On the Duality Gap of Constrained Cooperative Multi-Agent Reinforcement Learning

Ziyi Chen[1]   Yi Zhou[2]   Heng Huang[1]

[1]Department of CS, University of Maryland

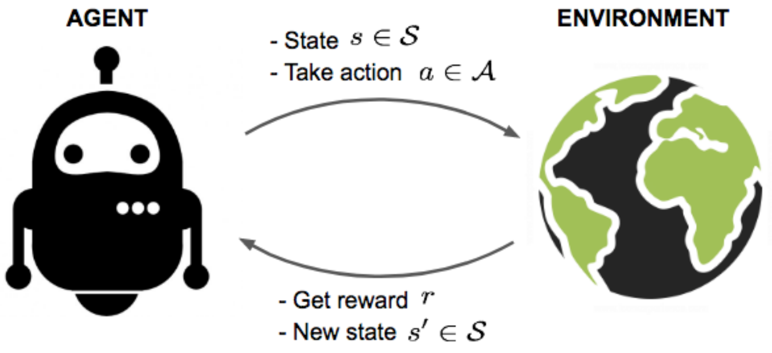
[2]Department of ECE, University of Utah

Outline

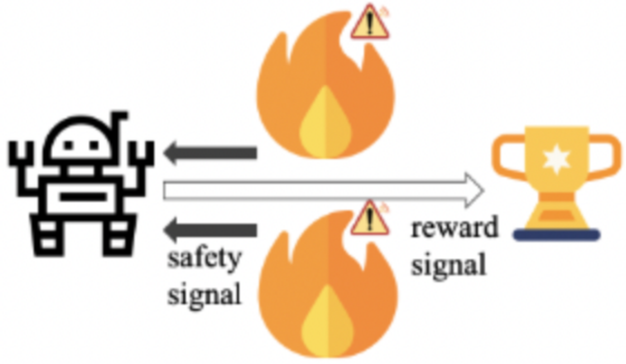
- **Problem Formulation**
- Challenges in Constrained Cooperative MARL
- Existing Primal-Dual Algorithm has Duality Gap>0
- Our Decentralized Primal Algorithm
- Numerical Examples: Neither Algorithm Outperforms

Problem Formulation

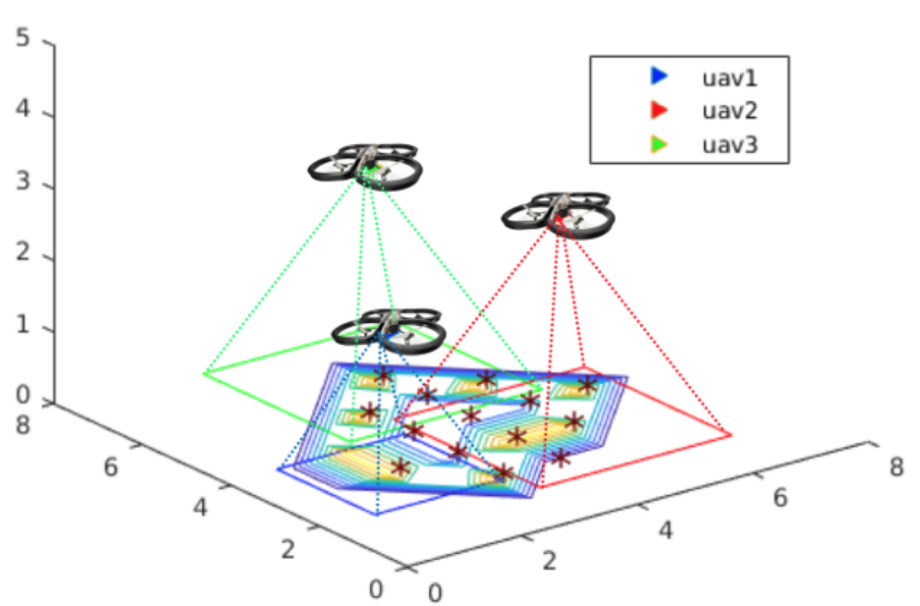
Reinforcement Learning (RL)



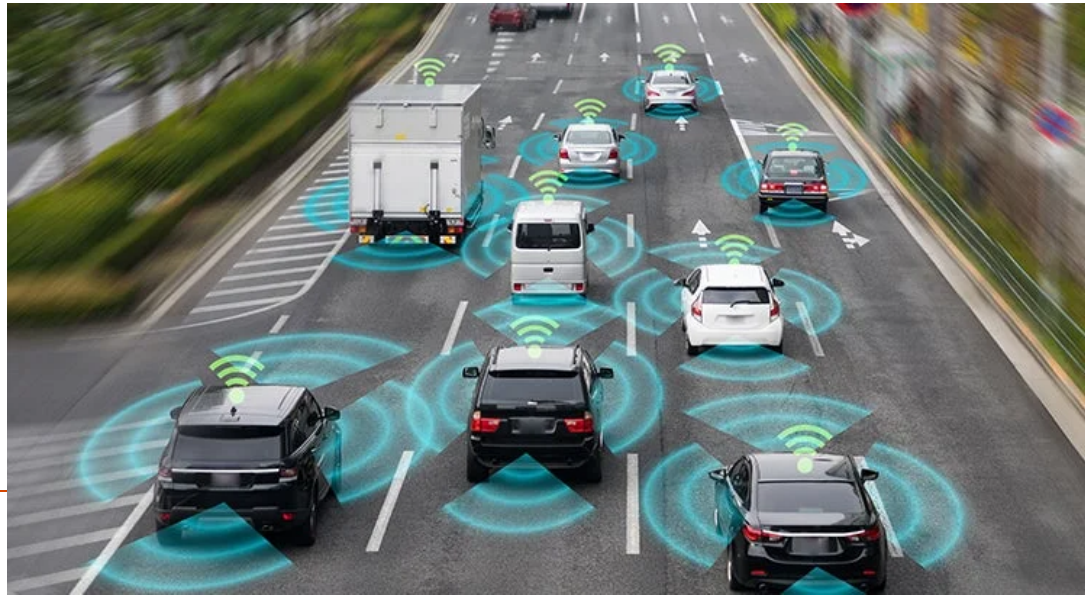
Constrained Reinforcement Learning



Cooperative Multi-agent Reinforcement Learning (Cooperative MARL)



Constrained Cooperative MARL (Our focus)



Problem Formulation: Constrained MARL

- Environmental state s_t.

Problem Formulation: Constrained MARL

- Environmental state s_t.
- Each agent $m \in \{1, \ldots, M\}$
- Takes action $a_t^{(m)} \sim \text{policy } \pi^{(m)}(\cdot \mid s_t)$.

Problem Formulation: Constrained MARL

- Environmental state s_t.
- Each agent $m \in \{1, \ldots, M\}$
  - Takes action $a_t^{(m)} \sim$ policy $\pi^{(m)}(\cdot \mid s_t)$.
  - Receives reward $r_{0,t}^{(m)} = r_0^{(m)}(s_t, a_t)$.

Problem Formulation: Constrained MARL

- Environmental state s_t.
- Each agent $m \in \{1,\ldots,M\}$
  - Takes action $a_t^{(m)} \sim \text{policy } \pi^{(m)}(\cdot \mid s_t)$.
  - Receives reward $r_{0,t}^{(m)} = r_0^{(m)}(s_t, a_t)$.
  - Receives K safety scores $r_{k,t}^{(m)} = r_k^{(m)}(s_t, a_t)$ $(k \in \{1,\ldots,K\})$.

Problem Formulation: Constrained MARL

- Environmental state s_t.
- Each agent $m \in \{1, \ldots, M\}$
  - Takes action $a_t^{(m)} \sim \text{policy } \pi^{(m)}(\cdot \mid s_t)$.
  - Receives reward $r_{0,t}^{(m)} = r_0^{(m)}(s_t, a_t)$.
  - Receives K safety scores $r_{k,t}^{(m)} = r_k^{(m)}(s_t, a_t)$ ($k \in \{1, \ldots, K\}$).
- Environment transitions to the next state $s_{t+1} \sim \text{transition kernel } P(\cdot \mid s_t, a_t)$.

Problem Formulation: Constrained MARL

- Environmental state s_t.
- Each agent $m \in \{1, \ldots, M\}$
  - Takes action $a_t^{(m)} \sim \text{policy } \pi^{(m)}(\cdot \mid s_t)$.
  - Receives reward $r_{0,t}^{(m)} = r_0^{(m)}(s_t, a_t)$.
  - Receives K safety scores $r_{k,t}^{(m)} = r_k^{(m)}(s_t, a_t)$ ($k \in \{1, \ldots, K\}$).
- Environment transitions to the next state $s_{t+1} \sim \text{transition kernel } P(\cdot \mid s_t, a_t)$.
- Objective:

$$\max_{\text{product policy } \pi} V_0(\pi) := \mathbb{E}_\pi\Big[\sum_{t=0}^{\infty} \gamma^t \bar{r}_{0,t} \Big| s_0 \sim \rho\Big], \qquad \text{(reward)}$$

$$\text{s.t. } V_k(\pi) := \mathbb{E}_\pi\Big[\sum_{t=0}^{\infty} \gamma^t \bar{r}_{k,t} \Big| s_0 \sim \rho\Big] \geq \xi_k, \quad k = 1, \ldots, K \qquad \text{(safety)}$$

Outline

- Problem Formulation
- **Challenges in Constrained Cooperative MARL**
- Existing Primal-Dual Algorithm has Duality Gap>0
- Our Decentralized Primal Algorithm
- Numerical Examples: Neither Algorithm Outperforms

Challenges in Constrained Cooperative MARL

- Occupation measure

$$\nu_\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_\pi(s_t = s, a_t = a \,|\, s_0 \sim \rho), \quad \nu_\pi(s) := \sum_a \nu_\pi(s,a)$$

Challenges in Constrained Cooperative MARL

- Occupation measure

$$\nu_\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_\pi(s_t = s, a_t = a \,|\, s_0 \sim \rho), \quad \nu_\pi(s) := \sum_a \nu_\pi(s,a)$$

- Constrained MARL is equivalent to

$$\max_\nu \frac{1}{1-\gamma}\sum_{s,a}\bar{r}_0(s,a)\nu(s,a)$$

Challenges in Constrained Cooperative MARL

- Occupation measure

$$\nu_\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_\pi(s_t = s, a_t = a \,|\, s_0 \sim \rho), \quad \nu_\pi(s) := \sum_a \nu_\pi(s,a)$$

- Constrained MARL is equivalent to

$$\max_\nu \frac{1}{1-\gamma}\sum_{s,a}\bar{r}_0(s,a)\nu(s,a)$$

s.t. (Occupation): $\nu \geq 0, \ \sum_{s,a}\nu(s,a) = 1, \ \sum_a \nu(s',a) = (1-\gamma)\rho(s') + \gamma\sum_{s,a}\nu(s,a)\mathscr{P}(s'\,|\,s,a); \ \forall s'$

Challenges in Constrained Cooperative MARL

- Occupation measure

$$\nu_\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_\pi(s_t = s, a_t = a \mid s_0 \sim \rho), \quad \nu_\pi(s) := \sum_a \nu_\pi(s,a)$$

- Constrained MARL is equivalent to

$$\max_\nu \frac{1}{1-\gamma}\sum_{s,a}\bar{r}_0(s,a)\nu(s,a)$$

s.t. (Occupation): $\nu \geq 0, \ \sum_{s,a}\nu(s,a) = 1, \ \sum_a \nu(s',a) = (1-\gamma)\rho(s') + \gamma\sum_{s,a}\nu(s,a)\mathscr{P}(s' \mid s,a); \ \forall s'$

(Safety): $\frac{1}{1-\gamma}\sum_{s,a}\bar{r}_k(s,a)\nu(s,a) \geq \xi_k; \ k = 1,2,\ldots,K$

Challenges in Constrained Cooperative MARL

- Occupation measure

$$\nu_\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}_\pi(s_t = s, a_t = a \,|\, s_0 \sim \rho), \quad \nu_\pi(s) := \sum_a \nu_\pi(s,a)$$

- Constrained MARL is equivalent to

$$\max_\nu \frac{1}{1-\gamma}\sum_{s,a}\bar{r}_0(s,a)\nu(s,a)$$

s.t. (Occupation): $\nu \geq 0, \ \sum_{s,a}\nu(s,a) = 1, \ \sum_a \nu(s',a) = (1-\gamma)\rho(s') + \gamma\sum_{s,a}\nu(s,a)\mathscr{P}(s'|s,a); \ \forall s'$

(Safety): $\frac{1}{1-\gamma}\sum_{s,a}\bar{r}_k(s,a)\nu(s,a) \geq \xi_k; \ k = 1,2,\ldots,K$

Quadratic!

No known algorithm in polynomial time!

(Product policy): $\nu(s,a)\sum_{a'}\nu(s,a') = \sum_{a'^{(m)}}\nu\left(s,[a'^{(m)},a^{(\backslash m)}]\right)\cdot\sum_{a'^{(\backslash m)}}\nu\left(s,[a^{(m)},a'^{(\backslash m)}]\right); \ \forall s,a$

Challenges in Constrained Cooperative MARL

❖ Constrained **single-agent** RL is equivalent to

$$\max_{\nu} \frac{1}{1-\gamma} \sum_{s,a} \bar{r}_0(s,a)\nu(s,a)$$

Linear Programming!

Can solve in polynomial time!

s.t. (Occupation): $\nu \geq 0, \ \sum_{s,a} \nu(s,a) = 1, \ \sum_{a} \nu(s',a) = (1-\gamma)\rho(s') + \gamma \sum_{s,a} \nu(s,a)\mathscr{P}(s'\,|\,s,a); \ \ \forall s'$

(Safety): $\frac{1}{1-\gamma} \sum_{s,a} \bar{r}_k(s,a)\nu(s,a) \geq \xi_k; \ \ k = 1,2,\ldots,K$

~~(Product policy): $\nu(s,a)\sum_{a'} \nu(s,a') = \sum_{a'^{(m)}} \nu\big(s,[a'^{(m)}, a^{(\backslash m)}]\big) \cdot \sum_{a'^{(\backslash m)}} \nu\big(s,[a^{(m)}, a'^{(\backslash m)}]\big); \ \forall s,a$~~

Challenges in Constrained Cooperative MARL

❖ Constrained **single-agent** RL is equivalent to

$$\max_{\nu} \frac{1}{1-\gamma} \sum_{s,a} \bar{r}_0(s,a)\nu(s,a)$$

Linear Programming!

Can solve in polynomial time!

s.t. (Occupation): $\nu \geq 0, \ \sum_{s,a} \nu(s,a) = 1, \ \sum_{a} \nu(s',a) = (1-\gamma)\rho(s') + \gamma \sum_{s,a} \nu(s,a)\mathscr{P}(s' \mid s,a); \ \forall s'$

(Safety): $\frac{1}{1-\gamma} \sum_{s,a} \bar{r}_k(s,a)\nu(s,a) \geq \xi_k; \ k = 1,2,\ldots,K$

~~(Product policy): $\nu(s,a) \sum_{a'} \nu(s,a') = \sum_{a'^{(m)}} \nu\big(s,[a'^{(m)}, a^{(\backslash m)}]\big) \cdot \sum_{a'^{(\backslash m)}} \nu\big(s,[a^{(m)}, a'^{(\backslash m)}]\big); \ \forall s,a$~~

❖ Cooperative MARL **(no constraints)**:

Greedy deterministic solution $\pi^*(s) \in \arg\max_{\pi} Q^*(s,a)$, **efficiently obtain!**

Outline

- Problem Formulation
- Challenges in Constrained Cooperative MARL
- **Existing Primal-Dual Algorithm has Duality Gap>0**
- Our Decentralized Primal Algorithm
- Numerical Examples: Neither Algorithm Outperforms

Duality Gap>0

❖ Constrained MARL is also equivalent to:

$$\max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda) := V_0(\pi) + \sum_{k=1}^{K} \lambda_k \left[V_k(\pi) - \xi_k \right]$$

Duality Gap>0

❖ Constrained MARL is also equivalent to:

$$\max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda) := V_0(\pi) + \sum_{k=1}^{K} \lambda_k \left[V_k(\pi) - \xi_k \right]$$

❖ Duality gap:

$$\Delta := \min_{\lambda \in \mathbb{R}_+^K} \max_{\pi} L(\pi, \lambda) - \max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda)$$

Duality Gap>0

- Constrained MARL is also equivalent to:

$$\max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda) := V_0(\pi) + \sum_{k=1}^{K} \lambda_k \left[V_k(\pi) - \xi_k \right]$$

- Duality gap:

$$\Delta := \min_{\lambda \in \mathbb{R}_+^K} \max_{\pi} L(\pi, \lambda) - \max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda)$$

- Primal-Dual Algorithm (exact version):

$$\pi_t = \arg\max_{\pi} L(\pi, \lambda_t)$$

$$\lambda_{t+1} = \mathrm{Proj}_{\Lambda}\left[\lambda_t - \beta \nabla_{\lambda} L(\pi_t, \lambda_t) \right]$$

Duality Gap>0

❖ Constrained MARL is also equivalent to:

$$\max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda) := V_0(\pi) + \sum_{k=1}^{K} \lambda_k \left[V_k(\pi) - \xi_k \right]$$

❖ Duality gap:

$$\Delta := \min_{\lambda \in \mathbb{R}_+^K} \max_{\pi} L(\pi, \lambda) - \max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda)$$

❖ Primal-Dual Algorithm (exact version):

$$\pi_t = \arg\max_{\pi} L(\pi, \lambda_t)$$

$$\lambda_{t+1} = \mathrm{Proj}_{\Lambda}\left[\lambda_t - \beta \nabla_{\lambda} L(\pi_t, \lambda_t) \right]$$

❖ Our convergence results for Cooperative MARL:

- Fact 1: $\Delta > 0$ for some examples. (VS. $\Delta = 0$ for constrained **single-agent** RL)

Duality Gap>0

❖ Constrained MARL is also equivalent to:

$$\max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda) := V_0(\pi) + \sum_{k=1}^{K} \lambda_k \left[V_k(\pi) - \xi_k \right]$$

❖ Duality gap:

$$\Delta := \min_{\lambda \in \mathbb{R}_+^K} \max_{\pi} L(\pi, \lambda) - \max_{\pi} \min_{\lambda \in \mathbb{R}_+^K} L(\pi, \lambda)$$

❖ Primal-Dual Algorithm (exact version):

$$\pi_t = \arg\max_{\pi} L(\pi, \lambda_t)$$

$$\lambda_{t+1} = \mathrm{Proj}_{\Lambda}\left[\lambda_t - \beta \nabla_{\lambda} L(\pi_t, \lambda_t) \right]$$

❖ Our convergence results for Cooperative MARL:

- Fact 1: $\Delta > 0$ for some examples. (VS. $\Delta = 0$ for constrained **single-agent** RL)
- Theorem 2: Constrained Violation $= O\left(1/\sqrt{T} + \Delta\right)$

  Optimality gap $= O\left(1/\sqrt{T}\right)$

Outline

- Problem Formulation
- Challenges in Constrained Cooperative MARL
- Existing Primal-Dual Algorithm has Duality Gap>0
- **Our Decentralized Primal Algorithm**
- Numerical Examples: Neither Algorithm Outperforms

Our Decentralized Primal Algorithm

- Our Decentralized Primal Algorithm (natural policy gradient update on $V_{k_t}(\pi)$):

$$\pi_{t+1}^{(m)}(a^{(m)} \mid s) \propto \pi_t^{(m)}(a^{(m)} \mid s) \exp\left[\alpha Q_{k_t}^{(m)}(\pi_t; s, a^{(m)})\right]$$

Our Decentralized Primal Algorithm

- Our Decentralized Primal Algorithm (natural policy gradient update on $V_{k_t}(\pi)$):

$$\pi_{t+1}^{(m)}(a^{(m)} \mid s) \propto \pi_t^{(m)}(a^{(m)} \mid s)\exp\left[\alpha\, Q_{k_t}^{(m)}(\pi_t; s, a^{(m)})\right]$$

(Case 1): Select $k_t = k \in \{1,\ldots,K\}$ if k-th safety constraint $V_k(\pi) \geq \xi_k$ is heavily violated

$$V_k(\pi_t) < \xi_k - \eta$$

Our Decentralized Primal Algorithm

- Our Decentralized Primal Algorithm (natural policy gradient update on $V_{k_t}(\pi)$):

$$\pi_{t+1}^{(m)}(a^{(m)} \mid s) \propto \pi_t^{(m)}(a^{(m)} \mid s)\exp\left[\alpha\, Q_{k_t}^{(m)}(\pi_t; s, a^{(m)})\right]$$

(Case 1): Select $k_t = k \in \{1,\ldots,K\}$ if k-th safety constraint $V_k(\pi) \geq \xi_k$ is heavily violated

$$V_k(\pi_t) < \xi_k - \eta$$

(Case 2): Select $k_t = 0$ (Objective $V_0(\pi)$) if no heavy violation.

Our Decentralized Primal Algorithm

- Local advantage function of m-th agent:

$$A_k^{(m)}(\pi; s, a^{(m)}) := Q_k^{(m)}(\pi; s, a^{(m)}) - V_k(\pi; s)$$

Our Decentralized Primal Algorithm

- Local advantage function of m-th agent:

$$A_k^{(m)}(\pi; s, a^{(m)}) := Q_k^{(m)}(\pi; s, a^{(m)}) - V_k(\pi; s)$$

- Global advantage function:

$$A_k(\pi; s, a) := Q_k(\pi; s, a) - V_k(\pi; s)$$

Our Decentralized Primal Algorithm

- Local advantage function of m-th agent:

$$A_k^{(m)}(\pi; s, a^{(m)}) := Q_k^{(m)}(\pi; s, a^{(m)}) - V_k(\pi; s)$$

- Global advantage function:

$$A_k(\pi; s, a) := Q_k(\pi; s, a) - V_k(\pi; s)$$

- Advantage gaps:

$$\zeta_k := \sup_{s,a,\pi} \left| A_k(\pi; s, a) - \sum_{m=1}^{M} A_k^{(m)}(\pi; s, a^{(m)}) \right|$$

Our Decentralized Primal Algorithm

- Local advantage function of m-th agent:

$$A_k^{(m)}(\pi; s, a^{(m)}) := Q_k^{(m)}(\pi; s, a^{(m)}) - V_k(\pi; s)$$

- Global advantage function:

$$A_k(\pi; s, a) := Q_k(\pi; s, a) - V_k(\pi; s)$$

- Advantage gaps:

$$\zeta_k := \sup_{s,a,\pi} \left| A_k(\pi; s, a) - \sum_{m=1}^{M} A_k^{(m)}(\pi; s, a^{(m)}) \right|$$

- Convergence of our Primal Algorithm (Theorem 3):

$$\text{Constrained Violation} = O\left(1/\sqrt{T} + \zeta_0\right)$$

$$\text{Optimality gap} = O\left(1/\sqrt{T} + \max_{1 \le k \le K} \zeta_k\right)$$

Outline

- Problem Formulation
- Challenges in Constrained Cooperative MARL
- Existing Primal-Dual Algorithm has Duality Gap>0
- Our Decentralized Primal Algorithm
- **Numerical Examples: Neither Algorithm Outperforms**

Numerical Examples: Neither Algorithm Outperforms

	Primal-Dual Algorithm	Our Primal Algorithm
Example 1	Infeasible policy ✗	Converges to optimal policy ✓
Example 2	Get optimal policy in 1 iteration ✓	Infeasible policy ✗
Constant Convergence Error Term	Duality gap	Advantage gap

Thank You