모델 예측 제어를 위한 시간차 학습 알고리즘에서 행동 선택 방법의 영향 분석

지창훈, 석영준, 임현교, 한연희+
한국기술교육대학교 미래융합공학전공
{koir5660, dsb04163, glenn89, yhhan}@koreatech.ac.kr

Analysis of the Impact of Action Selection Methods in Time Difference Learning for Model Predictive Control Algorithms

Chang-Hun Ji, Yeong-Jun Seok, Hyun-Kyo Lim, Youn-Hee Han+
Future Convergence Engineering, Korea University of Technology and Education

요 약

Model-based 강화 학습은 환경의 역학 모델을 통한 계획(planning)과 데이터 생성을 통해 높은 샘플 효율성을 가질 수 있다. 하지만 환경의 역학 모델의 정확도에 학습 성능이 지나치게 의존하고 계획 기간(planning horizon)이 길어지면 과도하게 연산량이 많아지기 때문에 연속적인 제어 작업에서 model-free 강화 학습에 비해 model-based 강화 학습은 성능 향상에 많은 어려움을 겪는다. 하지만 최근 Temporal Difference Learning for Model Predictive Control(TD-MPC) 알고리즘은 계획을 통해 행동을 결정하며, model-based 강화 학습의 약점을 극복하여 연속적인 제어 작업에서도 model-free 강화 학습보다 높은 성능을 보여주었다. 본 논문에서는 기존 TD-MPC가 훈련시키는 정책 모델을 통해서도 행동을 결정할 수 있는 방법을 제안하여 추가적인 네트워크의 변화 없이 Deep Mind Control 환경에서 기존의 TD-MPC보다 높은 성능을 나타냄을 보여준다.

Ⅰ. 서 론

강화 학습은 학습 중 환경의 역학 모델 이용 여부를 기준으로 model-free 강화 학습과 model-based 강화 학습으로 분류 할 수 있다. Model-based 강화 학습은 환경 모델을 이용하여 데이터 생성이 가능하며, 계획(planning) 또한 가능하기 때문에 model-free 강화 학습에 비해 샘플 효율성(sample efficiency)이 높다 [1]. 하지만 계획에 의해 학습되는 에이전트는 환경의 역학 모델의 영향을 많이 받기 때문에, 주어진 환경의 역학 모델이 정확하지 않은 경우 에이전트 학습에 악영향을 미칠 수 있다. 결과적으로 연속적인 제어 작업에서 model-free 강화 학습에 비해 model-based 강화 학습은 성능 향상에 많은 어려움을 겪는다.

Temporal Difference Learning for Model Predictive Control(TD-MPC) 알고리즘은 Model Predictive Control(MPC)와 Task-Oriented Latent Dynamics(TOLD) 모델 그리고 가치 함수의 시간차 학습을 통해 학습하는 방법으로 앞서 언급한 단점을 보완하였다 [2]. 하지만 TD-MPC는 MPC 알고리즘을 통해 계획에 활용하지만 에이전트의 정책 업데이트에 활용되는 데이터는 에이전트의 정책에 기반하지 않는다. 강화학습 에이전트는 MPC와 학습된 환경 모델에 기반한 데이터에 의해 학습이 이루어진다. 즉, 직접적으로 정책을 통해 데이터를 생성하지 않으며 학습되는 정책을 기반한 업데이트는 이루어지지 않는다. 따라서 본 논문에서는 MPC와 환경 모델에 기반하여 학습되는 강화학습 에이전트 업데이트에 정책을 직접적으로 활용하는 새로운 방법을 제안한다. 연속적인 제어 작업의 대표적인 benchmark인 DeepMind Control (DMControl)에서 baseline인 TD-MPC에 제안하는 방법을 적용한 알고리즘의 실험 평가를 진행한다 [3]. 비교 실험에서 제안하는 알고리즘은 baseline에 비해 높은 성능 향상을 보이며 연속적인 제어 작업에 더 효과적인 것으로 평가한다.

Ⅱ. 본 론

Data driven model predictive control은 model-free 알고리즘에 비해 샘플 효율성이 좋고, 계획을 통해 좋은 성능을 가질 수 있다는 장점이 있다. 하지만 계획 기간(planning horizon)을 길게 설정하면 연산량이 과도하게 많아지고, 환경의 역학 모델이 정확하게 학습되지 않는 경우 성능 향상이 안 되는 단점이 있다 [2]. Temporal Difference Learning for Model Predictive Control(TD-MPC) 알고리즘은 Task-Oriented Latent Dynamics(TOLD) 모델을 통해 짧은 기간의 계획은 환경의 역학 모델로부터 수행하고, 긴 기간의 계획은 가치 함수 모델로부터 bootstrapping 하여 과도한 연산량의 증가 없이 긴 계획 기간을 가지는 계획을 수행할 수 있게 하였다. TOLD 모델은 5개의 구성요소가 있다. 구성요소는 아래와 같다.

$$
\begin{aligned}
&\textit{Representation}: z_t = h_\theta(s_t)\\
&\textit{Latent dynamics}: z_{t+1} = d_\theta(z_t, a_t)\\
&\textit{Reward}: \hat{r_t} = R_\theta(z_t, a_t)\\
&\textit{Value}: \hat{q_t} = Q_\theta(z_t, a_t)\\
&\textit{Policy}: \hat{a_t} \sim \pi_\theta(z_t)
\end{aligned}
$$

s_t는 현재 스텝에서 state를 말하고, a_t는 현재 스텝에서 수행한 행동을 뜻한다. 학습 전반에 걸쳐 TD-MPC 에이전트는 환경과의 상호작용을 통해 수집된 데이터를 활용하여 TOLD 모델을 학습시키고, 학습된 TOLD

+ 한연희(Youn-Hee Han, yhhan@koreatech.ac.kr): 교신저자

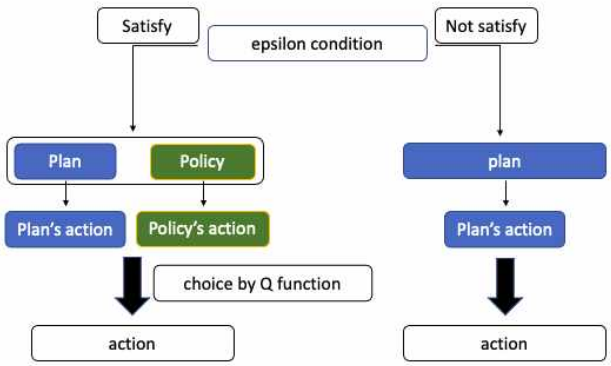
모델을 활용하여 환경에 적용될 행동을 계획을 통해 결정한다. 위의 일련의 과정을 반복하며 TD-MPC 에이전트는 학습하게 된다. 이 때 TOLD 모델의 손실 함수는 다음과 같다.

$$L(\theta;\Gamma_i) = c_1 \| R_\theta(z_i,a_i) - r_i \|_2^2 + c_2 \| Q_\theta(z_i,a_i) - (r_i + \gamma Q_{\bar{\theta}}(z_{i+1},\pi_\theta(z_{i+1}))) \|_2^2 + c_3 \| d_\theta(z_i,a_i) - h_{\bar{\theta}}(s_{i+1}) \|_2^2$$

Γ_i는 리플레이 버퍼에서 가져온 i번째 스텝부터 i+계획 기간 까지의 트레젝토리이고, $\bar{\theta}$가 파라미터로 있는 모델들은 타겟 모델들을 의미한다. 이 식에서 2번째 항을 만들 때 정책 모델이 없으면 $Q_{\bar{\theta}}(z_{i+1},\pi_\theta(z_{i+1}))$ 식을 $\max_{a_t} Q_{\bar{\theta}}(z_t,a_t)$ 식으로 대체하여야 한다. 이 때 행동은 계획에 의해서만 결정된다. 하지만 이 방법은 연산량이 매우 높아서 실제로 사용하기 어렵다 [4]. 따라서 정책 모델을 다음 목적 함수식으로 학습시켜 TOLD 모델을 업데이트할 때 사용한다. 이 식에서 sg는 the stop-grad operator를 뜻한다.

$$J_\pi(\theta;\Gamma) = -\sum_{i=t}^{t+H} \lambda^{i-t} Q_\theta(z_i,\pi_\theta(sg(z_i)))$$

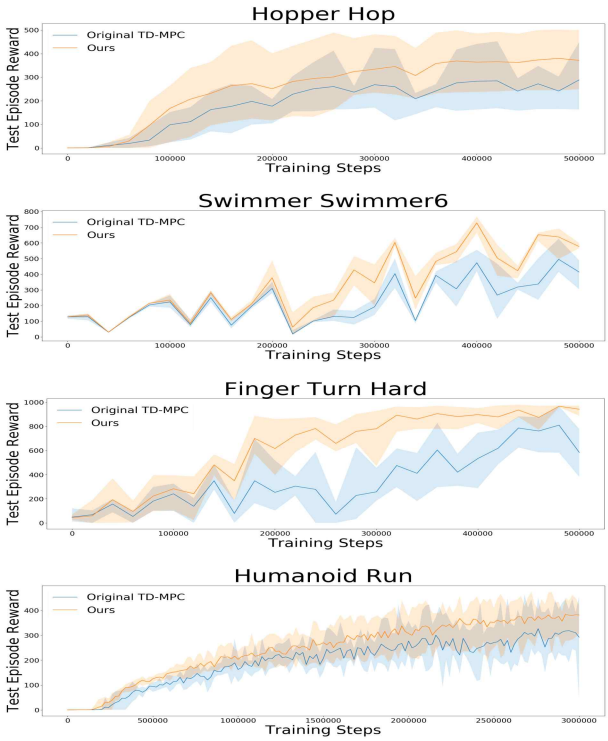
TD-MPC 알고리즘은 TOLD 모델 내에서 정책 모델을 명시적으로 학습을 시키고 있음에도, 환경에 적용되는 행동은 정책 모델을 통해 생성하지 않는다. 본 논문에서 제안하는 방법은 계획에서 나온 행동과 정책 모델에서 나온 행동 중 가치 함수의 결과값이 높은 행동을 환경에 적용시키는 방법이다. 하지만 가치 함수의 결과값에 따라 정책 모델의 행동을 결정하는 비율이 높게 되면 앞서 언급한 MPC의 장점이 사라지기 때문에 Deep-Q Network(DQN)의 epsilon 기법을 사용하여 계획의 행동과 정책의 행동을 균등하게 결정하고자 하였다 [5].



[그림 1]. 제안하는 TD-MPC 행동 결정 개요도

Ⅲ. 실험 및 실험 평가

본 논문에서는 연속적인 제어 작업의 대표적인 benchmark인 DeepMind Control (DMControl)에서 실험을 진행하였다 [3]. DMControl은 연속적인 행동 공간을 가지며, 로보틱스 제어의 다양한 환경을 제공한다. 실험은 DMControl의 hopper-hop, swimmer-swimmer6, finger-turn-hard, humanoid-run 환경에서 진행하였고, 환경 20000 스텝 진행할 때마다 test를 진행한 결과 그래프이다. Test는 각 환경마다 5번씩 진행하였다. 실험 결과 기존 TD-MPC 보다 높은 성능을 가지고 있음을 알 수 있다.



[그림 2]. Training Step에 따른 Test Episode Reward 실험 결과

Ⅳ. 결론

본 논문에서는 최근 높은 성능을 가지고 있는 TD-MPC 알고리즘을 소개하고 새로운 TD-MPC의 행동 결정 방법을 소개하였다. 새로운 TD-MPC 행동 결정 방법은 계획과 정책 모델이 결정한 행동을 가치 함수로 판단하여 더 좋은 행동을 환경에 적용시키는 방법이다. 또한 DQN 알고리즘의 epsilon 기법을 통해 계획의 행동과 정책 모델의 행동을 균등하게 사용하고자 하였다. 그리고 DMControl 환경에서 실험으로 제안한 방법의 효능을 검증하였다.

ACKNOWLEDGMENT

이 논문은 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행 된 기초연구사업임 (No. 2018R1A6A1A03025526 & No. 2020R1I1A3065610).

참 고 문 헌

[1] Polydoros, Athanasios S., and Lazaros Nalpantidis. "Survey of model-based reinforcement learning: Applications on robotics." Journal of Intelligent & Robotic Systems 86.2 (2017): 153-173.

[2] Hansen, Nicklas, Xiaolong Wang, and Hao Su. "Temporal Difference Learning for Model Predictive Control." arXiv preprint arXiv:2203.04955 (2022).

[3] Tunyasuvunakool, Saran, et al. "dm_control: Software and tasks for continuous control." Software Impacts 6 (2020): 100022.

[4] Lowrey, Kendall, et al. "Plan online, learn offline: Efficient learning and exploration via model-based control." arXiv preprint arXiv:1811.01848 (2018).

[5] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." nature 518.7540 (2015): 529-533.