

# CarMaker 환경에서 최적 성능 달성을 위한 강화학습 기반 운전자 모델 개발

† 한재웅

## Developing a Reinforcement Learning-based Driver Model for Achieving Optimal Performance in the CarMaker Environment

† Jaewoong Han

*Key Words : Reinforcement Learning(강화학습), CarMaker(카메이커), IPG Driver(IPG Driver),*

### ABSTRACT

1930s for the investigation of air density. Today it is used in mobility industries. apparatus like localizer and glide slope. So this paper focus on LiDAR sensor for the aircraft's active autonomous landing.

#### 1. 서 론

차량 연구 분야에서 차량 시뮬레이션의 역할은 점점 중요해지고 있다. 이러한 맥락에서 차량 동역학 시뮬레이션 프로그램인 독일 IPG Automotive사의 CarMaker는 차량의 성능을 정밀하게 고려할 수 있는 강력한 도구로 자리잡고 있다. CarMaker는 차량의 다양한 동적 특성을 정밀하게 묘사할 수 있는 기능을 갖추고 있으며, 이를 통해 자동차 설계 및 테스트에 있어 중요한 역할을 한다. 사용자는 차량, 도로 상태 등의 변수를 설정한다. 그리고 이 변수에 따라 CarMaker 내부의 운전자 모델 IPG 테스트 드라이버를 이용하여 차량 주행 결과를 확인할 수 있다. 또한, CarMaker는 MATLAB/Simulink를 통해 차량을 사용자가 직접적으로 조종할 수 있기 때문에 차량 시뮬레이션 분야에서 적극적으로 활용되고 있다.

그러나, IPG 테스트 드라이버가 제공하는 동역학 모델은 몇 가지 한계를 지니고 있는데, 그중 횡방향 차량 동역학 제어에서 뚜렷한 한계가 드러난다. 이러한 한계는 차량 성능을 평가하는 테스트 과정에서 두드러지게 나타난다. 예를 들어, 실제 차량 시험에서 성공적으로 통과된 횡방향 테스트 시나리오에서 IPG 테스트 드라이버는 실패하는 사례가 발견되었다.<sup>(1)</sup> IPG 테스트 드라이버의 정밀도와 현실적 적용 가능성에 대한 간극을 보여주며, CarMaker 내의 환경을 IPG 테스트 드라이버의 동역학 모델이 적절히 활용하지 못한다는 점을 시사한다. 따라서, 현실 차량 테스트 환경을 정밀하게 모

사하기 위해서는 IPG 테스트 드라이버 모델의 한계를 극복할 수 있는 새로운 접근 방식이 필요함을 암시한다.

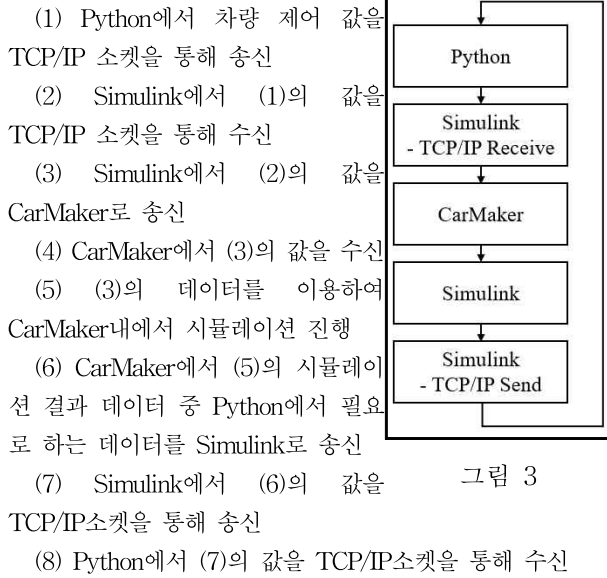
본 연구는 이러한 문제를 해결하기 위해 강화학습 기반의 새로운 운전자 모델을 개발하고자 한다. 강화학습은 머신러닝의 한 분야로, 에이전트가 환경과 상호작용하면서 보상을 최대화하는 방향으로 행동을 학습하는 과정이다. 강화학습을 이용하면 실제 운전자의 반응과 유사한 방식으로 차량을 제어할 수 있는 모델을 개발할 수 있으며, 이를 통해 더 정밀하고 현실적인 시뮬레이션 결과를 얻을 수 있을 것으로 기대된다.

본 연구의 목적은 CarMaker 환경에서 적용 가능한 강화학습 기반 운전자 모델을 개발하고, 이를 기존의 IPG 테스트 드라이버 모델과 비교 평가하여 성능을 뛰어넘는 결과를 도출하는 것이다. 특히, 횡방향 차량 동역학 제어 성능 비교를 위해 차량 핸들링 시나리오를 통해 비교할 예정이다. 또한, Rear Wheel Steering (RWS) 시스템을 포함하는 다양한 차량 핸들링 시나리오에 대한 적용 가능성을 탐구할 예정이다. RWS는 후륜 조향 시스템으로, 차량의 주행 상황에 따라 능동적으로 후륜 조향각을 제어하는 기술이다. 이를 통해 다양한 task에서도 적용 가능하며 IPG 테스트 드라이버의 횡방향 차량 동역학 제어 성능을 뛰어넘는 강화학습 기반 운전자 모델을 개발하고자 한다.

## 2. 연구 방법

### 2.1 CarMaker 차량 직접 제어

본격적으로 강화학습 모델을 제작하기 전, CarMaker의 차량 동역학 모델을 직접적으로 제어함과 동시에 시뮬레이션 결과를 받아올 수 있도록 설계해야 한다. 이를 위해 MATLAB의 Simulink를 이용하였다. 또한, Simulink의 송수신 결과를 TCP/IP 소켓 통신을 이용하여 데이터를 python에서 직접적으로 관리할 수 있도록 제작하였다. 정리하자면, 다음과 같다.



이때, CarMaker와 Simulink의 데이터는 CarMaker 자체에서 제공하는 CarMaker4SL 라이브러리를 이용하여 데이터를 주고받는다.

### 2.2 강화학습 모델 선정

상태(state)에 해당하는 시뮬레이션 데이터, 행동(action)에 해당하는 차량 제어 값은 연속적인 값을 갖기 때문에 연속적인 환경에 사용이 가능한 강화학습 모델 중 Soft Actor-Critic (SAC) 모델을 선택하였다. SAC 모델은 Actor, Critic 두 가지의 네트워크를 사용하여 학습을 진행하며, 로봇 공학과 같은 복잡한 환경에서 효과적인 학습이 가능한 모델로 밝혀져 있다. 그 이유는 다음과 같다.

#### (1) Actor 네트워크

Actor 네트워크의 주된 목적은 주어진 상태에 대해 어떤 행동을 취할지 결정하는 것이다. 이 네트워크는 현재 환경의 상태를 입력으로 받아, 이에 대응하는 행동을 출력한다. Actor 네트워크의 목적함수는 다음과 같다:

$$J(\pi) = \sum_{t=0}^T \mathbb{E}_{(s_t, a_t) \sim \rho_{\pi}} \left[ r(s_t, a_t) + \alpha \mathcal{H}(\pi(\cdot | S_t)) \right]$$

Actor 네트워크의 목적 함수는 정책(policy)의 성능을 최대화하는 것이다. 이때, SAC는 엔트로피 보상을 포함한 목적 함수를 사용한다. 각 항은 다음과 같은 의미를 지닌다.

(a)  $\mathbb{E}_{(s_t, a_t) \sim \rho_{\pi}}$ : 기대값 연산자로, 정책  $\pi$ 에 따라 분포  $\rho_{\pi}$ 에서 샘플링된 상태-행동 쌍에 대한 기대값을 의미한다.

(b)  $r(s_t, a_t)$ : 시간스텝  $t$ 에서 상태  $s_t$ 와 행동  $a_t$ 를 취했을 때 받는 즉각적인 보상을 의미한다.

(c)  $\mathcal{H}(\pi(\cdot | S_t))$ : 엔트로피 항으로, 정책의 불확실성을 나타낸다. 이를 통해 알고리즘이 탐험을 장려하도록 한다. 높은 엔트로피는 더 많은 탐험을 의미하며, 낮은 엔트로피는 더 결정적인 행동을 의미한다.

(d)  $\alpha$ : 엔트로피 가중치로, 정책의 확률적 특성을 얼마나 중요시할지 결정한다.

Actor 네트워크의 목적은 최대의 목적함수를 갖는 것이다. 따라서, 시간에 걸쳐 기대되는 반환값을 최대화하는 동시에, 정책의 엔트로피를 증가시켜 탐험을 장려한다. 특히, 이러한 엔트로피의 특성 때문에 복잡한 환경에서 자주 사용되는 강화학습 모델이다.

#### (2) Critic 네트워크

Critic 네트워크의 주요 목적은 특정 정책에 따른 행동의 가치를 평가하는 것이다. 즉, 특정 상태에서 특정 행동을 취할 때 기대할 수 있는 반환값(return)을 예측한다.

$$J_Q(\theta) = \mathbb{E}_{(s_t, a_t) \sim \mathcal{D}} \left[ \frac{1}{2} \left( Q_{\theta}(s_t, a_t) - \hat{Q}(s_t, a_t) \right)^2 \right]$$

Critic 네트워크의 목적함수는 정책에 따른 행동의 가치를 정확하게 추정하는 것이다. 각 항은 다음과 같은 의미를 지닌다.

(a)  $Q_{\theta}(s_t, a_t)$ : 파라미터  $\theta$ 를 가진 critic 네트워크가 추정하는 상태  $s_t$ 에서 행동  $a_t$ 를 취했을 때의 가치이다.

(b)  $\hat{Q}(s_t, a_t)$ : 타겟 가치로, 일반적으로 실제 보상과 다음 상태에서의 가치 추정을 바탕으로 계산된다.

(c)  $\mathbb{E}_{(s_t, a_t) \sim \mathcal{D}}$ : 경험 리플레이 버퍼  $\mathcal{D}$ 에서 생성된 샘플링 데이터에 대한 기대값이다.

Critic 네트워크의 목적은 추정된 Q-값과 타겟 Q-값 사이의 평균 제곱 오차를 최소화하여, 실제 반환값을 정확하게 예측하는 것이다. 이를 통해 Actor 네트워크가 보다 정확한 기대 반환값에 따라 행동을 선택하도록 한다. 특히, 경험 리플레이 버퍼를 사용하는 off-policy 기법을 이용하여 이전의 정책에서 얻은 경험도 학습에 재사용 함으로써 학습의 효율을 높인다.

## 2.2 강화학습 환경 제작

### 2.2.1 코드별 구조

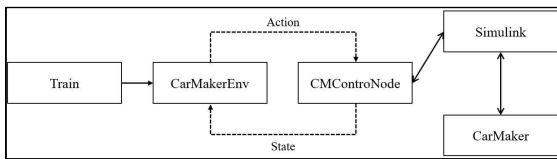
2.1 CarMaker 차량 직접 제어 결과를 토대로 강화학습 환경을 제작하였다. 코드는 크게 세 가지로 구성된다.

(1) Train: stable-baselines3 모듈의 SAC 모델을 이용하여 학습을 진행하는 코드이다.

(2) CarMakerEnv: gymnasium의 Env를 이용하여 제작된 custom 환경이다. CMControlNode를 통해 데이터를 주고받는다.

(3) CMControlNode: 2.1 CarMaker 차량제어에서 구현한 데이터 송수신을 담당하는 코드이다.

각 코드의 작동 방식은 그림 #과 같다.



Train 코드를 통해 SAC로 구성된 모델이 학습을 시작하면, CarMakerEnv 환경에서 선택된 행동 값을 CMControlNode 내부의 TCP/IP 소켓을 통해 Simulink로 송신한다. 이 값은 CarMaker로 전송되며, 시뮬레이션이 진행된다. 시뮬레이션 결과 중 강화학습 환경의 상태에 필요한 값을 Simulink가 수신하여 TCP/IP 소켓을 통해 CMControlNode로 송신한다. 이 값을 CarMakerEnv 환경에서 보상을 계산하여 학습이 진행된다.

### 2.2.2 강화학습 환경 설명

강화학습 환경을 담당하는 CarMakerEnv의 상태, 행동, 보상은 아래와 같다:

(1) 행동: 횡방향 차량 동역학 제어를 위해 차량의 조향각 변화량(difference in steering angle)을 제어하며, -0.15에서 +0.15의 값을 가질 수 있다.

(2) 상태: 차량에 대한 정보와 전방 경로(lookahead trajectory)를 상태정보로 활용한다. 각 항에 대한 설명은 아래와 같다:

(a) 차량 정보: 속도, 요(yaw), 조향각 및 조향속도, 각 바퀴별 조향각

(b) 전방 경로: 전방 2 m, 4 m, 6 m, 8 m, 10 m, 12 m의 경로 위치의 상대좌표 값

추가로, 2.3에서 구현한 RWS 차량의 경우 뒷바퀴 조향각에 대한 추가적인 정보를 상태 값으로 제공하였다.

(3) 보상: 차량의 중심과 경로간 떨어진 거리 및 각도의 합을 보상으로 제공하였다.

## 2.3 RWS 시스템 적용 차량 구현

RWS 시스템을 구현하여 CarMaker 차량의 후륜 조향각을 제어하기 위해 MATLAB을 이용하였다.

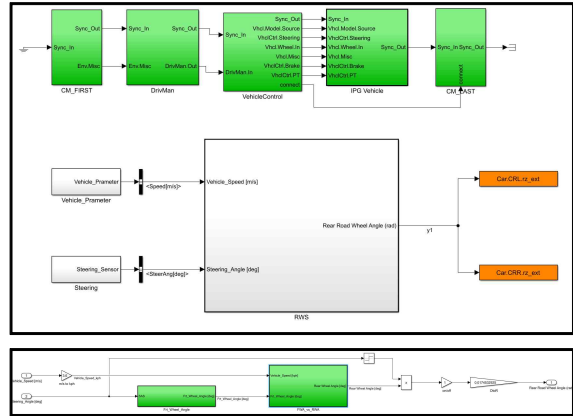


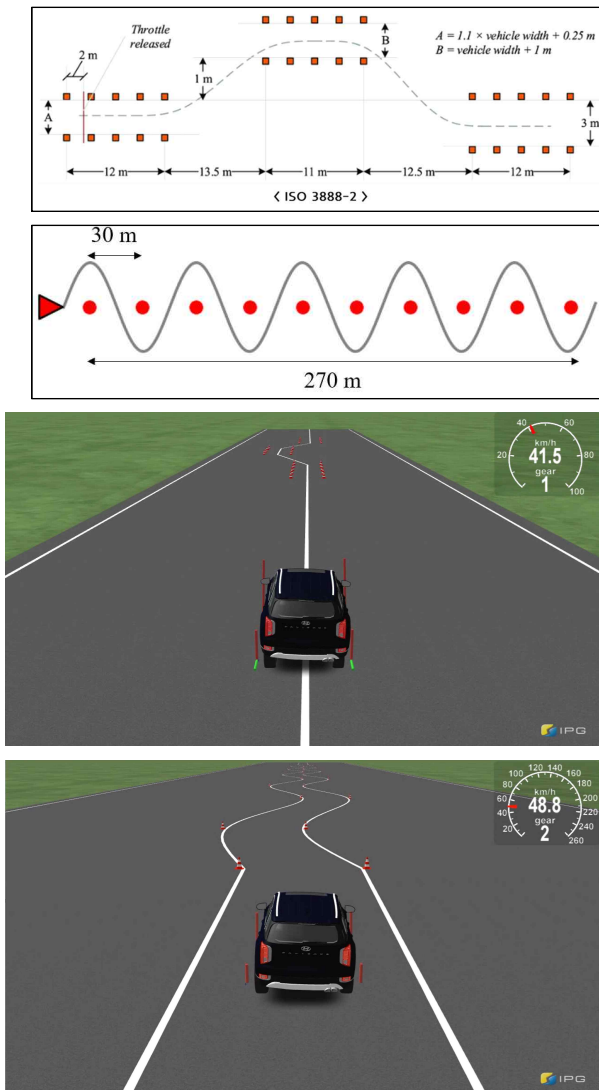
그림 #은 그림 #은 중앙 RWS 박스 부분의 내부 모습이다. 차량의 속도를 고려하여 뒷바퀴 조향각을 제어하도록 설계되었다.

## 2.4 테스트 시나리오 선정 및 제작

핸들링 시험에 대한 차량의 적합성을 확인하기 위하여 30 M Slalom과 ISO 3888-2(Double Lane Chagne)의 규격에 맞춰 그림# 및 그림 #과 같이 코스를 구성하였으며, 표#과 그림 # 및 그림 #과 같이 시뮬레이션을 수행하였다<sup>(1)</sup>. 추가로, ISO 3888-2의 경우 IPG 테스트 드라이버와 RWS 시스템 적용 차량 및 RWS 시스템 미적용 차량 세 가지에 대해 비교하였다.

Slalom	세팅	
차량	Hyundai Ionic	
초기 속도	50 kph	
Manuever	Longitudinal Dynamics	Lateral Dynamics
0 m ~	Speed Control, 50 kph	Not Specified

ISO 3888-2	세팅	
차량	Hyundai Palisade	
초기 속도	50 kph	
Maneuer	Longitudinal Dynamics	Lateral Dynamics
0 m ~ 52 m	Speed Control, 50 kph	Not Specified
52 m ~ 111 m	Manual, G/B/C/P 0, 0.2	
111 m ~	Speed Control, 50 kph	



두 테스트 시나리오에 대한 공통적인 목표는 IPG 테스트 드라이버 모델과 강화학습 모델의 경로 생성을 비교하는 것이다. 각 시나리오의 세부 목표는 아래와 같다:

- (1) Slalom: 콘과의 충돌이 일어나지 않으면서 차량의 움직임을 최소화하는, 즉, 횡방향 움직임을 최소화하는 것을 목표로 한다.
- (2) ISO 3888-2: 콘과의 충돌 수를 최소화하는 것을 목표로 한다.

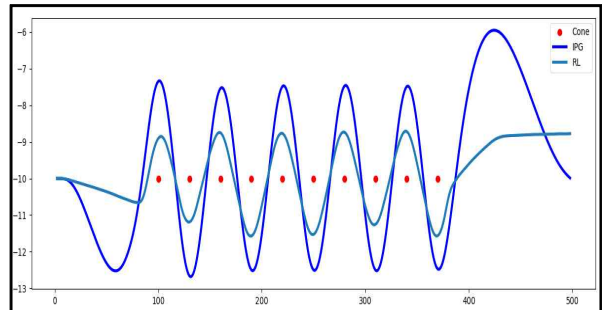
### 3. 연구 결과

각 테스트 시나리오별 300 만회의 학습을 진행하였다. 그 결과는 다음과 같다.

#### 3.1 SLALOM 테스트 시나리오 결과 및 결과 분석

SLALOM 테스트 시나리오에서의 차량 경로를 IPG 테스트

트 드라이버 모델과 강화학습 모델을 비교한 결과는 다음과 같다.



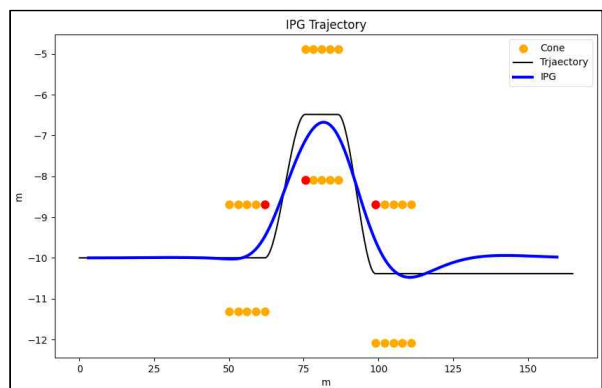
IPG 테스트 드라이버 운전자 모델은 상대적으로 불필요한 횡방향 움직임이 나타나는 것으로 관찰되었다. 반면, 강화학습 운전자 모델은 횡방향 움직임이 상당히 줄어든 것을 확인할 수 있다. 이러한 차이는 y축 이동량 비교에서 더 명확히 드러난다.

y축 이동량 (m)	
IPG 운전자 모델	64
강화학습 운전자 모델	29
비교 (%)	54

IPG 운전자 모델은 64 m의 y축 이동량을 보인 반면, 강화학습 운전자 모델은 29 m의 y축 이동량을 보였다. 이는 강화학습 운전자 모델이 IPG 모델에 비해 약 54 %더 효율적인 횡방향 제어를 달성했음을 의미한다.

#### 3.2 ISO 3888-2 (Double Lane Change) 테스트 시나리오 결과 및 결과 분석

IPG 3888-2 테스트 시나리오에서의 차량 경로를 IPG 테스트 드라이버 모델과 강화학습 모델을 비교한 결과는 다음과 같다.



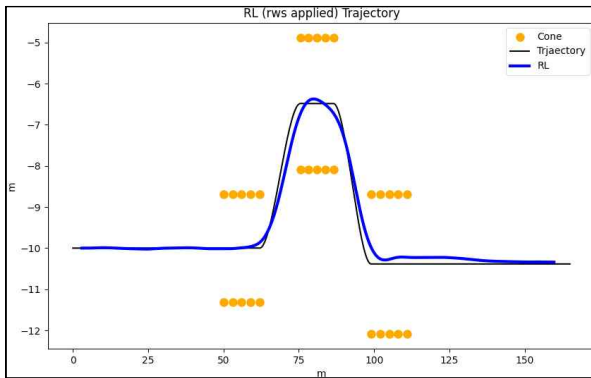
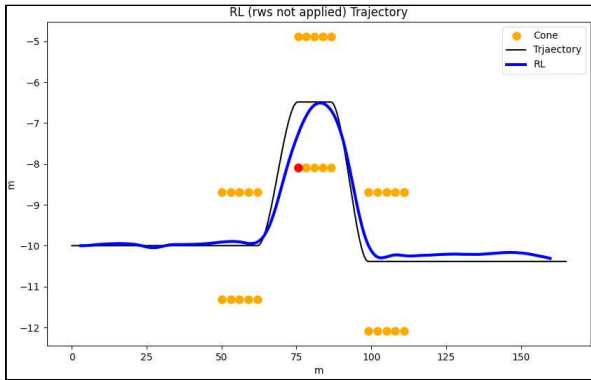


그림 #, #, #은 각 차량의 움직임에 대한 경로이며, 빨간색 원은 해당 위치의 콘과 충돌했음을 의미한다. 결과는 표#와 같다.

운전자 모델	충돌 횟수	충돌지점
IPG	3	(62, -8.685), (75.5, -8.085), (99, -8.685)
강화학습 - RWS 미적용	1	(75.5, -8.085)
강화학습 - RWS 적용	0	-

IPG 테스트 드라이버 운전자 모델의 경우 세 개의 콘과 충돌하였으며, RWS가 미적용된 강화학습 운전자 모델의 경우 한 개의 콘과 충돌한 반면, RWS가 적용된 강화학습 운전자 모델의 경우 충돌하지 않았다.

또한, 이동 경로에서도 세 개의 운전자 모델은 차이를 보인다. 먼저, IPG 테스트 드라이버 운전자 모델의 경우 Lane Change 구간인 62 m ~ 75.5 m, 86.5 m ~ 99 m 지점에서 급격한 조향각 제어에 대응하지 못하고 충돌하는 모습을 보인다. 그러나, 강화학습 운전자 모델의 경우 이에 대응하는 모습을 보이며, RWS가 적용된 차량의 경우 완전히 대응하는 것을 확인할 수 있다.

#### 4. 토의

#### 5. 결론

#### 참고문헌

- (1) Kim, H., et al. (2020). A Study on the Development and Correlation of Full Vehicle Model for Virtual Vehicle Dynamics Simulation Using CarMaker. Korea Automotive Technology Institute.
- (2) Cheok Jun Hong, Vimal Rau Apraow (2021), System configuration of human-in-loop simulation for Level 3 autonomous vehicle using IPG CarMaker, IEEE International Conference on Internet of Things and Intelligence Systems, pp.215-221
- (3) Tuomas Haarnoja., et al, Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor, arXiv:1801.01290v2, 2018.
- (4) Sampo Kuutti., et al (2019), End-to-end reinforcement learning for autonomous longitudinal control using advantage actor critic with temporal context, IEEE Intelligent Transportation Systems Conference, pp.2456-2462
- (5) Cheok Jun Hong, Vimal Rau Apraow, System configuration of human-in-loop simulation for Level 3 autonomous vehicle using IPG CarMaker, IEEE International Conference on Internet of Things and Intelligence Systems, 2021, pp.215-221
- (6) Sampo Kuutti., et al, End-to-end reinforcement learning for autonomous longitudinal control using advantage actor critic with temporal context, IEEE Intelligent Transportation Systems Conference, 2019, pp.2456-2462
- (7) H.Yong., et al, "Suspension control strategies using switched soft actor-critic models for real roads," in IEEE Transactions on Industrial Electronics., 2023, pp.824-832
- (8) Inuzuka, S., Zhang, B., & Shen, T. (2021), Real-time HEV energy management strategy considering road congestion based on deep reinforcement learning. energies, 2021,14.