Efficient Reinforcement Learning for Autonomous Racing with Imperfect Demonstrations

Video Link

Heeseong Lee, Sungpyo Sagong, Minhyeong Lee, and Dongjun Lee
Department of Mechanical Engineering, Seoul National University

Motivation

- Autonomous car racing presents challenges in robotics :
  - Handling highly nonlinear dynamics under extreme actions.
  - Rapid change in characteristics of the vehicle's behavior. (e.g. tire wear, fuel consumption, varying track conditions.)
  - Executing strategic maneuvers.

  ⇒ Hard to design a real-time controller w/ traditional approaches.
- Our environment, "Assetto Corsa" is unable to replicate or fast-forward.

  ⇒ Poor sample efficiency due to low sampling speed.
- Require long and precise actions to successfully accomplish a lap.

  ⇒ Agent may be impeded or even unable to find a solution w/ terminal reward structure.

Fig. 1. Our environment, Assetto Corsa, a widely renowned simulator in STEAM for its realistic modeling of car dynamics and high-quality rendering.

Contribution

- Discriminator Augmented Q-function (DAQ) aided RL algorithm is proposed.
  - Integrated with off-policy algorithms to enhance sample efficiency.
  - Capable of utilizing low-quality (sub-optimal) demonstrations and even outperform their performance.
- Applied to car racing task, it achieves state-of-the-art performance.
  - Exhibits the fastest learning speed and the best final performance in sparse reward settings compared to existing LfD methods.

Method

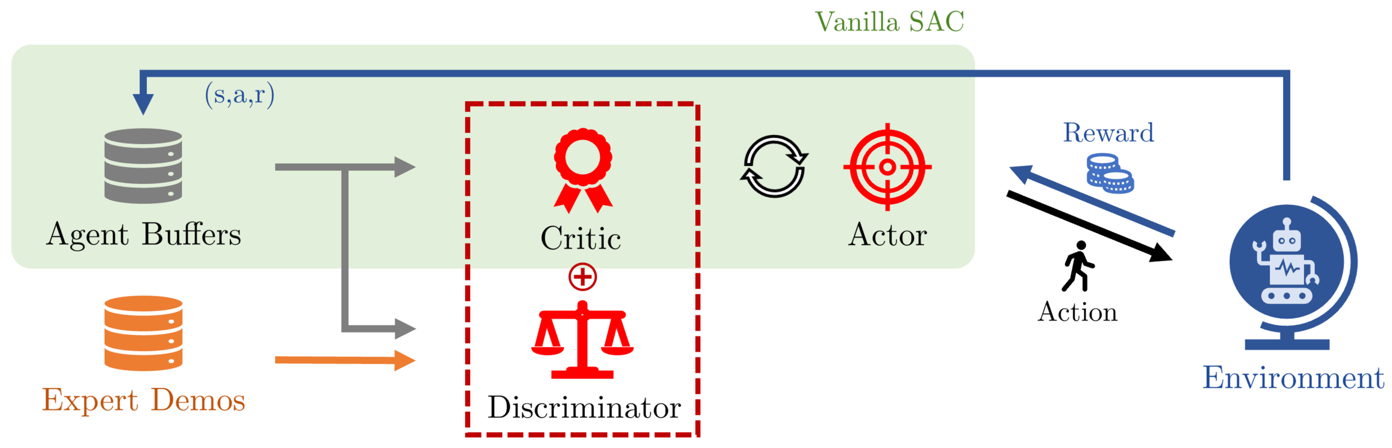

Fig. 2. Overview of the proposed DAQ-SAC algorithm.

- DAQ-SAC : DAQ aided Soft Actor-Critic algorithm.
  - Combine RL and IL using a discriminator ; Learning objective is defined as :

$$\min_{\theta}\max_{w}\mathcal{L}_{\pi_\theta}=\mathbb{E}_{\pi_\theta}\left[\alpha\log\pi_\theta(\tilde{a}_\theta(s)|s)-\min_{i=1,2}Q'_{\phi_i}(s,\tilde{a}_\theta(s))\right]+\lambda_1\mathbb{E}_{\pi_E}\left[\log(D_w(s,a))\right]$$

  - Agent is guided by the augmented Q-function :

$$\underbrace{Q'_{\phi_i}(s,\tilde{a}_\theta(s))}_{\text{Augmented Q-function}}=\underbrace{Q_{\phi_i}(s,\tilde{a}_\theta(s))}_{\text{Original Q-function}}-\lambda_1\underbrace{\log(1-D_w(s,\tilde{a}_\theta(s)))}_{\text{Guidance of discriminator}}$$

  - Additionally, positive-unlabeled reward learning is adopted for the discriminator.

    ⇒ Enable continual improvement of the positive datasets.
  - Then, the final practical algorithm :
    1. Fix actor and critics, update discriminator by gradient ascent step w/

$$\eta\nabla_w\mathbb{E}_{\mathcal{B}}\left[\log(D_w(s,a,\log\pi_\theta(a|s)))\right]+\nabla_w\mathbb{E}_{\mathcal{D}}\left[\log(1-D_w(s,a,\log\pi_\theta(a|s)))\right]$$
$$-\eta\nabla_w\mathbb{E}_{\mathcal{B}}\left[\log(1-D_w(s,a,\log\pi_\theta(a|s)))\right]$$

    2. Fix discriminator and actor, update critics by gradient descent step w/

$$\nabla_{\phi_i}\mathbb{E}_{\pi_\theta}[Q'_{\phi_i}(s,a)-y'(r,s',d)]^2$$

    3. Fix discriminator and critics, update actor by gradient ascent step w/

$$\nabla_\theta\mathbb{E}_{\pi_\theta}[\alpha\log\pi_\theta(\tilde{a}(s)|s)-\min_{i=1,2}Q'_{\phi_i}(s,\tilde{a}(s))]$$

Experiment Setup

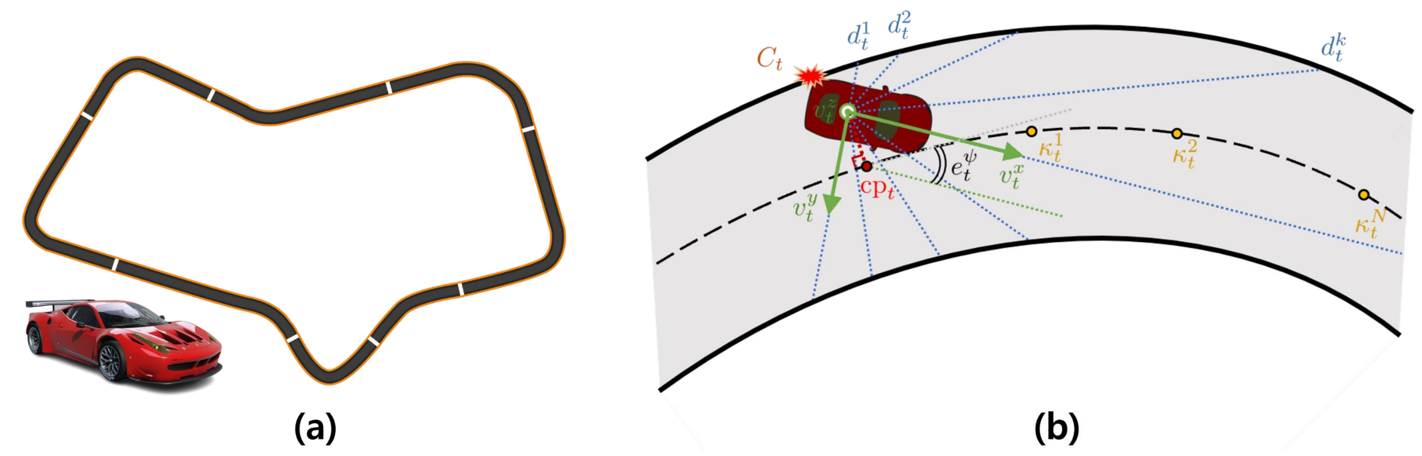

Fig. 3. (a) Car and track used in our experiment. (b) Subset of the observation fed to the networks.

- Ferrari 458 GT2 is selected to drive the Silverstone1967 track.
- MDP settings
  - Observations : $\mathbf{o}_t=[\mathbf{v}_t,\dot{\mathbf{v}}_t,e^{\psi}_t,C_t,\mathbf{d}_t,\boldsymbol{\kappa}_t,\delta_{t-1}]$
  - Actions : $\mathbf{a}_t=\boldsymbol{\mu}_t+\boldsymbol{\sigma}_t\cdot\epsilon\,,\quad\epsilon\sim\mathcal{N}(0,1)$
  - Rewards : $r_t=\sum_{i=1}^{5}\lambda_i r_{t,i}$
    1. Track progress reward : $r_{t,1}=+1$ for every ckpt*
    2. Time penalty : $r_{t,2}=-1$ for each step
    3. Under-pace penalty : $r_{t,3}=-1$ if $|v|<|v|_{\text{thres}}$
    4. Tire-off-track penalty :

$$r_{t,4}=\begin{cases}-10 & \text{if numTyresOffTrack}>2\\ -1 & \text{elseif numTyresOffTrack}>0\end{cases}$$

    5. Collision penalty : $r_{t,5}=-C_t$

$\mathbf{v}_t\in\mathbb{R}^3$: velocity
$\dot{\mathbf{v}}_t\in\mathbb{R}^3$: acceleration
$e^{\psi}_t\in(-\pi,\pi]$: yaw error w.r.t centerline
$C_t\in\{0,1\}$: wall contact flag
$\mathbf{d}_t\in\mathbb{R}^M$: distance of each M rangefinder
$\boldsymbol{\kappa}_t\in\mathbb{R}^N$: N sampled curvature of centerline
$\delta_{t-1}\in[-1,1]$: previous steering command

$\boldsymbol{\mu}_t=[\mu^{\tau}_t,\mu^{\delta}_t]\in\mathbb{R}^2,\quad\boldsymbol{\sigma}_t=[\sigma^{\tau}_t,\sigma^{\delta}_t]\in\mathbb{R}^2$
where, τ : throttle-brake, δ : steering

* ckpt : checkpoint

- Demonstrations are collected using MPC w/ simple kinematic bicycle model.

Results

- Training efficiency comparison
  - DAQ-SAC exhibits the SOTA performance in two aspects.
    1. Learning speed : required training steps until the first lap completion.
    2. Effectiveness : episode return upon the first lap completion.

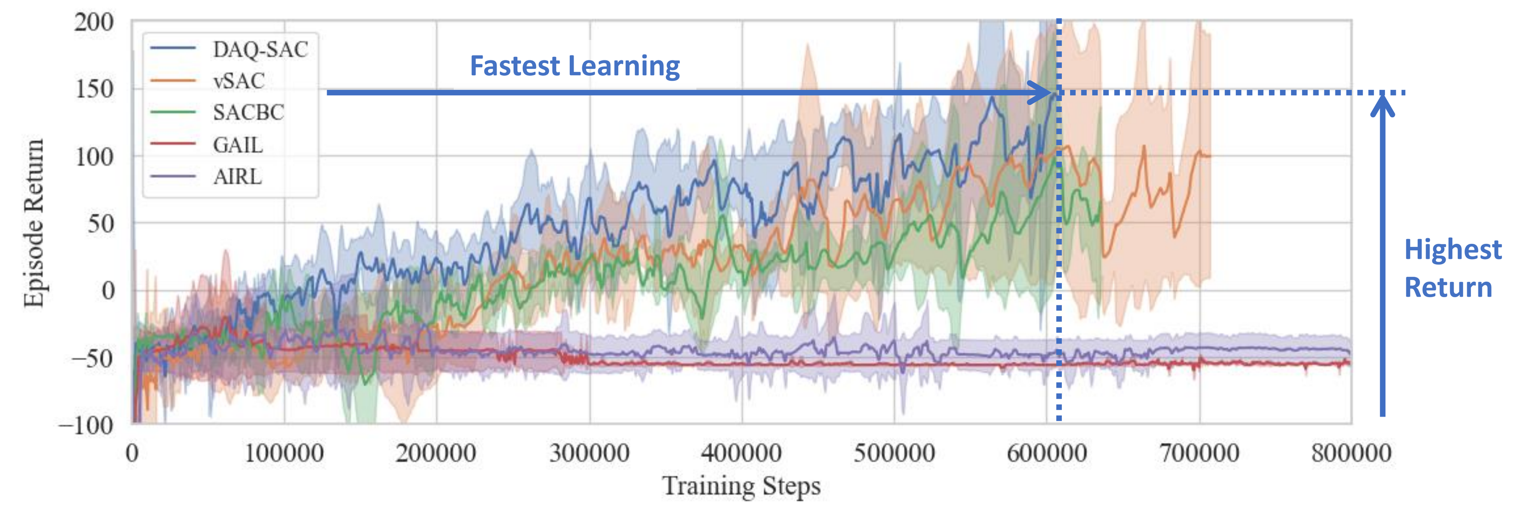

Fig. 4. Experiment results comparing training efficiency. The graph shows the episode return over training steps until the first lap completion.

- Final performance comparison (~ 500,000 training steps)
  - Agent learned w/ DAQ-SAC shows the fastest lap time.

	Demo	DAQ-SAC(Ours)	vSAC	SACBC	GAIL	AIRL
Lap time	1:37:330	**1:29:767**	1:39:624	1:38:539	- (fail)	- (fail)

- Learned driving behaviors
  - Agent learns to effectively use full width of track to minimize the curvature and maximize its speed, i.e. "out-in-out" trajectory.

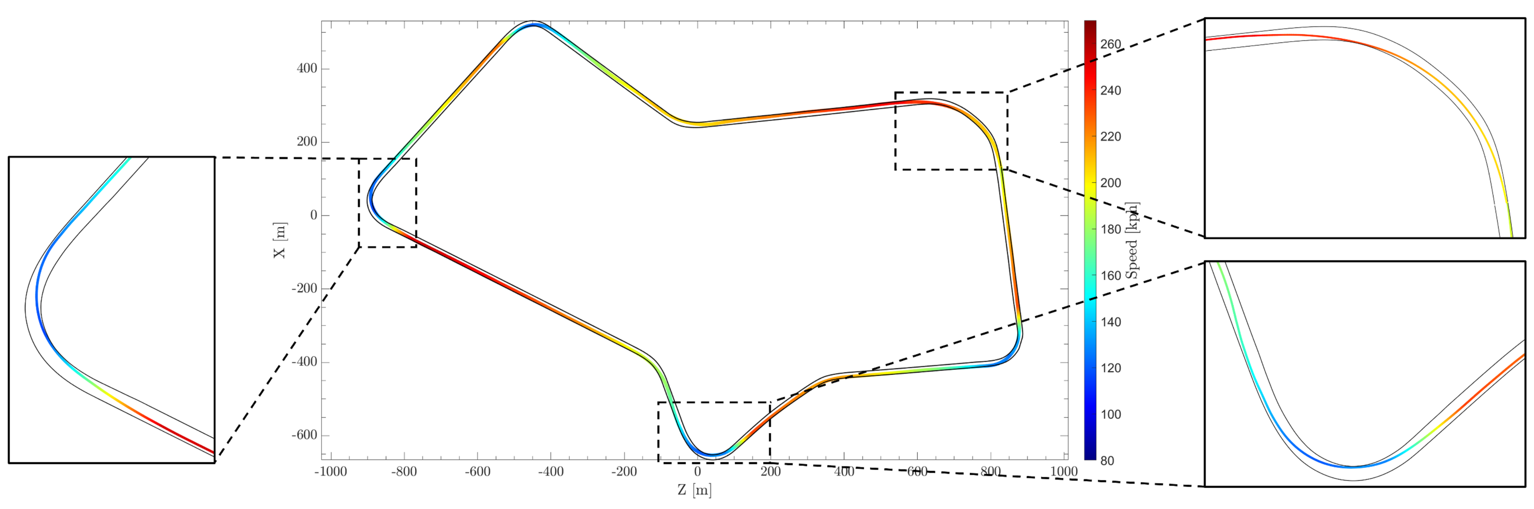

Fig. 5. Speed profile of the DAQ-SAC agent along the track. Three corners with different curvatures are selected to closely visualize the trajectory.

Acknowledgement

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant code RS-2021-KA162182), and the Technology Innovation Program (20024355 and 1415187329, Development of autonomous driving connectivity technology based on sensor-infrastructure cooperation) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea).

IROS '24
ABU DHABI