

## A Proof for Propositions and Theorems

**Proposition 3.1** Denote  $v$  as the probability ratios  $\frac{q(a|s)}{p_\pi(a|s)}$  calculated from sampled trajectories. If there are sufficient number of sampled  $v$ , we have  $\mathbb{E}[v] = 1$  and  $\mathbb{E}[v \log v] \leq \text{Var}(v - 1)$ .

*Proof.* Denote  $p_\pi(s, a)$  and  $q(s, a)$  are the probability density function of the state-action distribution under different policies. Considering the divergence between  $q$  and  $p_\pi$  are small, we assume that the policy change will not cause the change in state distribution. We denote  $d(s)$  as the probability density function of the state distribution. From the definition of the probability density function, we know that  $\int p_\pi(s, a) d(s, a) = 1$ . Considering the current trajectories are sampled under policy  $p_\pi$ , we can obtain that

$$\begin{aligned} \mathbb{E}[v] &= \int p_\pi(s, a) \frac{q(a|s)}{p_\pi(a|s)} d(s, a) \\ &= \int p_\pi(s, a) \frac{q(a|s) \times d(s)}{p_\pi(a|s) \times d(s)} d(s, a) \\ &= \int p_\pi(s, a) \frac{q(s, a)}{p_\pi(s, a)} d(s, a) \\ &= \int q(s, a) d(s, a) = 1, \end{aligned} \tag{10}$$

as  $p(s, a)$  is a probability density function. Therefore,  $\mathbb{E}[v] = 1$  is proven. Q.E.D  $\square$

**Theorem 3.7** For a probability ratio vector  $\bar{v}$ , if the variance of  $\bar{v}$  is constant, then the upper bound of the approximated forward KL divergence  $D_{\text{KL}}(\pi \parallel \pi_\theta)$ , will decrease as the element-wise lower bound of  $\bar{v}$  increase.

*Proof.* Using the same symbol in **Proof of Proposition 3.1**,  $\bar{v}$  is the vector consists of  $v = \frac{p_{\pi_\theta}(s, a)}{p_\pi(s, a)}$  and the definition of the forward KL divergence  $D_{\text{KL}}(\pi \parallel \pi_\theta)$  can be expressed as

$$\begin{aligned} D_{\text{KL}}(\pi \parallel \pi_\theta) &= \int p_\pi(s, a) \log \frac{p_\pi(s, a)}{p_{\pi_\theta}(s, a)} \\ &= -\mathbb{E}[\log v] \\ &= -\frac{\sum \log v}{N} \\ &= -\log\left(\prod_{i=1}^N v_i\right)^{\frac{1}{N}}, \end{aligned} \tag{11}$$

where  $N$  is the number of elements in  $\bar{v}$ . According to the Theorem in Cartwright & Field (1978), we obtain that

$$\mathbb{E}(v) - \prod_{i=1}^N v_i^{\frac{1}{N}} \leq \frac{1}{2N \min v_i} \sum_{i=1}^N (v_i - \mathbb{E}(v))^2. \tag{12}$$

As we know  $\mathbb{E}(v) = 1$  from Proposition 3.1,  $\prod_{i=1}^N v_i^{\frac{1}{N}} > 0$ , and  $\sum_{i=1}^N (v_i - \mathbb{E}(v))^2 = N \cdot \text{Var}(\bar{v})$ , we have

$$\begin{aligned} \prod_{i=1}^N v_i^{\frac{1}{N}} &\geq 1 - \frac{\text{Var}(\bar{v})}{2 \min v_i} \\ \log\left(\prod_{i=1}^N v_i^{\frac{1}{N}}\right) &\geq \log\left(1 - \frac{\text{Var}(\bar{v})}{2 \min v_i}\right) \\ D_{\text{KL}}(\pi \parallel \pi_\theta) &\leq -\log\left(1 - \frac{\text{Var}(\bar{v})}{2 \min v_i}\right) \approx \frac{\text{Var}(\bar{v})}{2 \min v_i}. \end{aligned} \tag{13}$$

As  $\text{Var}(\bar{v})$  is a constant, Equation (13) proves the upper bound of  $D_{\text{KL}}(\pi \parallel \pi_\theta)$  is  $\frac{\text{Var}(\bar{v})}{2 \min v_i}$ , showing that the upper bound of  $D_{\text{KL}}(\pi \parallel \pi_\theta)$ , will decrease as the element-wise lower bound of  $\bar{v}$ ,  $\min v_i$ , increase.

Q.E.D  $\square$

**Theorem 3.4** Given a feasible optimization problem of the form:

$$\begin{aligned} &\underset{\bar{v}}{\text{maximize}} \quad \bar{v} \cdot \mathbf{A} \\ &\text{s.t.} \quad \bar{v} \cdot \mathbf{A}_c \leq D \\ &\quad \|\bar{v}\|_2 \leq 2N\delta \quad \mathbb{E}(\bar{v}) = \mathbb{E}(\mathbf{A}) = \mathbb{E}(\mathbf{A}_c) = 0 \end{aligned}$$

where  $\bar{\mathbf{v}}$ ,  $\mathbf{A}$ , and  $\mathbf{A}_c$  are  $N$ -dimensional vectors, then the optimal solution  $\bar{\mathbf{v}}$  will lie in the  $A$ - $A_c$  plane determined by  $\mathbf{A}_c$  and  $\mathbf{A}$ .

*Proof.* Assuming  $\bar{\mathbf{v}}$ ,  $\mathbf{A}$ , and  $\mathbf{A}_c$  can be represented by three orthonormal basis vectors  $\mathbf{i}$ ,  $\mathbf{j}$ , and  $\mathbf{k}$ , where  $\bar{\mathbf{v}} = a_1\mathbf{i} + b_1\mathbf{j} + c_1\mathbf{k}$ ,  $\mathbf{A} = a_2\mathbf{i} + b_2\mathbf{j}$ , and  $\mathbf{A}_c = a_3\mathbf{i}$ , then the optimization problem becomes:

$$\begin{aligned} & \underset{a_1, b_1}{\text{maximize}} && a_1 a_2 + b_1 b_2 \\ & \text{s.t.} && a_1 \leq D/a_3 \\ & && a_1^2 + b_1^2 \leq 4N^2\delta^2 - c_1^2 \end{aligned} \quad (14)$$

From the geometric interpretation, we can find the optimal solution of the above problem always exists on the circle  $a_1^2 + b_1^2 = 4N^2\delta^2 - c_1^2$ . By increasing the radius of the circle, the line  $a_1 a_2 + b_1 b_2$  will have a larger intercept. Thus, the aforementioned problem will get its optimal solution when  $c_1 = 0$ , i.e.,  $\bar{\mathbf{v}}$  will lie in the  $A$ - $A_c$  plane determined by  $\mathbf{A}_c$  and  $\mathbf{A}$ .

Q.E.D □

## B Derivation in EM framework

### B.1 Derivation of evidence lower bound

Following the definition in Section 3.1, we have  $p(O = 1|s, a) \propto \exp(A(s, a)/\alpha)$ . Assume the likelihood of acting  $a$  under  $s$  and  $\theta$  is  $p(a|s, \theta) = p_{\pi_\theta}(a|s) * p(\theta)$ . Then we can obtain following evidence lower bound(ELBO)

$$\begin{aligned} \log p_{\pi_\theta}(O = 1) &= \log \int p(O = 1|s, a) * p_{\pi_\theta}(s, a) * p(\theta) d(s, a) \\ &= \log \mathbb{E}_{s \sim d^q, a \sim q} \left[ \frac{p(O = 1|s, a) * p_{\pi_\theta}(s, a) * p(\theta)}{q(s, a)} \right] \\ &\geq \mathbb{E}_{s \sim d^q, a \sim q} \left[ \log p(O = 1|s, a) + \log \frac{p_{\pi_\theta}(s, a)}{q(s, a)} + \log p(\theta) \right] \end{aligned} \quad (15)$$

where  $d^q$  is the state distribution under theoretical optimal distribution  $q$ . If we assume that the sampled policy  $\pi$  and  $q$  is enough close that  $d^\pi = d^q$ , then

$$\begin{aligned} \log p_{\pi_\theta}(O = 1) &\geq \mathbb{E}_{s \sim d^q, a \sim q} \log p(O = 1|s, a) + \mathbb{E}_{s \sim d^q, a \sim q} \log \frac{p_{\pi_\theta}(s, a)}{q(s, a)} + \log p(\theta) \\ &\propto \mathbb{E}_{s \sim d^\pi, a \sim q} [A(s, a)] + \alpha \mathbb{E}_{s \sim d^\pi, a \sim q} \log \frac{p_{\pi_\theta}(a|s)}{q(a|s)} + \log p(\theta) \\ &= \mathbb{E}_{s \sim d^\pi, a \sim \pi} \left[ \frac{q(a|s)}{p_\pi(a|s)} A(s, a) \right] - \alpha D_{\text{KL}}(q \parallel \pi_\theta) + \log p(\theta) \end{aligned} \quad (16)$$

Thus, the ELBO in Equation (1) is obtained.

### B.2 Derivation in M-step

Recall Equation (7) in Section 3.4, we have following optimization problem

$$\underset{\theta}{\text{maximize}} \quad -\alpha D_{\text{KL}}(q \parallel \pi_\theta) + \log p(\theta). \quad (17)$$

Consider  $\theta$  is a Gaussian prior around the policy parameter of sampled policy  $\hat{\theta}$ , i.e.,  $\theta \sim \mathcal{N}(\hat{\theta}, \frac{F_{\hat{\theta}}}{\alpha\beta})$ . Therefore, the problem above will become

$$\underset{\theta}{\text{maximize}} \quad -\alpha D_{\text{KL}}(q \parallel \pi_\theta) - \alpha\beta(\theta - \hat{\theta})^T F_{\hat{\theta}}^{-1}(\theta - \hat{\theta}). \quad (18)$$

Note that  $(\theta - \hat{\theta})^T F_{\hat{\theta}}^{-1}(\theta - \hat{\theta})$  is the second order estimation of  $D_{\text{KL}}(\pi \parallel \pi_\theta)$ , we have

$$\underset{\theta}{\text{maximize}} \quad -D_{\text{KL}}(q \parallel \pi_\theta) - \beta D_{\text{KL}}(\pi \parallel \pi_\theta). \quad (19)$$

By converting the soft KL constraint into a hard constraint, we can obtain

$$\begin{aligned} & \underset{\theta}{\text{minimize}} && D_{\text{KL}}(q \parallel \pi_\theta) \\ & \text{s.t.} && D_{\text{KL}}(\pi \parallel \pi_\theta) \leq \delta, \end{aligned} \quad (20)$$

which is the same optimization problem as in Equation (8).

## C Details in heuristic algorithm and M-step

### C.1 Heuristic algorithm

The detailed steps of iteratively heuristic algorithm are shown in Algorithm 1. Note that, after masking, the masked elements are removed from the original vector, which means the size of  $\bar{\mathbf{v}}'$ ,  $\mathbf{A}'_c$ , and  $\mathbf{A}'$  is smaller than  $\bar{\mathbf{v}}$ ,  $\mathbf{A}_c$ , and  $\mathbf{A}$ .

---

#### Algorithm 1 Iteratively Heuristic Algorithm

---

**Input:** Advantage vector  $\mathbf{A}$ ,  $\mathbf{A}_c$

Using QR decomposition to orthonormalize  $\mathbf{A}$  and  $\mathbf{A}_c$  into orthogonal unit vectors  $\tilde{\mathbf{A}}_c = k\mathbf{A}_c$  and  $\tilde{\mathbf{A}}$ .

Find  $\theta'$  that makes  $\bar{\mathbf{v}} = 2N\delta'(\cos \theta' \tilde{\mathbf{A}}_c + \sin \theta' \tilde{\mathbf{A}})$  become the optimal solution of the problem in Theorem 3.4.

**while**  $\bar{\mathbf{v}}$  violates element-wise lower bound constraint **do**

Clip the value in  $\bar{\mathbf{v}}$  to element-wise lower bound.

Record the clipped values and corresponding cost advantage value in  $\bar{\mathbf{v}}^m$  and  $\mathbf{A}_c^m$ , mask these clipped values and their corresponding advantage values to obtain new vector  $\bar{\mathbf{v}}'$  and corresponding advantage vectors  $\mathbf{A}'_c$  &  $\mathbf{A}'$ .

Subtract the mean of  $\mathbf{A}'_c$  and  $\mathbf{A}'$  to obtain  $\mathbf{A}''_c$  &  $\mathbf{A}''$ .

Initial a new zero vector  $\bar{\mathbf{v}}''$  with the same size of  $\bar{\mathbf{v}}'$

Calculate the  $l_2$ -norm bound of  $\bar{\mathbf{v}}''$ , i.e.,  $\delta'$ , using  $D(\bar{\mathbf{v}}'') = \mathbb{E}(\bar{\mathbf{v}}''^2) - \mathbb{E}(\bar{\mathbf{v}}'')^2$ .

Using QR decomposition to orthonormalize  $\mathbf{A}''$  and  $\mathbf{A}''_c$  into orthogonal unit vectors  $\tilde{\mathbf{A}}''_c = k\mathbf{A}''_c$  and  $\tilde{\mathbf{A}}''$ .

Find  $\theta'$  that maximize  $\bar{\mathbf{v}}'' \mathbf{A}''$  while satisfy  $\bar{\mathbf{v}}'' \mathbf{A}''_c \leq Nd' - \bar{\mathbf{v}}^m \mathbf{A}_c^m - M \cdot \text{mean}(\bar{\mathbf{v}}') \cdot \text{mean}(\mathbf{A}'_c)$ , where  $M$  is the number of element in  $\bar{\mathbf{v}}^m$ ,  $\bar{\mathbf{v}}'' = 2N\delta'(\cos \theta' \tilde{\mathbf{A}}''_c + \sin \theta' \tilde{\mathbf{A}}'')$ .

Concatenate  $\bar{\mathbf{v}}^m$  and  $\bar{\mathbf{v}}'' + \text{mean}(\bar{\mathbf{v}}')$  according to the recorded location to obtain the new  $\bar{\mathbf{v}}$

**end while**

Obtain optimal probability ratio  $\mathbf{v} = \bar{\mathbf{v}} + 1$ .

---

### C.2 Modified update gradient in M-step when conducting recovery update

In the recovery update process described in Section 3.2.2, the gradient update in the M-step is modified from  $(v - \frac{p\pi_\theta}{p\pi}) \frac{\partial \pi_\theta}{\partial \theta}$  to  $((\beta(v - \frac{p\pi_\theta}{p\pi}) + (1 - \beta)\mathbf{A}'_c) \frac{\partial \pi_\theta}{\partial \theta})$ , where  $\mathbf{A}'_c$  is the projection of  $v - \frac{p\pi_\theta}{p\pi}$  onto the cost advantage vector  $\mathbf{A}_c$ .

In Figure 5, the tracking trajectories with and without gradient modification are compared. The yellow target point represents the location of  $v$ , and the blue start point represents the initial location of  $\frac{p\pi_\theta}{p\pi}$ . The dashed optimal trajectory demonstrates that the optimal way to approach  $v$  is to first enter the feasible region quickly and then follow the zero reward boundary. This approach allows the agent to satisfy the constraint while preserving the reward return for most of the trajectory. The blue line represents the trajectory before gradient modification. In this case,  $\frac{p\pi_\theta}{p\pi}$  directly heads towards  $v$ , leading to a violation of the cost constraint during the initial part of the tracking. On the other hand, the orange line represents the trajectory after gradient modification, which closely follows the optimal path at the beginning of the tracking. This modification ensures that the agent can satisfy the constraint throughout the entire tracking path.

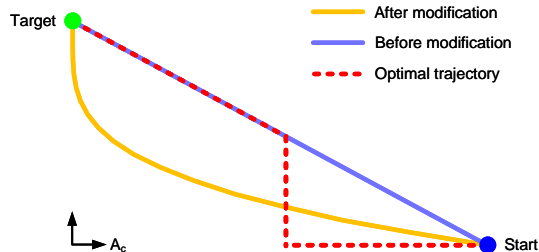


Figure 5: The tracking trajectories with and without modification.

### C.3 Clipping in M-step for constrain KL divergence

To satisfy the KL constraint in Equation (9), we employ a clipping technique similar to PPO to constrain the KL divergence in the M-step. In this case, we clip the lower bound of  $\frac{p_{\pi_{\theta}}}{p_{\pi}}$  to 0.6. Considering the original loss function  $\mathbb{E} \left[ \left( v - \frac{p_{\pi_{\theta}}}{p_{\pi}} \right)^2 \right]$ , after taking the derivative, it can be rewritten as  $-\mathbb{E} \left[ \left( v - \frac{p_{\pi_{\theta}}}{p_{\pi}} \right) \frac{p_{\pi_{\theta}}}{p_{\pi}} \right]$ . In this form,  $\left( v - \frac{p_{\pi_{\theta}}}{p_{\pi}} \right)$  can be treated as the advantage value in PPO, which does not require gradient. Therefore, following the clip technique in PPO, the new loss function can be expressed as  $-\mathbb{E} \left[ \min \left( \left( v - \frac{p_{\pi_{\theta}}}{p_{\pi}} \right) \frac{p_{\pi_{\theta}}}{p_{\pi}}, \left( v - \frac{p_{\pi_{\theta}}}{p_{\pi}} \right) \text{Clip} \left( \frac{p_{\pi_{\theta}}}{p_{\pi}}, 0.6 \right) \right) \right]$ , where Clip is a function that clips the value smaller than 0.6 to 0.6.

### C.4 The outline of CPPO method

---

#### Algorithm 2 CPPO Outline

---

**Input:** Policy network  $\pi_{\theta}$ , Value network  $V$ ,  $V_c$   
**while** Stopping criteria not met **do**  
    Rollout sampling from the environment, generate trajectories  $\tau \sim \pi_{\theta}$ .  
    Calculate advantage value  $A$  and  $A_c$  from  $\tau$ .  
    **if** Current policy violates the constraint **then**  
        Conduct **recovery update** in **E-step** to optimal policy  $\mathbf{v}$ .  
    **else**  
        Conduct **normal update** in **E-step** to optimal policy  $\mathbf{v}$ .  
    **end if**  
    Conduct **M-step** according to Equation (9) to update policy parameter  $\theta$  based on  $\mathbf{v}$ .  
    Update value networks using GAE.  
**end while**

---

## D Details about test environments

The environment parameters used in our experiments are listed in Table 1. The implementation of the Safety Gym environment can be found at <https://github.com/openai/safety-gym> as an open-source project. Similarly, the open-source implementation of the Circle environment can be found at <https://github.com/ymzhang01/mujoco-circle>. The PointCircle environment was created based on this open-source implementation, following the same settings as described in Achiam et al. (2017).

TABLE 1: THE ENVIRONMENT PARAMETERS

ENVIRONMENT	CARPUSH	POINTGOAL	POINTPUSH	POINTCIRCLE	ANTCIRCLE
BATCH SIZE	$3 \times 10^4$	$3 \times 10^4$	$3 \times 10^4$	1000	$3 \times 10^4$
TOTAL STEPS	$1 \times 10^7$	$1 \times 10^7$	$1 \times 10^7$	$2 \times 10^5$	$1 \times 10^7$
ROLLOUT LENGTH	1000	1000	1000	50	500
CONSTRAINT	25	25	25	5	50

## E Details for experiments

The hyperparameters of proposed method and baseline methods are shown in Table 2. The baseline methods are modified from <https://github.com/openai/safety-starter-agents> to a Pytorch version. The experiments are conducted on a HPC with 24 nodes, each node has 32 CPU cores and 2 Nvidia A100 GPUs.

Note that, in CPPO, setting the KL divergence constraint to 0.02 does not directly determine the value of  $\delta'$  in Equation (5). Although Proposition 3.1 states that  $\text{Var}(v)$  determines the upper bound of the reverse KL divergence, it does not provide a lower bound for the reverse KL divergence. Consequently, the update step may become very small. To address this issue, we can consider the inequality

$$(2 \log 2 - 1)(x - 1)^2 + (x - 1) \leq x \log x, \quad (21)$$

which holds for  $x$  values smaller than 2. This inequality implies that  $(2 \log 2 - 1)\text{Var}(v)$  could serve as a lower bound for the reverse KL divergence. In order to prevent the KL divergence from becoming too small, we

TABLE 2: HYPERPARAMETERS SETTING FOR EACH ALGORITHM IN EXPERIMENT

ALGORITHM	PPO-LAG	TRPO-LAG	CPO	CPPO
POLICY NETWORK	(64,64)	(64,64)	(64,64)	(64,64)
VALUE NETWORK	(64,64)	(64,64)	(64,64)	(64,64)
NETWORK ACTIVATION	tanh	tanh	tanh	tanh
DISCOUNT FACTOR $\gamma$ FOR RETURN	0.99	0.99	0.99	0.99
GAE $\lambda$ FOR RETURN	0.97	0.97	0.97	0.97
DISCOUNT FACTOR $\gamma_c$ FOR COST	0.99	0.99	0.99	0.99
GAE $\lambda_c$ FOR COST	0.95	0.95	0.95	0.95
LEARNING RATE FOR POLICY NETWORK	$3 \times 10^{-4}$	N/A	N/A	$1 \times 10^{-4}$
LEARNING RATE FOR VALUE NETWORK	$1 \times 10^{-3}$	$1 \times 10^{-3}$	$1 \times 10^{-3}$	$1 \times 10^{-3}$
LEARNING RATE FOR LAGRANGIAN MULTIPLIER	$5 \times 10^{-2}$	$5 \times 10^{-2}$	N/A	N/A
KL DIVERGENCE CONSTRAINT	0.01	0.01	0.01	0.02
CLIPPING COEFFICIENT	0.2	N/A	N/A	0.4 <sup>1</sup>
$\beta$ IN RECOVERY UPDATE	N/A	N/A	N/A	0.3

<sup>1</sup> THIS COEFFICIENT IS ONLY USED FOR CLIPPING THE LOWER BOUND, SEE APPENDIX C FOR DETAILS.

choose  $\delta' = 0.02/(2 \log 2 - 1)$ , ensuring that the reverse KL divergence of the optimal  $v$  lies within the range  $(0.02, 0.02/(2 \log 2 - 1))$ .

*Remark E.1.* By applying Cantelli's inequality, we can derive the inequality  $\Pr(v \geq 2) \leq \frac{\text{Var}(v)}{\text{Var}(v)+1}$ . In the case where  $\text{Var}(v)$  is sufficiently small, this upper bound can be approximated as  $\text{Var}(v)$ . Since  $\text{Var}(v) = 0.02$  is a small value, it validates the aforementioned assumption that  $v$  is smaller than 2.