{\epsilon^2}\left(\log\left(\frac{2H}{\delta}\right) + k\log\left(1 + \frac{32H|\mathcal{A}|^{C+1}C\phi B}{\epsilon}\right)\right),$$

and this completes the proof.

B.2. Proof Of Lemma 1

To facilitate our proof, we first state the following useful lemma. Recall the definition of $\widehat{\theta}_t$ from Eq. (2).

Lemma 2. *For any $(\mathcal{M}, \pi(\Theta))$ satisfying the EPW condition, and $\overline{\theta}(t)$ constructed by Algorithm 1 for any $t \in \{0, 1 \dots H-2\}$ when given sample size n, assume that $\mathbb{P}_{\overline{\theta}(t)}(s_t \in \mathcal{S}^\star \mid s_0) \geq 1-\alpha$ for some $\alpha \in (0,1)$. Then the event*

$$\mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+C+1} \in \mathcal{F} \mid s_t)\right] \leq 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha$$

holds with probability at least $1-\delta'\left(1+16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$.

We will return to prove Lemma 2 in Appendix B.3. For now, let us return to the proof of Lemma 1. By the result of Lemma 2, the event

$$\mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+C+1} \in \mathcal{F} \mid s_t)\right] \leq 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha$$

holds with probability at least $1-\delta'\left(1+16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. Let us denote this event as $\mathcal{E}$. Then on this event, we have

$$\begin{aligned}
\mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \mid s_t)\right] &= \mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \wedge s_{t+C+1} \in \mathcal{F} \mid s_t) + \mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \wedge s_{t+C+1} \notin \mathcal{F} \mid s_t)\right] \\
&\overset{(i)}{\leq} \mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+C+1} \in \mathcal{F} \mid s_t) + \mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \wedge s_{t+C+1} \notin \mathcal{F} \mid s_t)\right] \\
&\overset{(ii)}{\leq} 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha + \mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \wedge s_{t+C+1} \notin \mathcal{F} \mid s_t)\right] \\
&\overset{(iii)}{=} 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha.
\end{aligned}$$

Here, step (i) is trivial as $\mathbb{P}(A \wedge B) \leq \mathbb{P}(A)$. Step (ii) follows from the definition of $\mathcal{E}$, and finally, step (iii) follows from the EPW condition and definition of C. It remains to note that

$$\mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+1} \notin \mathcal{S}^\star \mid s_t)\right] = \mathbb{P}_{\overline{\theta}(t+1)}(s_{t+1} \notin \mathcal{S}^\star \mid s_0),$$

which follows directly from the definition $\overline{\theta}(t+1) = [\widehat{\theta}_0, \widehat{\theta}_1 \dots, \widehat{\theta}_{t-1}, \widehat{\theta}_t, \theta_{\text{rand}}, \dots, \theta_{\text{rand}}]$ and the Law of Total Expectation.

B.3. Proof Of Lemma 2

To facilitate this proof, we first state the following two auxiliary lemmas.

Lemma 3. *For any $(\mathcal{M}, \pi(\Theta))$ satisfying the EPW condition, and $\overline{\theta}(t)$ constructed by Algorithm 1 for any $t \in \{0, 1 \dots H-2\}$, we have*

$$L_t(\theta) = \mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_\theta(s_{t+C+1} \in \mathcal{F} \mid s_t)\right],$$

where the expectation is taken with respect to the marginal distribution of s_t while sampling trajectories from $\pi(\overline{\theta})$.

Lemma 4. *For any $(\mathcal{M}, \pi(\Theta))$ satisfying the EPW condition, and $\overline{\theta}(t)$ constructed by Algorithm 1 for any $t \in \{0, 1 \dots H-2\}$, the functions L_t, $\widehat{L}_t$ are each Lipschitz with Lipschitz constant $|\mathcal{A}|^{C+1}(C+1)\phi$.*

We shall return to prove these lemmas in Appendices B.4 and B.5 respectively. Let us now return to the proof of Lemma 2. Recall from Lemma 3 that $\mathbb{E}_{s_t \sim \pi(\overline{\theta}(t))}\left[\mathbb{P}_{\widehat{\theta}_t}(s_{t+C+1} \in \mathcal{F} \mid s_t)\right] = L_t(\widehat{\theta}_t)$. So it is sufficient to show that the event

$$L_t(\widehat{\theta}_t) \leq 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha$$

holds with probability at least $1-\delta'\left(1+16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$, and we will devote the remainder of the proof to showing this. First, we use the characterization of L_t derived in Lemma 3 to the helpful fact that

$$\begin{aligned}
L_t(\theta^\star) &= \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{P}_{\theta^\star}(s_{t+C+1}\in\mathcal{F}\mid s_t)\right] \\
&= \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{P}_{\theta^\star}(s_{t+C+1}\in\mathcal{F}\mid s_t)\mathbb{I}_{s_t\in\mathcal{S}^\star} + \mathbb{P}_{\theta^\star}(s_{t+C+1}\in\mathcal{F}\mid s_t)\mathbb{I}_{s_t\notin\mathcal{S}^\star}\right] \\
&= \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{P}_{\theta^\star}(s_{t+C+1}\in\mathcal{F}\mid s_t)\mathbb{I}_{s_t\notin\mathcal{S}^\star}\right] \\
&\leq \alpha,
\end{aligned} \tag{7}$$

where the final equality follows from the definition of $\mathcal{S}^\star$ and the inequality follows from the assumption that $\mathbb{P}_{\overline{\theta}(t)}(s_t\in\mathcal{S}^\star\mid s_0)\geq 1-\alpha$.

By the Regularity of $\pi(\Theta)$, we are guaranteed that Θ is contained in the Euclidean ball of radius B. For any $\gamma>0$, we use $\mathcal{N}(\gamma)$ to denote a minimal γ-covering of Θ. Recall that $\Theta\subset\mathbb{R}^k$. Also recall the standard fact that $|\mathcal{N}(\gamma)|\leq\left(1+\frac{2B}{\gamma}\right)^k$ (Vershynin, 2018).

Now for any fixed $\theta\in\mathcal{N}(\gamma)$, we know from Hoeffding's inequality that the bound

$$|\widehat{L}_t(\theta)-L_t(\theta)|\leq\frac{|\mathcal{A}|^{C+1}}{2}\sqrt{\frac{\log(2/\delta')}{n}} \tag{8}$$

holds with probability at least $1-\delta'$. Hence, applying a union bound, we know that the above bound holds for every $\theta\in\mathcal{N}(\gamma)$ with probability at least $1-\delta'|\mathcal{N}(\gamma)|\geq 1-\delta'\left(1+\frac{2B}{\gamma}\right)^k$.

For any $\theta\in\Theta$, let θ_γ be an element of $\mathcal{N}(\gamma)$ such that $\|\theta-\theta_\gamma\|_2\leq\gamma$. We now argue that with probability at least $1-\delta'\left(1+\frac{2B}{\gamma}\right)^k$, any $\theta\in\Theta$ satisfies the bound

$$\begin{aligned}
|\widehat{L}_t(\theta)-L_t(\theta)| &\overset{(i)}{\leq} |\widehat{L}_t(\theta)-\widehat{L}_t(\theta_\gamma)| + |\widehat{L}_t(\theta_\gamma)-L_t(\theta_\gamma)| + |L_t(\theta_\gamma)-L_t(\theta)| \\
&\overset{(ii)}{\leq} 2|\mathcal{A}|^{C+1}(C+1)\phi\gamma + |\widehat{L}_t(\theta_\gamma)-L_t(\theta_\gamma)| \\
&\overset{(iii)}{\leq} 2|\mathcal{A}|^{C+1}(C+1)\phi\gamma + \frac{|\mathcal{A}|^{C+1}}{2}\sqrt{\frac{\log(2/\delta')}{n}},
\end{aligned}$$

where step (i) follows from the triangle inequality, step (ii) is due to the Lipschitz property of $L_t,\widehat{L}_t$ we derived in Lemma 4, and step (iii) is due to Eq. (8). Now set $\gamma=\frac{1}{4(C+1)\phi}\sqrt{\frac{\log(2/\delta')}{n}}$. Then the bound

$$|\widehat{L}_t(\theta)-L_t(\theta)|\leq|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} \tag{9}$$

holds for uniformly for each $\theta\in\Theta$ with probability at least $1-\delta'\left(1+16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. Let $\mathcal{E}$ denote the event that Eq. (9) holds uniformly for each $\theta\in\Theta$.

We now use this uniform bound to control the quantity $L_t(\widehat{\theta}_t)$. Concretely, on the event $\mathcal{E}$ we have that

$$\begin{aligned}
L_t(\widehat{\theta}_t) &= L_t(\widehat{\theta}_t)-\widehat{L}_t(\widehat{\theta}_t)+\widehat{L}_t(\widehat{\theta}_t)-\widehat{L}_t(\theta^\star)+\widehat{L}_t(\theta^\star) \\
&\overset{(iv)}{\leq} L_t(\widehat{\theta}_t)-\widehat{L}_t(\widehat{\theta}_t)+\widehat{L}_t(\theta^\star) \\
&\overset{(v)}{\leq} |\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}}+\widehat{L}_t(\theta^\star),
\end{aligned}$$

where step (iv) follows from the definition $\widehat{\theta}_t \in \operatorname{argmin}_\theta \widehat{L}_t(\theta)$, and step (v) follows from Eq. (9). To obtain control on $L_t(\widehat{\theta}_t)$, it remains to bound $\widehat{L}_t(\theta^\star)$. Simply observe that on the event $\mathcal{E}$ we have

$$\begin{aligned}\widehat{L}_t(\theta^\star) &= \widehat{L}_t(\theta^\star) - L_t(\theta^\star) + L_t(\theta^\star) \\ &\overset{(vi)}{\leq} |\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + L_t(\theta^\star) \\ &\overset{(vii)}{\leq} |\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha.\end{aligned}$$

Steps (vi) and (vii) follow from Eq. (9) and Eq. (7) respectively. Putting the previous two equations together and recalling the definition of $\mathcal{E}$, we have demonstrated that the bound

$$L_t(\widehat{\theta}_t) \leq 2|\mathcal{A}|^{C+1}\sqrt{\frac{\log(2/\delta')}{n}} + \alpha$$

holds with probability at least $1 - \delta'\left(1 + 16\sqrt{\frac{n}{\log(2/\delta')}}C\phi B\right)^k$. As argued earlier, this is sufficient to complete the proof.

B.4. Proof Of Lemma 3

Recall that notation $\tau = \left\{(s_h, a_h)_{h=0}^{H-1}\right\}$. Also recall from Algorithm 1 that at timestep t onwards, $\pi(\overline{\theta}(t))$ executes θ_{rand}, implying it selects actions uniformly at random regardless of the state. This fact allows us to decompose $L_t(\theta)$ as follows

$$\begin{aligned}L_t(\theta) &= |\mathcal{A}|^{C+1} \cdot \mathbb{E}_{\tau\sim\pi(\overline{\theta}(t))}\left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}}\prod_{j=0}^{C}\pi_{s_{t+j}}^{a_{t+j}}(\theta)\right] \\ &= |\mathcal{A}|^{C+1} \cdot \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{E}_{\pi(\theta_{\text{rand}})}\left(\mathbb{I}_{s_{t+1+C}\in\mathcal{F}}\prod_{j=0}^{C}\pi_{s_{t+j}}^{a_{t+j}}(\theta)\;\middle|\; s_t\right)\right] \\ &= \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{E}_{\pi(\theta)}\left(\mathbb{I}_{s_{t+1+C}\in\mathcal{F}}\;\middle|\; s_t\right)\right] \\ &= \mathbb{E}_{s_t\sim\pi(\overline{\theta}(t))}\left[\mathbb{P}_\theta(s_{t+C+1}\in\mathcal{F}\mid s_t)\right].\end{aligned}$$

This completes the proof.

B.5. Proof Of Lemma 4

We first note that the product of $m \geq 2$ functions $\{f_i\}_{i=1}^m$ which are bounded by 1 and Lipschitz continuous with constant L is also Lipschitz continuous with constant mL. This can be proved by induction. Consider the base case when $m = 2$. For any x, y in the domain, we have

$$\begin{aligned}|f_1(x)f_2(x) - f_1(y)f_2(y)| &= |f_1(x)f_2(x) - f_1(x)f_2(y) + f_1(x)f_2(y) - f_1(y)f_2(y)| \\ &\overset{(i)}{\leq} |f_1(x)|\,|f_2(x) - f_2(y)| + |f_2(y)|\,|f_1(x) - f_2(y)| \\ &\overset{(ii)}{\leq} |f_2(x) - f_2(y)| + |f_1(x) - f_1(y)| \\ &\overset{(iii)}{\leq} 2L\|x - y\|_2,\end{aligned}$$

where in steps (i), (ii) and (iii) we have used the triangle inequality, the fact that f_1, f_2 are bounded by 1 and the Lipschitz continuity of f_1, f_2 respectively.

Now assume that $g_{[k]} = f_1 \dots f_k$ is Lipschitz continuous with constant kL for some $k \geq 2$. Following the same steps from above, we see that $g_{[k]}f_{k+1}$ is $(k+1)L$-Lipschitz. This completes the proof of the fact.

We now prove the statement of the lemma. Let θ, θ' be two distinct policy parameters. For any $h \in [H]$, we have

$$
\begin{aligned}
|L_h(\theta) - L_h(\theta')| &= |\mathcal{A}|^{C+1} \cdot \left| \mathbb{E}_{\tau\sim\pi(\bar{\theta})} \left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}} \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta) \right] - \mathbb{E}_{\tau\sim\pi(\bar{\theta})} \left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}} \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta') \right] \right| \\
&= |\mathcal{A}|^{C+1} \cdot \left| \mathbb{E}_{\tau\sim\pi(\bar{\theta})} \left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}} \left(\prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta) - \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta') \right) \right] \right| \\
&\overset{(iv)}{\leq} |\mathcal{A}|^{C+1} \cdot \mathbb{E}_{\tau\sim\pi(\bar{\theta})} \left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}} \left(\left| \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta) - \prod_{j=0}^{C} \pi_{s_{t+j}}^{a_{t+j}}(\theta') \right| \right) \right] \\
&\overset{(v)}{\leq} |\mathcal{A}|^{C+1} \cdot \mathbb{E}_{\tau\sim\pi(\bar{\theta})} \left[\mathbb{I}_{s_{t+1+C}\in\mathcal{F}} \cdot ((C+1)\phi\|\theta - \theta'\|_2) \right] \\
&\leq |\mathcal{A}|^{C+1}(C+1)\phi\|\theta - \theta'\|_2.
\end{aligned}
$$

Step (iv) is due to Jensen's inequality $|\mathbb{E}[X]| \leq \mathbb{E}[|X|]$ for any random variable X, and step (v) is due the Lipschitz continuity of a product of Lipschitz continuous functions bounded by 1 which was shown earlier. Since the functions are policy probabilities, they are bounded by 1, and are also ϕ-Lipschitz due to the Regularity of $\pi(\Theta)$.

Analogously, we can show the Lipschitz continuity of $\widehat{L}_h$ for any $h \in \{0, 1 \ldots, H-2\}$, by replacing the expectation with the empirical average over trajectory samples, and this completes the proof of the Lemma.

C. Lower Bound Proof Sketch

As discussed earlier, this result follows almost directly from the results of Du et al. (2020a), and so we only sketch the proof. First we note it is well known that softmax linear policies are Lipschitz (Agarwal et al., 2020b). In our proof, we use Θ as the scaled unit ball in $\mathbb{R}^k$, where the scaling factor is polynomial in $H, \mathcal{A}$. Hence the policy class $\pi(\Theta)$ is indeed Regular. For the construction of $\mathcal{M}$, we use the same construction that was given in the proof of Theorem 4.1 in Du et al. (2020a). Recall this construction is defined by a horizon H MDP $\mathcal{M}$ whose states, actions and transitions are defined by a binary tree with H levels. There is a single state on the final level with unit reward, and all the other states in the tree have zero reward. To cast this construction in our Generic Game framework, we only need to make a slight modification. For each state on the penultimate level whose child does not have reward, modify it transitions so that taking any action from here deterministically exits the MDP. Then discard each of the now unreachable states on the final level, so the final level only contains a state with unit reward. The set $\mathcal{F}$ is precisely defined by the states on the penultimate level of the tree from where taking any action exits the MDP. The binary rewards property is true by definition. And the complete policy class property is true directly by the proof of Du et al. (2020a). Note here, that since we have a bounded Θ and are using a softmax linear policy class $\pi(\Theta)$, the θ^* does not lead to $\mathcal{S}_{H-1} - \mathcal{F}$ almost surely. However, since B is polynomial in $H, \mathcal{A}$, using θ^* will lead to $\mathcal{S}_{H-1} - \mathcal{F}$ with probability exponentially large in H. Our main Theorem 1 easily handles this.

It remains to show the existence of $f : \mathbb{R}^d \to \mathbb{R}$ where f is a linear combination of two neurons, and $V^\star_{\mathcal{M}}(s) = f(s)$ for each $s \in \mathcal{S}$. Again, this follows almost immediately from the proof provided by Du et al. (2020a). Recall that for d sufficiently large, their proof demonstrates the existence of $\theta^{\star\star}$ such that $s^T\theta^{\star\star} = 1$ if $V^\star_{\mathcal{M}}(s) = 1$ and $s^T\theta^{\star\star} \leq 0.25$ if $V^\star_{\mathcal{M}}(s) = 0$. It remains to observe that the function f defined as

$$
f(x) = \mathrm{ReLU}(2x^T\theta^{\star\star} - 1) - \mathrm{ReLU}(-2x^T\theta^{\star\star} - 1)
$$

exactly satisfies the claim.