Model Predictive Control : A Reinforcement Learning-based Approach

Model Predictive Control : A Reinforcement Learning-based Approach

Xia Pan[1,2*], Xiaowei Chen[1,2], Qingyu Zhang[1,2], Nannan Li[3]

[1]China Automotive Technology & Research Center, Tianjin, 300300, China

[2]Automotive Data of China (Tianjin) Co., Ltd. , Tianjin, 300300, China

[3]Beijing Institute of Control and Electronics Technology, Beijing, 100045, China

[*]Email: panxia@catarc.ac.cn

Abstract. This article proposes a method of model predictive control, which combine the excellent data-driven optimization ability of reinforcement learning and model predictive control to design the controller. Different from the off-line design of MPC, reinforcement learning is based on the adaptation of on-line data to achieve the purpose of control strategy optimization. The reinforcement learning-based model predictive control can improve the control performance effectively. And the numerical simulations are given to demonstrate the effectiveness of the proposed approach.

1. Introduction

The unmanned systems is mainly divided into three layers: perception layer, decision layer and control layer. In control layer, the performance of the controller plays a very important role in driving safety. Usually, the control system will be constrained. If the system is not in the constraints, it may seriously affect the normal operation. Model predictive control (MPC) can clearly deal with system constraints and provide the optimal solution. It has achieved great success in recent decades and has been widely used in different fields[1-3].

As we all know, the early stability results of MPC adopt the zero state terminal equality constraint. Then, by taking the control Lyapunov function as the terminal cost[4], it is extended to the use of terminal inequality constraints, so as to establish the well-known stability framework. According to the framework, a commonly used MPC design paradigm involves finding appropriate terminal costs and controller. However, due to the large amount of calculation, the mature MPC method in theory is difficult to apply in practice. On the contrary, MPC without terminal cost and terminal constraints is widely welcomed. Its closed-loop stability is generally not guaranteed, but it is only applicable to a sufficiently large prediction range. Moreover, the optimal performance requires the correct objective function, and the design paradigm can not provide any guidance to solve this problem in the case of nonlinearity, which makes it difficult to achieve satisfactory performance. The cost function in the above offline MPC design remains basically unchanged during system operation, which makes it more difficult to maintain optimization. Based on the above analysis, we aim to develop a learning based MPC scheme to learn the correct objective function and eliminate the complex off-line design process.

Recently, reinforcement learning has gradually played its advantages in the field of control. Different from the off-line design of MPC, reinforcement learning (RL) is based on the adaptation of on-line data to achieve the purpose of control strategy optimization. The core idea of RL is temporal difference (TD) learning[5], which estimates the value function directly from raw experience in a bootstrapping way, without waiting for a final outcome. By using the excellent data-driven optimization ability of RL,combine MPC to design the controller, it is expected to improve the control performance, which urges people to try the combination of the two. However, the current research progress in this field is still shallow. For example, in [6], a novel "online planning, offline learning" framework is proposed by using MPC as the trajectory optimizer of RL. The results show that MPC can achieve more effective approximation, so as to accelerate the convergence speed and reduce the approximation error in the value function. In [7], the problem of stability is mainly emphasized, and the economic MPC is used as the approximate function in RL. In [8-9], Learning-based model predictive control towards safe and its practical application are proposed.

In this article, we propose Reinforcement learning based model predictive control. The rest of the paper is orgnized as follows. In section 2, some preliminaries and problem formulation are given. Next, we presented the optimization problem and the value function approximation. In section 4, some results are given. And numerical simulations are given in section 5.

2. Problem formulation

2.1. Optimal control problem

Consider a dynamical system can be discribed by affine state space difference equation as follows:

$$\mathrm{x}_{k+1} = f(x_k) + g(x_k)u_k \tag{1}$$

Where $\mathrm{x}_\mathrm{k} \in R^\mathrm{n}$ is the system state and $\mathrm{u}_k \in R^m$ is the control input, respectively, $\mathrm{f} : R^n \to R^n$ with f(**0**)=**0** and $\mathrm{g} : R^n \to R^{m\times n}$ are continuous functions. The system is subject to the constraints:

$$\mathrm{x}_k \in X, u_k \in U \tag{2}$$

Where *X* and *U* are compact sets and contain zero as an interior point. They are represenable by $X = \{x \in R^n : h_X(x) \prec 0\}$ and $U = \{\mathrm{u} \in R^m : h_U(u) \prec 0\}$ where $\mathrm{h}_X : R^n \to R^r$ and $\mathrm{h}_U : R^m \to R^q$ are continuous and convex in x and u, respectively, with suitable r and q. The operator$\prec$ denotes element-wise inequalities. Suppose that system (1) is stabilizable under constraints (2). The optimal control objective can be described as following problem.

Problem 1 Stabilize system (1) subject to (2) by selecting a policy $\mathrm{u}_k = \pi(x_k)$ that minimize the cost as follows:

$$V\text{（}\mathrm{x}_k\text{）} = \sum_{i=k}^{\infty} l(x_i, u_i) \tag{3}$$

Where $l(x_i, u_i)$ is the running cost. We define $l(x_i, u_i) = Q\text{（}x_i\text{）} + \mathrm{u}_k^T R u_k$ for simplicity and require $Q\text{（}x_i\text{）}$ and R to be positive definite.

3. Reinforcement learning MPC

3.1. Optimization problem

In this subsection, we formulate the optimization problem to be solved online at each step. One of the most important issue brought by removing the terminal constraints is recursive feasibility. For this, we introduce a sequence of slacking variables $v_k = \{v_{1|k}, v_{2|k}, ..., v_{N|k}\}$ with $v_{\mathrm{i}|k} \geq 0$ for i=1,2,...,N, to transform the hard constraint in OP 1 into soft ones. Moreover, we replace the terminal cost in (4) with an estimated one. The cost is then

$$\hat{J}(x_k, u_k, v_k) = \sum_{i=0}^{N-1} [l(x_{i|k}, u_{i|k}) + \beta v_{i+1|k}] + \hat{V}(x_{N|k}) \tag{4}$$

where β is a weighted scalar which is chosen by control designer, $\hat{V}(x_{N|k})$ will be updated online, and the next subsection will introduce in detail. The optimization problem is now rewritten as follows.

OP 1 in the following program, we need find the optimal optimizer u_k^* and v_k^*

$$\begin{aligned}
\min_{\mathrm{u}_k,\mathrm{v}_k} \quad & \hat{J}(x_k,u_k,v_k) && (5a)\\
s.t. \quad & x_{0|k}=x_k && (5b)\\
& x_{i+1|k}=f(x_{i|k})+g(x_{i|k})u_{i|k}, 0\le i\le N-1 && (5c)\\
& h_U(u_{i|k})\prec 0, 0\le i\le N-1 && (5d)\\
& h_X(x_{i|k})\prec 0, 1\le i\le N && (5e)\\
& v_{i|k}\succ 0 && (5f)
\end{aligned}$$

The first element in u_k^* is implemented to control system (1) for one step.

It is important to note that the terminal constraint in OP1 is a soft one (5e). This implies OP 1 is always feasible and the selection of the prediction becomes more flexible. In particular, by selecting a sufficient high value of β, the optimizer v_j^* turns out to be negligible, and the first element in v_j^*, i.e., $\mathrm{v}_{1|k}^*$, indicates the level of constraint to be violated at k + 1.

3.2. Value function approximation

According to the Bellman optimality principle, define, the optimal value function as follows

$$V^*(x_k)=\min_{\mathrm{u}_k,v_k}[l(x_{i|k},u_{i|k})+\beta v_{i+1|k}]+V^*(x_{N|k}) \tag{6}$$

Following the higher-order Weierstrass approximation theory, we can use a single-layer network to approximate the optimal value function $\mathrm{V}^*(x_k)$ as

$$V^*(x_k)=W^{*\mathrm{T}}\phi(x_k)+\varepsilon(x_k) \tag{7}$$

Where $\phi(x_k)\in R^p$ and $W\in R^p$ are the polynomial basis function vector and the weighting vector, respectively, and $\varepsilon(x_k)$ is the approximation error.

Generally, the optimal weight W^* is not known in advance. We use the online data to learn it. Thus, letting $\hat{W}_\mathrm{k}$ denote the estimation of W^* at k, the estimated value is then

$$\hat{V}(x_k)=\hat{W}_\mathrm{k}^\mathrm{T}\phi(x_k) \tag{8}$$

which can serve as the terminal cost.

Following the N-step TD learning paradigm, the N-step TD error reads

$$e_N=\hat{J}(x_k,u_k^*,v_k^*)-\hat{W}_k^\mathrm{T}\phi(x_k) \tag{9}$$

If the Bellman equation holds, then $e_N=0$. If not, we aim to minimize $E=\frac{1}{2}e_N^\mathrm{T}e_N$. The weight of the network is updated in gradient descent rule

$$\begin{aligned}
W_{k+1}&=W_k-\frac{1}{2}\alpha\Delta[\hat{J}(x_k,u_k^*,v_k^*)-\hat{W}^\mathrm{T}\phi(x_k)]^2\\
&=W_k-\alpha[\hat{J}(x_k,u_k^*,v_k^*)-\hat{W}^\mathrm{T}\phi(x_k)]\phi(x_k)
\end{aligned} \tag{10}$$

4. Simulation example

Consider the regulation problem of a nonholonomic vehicle system

$$\begin{bmatrix} \mathrm{x}_{\mathrm{k+1}} \\ \mathrm{y}_{\mathrm{k+1}} \\ \theta_{\mathrm{k+1}} \end{bmatrix} = \begin{bmatrix} \mathrm{x}_{\mathrm{k}} \\ \mathrm{y}_{\mathrm{k}} \\ \theta_{\mathrm{k}} \end{bmatrix} + \delta \begin{bmatrix} \cos\theta_k & 0 \\ 0 & \sin\theta_k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_k \\ \omega_k \end{bmatrix} \tag{11}$$

where x_k, y_k and θ_k are the system states, and v_k, ω_k are the control inputs and δ is the sampling step size. Rewritten the system in a compact form and let $\chi_k = [x_k, y_k, \theta_k]^{\mathrm{T}}$ and $u_k = [v_k, \omega_k]^{\mathrm{T}}$. In the simulation, set $\delta = 0.1$. The control input is constrained by $|v_k| \leq 1, |\omega_k| \leq 4$ and the system state is assumed to be subject to $0 \leq x \leq 2$. The control objective is to stabilize the system at the origin while minimize the infinite horizon cost (3) with $l(\chi_k, u_k) = x_k^2 + y_k^2 + 0.01\theta_k^2 + 0.01v_k^2 + 0.005\omega_k^2$.

Note that there is no analytical solution for this problem. Besides, according to Brockett's theory, there doesn't exist contentious invariant feedback that is able to stabilize the system. It brings large challenges in the design of terminal controller and terminal set when using MPC. But we can use the proposed RLMPC framework. The prediction horizon is set to be N = 3. The basic functions are selected as $\Phi(\chi) = [x^4, y^4, \theta^4, x^3y, x^2y^2, xy^3, x^2y, x^2\theta^2, y^2\theta^2, xy^2, x^3, y^3, \theta^3, x^2, y^2, xy]$

The learning step size is selected by repeated simulations to obtain the convergence of weighting W_k.

In the first figure, we use MPC without terminal cost and terminal constraint. In the second figure, we use the RL-MPC which is proposed in this paper. And in the third and forth figures, we replay the task with the well trained network.

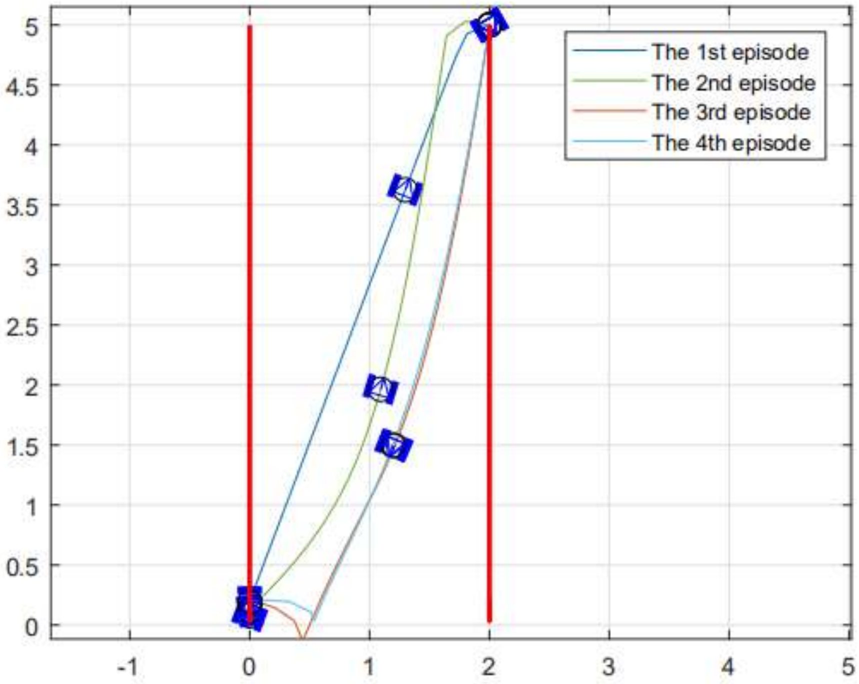

Figure 1. Trajectories

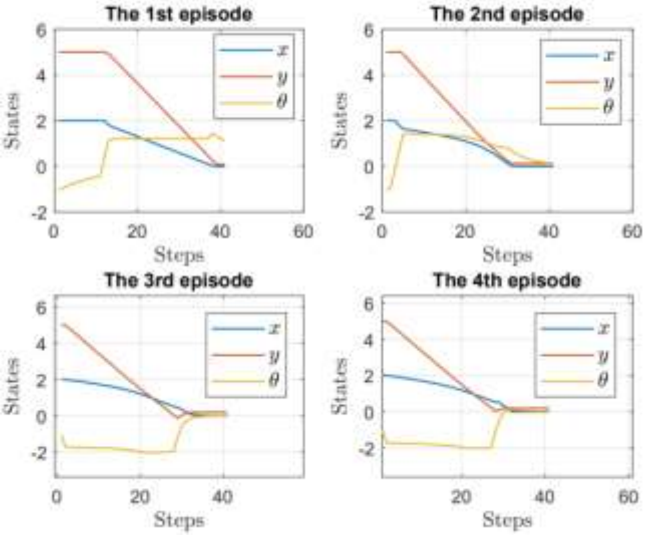

Figure 2. States

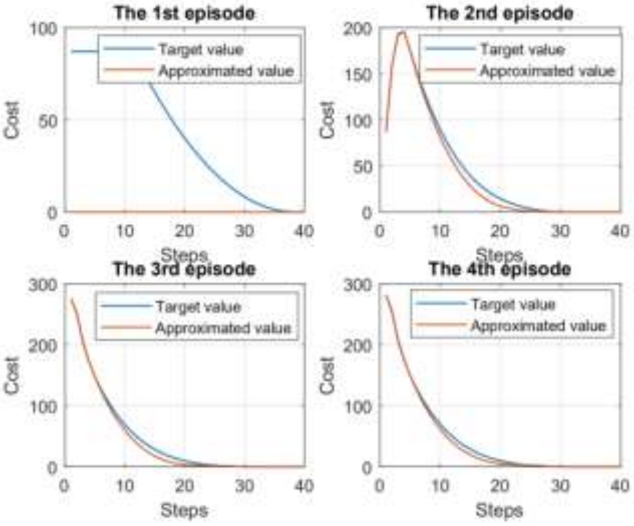

Figure 3. Target and approximated values

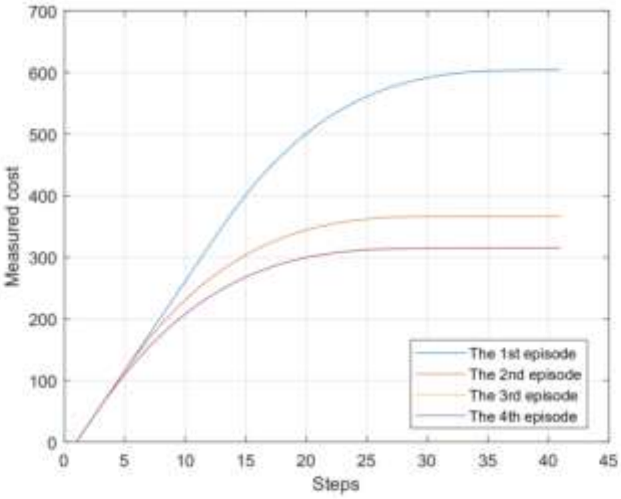

Figure 4. The accumulated costs

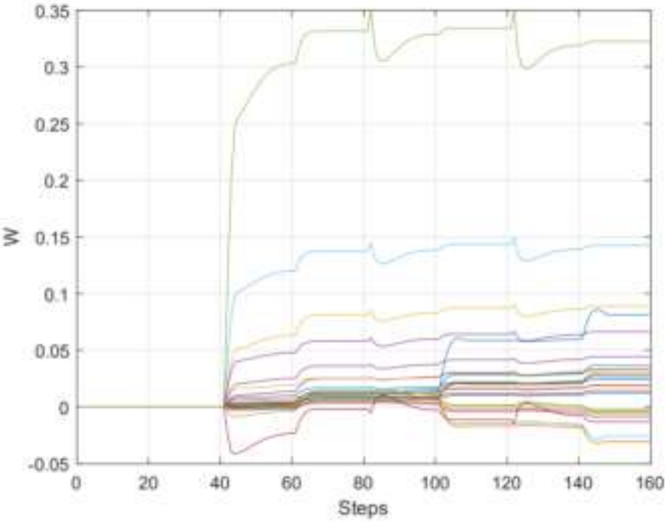

Figure 5. Weighting parameters

5. Conclusion

In order to free the design of the terminal cost, terminal controller and the terminal set in traditional MPC, and to improve the computational efficiency as well as the closed-loop performance, this paper developed a new approach by introducing reinforcement learning into traditional MPC for discrete-time systems. The reinforcement learning-based model predictive control (RLMPC) can improve the control performance effectively. And the numerical simulations demonstrate the effectiveness of the proposed approach.

References

[1] Darby M L and Nikolaou M 2012 MPC: Current practice and challenges *Control Engineering Practice* **20(4)** pp 328–342

[2] Re L, F.Allg¨ower F, Glielmo L, Guardiola C and Kolmanovsky I 2010 Automotive Model Predictive Control: Models, Methods and *Applications Lecture Notes in Control and Information Sciences* vol 402

[3] Rawlings J B and Amrit R 2009 Optimizing process economic performance using model predictive control *Nonlinear Model Predictive Control* pp 119–138

[4] Rawlings J B and Mayne D Q 2009 *Model Predictive Control: Theory and Design* Nob Hill Publishing

[5] Tesauro G 1992 Practical issues in temporal difference learning *Machine Learning* **8(3)** pp 257–277

[6] Lowrey K, Rajeswaran A, Kakade S, Todorov E and Mordatch I 2018 Plan online, learn offline: Efficient learning and exploration via model-based control arXiv preprint arXiv:1811.01848

[7] Gros S and Zanon M 2019 Data-driven economic NMPC using reinforcement learning *IEEE Transactions on Automatic Control* **65(2)** pp 636–648

[8] Hewing L, Wabersich K P, Menner M and Zeilinger M N 2020 Learning-based model predictive control: Toward safe learning in control *Annual Review of Control, Robotics, and Autonomous Systems* **3** pp 269–296

[9] Napat K, Valls M I, Hoeller D and Hutter M 2020 Practical reinforcement learning for MPC: Learning from sparse objectives in under an hour on a real robot *Proceedings of Annual Conference on Learning for Dynamics and Control*