Appendix

Certified Policy Smoothing for Cooperative Multi-Agent Reinforcement Learning

The appendix can be summarized as follows:

- Appendix A: We provide the proof for Proposition 1 and Corollary 1 in Section 3.2
- Appendix B: We provide the proof for Corollary 2 in Section 4.1
- Appendix C: We present the technical details of the tree-search-based method in single agent environment for comparison with the baseline.
- Appendix D: We provide the details for all environments we performed on in the Section 6.
- Appendix E: We provide the additional experimental results on the environment – " Traffic Junction" with four and ten agents.

A Proofs of Proposition 1 and Corollary 1

Proposition 1. *(restated)If the certification in Algorithm 1 returns the action set* $\mathbf{a} : (a^1, a^2, ..., a^N)$ *at time step t with certified radius D,*

$$D = \frac{\sigma}{2}\left(\Phi^{-1}\left(\underline{p_{\mathbf{a}^m}}\right) - \Phi^{-1}\left(\overline{p_{\mathbf{a}^r}}\right)\right)$$

then with probability at least $(1-\alpha)$*, smoothed policy* $\tilde{\pi}(s_t+\epsilon_t)$ *chooses the action* $\mathbf{a}$, $\forall ||\epsilon_t||_2 \leq D$*.*

Corollary 1. *(restated)(ertification for actions of each agent at each state) At state s, given the joint smoothed policy* $\tilde{\pi}(s) = \{\tilde{\pi}^1(z^1), ..., \tilde{\pi}^N(z^N)\}$*, we can obtain the certified bound at state s for each agent to guarantee* $\tilde{\pi}^n(z^n+\epsilon) := a^{m,n}, \forall ||\epsilon||_2 \leq d_n$*:*

$$d_n = \frac{\sigma}{2}(\Phi^{-1}(\underline{p_{a^{m,n}}}) - \Phi^{-1}(\overline{p_{a^{r,n}}})) \qquad (1)$$

Proof

First, we leverage the proof for the technique of randomized smoothing in (Cohen, Rosenfeld, and Kolter 2019). The following lemma is modified from the (Neyman and Pearson 1933) in statistical hypothesis testing.

Lemma 1. *Let X and Y be random variables in* $\mathbb{R}^d$ *with densities* μ_X *and* μ_Y*. Suppose h* $\mathbb{R}^d \rightarrow 0, 1$ *is a random or deterministic function. Then: If S:*$\{z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \leq t\}$ *for some* $t > 0$ *and* $\mathbb{P}(h(X) = 1) \geq \mathbb{P}(X \in S)$*, then* $\mathbb{P}(h(Y) = 1) \geq \mathbb{P}(Y \in S)$*. On the other hand, if* $\{z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \geq t\}$ *for some* $t > 0$ *and* $\mathbb{P}(h(X) = 1) \leq \mathbb{P}(X \in S)$*, then* $\mathbb{P}(h(Y) = 1) \leq \mathbb{P}(Y \in S)$*.*

Proof. Assume that there is no loss of generality, and h is a random variable, we define $h(1|x)$ as the probability that $h(x) = 1$. First, we can prove the argument $S : z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \leq t$. The complement of S is represented as S^c.

$$\begin{aligned}
\mathbb{P}(h(Y)=1) - \mathbb{P}(Y \in S) &= \int_{\mathbb{R}^d} h(1 \mid z)\mu_Y(z)dz - \int_S \mu_Y(z)dz \\
&= \left[\int_{S^c} h(1 \mid z)\mu_Y(z)dz + \int_S h(1 \mid z)\mu_Y(z)dz\right] \\
&- \left[\int_S h(1 \mid z)\mu_Y(z)dz + \int_S h(0 \mid z)\mu_Y(z)dz\right] \\
&= \int_{S^c} h(1 \mid z)\mu_Y(z)dz - \int_S h(0 \mid z)\mu_Y(z)dz \\
&\geq t\left[\int_{S^c} h(1 \mid z)\mu_X(z)dz - \int_S h(0 \mid z)\mu_X(z)\right] \\
&= t\Bigg[\int_{S^c} h(1 \mid z)\mu_X(z)dz + \int_S h(1 \mid z)\mu_X(z)dz \\
&- \int_S h(1 \mid z)\mu_X(z)dz - \int_S h(0 \mid z)\mu_X(z) \\
&= t\left[\int_{\mathbb{R}^d} h(1 \mid z)\mu_X(z)dz - \int_S \mu_X(z)dz\right] \\
&= t[\mathbb{P}(h(X)=1) - \mathbb{P}(X \in S)] \\
&\geq 0
\end{aligned}$$

where the inequality at the middle is caused by $\mu_Y(z) \leq t\mu_X(z) \forall z \in S$ and $\mu_Y(z) > t\mu_X(z) \forall z \in S^c$, and the inequality in the end is because that we assume that both terms are non-negative in the product.

Then, to prove the argument for $S : z \in \mathbb{R}^d : \frac{\mu_Y(z)}{\mu_X(z)} \leq t$, we can change all " $\geq$ " to " $\leq$ "

Suppose the input comes from one of two distributions, which can be defined as null distribution X or the alternative distribution Y, then the Neyman-Pearson Lemma (Neyman and Pearson 1933) can be applied to determine which distribution the sample came from. The test statistic $\frac{\mu_Y(z)}{\mu_X(z)}$ is denoted as the likelihood ratio statistic and the test will reject H0 : the sample comes from distribution X, if the likelyhood ratio is small.

If the X and Y are isotropic Gaussians, this can be interpreted as the special case of Lemma 1:

Lemma 2. *Let X and Y be random variables from Gaussian distributions,* $X \sim \mathcal{N}\left(x, \sigma^2 I\right)$ *and* $Y \sim \mathcal{N}\left(x+\delta, \sigma^2 I\right)$*. Suppose h* $\mathbb{R}^d \rightarrow 0, 1$ *is a random or deterministic function.*

Then we can have:
a. If $S = \left\{z \in \mathbb{R}^d : \delta^T z \le \beta\right\}$ for some β and $\mathbb{P}(h(X) = 1) \ge \mathbb{P}(X \in S)$, then $\mathbb{P}(h(Y) = 1) \ge \mathbb{P}(Y \in S)$.
b. if $S = \left\{z \in \mathbb{R}^d : \delta^T z \ge \beta\right\}$ for some $t > 0$ and $\mathbb{P}(h(X) = 1) \le \mathbb{P}(X \in S)$, then $\mathbb{P}(h(Y) = 1) \le \mathbb{P}(Y \in S)$.

Proof Lemma 2 can be viewed as a special case of Lemma 1, when X and Y are isotropic Gaussian with means x and $x + \sigma$. Based on Lemma 1, it suffices to show that for ang β, there is some $t > 0$ for:

$$\left\{z : \delta^T z \le \beta\right\} = \left\{z : \frac{\mu_Y(z)}{\mu_X(z)} \le t\right\}$$
$$\left\{z : \delta^T z \ge \beta\right\} = \left\{z : \frac{\mu_Y(z)}{\mu_X(z)} \ge t\right\}$$

where the likelihood ratio can be represented as:

$$\begin{aligned}\frac{\mu_Y(z)}{\mu_X(z)} &= \frac{\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^d \left(z_i - (x_i + \delta_i)\right)^2\right)\Big)}{\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^d \left(z_i - x_i\right)^2\right)} \\ &= \exp\left(\frac{1}{2\sigma^2}\sum_{i=1}^d 2z_i\delta_i - \delta_i^2 - 2x_i\delta_i\right) \\ &= \exp\left(a\delta^T z + b\right)\end{aligned}$$

where $a > 0$ and b are constants w.r.t z.

$$\begin{cases} a = \frac{1}{\sigma^2} \\ b = \frac{-\left(2\delta^T x + \|\delta\|^2\right)}{2\sigma^2} \end{cases}$$

Thus, for any β we can define $t = \exp(a\beta + b)$ to have

$$\begin{aligned}\delta^T z \le \beta &\Longleftrightarrow \exp\left(a\delta^T z + b\right) \le t \\ \delta^T z \ge \beta &\Longleftrightarrow \exp\left(a\delta^T z + b\right) \ge t\end{aligned}$$

Proof of proposition 1

To prove that the smoothed policy $\tilde{\pi}(s_t + \epsilon_t)$ returns the action $\mathbf{a}^m$, the following constraint should be certified:

$$\mathbb{P}\left(\tilde{\pi}(s_t + \epsilon_t) := \mathbf{a}^m\right) > \max_{\mathbf{a}^m \ne \mathbf{a}^r} \mathbb{P}\left(\tilde{\pi}(s_t + \epsilon_t) := \mathbf{a}^r\right)$$

Therefore, we need to guarantee $\mathbb{P}\left(\tilde{\pi}(s_t + \epsilon_t) := \mathbf{a}^m\right) > \mathbb{P}\left(\tilde{\pi}(s_t + \epsilon_t) := \mathbf{a}^r\right)$ to make sure the action set $\mathbf{a}^m \ne \mathbf{a}^r$. Suppose we have two random variables $\mathbf{X}$ and $\mathbf{Y}$:

$$\begin{aligned}\mathbf{X} &:= s_t + \Delta = \mathcal{N}\left(s_t, \sigma^2 I_N\right) \\ \mathbf{Y} &:= s_t + \varepsilon_t + \Delta = \mathcal{N}\left(s_t + \epsilon, \sigma^2 I_N\right)\end{aligned}$$

As we have obtained the the lower bound on the probability ($\underline{p_{\mathbf{a}^m}}$) of the most frequently selected action set, $\mathbf{a}^m$, and the upper bound on probability ($\overline{p_{\mathbf{a}^r}}$) of the second most frequently selected ("runner-up") action, $\mathbf{a}^r$, with confidence (1-α) calculated by the MULTICONBND function based on a chi-squared approximation (Goodman 1965), which takes the number of observations for each category as input and output the confidence levels.

In this domain, we get

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{X}) := \mathbf{a}^m\right) \ge \underline{p_{\mathbf{a}^m}}$$

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{X}) := \mathbf{a}^r\right) \le \overline{p_{\mathbf{a}^r}},$$

As we are going to show:

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{Y}) := \mathbf{a}^m\right) > \mathbb{P}\left(\tilde{\pi}(\mathbf{Y}) := \mathbf{a}^r\right) \tag{2}$$

the half-spaces can be defined as:

$$\begin{aligned}A &:= \left\{y : \epsilon_t^T(y - s_t) \le \sigma\|\epsilon_t\|\Phi^{-1}\left(\underline{p_{\mathbf{a}^m}}\right)\right\} \\ B &:= \left\{y : \epsilon_t^T(y - s_t) \ge \sigma\|\epsilon_t\|\Phi^{-1}\left(1 - \overline{p_{\mathbf{a}^r}}\right)\right\}\end{aligned} \tag{3}$$

Noticing that $\mathbb{P}(\mathbf{X} \in A) = \underline{p_{\mathbf{a}^m}}$, we can conclude $\mathbb{P}\left(\pi(\mathbf{X}) := \mathbf{a}^m\right) \ge \mathbb{P}(\mathbf{X} \in A)$. Hence, in Lemma 2, let $h(y) := \mathbf{1}\left[\pi(y) := \mathbf{a}^m\right]$, we can have

$$\mathbb{P}(\pi(\mathbf{X}) = \mathbf{a}^r) \ge \mathbb{P}(\mathbf{Y} \in A) \tag{4}$$

Similarly, given $\mathbb{P}(\mathbf{X} \in B) = \overline{p_{\mathbf{a}^r}}$, and $\mathbb{P}\left(\pi(\mathbf{X}) = \mathbf{a}^r\right) \le \mathbb{P}(\mathbf{X} \in B)$, we can define $h(y) := \mathbf{1}\left[\pi(y) := \mathbf{a}^r\right]$and conclude that

$$\mathbb{P}\left(\pi(\mathbf{Y}) := \mathbf{a}^r\right) \le \mathbb{P}(\mathbf{Y} \in B) \tag{5}$$

To ensure Eq. 2, based on the Eq. 4 and 5, it suffices to show that

$$\mathbb{P}(\mathbf{Y} \in A) > \mathbb{P}(\mathbf{Y} \in B),$$

which completes the chain of inequalities:

$$\mathbb{P}\left(\pi(\mathbf{Y}) := \mathbf{a}^m\right) \ge \mathbb{P}(\mathbf{Y} \in A) > \mathbb{P}(\mathbf{Y} \in B) \ge \mathbb{P}\left(\pi(\mathbf{Y}) := \mathbf{a}^r\right) \tag{6}$$

Finally, we can calculate the following equations:

$$\begin{aligned}\mathbb{P}(\mathbf{Y} \in A) &= \Phi\left(\Phi^{-1}\left(\underline{p_{\mathbf{a}^m}}\right) - \frac{\|\epsilon_t\|}{\sigma}\right) \\ \mathbb{P}(\mathbf{Y} \in B) &= \Phi\left(\Phi^{-1}\left(\overline{p_{\mathbf{a}^r}}\right) + \frac{\|\epsilon_t\|}{\sigma}\right)\end{aligned} \tag{7}$$

Based on above analysis, $\mathbb{P}(\mathbf{Y} \in A) > \mathbb{P}(\mathbf{Y} \in B)$ is true if and only if :

$$\|\epsilon_t\| < \frac{\sigma}{2}\left(\Phi^{-1}\left(\underline{p_{\mathbf{a}^m}}\right) - \Phi^{-1}\left(\overline{p_{\mathbf{a}^r}}\right)\right). \tag{8}$$

Proof of Corollary 1

Similar as the proof for **Proposition 1**, we modified the certification goal as:

$$\mathbb{P}\left(\tilde{\pi}(z^n + \epsilon) := a^{m,n}\right) > \max_{a^{m,n} \ne a^{r,n}} \mathbb{P}\left(\tilde{\pi}(z^n + \epsilon) := a^{r,n}\right)$$

Thus, we need to show

$$\mathbb{P}\left(\tilde{\pi}(z^n + \epsilon) := a^{m,n}\right) > \mathbb{P}\left(\tilde{\pi}(z^n + \epsilon) := a^{r,n}\right)$$

Now the two random variables $\mathbf{X}$ and $\mathbf{Y}$ are defined as :

$$\begin{aligned}\mathbf{X} &:= z^n + \delta^n = \mathcal{N}\left(z^n, \sigma^2 I\right) \\ \mathbf{Y} &:= z^n + \varepsilon + \delta^n = \mathcal{N}\left(z^n + \epsilon, \sigma^2 I\right)\end{aligned}$$

Following above proof, as we can calculate the lower bound $\underline{p_{a^{m,n}}}$ on probability $\mathbb{P}(\tilde{\pi}^n(z^n) := a^{m,n})$ and upper bound $\overline{p_{a^{r,n}}}$ for the probability of choosing the "runner up" action, $a^{r,n}$, we get

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{X}) := a^{m,n}\right) \ge \underline{p_{a^{m,n}}}$$

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{X}) := a^{m,r}\right) \le \overline{p_{a^{m,n}}},$$

We are going to show:

$$\mathbb{P}\left(\tilde{\pi}(\mathbf{Y}) := a^{m,n}\right) > \mathbb{P}\left(\tilde{\pi}(\mathbf{Y}) := a^{m,r}\right) \tag{9}$$

The half-spaces for each agent can be defined as:

$$\begin{aligned} A &:= \left\{y : \epsilon^T(y - z^n) \le \sigma\|\epsilon\|\Phi^{-1}\left(\underline{p_{a^{m,n}}}\right)\right\} \\ B &:= \left\{y : \epsilon^T(y - z^n) \ge \sigma\|\epsilon\|\Phi^{-1}\left(1 - \overline{p_{a^{m,r}}}\right)\right\} \end{aligned} \tag{10}$$

Hence, following the proof pro proposition1, in 4 and 5, we can replace the $\mathbf{a}^m$ and $\mathbf{a}^r$ with $a^{m,n}$ and $a^{m,r}$ to show

$$\mathbb{P}\left(\pi(\mathbf{Y}) := a^{m,n}\right) \ge \mathbb{P}(\mathbf{Y} \in A) > \mathbb{P}(\mathbf{Y} \in B) \ge \mathbb{P}\left(\pi(\mathbf{Y}) := a^{m,r}\right) \tag{11}$$

The, we can conclude that, $\mathbb{P}(\mathbf{Y} \in A) > \mathbb{P}(\mathbf{Y} \in B)$ is true if and only if :

$$\|\epsilon\| < \frac{\sigma}{2}\left(\Phi^{-1}\left(\underline{p_{a^{m,n}}}\right) - \Phi^{-1}\left(\overline{p_{a^{m,n}}}\right)\right). \tag{12}$$

B Proof for Corollary 2

Definition 1. *Given the obtained certified bound for each agent at state s, $\{d_1, d_2, ..., d_N\}$, the certified bound at this state for all agents is determined by the least robust agent: $D = min\{d_1, d_2, ..., d_N\}$.*

Corollary 2. *(Certified bound per state) At state s, given N agents with action* $\mathbf{a}$, *the joint policy is* $\pi(s) = \{\pi^1(z^1), ..., \pi^N(z^N)\}$. *Suppose the observation of each agent is perturbed by the randomized noise δ^n, where $\delta^n \sim \mathcal{N}(0, \sigma^2 I)$. $\forall n \in N$, if $\mathbb{P}(\pi^n(z^n + \delta^n) := a^{m,n}) \ge 0.5$, we can compute the certified bound via **Definition 1**.*

Proof For every agent n $\in$ N, with confidence $1 - \alpha$, the $p_{a^{m,n}} = \mathbb{P}(\pi^n(z^n + \delta^n) := a^{m,n}) \ge 0.5$ is satisfied. Then we invoke Corollary 1 to compute the certified bound d_n and guarantee that $\tilde{\pi}^n(z^n + \epsilon) := a^{m,n}, \forall ||\epsilon||_2 \le d_n$. therefore, with $D = min\{d_1, d_2, ..., d_N\}$, we can show that $\forall n \in N, \tilde{\pi}^n(z^n + \epsilon) := a^{m,n}, \forall ||\epsilon||_2 \le D \le d_n$. D is determined as the certified bound per state that holds for and agents satisfies $p_{a^{m,n}} \ge 0.5$

C Adapted T-CRGR (AT-CRGR) for comparison in single agent environment

In order to compare our method on single agent environment with the state-of-the-art work (Wu et al. 2021), we adapted our method to the similar structure of the tree used in the baseline.

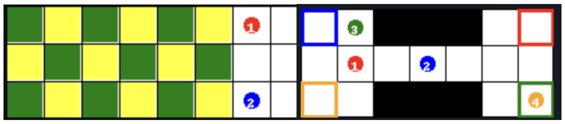
As algorithm 1 shows, our adapted tree AT-CRGR aims to explore all possible trajectories by progressively grow it as CROP-LoRE that used in the baseline. First, the root node, which is at depth 0 of the tree, represents the initial state $s0$. As each node of the tree denotes the state, for each node, we can calculate a sequence of non-decreasing bound of perturbation $d^k(s)$, where k is the number of all possible actions at state s. If current d_{lim} is between the $d^i(s)$ and $d^{i+1}(s)$, we can grow (i+1) branches for state s. We repeat same procedure for the newly expanded branch to expand the tree, until achieving the terminate state or the number of node depth to H. We keep track of the cumulative reward for each trajectory as we grow the tree and update the lower bound of the reward at the end of the trajectory.

Following the setting in the baseline, we apply the perturbation magnitude growth to find the certification under larger perturbation. The priority queue is employed to

Algorithm 1: Adapted Search-Tree based Certified robustness bound and global reward (AT-CRGR)

Input: Trained Q^π; N agents; confidence parameter α
Parameter: sampling times M; Gaussian distribution parameter σ

```
function GETNODE($s, d_{lim}, R$)
    $Que \leftarrow \emptyset$ ▷ Initialize the empty Que containing the tuple (s,a,d,R)
    $A\_dic \leftarrow \emptyset$
    $Actlists \leftarrow$ SMOOTHING$(M, Q^\pi, \alpha, \sigma)$
    for $n \in N$ do
        $poss\_action \leftarrow Sort(Counts(Actlist[n]))$ ▷ Get all possibles actions for agent n
        $a^* \leftarrow argmax(Counts(Actlist[n]))$
        $A\_dic[n] \leftarrow A\_dic[n] \cup \{a^*\}$
        $ct_1^n, ct_2^n \leftarrow$ Top two in $Counts(Actlist[n])$
        for $a$ in $poss\_action$ do
            if $a == a^*$ then
                $pv \leftarrow BioPVALUE(ct_1^n, ct_1^n + ct_2^n, \alpha)$
                if $pv > \alpha$ then
                    d=0
                else
                    $\underline{p} \leftarrow BioConBnd(ct_1^n, M, 1 - \alpha)$
                    d = $\sigma\Phi^{-1}\left(\underline{p}\right)$
            else
                $ct_a^n \leftarrow Counts(Actlist[n])[a]$
                $pv \leftarrow BioPVALUE(ct_1^n, ct_1^n + ct_a^n, \alpha)$
                if $pv > \alpha$ then
                    d=0
                else
                    $\underline{p_{a^*}}, \overline{p_a} \leftarrow MultiConBnd(Counts(Actlist[n]), \alpha)$
                    $d = \frac{\sigma}{2}\Phi^{-1}\left(\underline{p_{a^*}}\right) - \Phi^{-1}\left(\overline{p_a}\right)$
            if $d \le d_{lim}$and $a$ not in $A\_dic[n]$ then
                $A\_dic[n] \leftarrow A\_dic[n] \cup \{a\}$
            else
                $Que.push(s, a, d, R)$
    Return $A\_dic$
function SEARCH($s, d_{lim}, R, done$):
    if $R \ge R_{tot}$ then ▷ Prune the tree
        return 0
    if $done$ then
        $R_{tot} \leftarrow min(R, R_{tot})$
        return 0
    $A\_dic \leftarrow$ GETNODE$(s, d_{lim}, R)$
    for a in $A\_dic$ do
        $s_{new}, done \leftarrow env.step(s, \mathbf{a})$
        SEARCH$(s_{new}, d_{lim}, R + Q(s, \mathbf{a}), done)$
SEARCH$(s_0, d_{lim} = 0, R = 0)$ ▷ Construct initial trajectory
while True do
    if $Que = \emptyset$ then
        break
    $(s, \mathbf{a}, d, R) \leftarrow Que.pop()$ ▷ Pop out the first element in the Que
    $(\_, \_, d', \_) \leftarrow Que.top()$ ▷ Check the next element
    $Map[d] \leftarrow R_{tot}$
    SEARCH$(env.step(s, \mathbf{a}), d', R + Q(s, \mathbf{a}))$
```



(a) Checkers (b) Switches

Figure 1: Checkers and Switches environments.

choose the next critical d_{lim} effectively. When exploring the trajectory, we will store the possible action with their certified bound $d^k(s)$ into the priority queue. As the perturbation grows, these actions whose certified bound below the perturbation are going to be explored. Thus, after all trajectories for d_{lim} are explored, we will grow the d_{lim} to next perturbation magnitude by popping out the element in teh priority queue.

Comparison on time complexity

The time cost is related to the number of agents and complexity of the network, ranging from 30 mins to 2 days. For CROP-LoRE, the complexity is $O(H|S_{explored} \times (log|S_{explored} + |A|T))$, where $|S_{explored}|$ is the number of states that have been explored in the search procedure. H represents the horizon length, A denotes the size of action set and T is the time spent to sampling the noise.

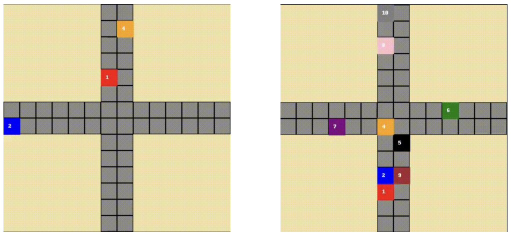
Compared with the complexity of CROP-LoRE, our method can reduce the size of action space $|A|$ to limit it by the number actions have been explored, which decrease the total complexity to $O(H|S_{explored} \times (log|S_{explored} + |A_{possible}|T))$.

D Environments settings

For the single agent environment, we follow the settings in the baseline. The Freeway is an Atari-2600 environments that can be implemented by OpenAI Gym (Brockman et al. 2016) on the top of the Arcade Learning Environments. The input states are high-dimentional images with shape $210 \times 160 \times 3$ and the actions are discrete actions. The experiments use NoFrameSkip-v4 version, where the randomness of the environments are fully controlled by setting the environments' random seed at the beginning.

As for the c-MARL environments, we choose two maps shown in Figure 2 from the ma-gym (Koul 2019): Checkers with two agents and switch with four agents, which are used in (Sunehag et al. 2017). The observations are byte values of size $3 \times 5 \times 5$ RGB images. The action spaces are: 0(Down), 1(Left), 2(Up), 3(Right), 4(Noop). The version used are Checkers-v0 and Switch4-v0, where each agent observes only it's local position.

Checkers The map of checkers contains apples and lemons. The red agent will give higher score for the team: +10 for the apple (green) and -10 for the lemon (yellow). The blue agent will give +1 for the apple (green) and -1 for the lemon. In this game, the red agent is expected to consume all apples and the blue agent need to leave them to its mate and eats the obstructing lemons.



(a) Four agents (b) Ten agents

Figure 2: Traffic Junction with four and ten agents environments

Switch This game is a grid world environment with four agents. Each agent expects to move to their home which is marked in the box with same outlined color. The challenge is how to pass through the narrow corridor, since at each time, only one agent can pass through. Whenever the agent reaches their home block, +5 score will be awarded, until all agents arrive their home or reach the maximum of 100 steps.

E Extra experiments

In this section, we show the performance of our certification method on *Traffic Junction* environments. On a grid of 14 by 14, there are a 4-way intersection. New cars (agents) join the grid with a probability from each of the four directions at each time step. However, there can never be more than $Nmax$ cars on the road at once. At each time, the car occupies a single block and is allotted one of three potential paths at random. Each agent has two options at a time step: gas, which moves it forward one cell on its path, or stop, which keeps it in place. Once a car arrives at its destination at the grid's edge, it will be eliminated.

The overlap of two cars' locations denotes the collision, which incurs a reward of -10. To prevent a traffic jam, each car receives reward of $\tau * r_{time} = -0.01\tau$ at every time step, where τ is the number time steps passed.

The results for per-state action certification in Traffic Junction environment are presented figure 3 and 4, where contains four and ten agents respectively. The first column of each figure shows the certification bound for each agent per state and the second column denotes the per-state bound for all agents. In table 1, we show the certified lower bound of global reward under the certified perturbation bound. As all global rewards are below zero, we will run the trajectory to the end without pruning to obtain the certified reward for all environments.

References

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Open-AI gym. *arXiv preprint arXiv:1606.01540*.

Cohen, J.; Rosenfeld, E.; and Kolter, Z. 2019. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, 1310–1320. PMLR.

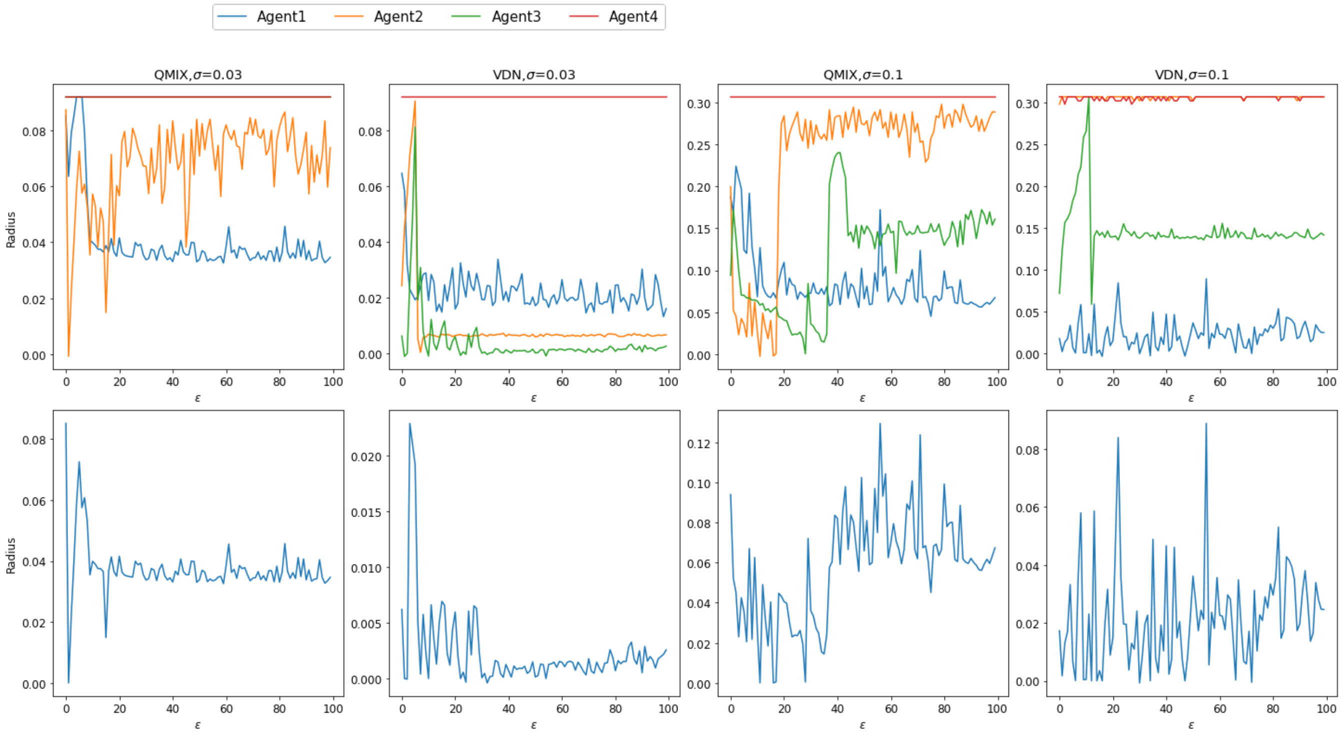


Figure 3: Certified robustness bound of perturbation for per step action in TrafficJunction with four agents.

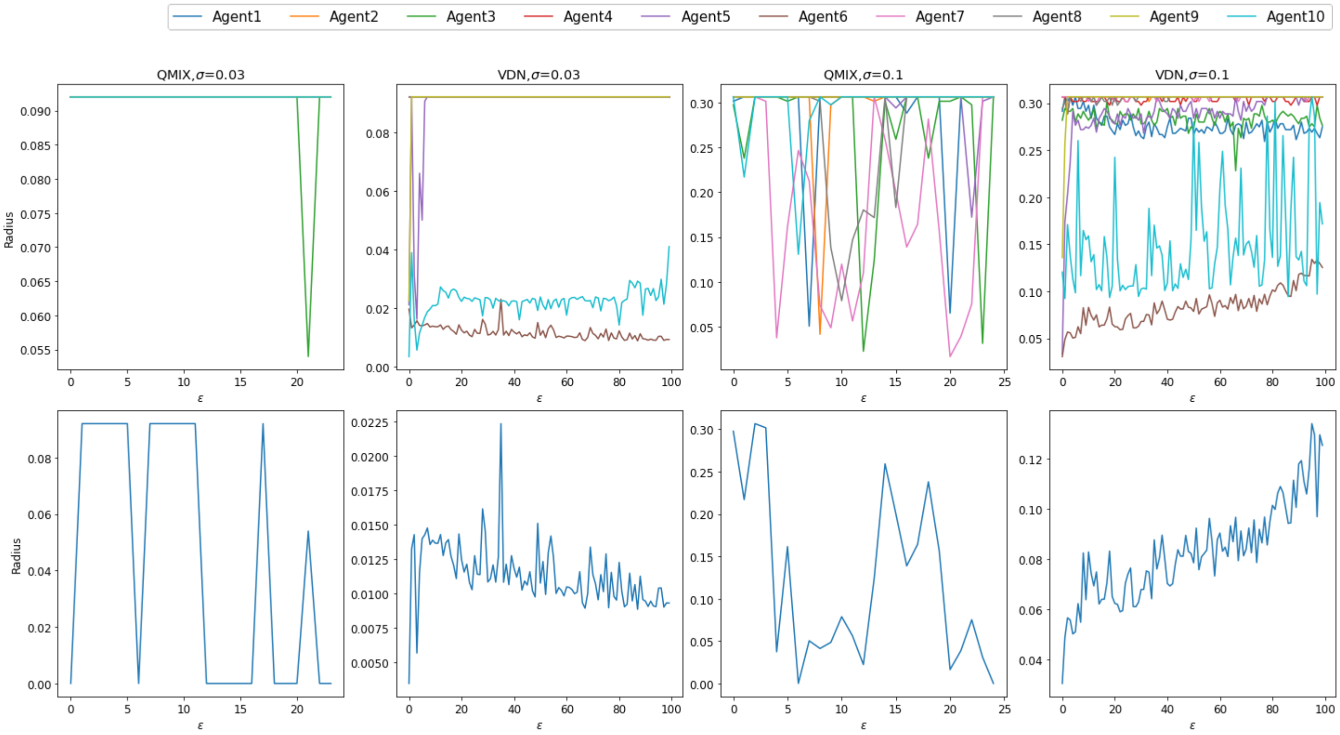


Figure 4: Certified robustness bound of perturbation for per step action in TrafficJunction with ten agents.

Models	Game	No.agent	$\sigma = 0.03$			$\sigma = 0.06$			$\sigma = 0.1$		
			ϵ_{cert}	Reward		ϵ_{cert}	Reward		ϵ_{cert}	Reward	
				Our	PGD		Our	PGD		Our	PGD
VDN	TrafficJunction	4	0.0171	-103.1	-102.83	0.0310	-103.25	-103.25	0.0536	-105.36	-105.36
QMIX	TrafficJunction	4	0.0150	-146.24	-134.5	0.0315	-109.94	-109.19	0.0501	-189.4	-172.7
VDN	TrafficJunction	10	0.0164	-202.0	-202.0	0.0347	-202.0	-202.0	0.0618	-202.0	-202.0
QMIX	TrafficJunction	10	0.0165	-175.23	-137.74	0.0398	-175.23	-139.6	0.0556	-175.45	-79

Table 1: Lower bound of global reward under the minimum certified bound of perturbation ϵ_{cert}

Goodman, L. A. 1965. On simultaneous confidence intervals for multinomial proportions. *Technometrics*, 7(2): 247–254.

Koul, A. 2019. ma-gym: Collection of multi-agent environments based on OpenAI gym. https://github.com/koulanurag/ma-gym.

Neyman, J.; and Pearson, E. S. 1933. IX. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694-706): 289–337.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J. Z.; Tuyls, K.; et al. 2017. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*.

Wu, F.; Li, L.; Huang, Z.; Vorobeychik, Y.; Zhao, D.; and Li, B. 2021. Crop: Certifying robust policies for reinforcement learning through functional smoothing. *arXiv preprint arXiv:2106.09292*.