2025 lEEE 14th Data Driven Control and Learning Systems Conference
May 9-11, 2025, Wuxi, China

Model-free Iterative Learning Control based on Adaptive Dynamic Programming

Simin Liu, Ronghu Chi*
School of Automation & Electronic Engineering, Qingdao University of Science & Technology, Qingdao 266061, P. R. China
E-mail: ronghu_chi@hotmail.com

2025 IEEE 14th Data Driven Control and Learning Systems (DDCLS) | 979-8-3503-5731-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/DDCLS66240.2025.11065403

Abstract: An exploration of the model-free iterative learning control based on adaptive dynamic programming is developed for nonlinear nonaffine discrete-time systems. A data-driven iterative dynamic linearization model is developed for original nonlinear systems only using input/output data. Subsequently, a novel performance index function incorporating the error information of future batches is formulated along the iteration axis through dynamic programming, where neural networks are employed to estimate future information. The gradient descent algorithm is employed to facilitate the weight updating of the neural network. Ultimately, a variable penalty factor is developed. This synthesis collectively enhances the convergence rate of tracking errors. The effectiveness of this method is substantiated through two simulation examples.

Key Words: Adaptive Dynamic Programming, Nonlinear Nonaffine Systems, Moder-free, Iterative Learning Control

1 Introduction

In recent years, artificial intelligence has experienced rapid advancements, and the concept of "learning" has been extensively explored across multiple disciplines. Within the domain of practical control engineering, there exist systems characterized by repetitive operations, including welding, transport robots, and fixed-task industrial cutting machines. The control performance can be enhanced through the implementation of "learning" methodologies. Consequently, iterative learning control has attracted significant research attention within the control engineering community [1] [2]. In contrast to conventional control methods over time, iterative learning control exhibits perfect tracking performance across the entire operational envelope rather than achieving merely asymptotic tracking over time.

Iterative learning control systematically utilizes operational data from prior batch processes to progressively refine control strategies, thereby achieving predefined objectives. In light of this, scholars devised numerous iterative learning control laws, including the PID-type learning law, the model reference learning law, and the optimal learning law, among others. To ensure the control performance and convergence rate of these learning laws, it is necessary to find suitable parameters, which give rise to adaptive iterative learning control. This method is widely used in uncertain nonlinear systems [3], switched nonlinear systems [4], multi-agent systems [5] and some practical systems [6], etc. It is imperative to acknowledge that the estimation and updating of parameters for adaptive iterative learning control strategies depend on first-principle model information. In practice, acquiring precise model information frequently poses a significant challenge. Meanwhile, the challenge of designing a controller remains formidable for nonlinear systems, even with an accurate model. Therefore, the data-driven methodology has emerged as a reliable solution to these vexing problems.

Chi and Hou [7] proposed a model-free adaptive iterative learning control (MFAILC) framework in 2007. This method involves reconstructing the first-principle model into an equivalent data model through the dynamic linearization approach along the iteration axis, a process that employs only the input/output data. It facilitates the subsequent design of the controller. The MFAILC method is characterized by simplicity, low computational burden, and independence from any model information, which renders it well-suited for addressing control problems in practical systems. Furthermore, it has the capacity to effectively suppress repetitive disturbances and is more robust. In recent years, data-driven adaptive iterative learning control has undergone further development in conjunction with the problems of event-triggered [8], network attacks [9], disturbances [10], fault-tolerant control [11] etc.

Dynamic programming is a common methodology for solving optimal control problems. However, it also confronts challenges, with the "curse of dimensionality" of recursive equations being the most prominent one. To address this challenge, novel control schemes collectively referred to as "Approximate Dynamic Programming" or "Adaptive Dynamic Programming" (ADP) are proposed, which currently serve as a primary focus in academic research. The ADP method integrates the ideas of optimal control, adaptive control, neural networks, and reinforcement learning. It offers an effective solution to the optimal control problems posed by large-scale, complex, nonlinear systems. In the study of the ADP learning algorithms, it is categorized into value iteration [12] and policy iteration [13] by setting different initial approaches. It is important to note that the above two types of schemes seeking optimal control laws only consider future information on the time axis and do not utilize the a priori operational information of the repetitive system. The element serves to restrict the control effect, consequently making it unfeasible to attain perfect tracking. Further exploration is necessary to investigate how the ADP approach can be combined to study the iterative learning control, so as to realize a globally optimal control strategy on the iterative axis.

In the present paper, the data-driven optimal control problem for nonlinear nonaffine repetitive systems along the iteration axis is studied. The unknown nonlinear model is transformed into a virtual data model by applying the iterative

* Corresponding author.
This work was supported in part by the National Natural Science Foundation of China 62273192.

dynamic linearization based on input/output measurements. Building upon this foundation, we innovatively integrate the ADP framework with iterative learning mechanisms to design a novel performance index function. Neural networks are subsequently adopted to approximate the formulated performance index. The control law is derived through rigorous application of Bellman's optimality principle, accompanied by the design of time-iteration dual-axis varying penalty factors to enhance convergence properties.

2 Problem Formulation

Consider a class of nonaffine nonlinear systems that operates repeatedly over a finite interval:

$$\begin{aligned} y(p+1,q) =& f(y(p,q),\cdots,y(p-n_y,q), \\ & u(p,q),\cdots,u(p-n_u,q)) \end{aligned} \tag{1}$$

where $p \in \{0,1,2,\cdots,\mathcal{T}\}$ denotes the operation instant with $\mathcal{T}$ being the terminal instant. $q \in \{1,2,\cdots\}$ donates the iterative number. $u(p,q) \in R$ and $y(p,q) \in R$ denote the control input and output at the pth operation instant of the qth iteration, respectively. $f(\cdot)$ donates an unknown nonlinear function. n_y and n_u are two unknown positive integers.

The control objective is, given the desired trajectory $y_d(p), p \in \{0,1,2,\cdots,\mathcal{T}\}$, to find optimal control input $u(p,q)$ the such that the tracking error $e(p+1,q) = y_d(p+1) - y(p+1,q)$ converges to 0 when the iterative number tends to infinity.

Before proceeding with the analysis, the following assumptions and lemma are given.

Assumption (A1): The partial derivative of unknown nonlinear function $f(\cdot)$ with respect to control input $u(p,q)$ is continuous.

Assumption (A2): The nonlinear system (1) satisfies generalized Lipschitz condition along the iteration axis. For $\forall p \in \{0,1,2,\cdots,\mathcal{T}\}$ and $\forall q = 1,2,3,\cdots$, if $|\Delta u(p,q)| \neq 0$, then the following equation holds

$$|\Delta y(p+1,q)| \leqslant \bar{L}\,|\Delta u(p,q)| \tag{2}$$

where $\Delta y(p+1,q) = y(p+1,q) - \Delta y(p+1,q-1)$, $\Delta u(p,q) = u(p,q) - u(p,q-1)$, and $\bar{L}$ is a constant.

Lemma 1 [7]: Under the Assumptions A1-A2, if $|\Delta u(p,q)| \neq 0$, there exists a iteratively related time-varying parameter $\phi(p,q)$. The nonlinear system (1) can be transformed into a iterative dynamic linearization model as follows

$$y(p+1,q) = y(p+1,q-1) + \phi(p,q)\,\Delta u(p,q) \tag{3}$$

where pseudo partial derivative $\phi(p,q)$ satisfying $|\phi(p,q)| \leqslant \bar{L}$.

3 Control strategy

In order to calculate the optimal control input $u(p,q)$, it is necessary to construct the performance index function corresponding to the system (1),

$$\begin{aligned} & J[e(p+1,q)] \\ =& \sum_{n=q}^{\infty} \gamma(p,q)^{n-q} \\ & \times \left(|e(p+1,n)|^2 + \lambda\,|u(p,n) - u(p,n-1)|^2 \right) \\ =& |e(p+1,q)|^2 + \lambda\,|u(p,q) - u(p,q-1)|^2 \\ & + \gamma(p,q) J[e(p+1,q+1)] \\ =& |e(p+1,q-1) - \phi(p,q)(u(p,q) - u(p,q-1))|^2 \\ & + \lambda\,|u(p,q) - u(p,q-1)|^2 + \gamma(p,q) J[e(p+1,q+1)] \end{aligned} \tag{4}$$

where $\lambda > 0$ is a constant. $\gamma(\cdot)$ is penalty factor and it is designed later.

Based on Hamilton-Jacobi-Bellman equation, there exists the optimal control sequence $u(p,q), q = 1,2,\cdots$, at operation instant p when (4) is minimized. That is,

$$\begin{aligned} & J^*[e(p+1,q)] \\ =& \min_{u(p,q)} \{ |e(p+1,q)|^2 + \lambda\,|u(p,q) - u(p,q-1)|^2 \\ & + \gamma(p,q)\, J^*[e(p+1,q+1)] \}. \end{aligned} \tag{5}$$

Therefore, based on the optimization condition $\frac{\partial J[e(p+1,q)]}{\partial u(p,q)} = 0$, it is deduced that

$$\begin{aligned} u(p,q) =& u(p,q-1) + \rho \frac{\phi(p,q)\, e(p+1,q-1)}{\lambda + |\phi(p,q)|^2} \\ & - \frac{\gamma(p,q)}{2\lambda + 2\,|\phi(p,q)|^2} \frac{\partial J[e(p+1,q+1)]}{\partial u(p,q)} \end{aligned} \tag{6}$$

where $\rho \in (0,1]$ is a constant.

The control input (6) is not directly applicable since $\phi(p,q)$ and $J[e(p+1,q+1)]$ are unknown. Therefore, the development of algorithms to estimate $\phi(p,q)$ and $J[e(p+1,q+1)]$ are imperative.

In the first place, to derive an estimate $\hat{\phi}(p,q)$ of $\phi(p,q)$, the following parameter estimation criterion function is designed

$$\begin{aligned} J(\phi(p,q)) =& |\Delta y(p+1,q-1) - \phi(p,q)\,\Delta u(p,q-1)|^2 \\ & + \mu \left| \phi(p,q) - \hat{\phi}(p,q-1) \right|^2 \end{aligned} \tag{7}$$

where $\mu > 0$ is a constant.

According to the optimization condition $\frac{\partial J(\phi(p,q))}{\partial \hat{\phi}(p,q)} = 0$, we have

$$\begin{aligned} \hat{\phi}(p,q) =& \hat{\phi}(p,q-1) + \frac{\eta}{\mu + |\Delta u(p,q-1)|^2} \\ & \times \left(\Delta y(p+1,q-1) - \hat{\phi}(p,q-1)\,\Delta u(p,q-1) \right) \\ & \times \Delta u(p,q-1) \end{aligned} \tag{8}$$

where $\eta \in (0,1]$ is a constant.

To make the estimation algorithm (8) more traceable, the

resetting algorithm is given

$$\hat{\phi}(p,q)=\hat{\phi}(p,1)\text{ ,if }\left|\hat{\phi}(p,q)\right|\leqslant\bar{c}\text{ or }|\Delta u(p,q-1)|\leqslant\bar{c}\text{ or sign}\left(\hat{\phi}(p,q)\right)\neq\text{sign}\left(\hat{\phi}(p,1)\right)\quad(9)$$

where $\bar{c}>0$ is a small constant. $\hat{\phi}(p,1)$ is the first batch value at the pth instant.

Then, considering that the neural networks have a good approximation of unknown functions, we have

$$J^{*}\left[e(p+1,q)\right]=W^{*\mathrm{T}}\sigma\left[Y^{\mathrm{T}}e(p+1,q-1)\right]+\delta(p,i)\quad(10)$$

$$J^{*}\left[e(p+1,q+1)\right]=W^{*\mathrm{T}}\sigma\left[Y^{\mathrm{T}}e(p+1,q)\right]+\delta(p,q+1)\quad(11)$$

where W^{*} is the ideal weight vector. $\sigma[\cdot]$ is a hyperbolic tangent function. $|\delta(\cdot)|\leqslant\delta_m$ is the approximation error of the neural network. Y is the constant weight vector with suitable dimensions.

According to (10) and (11), the performance index functions can be estimated as

$$\hat{J}\left[e(p+1,q)\right]=\hat{W}^{\mathrm{T}}(p,q)\,\sigma\left[Y^{\mathrm{T}}e(p+1,q-1)\right]\quad(12)$$

$$\hat{J}\left[e(p+1,q+1)\right]=\hat{W}^{\mathrm{T}}(p,q)\,\sigma\left[Y^{\mathrm{T}}e(p+1,q)\right]\quad(13)$$

where $\hat{W}(p,q)$ is the estimation of W^{*}.

At the operation instant p of the ith iteration, the control input $u(p,q)$ is unknown. It is replaced by the selection of an admissible control $u_0(p)$. Based on this, the estimation error $e(p+1,q)$ becomes $\tilde{e}(p+1,q)=y_d(p+1)-\tilde{y}(p+1,q)$ with $\tilde{y}(p+1,q)=y(p+1,q-1)+\hat{\phi}(p,q)\left(u_0(p)-u(p,q-1)\right)$.

The approximation error $\xi(p+1,q)$ of neural networks can be obtained by subtracting (5) from (12), that is

$$\begin{aligned}\xi(p+1,q)=&\hat{J}\left[e(p+1,q)\right]-J^{*}\left[e(p+1,q)\right]\\=&\hat{J}\left[e(p+1,q)\right]-\gamma(p,q)\hat{J}\left[\tilde{e}(p+1,q+1)\right]\\&-|\tilde{e}(p+1,q)|^{2}-\lambda|u_0(p)-u(p,q-1)|^{2}\\=&\hat{W}^{\mathrm{T}}(p,q)\,\theta\left[e(p+1,q)\right]\\&-|\tilde{e}(p+1,q)|^{2}-\lambda|u_0(p)-u(p,q-1)|^{2}\end{aligned}\quad(14)$$

where $\theta\left[e(p+1,q)\right]=\sigma\left[Y^{\mathrm{T}}e(p+1,q-1)\right]-\gamma(p,q)\sigma\left[Y^{\mathrm{T}}\tilde{e}(p+1,q)\right]$.

Constructing a loss function $E(p+1,q)=\frac{1}{2}\xi(p+1,q)^{2}$, using gradient descent algorithm, the weight updating law for $\hat{W}(p,q)$ is derived as

$$\hat{W}(p,q+1)=\hat{W}(p,q)-\frac{\alpha\theta\left[e(p+1,q)\right]\xi(p+1,q)}{\left\|1+\theta[e(p+1,q)]^{\mathrm{T}}\theta[e(p+1,q)]\right\|}.\quad(15)$$

Based on the above analysis, the optimal control input is

$$\begin{aligned}u(p,q)=&u(p,q-1)+\rho\frac{\hat{\phi}(p,q)\,e(p+1,q-1)}{\lambda+\left|\hat{\phi}(p,q)\right|^{2}}\\&-\frac{\gamma(p,q)}{2\lambda+2\left|\hat{\phi}(p,q)\right|^{2}}\frac{\partial\hat{J}\left[e(p+1,q+1)\right]}{\partial u(p,q)}\end{aligned}\quad(16)$$

where the penalty factor $\gamma(p,q)$ is constructed as $\gamma(p,q)=\beta\frac{|e(p+1,q-1)|}{|Y||\hat{W}(p,q)|}$ and $\beta>0$ is a constant.

Based on the analysis in the previous section, we propose a model-free adaptive iterative learning control strategy based on adaptive dynamic programming (ADPMFAILC), which including a learning control algorithm (16), an iterative parameter estimation algorithm (8), a parameter resetting algorithm (9), and a weight updating algorithm (15).

4 Simulation Results

Two simulation case studies are conducted to validate the effectiveness of the ADPMFAILC strategy for discrete-time systems. Notably, a mathematical model is exclusively employed to generate input-output data during simulations, while the controller synthesis process remains entirely model-free.

Example 1 (numerical example): Consider the following discrete-time nonlinear system,

$$y(p+1)=\begin{cases}\dfrac{y(p)}{1+y(p)^{2}}+u(p)^{3},0\leqslant p\leqslant 50\\\dfrac{y(p)y(p-1)y(p-2)u(p-1)\left(y(p-2)-1\right)}{1+y(p-1)^{2}+y(p-2)^{2}}\\+\dfrac{s(p)\,u(p)}{1+y(p-1)^{2}+y(p-2)^{2}},50\leqslant p\leqslant 100\end{cases}\quad(17)$$

where $s(p)=1+\text{round}\left(p/50\right)$.

The desired trajectory of the system (17) is given as

$$y_d(p+1)=\begin{cases}0.5\times(-1)^{\text{round}\left(p/10\right)},0\leqslant p\leqslant 30\\0.5\sin\left(\dfrac{p\pi}{10}\right)+0.3\cos\left(\dfrac{p\pi}{10}\right),30<p\leqslant 70\\0.5\times(-1)^{\text{round}\left(p/10\right)},70<p\leqslant 100\end{cases}\quad(18)$$

The parameters of the control strategy are selected as: $\lambda=1.15$, $\mu=15$, $\eta=1$, $\rho=1.5$, $\beta=1$, $\alpha=0.5$. The admissible control $u_0(p)$ are given as 0.5. The initial value are given as $u(0,q)=0,q=1,\cdots,100$, $y(0)=0$, $\hat{\phi}(p,1)=0.5$,

$$\begin{aligned}Y=&[0.1\ 0.5\ 0.1378\ 0.126\ 0.89\\&0.23\ 0.75\ 0.432\ 0.521\ 0.012],\\\hat{W}^{\mathrm{T}}(p,1)=&[0.352\ 0.518\ 0.734\ 0.425\ 0.133\\&0.16\ 0.465\ 0.634\ 0.879\ 0.05].\end{aligned}$$

The system is repeated for 100 iterations and the simulation results are presented in Fig. 1, Fig.2, and Fig. 3. Fig. 1 depicts the tracking performance. Fig. 2 compares the maximum tracking error $e_{\max}(q)=\max_{p\in\{1,\cdots,100\}}|e(p,q)|$ for each iteration of different control strategies. Fig. 3 depicts the control input for different batches.

Example 2 (practical example): A mechanistic model of a DC motor-driven single-link manipulator [14] is given as follows

$$J\ddot{\nu}+f\dot{\nu}+\left(\frac{1}{2}m+M\right)gl\sin(\nu)=u\quad(19)$$

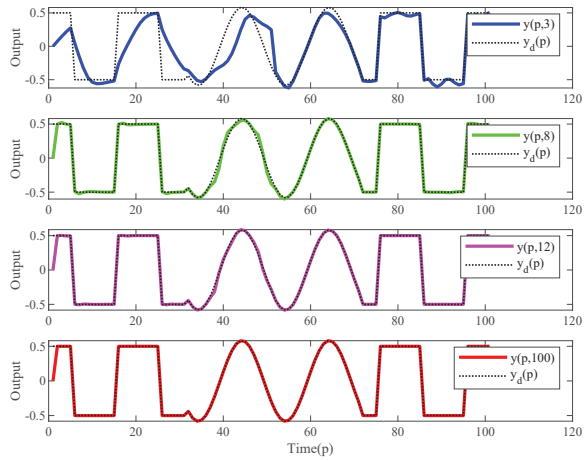

Fig. 1: The system output for different iterative numbers and the desired trajectory $y_d\,(p)$ of Example 1

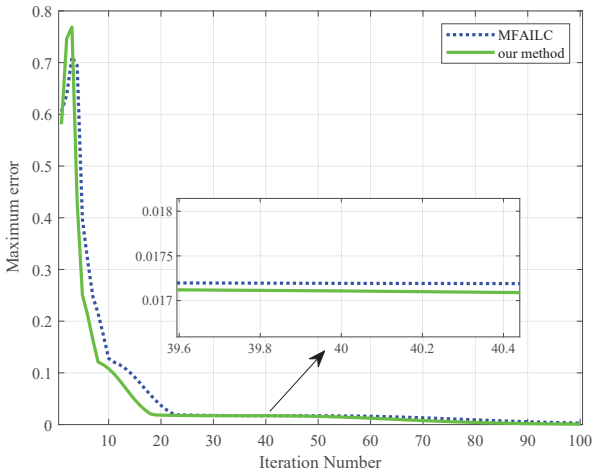

Fig. 2: The maximum tracking error $e_{\max}\,(q)$ of Example 1

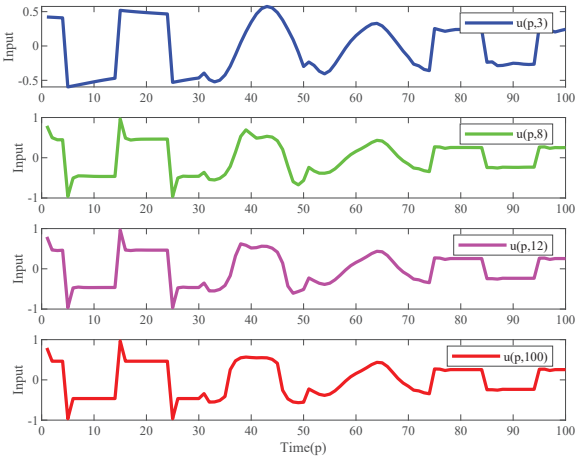

Fig. 3: The control input for different iterative numbers of Example 1

where $\ddot{\nu}$, $\dot{\nu}$, ν are the acceleration,velocity and angular displacement of the moter, respectively. $J = Ml^2 + \left(1/3\right) ml^2$ is the moment of inertia. u is the driving torqueu as well as the system input. $\dot{\nu}$ is the system output. Other physical quantities are explained in Reference [15]. The parameters of the system are $m = 1\,\text{kg}$, $M = 2\,\text{kg}$, $l = 0.5\,\text{m}$, $f = 3\,\text{kg}\cdot\text{m}^2/\text{s}$, $g = 9.8\,\text{m/s}^2$.

The desired tracking trajectory is $y_d\,(p) = 0.5\sin\left(\pi p\right)$ with $p \in [0s, 1s]$. Choose the sampling interval as 0.01s.

The parameters of the control strategy are selected as: $\lambda = 0.1$, $\mu = 10$, $\eta = 1$, $\rho = 4$, $\beta = 6$, $\alpha = 0.5$. The admissible control $u_0\,(p)$ are 0.5. The initial value are given as $u\,(0, q) = 0, q = 1, \cdots, 100$, $y\,(0) = 0.025$, $\hat{\phi}\,(p, 1) = 0.5$,

$$Y = [0.1\ 0.5\ 0.1378\ 0.126\ 0.89\ 0.23\ 0.75\ 0.432\ 0.521\ 0.012],$$

$$\hat{W}^{\mathrm{T}}\,(p, 1) = [0.352\ 0.518\ 0.734\ 0.425\ 0.133\ 0.16\ 0.465\ 0.634\ 0.879\ 0.05].$$

The ADPMFAILC strategy is applied to the practical system, and the simulation results are presented in Fig. 4, Fig. 5, and Fig. 6. Fig. 4 depicts the tracking performance at the 100th iteration. Fig. 5 compares the maximum tracking error $e_{\max}\,(q) = \max\limits_{p\in\{1,\cdots,100\}} |e\,(p, q)|$ for each iteration of different control strategies. Fig. 6 depicts the control input for different batches.

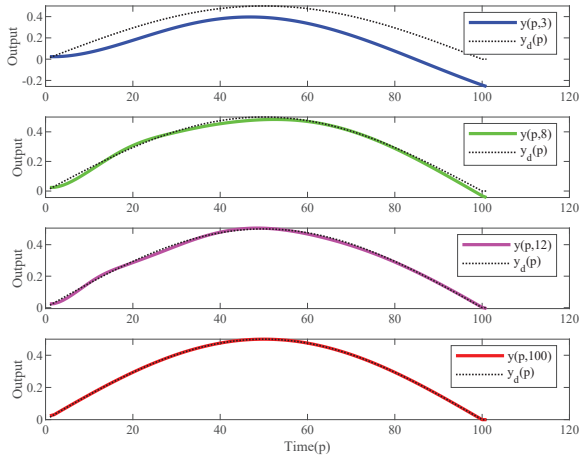

Fig. 4: The system output different iterative numbers and the desired trajectory $y_d\,(p)$ of Example 2

Fig. 1 and Fig. 4 show that as the number of iterations increases, the system outputs are able to track the desired signal in a pointwise manner, which proves the effectiveness of the control strategy. According to Fig. 2 and Fig. 5, it can be seen that both the MFAILC strategy and the ADPMFAILC strategy can converge to near 0 for the maximum tracking error $e_{\max}\,(q)$ after many iterations. However, the strategy proposed in this paper converges faster than the MFAILC strategy. It is due to the fact that the tracking error information for future iterations is incorporated into the controller design based on the dynamic programming theory. As evident from Fig. 3 and Fig. 6, the control input under-

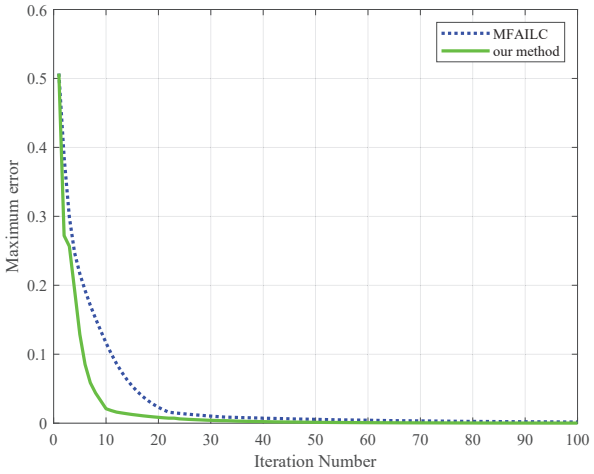

Fig. 5: The maximum tracking error $e_{\max}(q)$ of Example 2

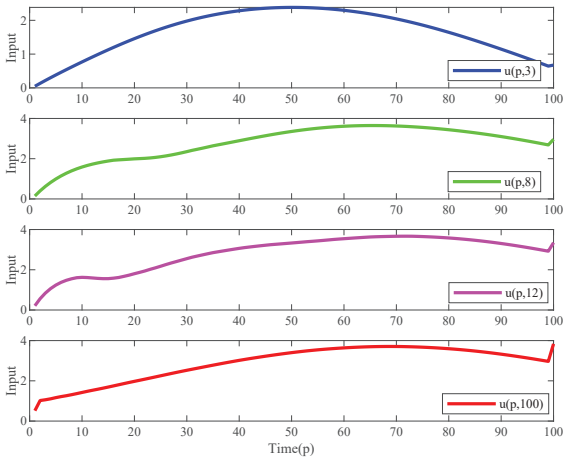

Fig. 6: The control input for different iterative number of Example 2

goes continuous adjustments across iterations while exhibiting dynamic variations from time step 1 to 100.

5 Conclusions

In this article, a novel control strategy is proposed by combining the MFAILC and ADP for nonlinear nonaffine discrete-time systems. This approach enhances the convergence rate of the iterative learning tracking error along the iteration axis. The strategy is initiates by converting the unknown nonlinear nonaffine system, through input/output data, into an equivalent dynamic linearization model. Subsequently, the neural networks are used to estimate the optimal performance index functions. The weights of the neural networks are subject to continuous updates during iterations. Moreover, the penalty factor is carefully designed as a variable that changes with each iteration and over the course of time, rather than remaining constant. Finally, two simulations are presented to illustrate the effectiveness of the proposed control strategy.

References

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, Bettering operation of robots by learning, *Journal of Robotic Systems*, 1(2): 123–140, 1984.

[2] R. Chi, Y. Hui, R. Wang, B. Huang, and Z. Hou, Discrete-time distributed adaptive ILC with nonrepetitive uncertainties and applications to building HVAC systems, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(8): 5068–5080, 2022.

[3] A. Tayebi and C. Chien, A unified adaptive iterative learning control framework for uncertain nonlinear systems, *IEEE Transactions on Automatic Control*, 52(10): 1907–1913, 2007.

[4] Y. Geng, X. Ruan and J. Xu, Adaptive iterative learning control of switched nonlinear discrete-time systems with unmodeled dynamics, *IEEE Access*, 7: 118370–118380, 2019.

[5] J. Chen, J. Li, W. Chen, S. Zhang and J. Zhang, Iterative learning control for nonlinear uncertain parameterized multi-agent systems with non-identical partially unknown control directions, *IEEE Transactions on Network Science and Engineering*, 11(4): 3358–3369, 2024.

[6] T. Zhang, X. Jiao and Y. Zhang, Internal-model-principle-based fast adaptive iterative learning trajectory tracking control for autonomous farming vehicle under alignment condition and input constraint, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(6): 3588–3599, 2023.

[7] R. Chi, Z. Hou, Dual-stage optimal iterative learning control for nonlinear non-affine discrete-time systems, *Acta Automatica Sinica*, 33(10): 1061–1065, 2007.

[8] N. Lin, R. Chi, B. Huang, Event-triggered ILC for optimal consensus at specified data points of heterogeneous networked agents with switching topologies, *IEEE Transactions on Cybernetics*, 52(9): 8951–8961, 2022.

[9] H. Liu and L. Hao, An improved data-driven iterative learning secure control for intelligent marine vehicles with DoS attacks, *IEEE Transactions on Intelligent Vehicles*, 9(2): 2160–2170, 2024.

[10] J. Zheng and Z. Hou, ESO-based model-free adaptive iterative learning energy-efficient control for subway train with disturbances and over-speed protection, *IEEE Transactions on Intelligent Transportation Systems*, 24(8): 8136–8148, 2023.

[11] G. Liu and Z. Hou, Cooperative adaptive iterative learning fault-tolerant control scheme for multiple subway trains, *IEEE Transactions on Cybernetics*, 52(2): 1098–1111, 2022.

[12] Q. Wei, D. Liu, H. Lin, Value iteration adaptive dynamic programming for optimal control of discrete-time nonlinear systems, *IEEE Transactions on Cybernetics*, 46(3): 840–853, 2016.

[13] B. Gravell, K. Ganapathy, T. Summers, Policy iteration for linear quadratic games with stochastic parameters, *IEEE Control Systems Letters*, 5(1): 307–312, 2021.

[14] R. Chi, Y. Hui, B. Huang, Z. Hou and X. Bu, Data-driven adaptive consensus learning from network topologies, *IEEE Transactions on Neural Networks and Learning Systems*, 33(8): 3487–3497, 2022.

[15] R. Chi, Y. Wei, R. Wang, and Z. Hou, Observer based switching ILC for consensus of nonlinear nonaffine multi-agent systems, *Journal of the Franklin Institute*, 358(12): 6195–6216, 2021.