

JOINT PERFECT TRACKING AND DISTURBANCE REJECTION FOR NONLINEAR ADAPTIVE DIRECT DATA DRIVEN CONTROL

XIAO LUO¹, SHIGEYUKI TATENO¹ AND JIANHONG WANG²

¹Graduate School of Information, Production and Systems
Waseda University
2-7 Hibikino, Wakamatsu-Ku, Kitakyushu-shi, Fukuoka 808-0135, Japan
yongluo@qq.com; rashou@ruri.waseda.jp

²School of Electronic Engineering and Automation
Jiangxi University of Science and Technology
No. 86, Hongqi Road, Ganzhou 341000, P. R. China
wangjianhong@nuaa.edu.cn

Received October 2025; revised February 2026

ABSTRACT. *This paper presents nonlinear adaptive direct data driven control for perfect tracking and disturbance rejection, while combining adaptation and nonlinearity. After reviewing closed loop system with unknown plant and controller, classical control within dual missions depends on the unknown plant, meaning the first premise is system modeling. To bypass the need for system modeling for unknown plant, direct data driven control is proposed from these dual missions of perfect tracking and disturbance rejection, respectively, i.e., designing controller only through data samples. For the special parameterized controller, adaptation is introduced to identify these controller parameters within the framework of data driven strategy. Furthermore, due to nonlinearity, instead of designing one ideal linear controller, multidimensional Taylor network approximation is applied to replacing the original nonlinear controller. This approximation remains nonlinear structure into one regressor vector, also shows linear relation with respect to network weights. The approximation accuracy can be improved through adjusting these network weights, identified by learning algorithm. Finally, the method is illustrated by a more practical case study.*

Keywords: Direct data driven control, Perfect tracking, Disturbance rejection, Multidimensional Taylor network, Nonlinearity, Adaptation

1. Introduction. Speaking informally, the major systems, used in modern practical engineering applications, all render their complex dynamical behaviors that may be described by their corresponding mathematical models, thus leading to linear and nonlinear models. To do the controller design process or other interesting work, such as fault detection and mission allocation, the premise is system modeling by first principle or physical modeling and system identification [1]. Whatever linear model or nonlinear model is used to exhibit the considered system's motion, it is well known that if the operating range of the control system is small and if the system nonlinearities are smooth, i.e., no chaos or saturation, then the control system can be locally approximated by a linearized system about a given operating condition and well established linear multivariable control methods can be used to maintain local stability and performance [2]. Due to the explicit form and solvable property, existing research about linear system identification and linear control is very mature, meaning lots of linear system theories exist. However, in modern high tracking precision engineering applications such as unmanned aircraft vehicle (UAV), advanced

traffic network system and human brain, the locally approximated linearized system cannot cover the operating range of nonlinear models [3], i.e., the ideal linear model cannot describe completely some unavoidable nonlinear factors, for example, damping, friction, and seeping [4]. For this reason, nonlinear model is more accurate than linear model, so up to now, general nonlinear system identification and nonlinear control have been studied widely and remain two open problems.

To deal with system modeling, the previous adopted solution depends on first principle or physical principle. However, as it needs strong background of dynamic analysis and kinematic analysis of the control system, resulting in more difficulties for engineer, the community turns to system identification, i.e., extracting some useful information for the control system through statistical analysis, machine learning and artificial intelligence references. From [5], the main essence of system identification includes four steps, i.e., experiment design, model choice, parameter identification and model validation, corresponding to dealing with data and data driven identification. On the one side, since data includes important knowledge about the control system, it also holds for the unknown controller, i.e., constructing an explicit form for controller directly from data, named direct data driven control [6]. The difference between indirect and direct data driven control is whether system modeling is needed. During this big data era, as data is easily collected through some physical devices, more ongoing research is concentrated on direct data driven control from academic theory and engineering application, such as robustness, adaptation, nonlinearity, safety and learning [7]. Thanks to the fact that data is indeed abundant and even freely accessible, direct data driven control is worth studying in depth, while combining more advanced ideas together, for example, data science, deep learning and reinforcement learning. On the other side, except the theoretic field about direct data driven control, it is also applied in lots of practical engineering widely (e.g., aerospace, automatic, energy, glass factory, garbage plant, and robot) [8].

The present paper is motivated by the realization that, although there are more advantages in direct data driven control, there is still a requirement for studying direct data driven control strategy deeply that can extend to more complex situations, and keep it more intelligent or smart. More specifically, firstly, this paper synthesizes dual control performances, i.e., perfect tracking and disturbance rejection together, guaranteeing two different transfer functions approach their own expected or desired models within the framework of direct data driven control, respectively. To explain these two control properties, perfect tracking and disturbance rejection are analyzed in detail, and then adaptation idea is introduced to identify the unknown controller parameters, i.e., establishing one parameter adjustment mechanism to generate controller parameters adaptively. Furthermore, our parameter adjustment mechanism is dependent on the collected data, not on the considered control system, thus suiting to the essence of data science. Secondly, based on adaptive direct data driven control, we continue to exploit the deep nonlinear adaptive direct data driven control, i.e., applying the collected data to devising one nonlinear controller. To embody the adaptation idea, i.e., generating controller parameters adaptively, multidimensional Taylor network structure (MTN) is proposed to replace that unknown or implicit nonlinear controller. Due to the nice property of multidimensional Taylor network structure that it can approximate any nonlinear system well on condition of the number of network [9], nonlinear control is transferred to an easy network weight identification problem. From [10], MTN controller features the following. 1) It is a fixed structure convenient for controller design and implementation. 2) It can realize asymptotic tracking of the reference through adjusting those network weights accurately. 3) Its dynamic regulation of system output is achieved with the help of a given index. Combination of stochastic model predictive control and MTN for nonlinear system is established

by virtue of the optimal transportation theory [11]. Making use of MTN approximation of the original nonlinear controller, our previously proposed learning algorithm in [12] is adopted here to identify those unknown network weights adaptively. Specifically, the commonly used proportional integral derivative (PID) control is a special case of MTN controller.

Generally, the main contributions of this paper are formulated as follows.

- 1) After reviewing the main essence of direct data driven control strategy, its dual control performances are achieved simultaneously, i.e., perfect tracking and disturbance rejection.
- 2) To embody the adaptation idea for the parameterized controller, a nice parameter adjustment mechanism, depending only on data, is constructed to adjust those controller parameters.
- 3) Considering nonlinear controller design, MTN structure is proposed to transform the original nonlinear controller design problem into one network weight identification problem, and then learning algorithm is applied to identifying network weights adaptively only through the collected data, thus resulting in our named innovative nonlinear adaptive direct data driven control.

This paper is organized as follows. In Section 2, closed loop system structure with the unknown plant and unknown controller is reviewed. After given two expected transfer functions, dual control goals of perfect tracking and disturbance rejection are explained. Bypass the need for system modeling for that unknown plant, Section 3 proposes direct data driven control strategy to achieve above dual control missions, while jointly considering adaptation idea. To consider nonlinear controller design only by virtue of data driven idea, multidimensional Taylor network approximation is shown in Section 4 to replace the original nonlinear controller. Then for approximation accuracy, learning algorithm is proposed to identify those network weights to approximate that original nonlinear controller well. Section 5 gives a practical example to prove our theoretical results. Finally, Section 6 ends the paper with a final conclusion and points out next work. Generally, both our main contributions, proposed in Sections 3 and 4, are to extract the controller only through data science directly, not dependent on the unknown plant.

2. Closed Loop System Structure Description. Two control systems exist: open loop system and closed loop system for academic community and practical engineering, respectively. Due to unstable or other bad properties of open loop system, now closed loop system is more widely used everywhere. For example, aircraft flight control system corresponds to a closed loop system with three feedback loops, thus needing to design three feedback controllers. Consequently, closed loop system structure is considered here, and plotted in Figure 1.

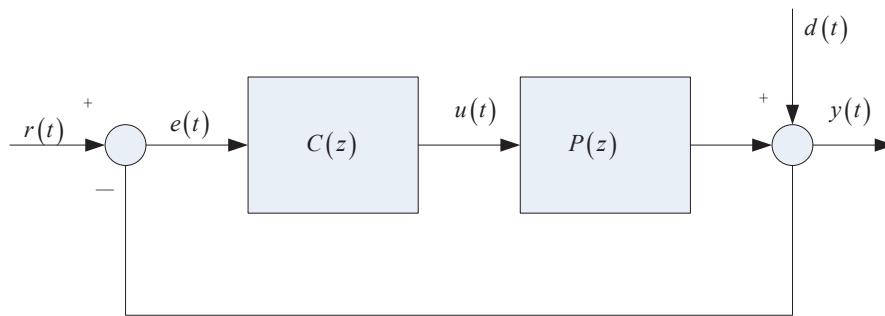


FIGURE 1. Closed loop system

In Figure 1, all physical variables are defined as follows. $\{P(z), C(z)\}$ are the unknown plant and unknown feed-forward controller. $\{r(t), u(t), y(t)\}$ are external input signal, plant input and closed loop output signal, respectively. $d(t)$ is the external disturbance or noise. z is the shift operator. Making use of the negative feedback, error signal is $e(t) = r(t) - y(t)$.

Observing Figure 1 again, some existing mathematical relations hold easily.

$$y(t) = P(z)u(t) + d(t) = P(z)C(z)r(t) - P(z)C(z)y(t) + d(t) \quad (1)$$

i.e.,

$$y(t) = \frac{P(z)C(z)}{1 + P(z)C(z)}r(t) + \frac{1}{1 + P(z)C(z)}d(t) \quad (2)$$

and

$$P(z)u(t) = y(t) - d(t) = \frac{P(z)C(z)}{1 + P(z)C(z)}r(t) - \frac{P(z)C(z)}{1 + P(z)C(z)}d(t) \quad (3)$$

i.e.,

$$u(t) = \frac{P(z)C(z)}{1 + P(z)C(z)}[r(t) - d(t)] \quad (4)$$

As each control strategy has its own control mission, in this paper, our control mission is to design an approximate feed-forward controller $C(z)$ to satisfy the dual perfect tracking and disturbance rejection. Specifically, our mission is to guarantee two transfer functions from external input $r(t)$ and external disturbance $d(t)$ for closed loop output $y(t)$ to converge to two given transfer functions $\{M(z), S(z)\}$ respectively, i.e.,

$$\begin{aligned} \frac{P(z)C(z)}{1 + P(z)C(z)} &\rightarrow M(z); \text{ Perfect tracking} \\ \frac{1}{1 + P(z)C(z)} &\rightarrow S(z); \text{ Disturbance rejection} \end{aligned} \quad (5)$$

where two transfer functions $\{M(z), S(z)\}$ are given in prior.

For above dual control missions, the feed-forward controller $C(z)$ is designed such that

$$\frac{P(z)C(z)}{1 + P(z)C(z)} = M(z); \quad \frac{1}{1 + P(z)C(z)} = S(z) \quad (6)$$

i.e.,

$$\frac{P(z)C(z)}{1 + P(z)C(z)} = M(z); \quad P(z)C(z)S(z) = M(z); \quad C(z) = \frac{M(z)}{P(z)S(z)} \quad (7)$$

Above Equation (7) also means model matching problem, i.e.,

$$C(z) = \arg \min_{C(z)} \left\| \frac{P(z)C(z)}{1 + P(z)C(z)} - M(z) \right\|_2 + \|P(z)C(z)S(z) - M(z)\|_2 \quad (8)$$

where notation $\|\cdot\|$ means the common Euclidean norm.

Remark 2.1. Observing the above model matching problem for two terms, corresponding to perfect tracking and disturbance rejection, as the unknown plant $P(z)$ exists in above both two terms, before solving above optimization problem, the premise is plant modeling, i.e., identifying the unknown plant $P(z)$ firstly, then designing the feed-forward controller $C(z)$. Furthermore, the ideal feed-forward controller is $C(z) = \frac{M(z)}{P(z)S(z)}$, where $\{M(z), S(z)\}$ are known a priori as two reference models, so the feed-forward controller $C(z)$ is dependent on the plant $P(z)$. Above analysis combines identification and control

together to be indirect data driven control, as identification is also to extract mathematical model for plant only through data.

3. Direct Data Driven Control. Data driven control includes two kinds, i.e., indirect and direct form. The above controller $C(z)$ from solving optimization problem (8) is indirect data driven control, whose two steps are identification and then simple substituting, i.e., $C(z) = \frac{M(z)}{P(z)S(z)}$. The foregoing analysis about indirect data driven control from the dual control missions of perfect tracking and disturbance rejection is shown in Figure 2, which will be analyzed in detail.

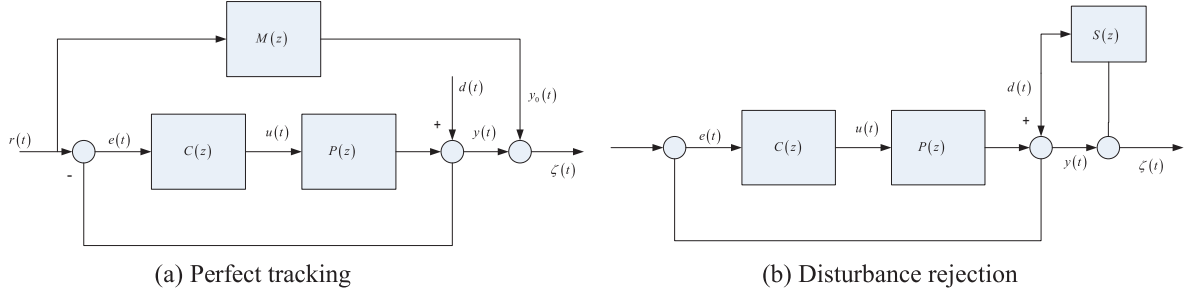


FIGURE 2. Dual control missions

3.1. Perfect tracking. Perfect tracking guarantees

$$\frac{P(z)C(z)}{1 + P(z)C(z)} = M(z) \quad (9)$$

Observing Figure 2(a), given a reference model $M(z)$, the ideal closed loop output $y_0(t) = M(z)r(t)$, and it holds

$$\begin{aligned} \xi(t) &= y(t) - y_0(t) = \left[\frac{P(z)C(z)}{1 + P(z)C(z)} - M(z) \right] r(t) + \frac{1}{1 + P(z)C(z)} d(t); \\ e(t) &= r(t) - y(t) \end{aligned} \quad (10)$$

Let closed loop output error $\xi(t) = 0$, i.e.,

$$\frac{P(z)C(z)}{1 + P(z)C(z)} \rightarrow M(z); \quad \frac{1}{1 + P(z)C(z)} \rightarrow 0 \quad (11)$$

To neglect the system modeling process for the unknown plant $P(z)$ in Equations (9)-(11), we replot Figure 2(a) in Figure 3, remaining perfect tracking.

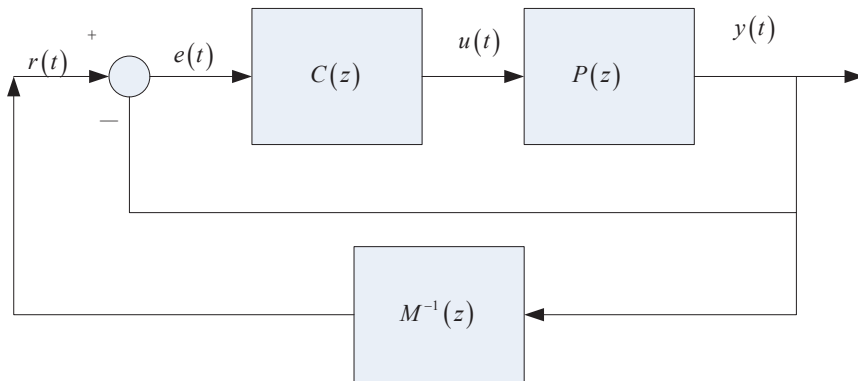


FIGURE 3. Direct data driven control with perfect tracking

In Figure 3, $M^{-1}(z)$ is an inverse model, and $d(t) = 0$.

As $y(t) = M(z)r(t)$, $r(t) = M^{-1}(z)y(t)$. To embody the nice essence of direct data driven control, the input-output data of unknown feed-forward controller $C(z)$ are collected as $\{u(t), y(t)\}_{t=1}^N$, where N is the total number of data. Also, we find $\{u(t), e(t)\}_{t=1}^N = \{u(t), r(t) - y(t)\}_{t=1}^N = \{u(t), (M^{-1}(z) - 1)y(t)\}_{t=1}^N$, where $M(z)$ is given and $\{u(t), y(t)\}_{t=1}^N$ are collected directly.

Due to

$$u(t) = C(z)e(t) = C(z)[r(t) - y(t)] = C(z)(M^{-1}(z) - 1)y(t), \quad t = 1, 2, \dots, N \quad (12)$$

all physical variables or data are known, only except the unknown feed-forward controller $C(z)$.

After collecting data $\{u(t), y(t)\}_{t=1}^N$ and using the inverse relation $M^{-1}(z)$, then both sides of feed-forward controller $C(z)$ are also generated as $\{u(t), r(t) - y(t)\}_{t=1}^N = \{u(t), (M^{-1}(z) - 1)y(t)\}_{t=1}^N$, also being as a data set. The ideal controller $C(z)$ must satisfy Equation (12), so it can be designed through minimizing the following optimization problem, constituted only by data set. Combining Equations (6) and (10), we have

$$C(z) = \arg \min \frac{1}{N} \sum_{t=1}^N [u(t) - C(z)y_1(t)]^2; \quad y_1(t) = (M^{-1}(z) - 1)y(t) \quad (13)$$

Equation (13) tells us that the feed-forward controller $C(z)$ is designed to satisfy Equation (12), and all physical variables are collected or known a priori, suited to the essence of direct data driven control.

3.2. Disturbance rejection. Since no external noise or disturbance $d(t)$ appears in Figure 3, the other control mission of disturbance rejection is considered here in Figure 4.

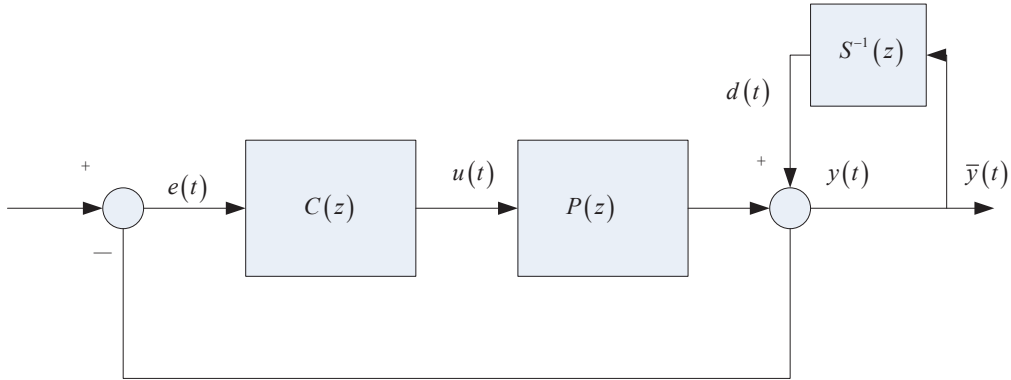


FIGURE 4. Direct data driven control with disturbance rejection

Figure 3 and Figure 4 consider our mentioned dual control missions separately, corresponding to Figure 2(a) and Figure 2(b), respectively. Observing Figure 4, $\bar{y}(t)$ is a noisy output sequence, i.e.,

$$\bar{y}(t) = y(t) + d(t) = S(z)d(t); \quad d(t) = \frac{y(t)}{S(z) - 1} \quad (14)$$

Then

$$\begin{aligned} u(t) &= C(z)e(t) = C(z)(-\bar{y}(t)) = -C(z)\bar{y}(t) \\ \bar{y}(t) &= y(t) + d(t) = y(t) + \frac{y(t)}{S(z) - 1} = \left[1 + \frac{y(t)}{S(z) - 1}\right] y(t) = \frac{S(z)}{S(z) - 1} y(t) \end{aligned} \quad (15)$$

After simple but tedious calculation, we get

$$u(t) = -C(z) \frac{S(z)}{S(z) - 1} y(t), \quad t = 1, 2, \dots, N \quad (16)$$

Based on this ideal Equation (16), we design the feed-forward controller $C(z)$ such that Equation (16) is satisfied, resulting in also the following optimization problem.

$$C(z) = \arg \min \frac{1}{N} \sum_{t=1}^N [u(t) + C(z)y_2(t)]^2; \quad y_2(t) = \frac{S(z)}{S(z) - 1} y(t) \quad (17)$$

Combining Equations (13) and (17) together yields our direct data driven control strategy, whose two terms define the dual control missions respectively.

$$C(z) = \arg \min_{C(z)} J(C(z)) = \arg \min \frac{1}{N} \sum_{t=1}^N [u(t) - C(z)y_1(t)]^2 + \frac{1}{N} \sum_{t=1}^N [u(t) + C(z)y_2(t)]^2$$

$$y_1(t) = (M^{-1}(z) - 1) y(t); \quad y_2(t) = \frac{S(z)}{S(z) - 1} y(t) \quad (18)$$

where $\{u(t), y(t)\}_{t=1}^N$ are collected and $\{M(z), S(z)\}$ are two reference models, so all physical variables other than $C(z)$ are known. Furthermore, these two terms in Equation (18) are independent of the unknown plant $P(z)$, i.e., avoiding the complex system modeling process.

Remark 3.1. From the above mathematical analysis, the controller design problem is turned to a mathematical optimization problem, whose physical variables are all dependent on data. When solving the mathematical optimization problem (8), some existing optimization algorithms can be applied directly, for example, gradient algorithm, and Newton algorithm.

3.3. Joint consideration of adaptation. To introduce the adaptation idea into direct data driven control strategy, adaptation means the controller parameters can be updated recursively and adaptively without human intervention, so consider the special linear parameterized controller as

$$C(z) = C(z, \theta) = \alpha(z)\theta \quad (19)$$

where $\alpha(z)$ is the basis function, for example, $\alpha(z) = [1, z, z^2, \dots, z^n]$, and θ denotes the controller parameter vector, $\theta = [\theta_1, \theta_2, \dots, \theta_n]$, where n is the order.

Substituting Equation (19) into Equation (18), we have

$$J(C(z)) = J(\theta) = \frac{1}{N} \sum_{t=1}^N [u(t) - \alpha(z)\theta y_1(t)]^2 + \frac{1}{N} \sum_{t=1}^N [u(t) + \alpha(z)\theta y_2(t)]^2 \quad (20)$$

By differentiating with respect to θ and by setting the derivative equal to zero, it holds

$$\frac{\partial J(\theta)}{\partial \theta} = \frac{2}{N} \sum_{t=1}^N [u(t) - \alpha(z)\theta y_1(t)]\alpha(z) + \frac{2}{N} \sum_{t=1}^N [u(t) + \alpha(z)\theta y_2(t)]\alpha(z) = 0 \quad (21)$$

i.e.,

$$\sum_{t=1}^N [-(\alpha(z)y_1(t))^T(\alpha(z)y_1(t)) + (\alpha(z)y_2(t))^T(\alpha(z)y_2(t))] \theta$$

$$= \sum_{t=1}^N [\alpha(z)y_1(t)u(t) + \alpha(z)y_2(t)u(t)] = \sum_{t=1}^N [(\alpha(z)y_1(t) + \alpha(z)y_2(t))u(t)] \quad (22)$$

For notational clarity, set

$$\phi_1(t) = \alpha(z)y_1(t); \quad \phi_2(t) = \alpha(z)y_2(t) \quad (23)$$

as two regression vectors or regressors, and then Equation (22) is simplified as

$$\sum_{t=1}^N [\phi_2^T(t)\phi_2(t) - \phi_1^T(t)\phi_1(t)] \theta = \sum_{t=1}^N [\phi_1(t) + \phi_2(t)]^T u(t) \quad (24)$$

Equation (24) is the common least squares form, and its solution is easily solved as

$$\hat{\theta} = \left[\sum_{t=1}^N [\phi_2^T(t)\phi_2(t) - \phi_1^T(t)\phi_1(t)] \right]^{-1} \left[\sum_{t=1}^N [\phi_1(t) + \phi_2(t)]^T u(t) \right] \quad (25)$$

where $\hat{\theta}$ is the parameter estimation.

Note that

$$\begin{aligned} \alpha(z)y_1(t) &= [y_1(t-1), y_1(t-2), \dots, y_1(t-n)]; \\ \alpha(z)y_2(t) &= [y_2(t-1), y_2(t-2), \dots, y_2(t-n)] \end{aligned} \quad (26)$$

meaning that

$$\begin{aligned} \sum_{t=1}^N (\alpha(z)y_1(t))^T (\alpha(z)y_1(t)) &= \begin{bmatrix} R_{y_1}(0) & R_{y_1}(1) & \cdots & R_{y_1}(n) \\ R_{y_1}(1) & R_{y_1}(0) & \cdots & R_{y_1}(n-1) \\ \vdots & \vdots & \cdots & \vdots \\ R_{y_1}(n) & R_{y_1}(n-1) & \cdots & R_{y_1}(0) \end{bmatrix}; \\ \sum_{t=1}^N (\alpha(z)y_2(t))^T (\alpha(z)y_2(t)) &= \begin{bmatrix} R_{y_2}(0) & R_{y_2}(1) & \cdots & R_{y_2}(n) \\ R_{y_2}(1) & R_{y_2}(0) & \cdots & R_{y_2}(n-1) \\ \vdots & \vdots & \cdots & \vdots \\ R_{y_2}(n) & R_{y_2}(n-1) & \cdots & R_{y_2}(0) \end{bmatrix} \end{aligned} \quad (27)$$

and

$$\begin{aligned} &\sum_{t=1}^N [\phi_2^T(t)\phi_2(t) - \phi_1^T(t)\phi_1(t)] \\ &= \begin{bmatrix} R_{y_2}(0) - R_{y_1}(0) & R_{y_2}(1) - R_{y_1}(1) & \cdots & R_{y_2}(n) - R_{y_1}(n) \\ R_{y_2}(1) - R_{y_1}(1) & R_{y_2}(0) - R_{y_1}(0) & \cdots & R_{y_2}(n-1) - R_{y_1}(n-1) \\ \vdots & \vdots & \cdots & \vdots \\ R_{y_2}(n) - R_{y_1}(n) & R_{y_2}(n-1) - R_{y_1}(n-1) & \cdots & R_{y_2}(0) - R_{y_1}(0) \end{bmatrix} \\ &\sum_{t=1}^N [\phi_1(t) + \phi_2(t)]^T u(t) = E \begin{bmatrix} y_1(t-1) + y_2(t-1) \\ y_1(t-2) + y_2(t-2) \\ \vdots \\ y_1(t-n) + y_2(t-n) \end{bmatrix} u(t) = \begin{bmatrix} R_{uy_2}(1) + R_{uy_1}(1) \\ R_{uy_2}(2) + R_{uy_1}(2) \\ \vdots \\ R_{uy_2}(n) + R_{uy_1}(n) \end{bmatrix} \end{aligned} \quad (28)$$

Then Equation (25) is simplified as

$$\hat{\theta} = \begin{bmatrix} R_{y_2}(0) - R_{y_1}(0) & R_{y_2}(1) - R_{y_1}(1) & \cdots & R_{y_2}(n) - R_{y_1}(n) \\ R_{y_2}(1) - R_{y_1}(1) & R_{y_2}(0) - R_{y_1}(0) & \cdots & R_{y_2}(n-1) - R_{y_1}(n-1) \\ \vdots & \vdots & \cdots & \vdots \\ R_{y_2}(n) - R_{y_1}(n) & R_{y_2}(n-1) - R_{y_1}(n-1) & \cdots & R_{y_2}(0) - R_{y_1}(0) \end{bmatrix}^{-1} \begin{bmatrix} R_{uy_2}(1) + R_{uy_1}(1) \\ R_{uy_2}(2) + R_{uy_1}(2) \\ \vdots \\ R_{uy_2}(n) + R_{uy_1}(n) \end{bmatrix} \quad (29)$$

Also Equation (29) can be rewritten as a matrix form, and set

$$\phi_1 = \begin{bmatrix} \phi_1(1) \\ \phi_1(2) \\ \vdots \\ \phi_1(N) \end{bmatrix}; \phi_2 = \begin{bmatrix} \phi_2(1) \\ \phi_2(2) \\ \vdots \\ \phi_2(N) \end{bmatrix}; u = \begin{bmatrix} u(1) \\ u(2) \\ \vdots \\ u(N) \end{bmatrix} \quad (30)$$

i.e.,

$$\hat{\theta} = (\phi_2^T \phi_2 - \phi_1^T \phi_1)^{-1} (\phi_1 + \phi_2)^T u \quad (31)$$

For the purpose of adaptation, rewrite Equation (20) as

$$\begin{aligned} J(\theta) &= \frac{1}{N} \sum_{t=1}^N e_1^2(t) + \frac{1}{N} \sum_{t=1}^N e_2^2(t) = \frac{1}{2} E_1^T E_1 + \frac{1}{2} E_2^T E_2 = \frac{1}{2} \|E_1\|^2 + \frac{1}{2} \|E_2\|^2 \\ e_1(t) &= [u(t) - \alpha(z)\theta y_1(t)]; e_2(t) = [u(t) + \alpha(z)\theta y_2(t)]; \\ E_1 &= \begin{bmatrix} e_1(1) \\ e_1(2) \\ \vdots \\ e_1(N) \end{bmatrix}; E_2 = \begin{bmatrix} e_2(1) \\ e_2(2) \\ \vdots \\ e_2(N) \end{bmatrix} \end{aligned} \quad (32)$$

The recursive parameter estimations at time instant $k+1$ and k are defined as follows

$$\hat{\theta}_{k+1} = \hat{\theta}_k - \gamma \left[E_1 \frac{\partial E_1}{\partial \theta} + E_2 \frac{\partial E_2}{\partial \theta} \right] / \hat{\theta}_k \quad (33)$$

where γ is a stepsize, and above differentiate expressions are computed.

$$\begin{aligned} \frac{\partial e_1}{\partial \theta} &= -\alpha(z)y_1(t); \frac{\partial e_2}{\partial \theta} = -\alpha(z)y_2(t); \\ \frac{\partial E_1}{\partial \theta} &= -\alpha(z) \begin{bmatrix} y_1(1) \\ y_1(2) \\ \vdots \\ y_1(N) \end{bmatrix}; \frac{\partial E_2}{\partial \theta} = -\alpha(z) \begin{bmatrix} y_2(1) \\ y_2(2) \\ \vdots \\ y_2(N) \end{bmatrix} \end{aligned} \quad (34)$$

So

$$\begin{aligned} \frac{\partial J(\theta)}{\partial \theta} &= E_1 \frac{\partial E_1}{\partial \theta} + E_2 \frac{\partial E_2}{\partial \theta} = \frac{2}{N} \sum_{t=1}^N e_1(t)(-\alpha(z)y_1(t)) + \frac{2}{N} \sum_{t=1}^N e_2(t)(\alpha(z)y_2(t)) \\ \hat{\theta}_{k+1} &= \hat{\theta}_k - \gamma \frac{\partial J(\theta)}{\partial \theta} / \hat{\theta}_k \end{aligned} \quad (35)$$

Combining the recursive expression for the controller parameter (33), our mentioned adaptive direct data driven control is plotted in Figure 5.

4. Nonlinear Case. Above Section 3 concerns on direct data driven control principle and its adaptive form. However, in practice, feed-forward controller $C(z)$ may be nonlinear form, so this section considers the problem of nonlinear direct data driven control.

4.1. Multidimensional Taylor network approximation. Observing Figure 2(a), the feed-forward controller $C(z)$ is a nonlinear controller, i.e., $C(z, e(t))$, and then the input-output relation of nonlinear controller is $u(t) = C(z, e(t))$, $t = 1, 2, \dots, N$. For analysis purpose to expand this nonlinearity $C(z, e(t))$, a multidimensional Taylor network (MTN) structure is proposed to approximate the original nonlinearity $C(z, e(t))$.

To the best of knowledge, multidimensional Taylor network is a new type of fixed structure controller based on Taylor series expansion. Its core concept is to use high order

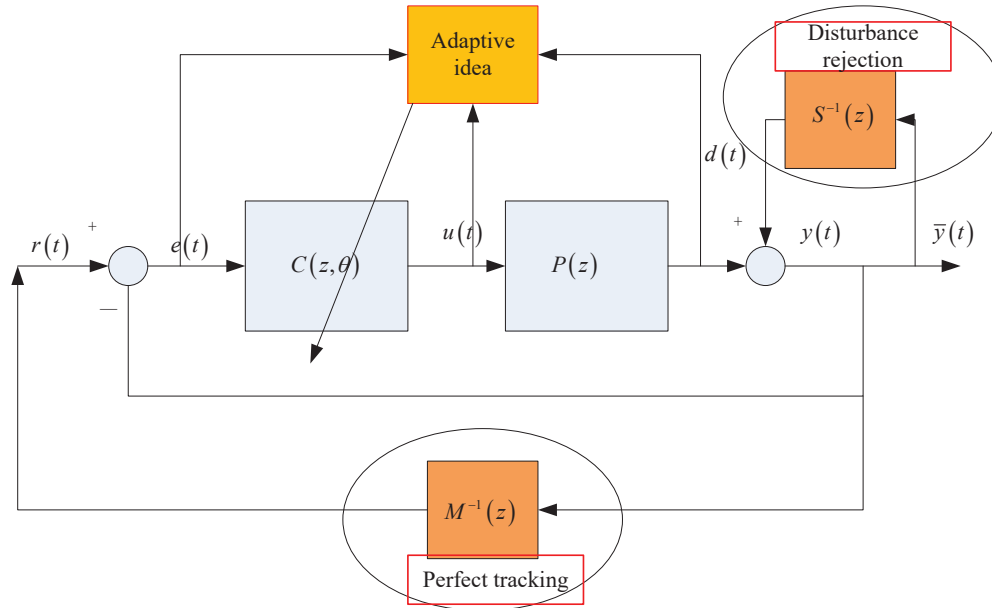


FIGURE 5. Adaptive direct data driven control

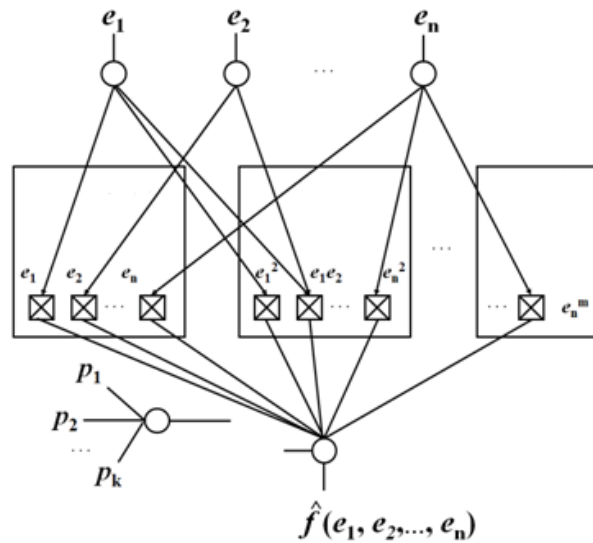


FIGURE 6. Multidimensional Taylor network approximation

nonlinear expansion to achieve the accurate approximation and control accuracy. Multidimensional Taylor network gives a combination of zero order terms, first order terms and high order terms for different high order external input signals and chooses the weighted sum of these terms to represent the consequent control laws. One multidimensional Taylor network is seen in Figure 6, where three layers exist, i.e., input layer, hidden layer and output layer.

More precisely, input layer contains p inputs to receive external information, and hidden layer is responsible for processing the input information of entire network, composed of the combination of p inputs from input layer. Output layer analyzes each layer to obtain the relationship among all nodes in each layer and the corresponding feedback mechanism. Due to space limit, references about MTN can refer to [9-11]. For our MTN approximation, we choose error $e(t)$, its differentiation $\dot{e}(t)$, and integration $\int e(t)$ as three input signals

to excite MTN. Assume the maximal output power is 3, and the total number of terms in the expansion reaches 16, i.e., representing the nonlinear MTN output in an explicit form.

$$\begin{aligned} e_1(t) &= e(t); \quad e_2(t) = \dot{e}(t); \quad e_3(t) = \int e(t); \\ u(t) &= \theta_1 e_1(t) + \theta_2 e_2(t) + \theta_3 e_3(t) + \theta_4 e_1(t)e_2(t) + \theta_5 e_1(t)e_3(t) + \theta_6 e_2(t)e_3(t) \\ &\quad + \theta_7 e_1^2(t) + \theta_8 e_2^2(t) + \theta_9 e_3^2(t) + \theta_{10} e_1(t)e_2^2(t) + \theta_{11} e_1(t)e_3^2(t) + \theta_{12} e_2(t)e_3^2(t) \\ &\quad + \theta_{13} e_1^3(t) + \theta_{14} e_2^3(t) + \theta_{15} e_3^3(t) + \theta_{16} e_1(t)e_2(t)e_3(t) = \varphi(t)^T \theta \end{aligned} \quad (36)$$

where the above nonlinear MTN output $u(t)$ is also a regression form, and the regressor $\varphi(t)$ and weight vector θ are defined as

$$\varphi(t) = [e_1(t) \quad e_2(t) \quad \cdots \quad e_1(t)e_2(t)e_3(t)]^T; \quad \theta = [\theta_1 \quad \theta_2 \quad \cdots \quad \theta_{16}]^T \quad (37)$$

Remark 4.1. Equation (36) tells us that although the MTN output is a nonlinear form, after reformulation, it is simplified into a linear relation with respect to the network weights θ and all nonlinear variables are stacked into the regressor vector $\varphi(t)$. Moreover, as all elements of regressor vector $\varphi(t)$ are composed of $\{e(t), \dot{e}(t), \int e(t)\}$ and $e(t) = r(t) - y(t) = (M^{-1}(z) - 1)y(t)$, based on data $\{u(t), y(t)\}_{t=1}^N$, variables $\{u(t), \varphi(t)\}_{t=1}^N$ are known or collected a priori, thus resulting in a linear least squares problem again.

4.2. Learning algorithm. To improve the approximation accuracy, an easy way is to adjust those unknown network weights, i.e., vector θ through solving the following least squares problem.

$$\begin{aligned} \hat{\theta} &= \arg \min_{\theta} J(\theta) = \arg \min_{\theta} \sum_{t=1}^N [u(t) - \varphi(t)^T \theta]^2 \\ \{u(t), y(t)\}_{t=1}^N &\rightarrow \{u(t), (M^{-1}(z) - 1)y(t)\}_{t=1}^N \rightarrow \{u(t), e(t)\}_{t=1}^N \\ &\rightarrow \{u(t), \varphi(t)\}_{t=1}^N \text{ are all known} \end{aligned} \quad (38)$$

For the above unconstraint optimization problem, the weight estimations are

$$\hat{\theta} = \left[\sum_{t=1}^N \varphi(t)^T \varphi(t) \right]^{-1} \left[\sum_{t=1}^N \varphi(t)^T u(t) \right] \quad (39)$$

Solution (39) is enough from an applied point of view. When to analyze its statistical property, straight forward differentiation gives

$$\frac{\partial J(\theta)}{\partial \theta} = -\frac{2}{N} \sum_{t=1}^N [u(t) - \varphi(t)^T \theta] \varphi(t); \quad \frac{\partial^2 J(\theta)}{\partial \theta^2} = \frac{2}{N} \sum_{t=1}^N \varphi(t)^T \varphi(t) \quad (40)$$

Then the recursive form is

$$\hat{\theta}_{k+1} = (1 - \gamma)\hat{\theta}_k + \gamma \left[\sum_{t=1}^N \varphi(t)^T \varphi(t) \right]^{-1} \left[\sum_{t=1}^N \varphi(t)^T u(t) \right] \quad (41)$$

Set one true weight vector θ^* as

$$\theta^* = \left[\sum_{t=1}^N \varphi(t)^T \varphi(t) \right]^{-1} \left[\sum_{t=1}^N \varphi(t)^T u(t) \right] \quad (42)$$

Then

$$\hat{\theta}_{k+1} = (1 - \gamma)\hat{\theta}_k + \gamma\theta^*; \quad \hat{\theta}_{k+1} - \theta^* = (1 - \gamma)(\hat{\theta}_k - \theta^*) \quad (43)$$

Set network weight error as $\tilde{\theta}_{k+1} = \hat{\theta}_{k+1} - \theta^*$. It holds

$$\tilde{\theta}_{k+1} = \hat{\theta}_{k+1} - \theta^* = (1 - \gamma) (\hat{\theta}_k - \theta^*) = \dots = (1 - \gamma)^{k+1} \tilde{\theta}_0; \tilde{\theta}_0 = \theta_0 - \theta^* \quad (44)$$

Finally, the statistical property of the least squares algorithm (39) is formulated in Theorem 4.1.

Theorem 4.1. *Considering the problem of identifying MTN weights θ , to guarantee the approximation accuracy of the least squares solution in Equation (39), stepsize γ must satisfy $0 < \gamma < 1$. Then we have $\lim_{t \rightarrow \infty} (1 - \gamma)^{k+1} = 0$ and $\tilde{\theta}_{k+1} = \hat{\theta}_{k+1} - \theta^* \rightarrow 0$, i.e., $\hat{\theta}_{k+1} \rightarrow \theta^*$.*

Remark 4.2. *If some constraints about the MTN weights θ are imposed on the least squares problem (38), our previous gradient projection algorithm is beneficial for solving this constrained optimization problem, seeing our paper [12].*

Generally, for clarity of presentation, the above nonlinear adaptive direct data driven control is plotted in Figure 7.

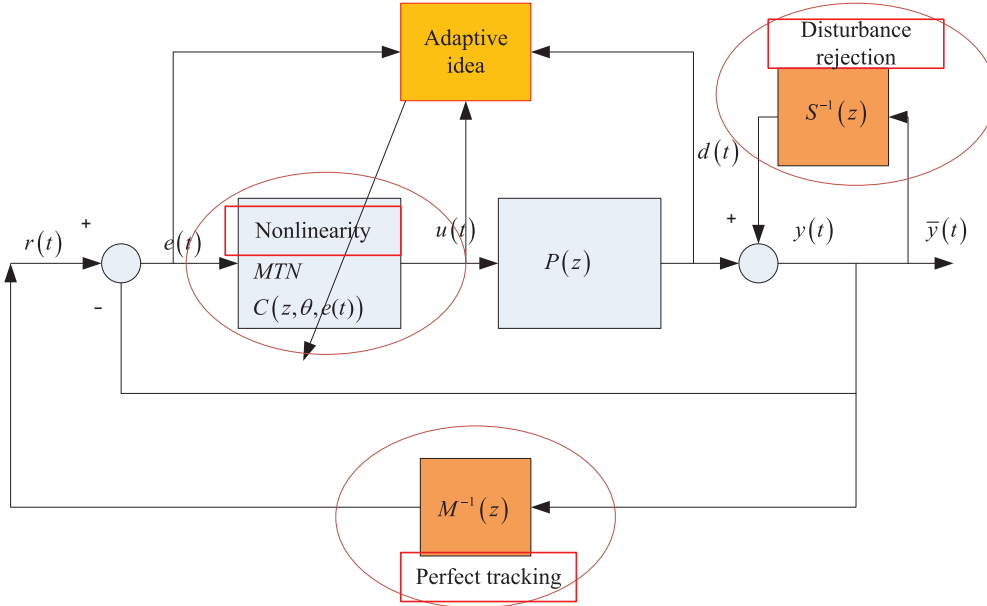


FIGURE 7. Nonlinear adaptive MTN approximation

5. Simulation Example. In this section, a simulation example of the unmanned aerial vehicle (UAV) is given to prove our proposed theoretical analysis.

5.1. Preliminary of UAV modeling. One practical real UAV is seen in Figure 8, which is a real platform for air force in our lab.

Before applying nonlinear adaptive data driven control, one mathematical model of UAV is needed through physical principle or system identification method. For convenience, Figure 9 gives a detailed force effect diagram. Through some forces analysis, we use the classical Lagrangian tool to construct the force balance equation.

More specifically, in Figure 9, F_L is lift, D is drag, Y is side force, and G is gravity. Furthermore, V_{aero} denotes air velocity, V_k is relative speed, and V_g represents speed of aircraft relative to the earth. If the thrust is moved to the center of mass, then the thrust is denoted by T , and the equivalent torque M_T is caused by the thrust movement.



FIGURE 8. Real UAV

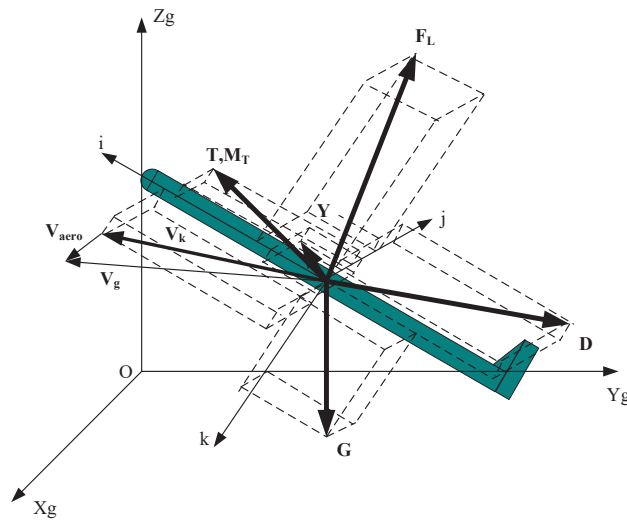


FIGURE 9. Force analysis diagram

The motion of UAV in air can be divided into two parts, i.e., the translational motion of UAV's center of mass and the fixed point rotation of UAV around the center of mass. Under certain assumptions, Newton's second law leads to the vector form of UAV dynamics equation.

$$F = m \frac{dN}{dt} \quad (45)$$

Then the vector forms along three axes are

$$\begin{cases} F = F_x i + F_y j + F_z k; V = ui + vj + wk \\ w = pi + qj + rk; M = Li + Mj + Nk \end{cases} \quad (46)$$

Due to a symmetrical plane on UAV, its dynamical equations are listed as follows.

$$\begin{cases} F_x = m(\dot{u} - vr + wq); F_y = m(\dot{v} - wp + ur); F_z = m(\dot{w} - up + vp) \\ L = I_x \dot{p} - I_{xz} \dot{r} + (I_x - I_y)qr - I_{xz}pq; M = I_y \dot{q} + (I_x - I_z)pr + I_{xz}(p^2 - r^2) \\ N = I_z \dot{r} - I_{xz} \dot{p} + (I_y - I_x)pq + I_{xz}qr \end{cases} \quad (47)$$

Furthermore, the thrust component and deflection angle in the airframe coordinate system satisfy the following geometric relations.

$$T^2 = T_x^2 + T_y^2 + T_z^2; \tan \beta_T = \frac{T_y}{T_x}; \tan \alpha_T = \frac{T_z}{T_x} \quad (48)$$

Then the three axis components of the thrust are expressed as

$$\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} T(1 + \tan^2 \alpha_T + \tan^2 \beta_T) \\ T(1 + \tan^2 \alpha_T + \tan^2 \beta_T) \tan \beta_T \\ T(1 + \tan^2 \alpha_T + \tan^2 \beta_T) \tan \alpha_T \end{bmatrix} \quad (49)$$

In practice, the flight velocity V is chosen as a state variable, and then the relation between flight velocity V and velocity component in body coordinate is as follows.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = S_{ba} \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix} \quad (50)$$

After differentiating both sides of Equation (50), it holds

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = S_{ba} \begin{bmatrix} \dot{V} \\ V\dot{\beta} \\ V\cos\beta\dot{\alpha} \end{bmatrix} = \begin{bmatrix} \dot{V}\cos\alpha\cos\beta - \dot{\beta}V\cos\alpha\sin\beta \\ \dot{V}\sin\beta + \dot{\beta}V\cos\beta \\ V\sin\beta\cos\beta + \dot{\alpha}V\cos\alpha\cos\beta \end{bmatrix} \quad (51)$$

Choose the state variable as

$$X = (V, \beta, \alpha, u, v, w) \quad (52)$$

Then the nonlinear model for UAV is

$$f(\dot{x}, x, u) = 0 \quad (53)$$

Equation (53) corresponds to a dynamical model for only one UAV.

5.2. Control principle. To guarantee UAV to track a given reference trajectory and maintain the desired attitude angle during the entire flight envelope, the control part of the whole flight system is important, while considering some necessary factors. From the above concise UAV modeling, there exist four real input signals and six output signals, thus leading to the considered system cannot reach its desired value for tracking control. In addition, external disturbance and other uncertainties make the controller design process more difficult. When UAV flies in air, we only concern its positioning function and yaw direction, and then a reasonable control principle is to track its three physical variables, i.e., x channel, y channel and z channel, described as follows:

$$x \rightarrow x_d; y \rightarrow y_d; z \rightarrow z_d; \quad (54)$$

where $\{x_d, y_d, z_d\}$ are three desired references for three channels $\{x, y, z\}$, respectively.

When designing UAV controller, a dual feedback control strategy is adopted in Figure 10, where the outer loop is the position loop and inner loop is attitude loop. Specifically, for the position loop, as the pitch angle and roll angle cannot approach their desired values, the position controller in the outer loop is applied in the computation process of so-called virtual attitude in order to obtain the virtual attitude angles that the pitch angle and roll angle want to achieve. The output signal of outer loop is determined by the output from inner loop, and the attitude angle data are yielded from Equations (49)-(51).

5.3. Simulation results. To compare our proposed nonlinear MTN controller and the classical PID controller, their corresponding simulinks from Matlab environment are all shown in Figure 11 and Figure 12, respectively, where these two simulinks include some sub-modules, for example, the signal generator, x position controller, y position controller and z position controller.

Through the computation of MTN controller, learning algorithm is applied to identifying those unknown network weights, thus using the three-layers network to approach the original nonlinear controller. After implementing the above different control simulinks,

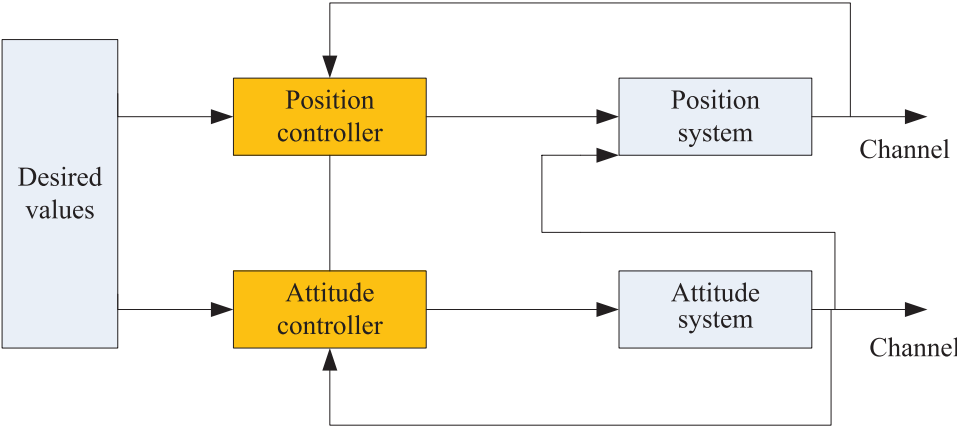


FIGURE 10. UAV flight control system

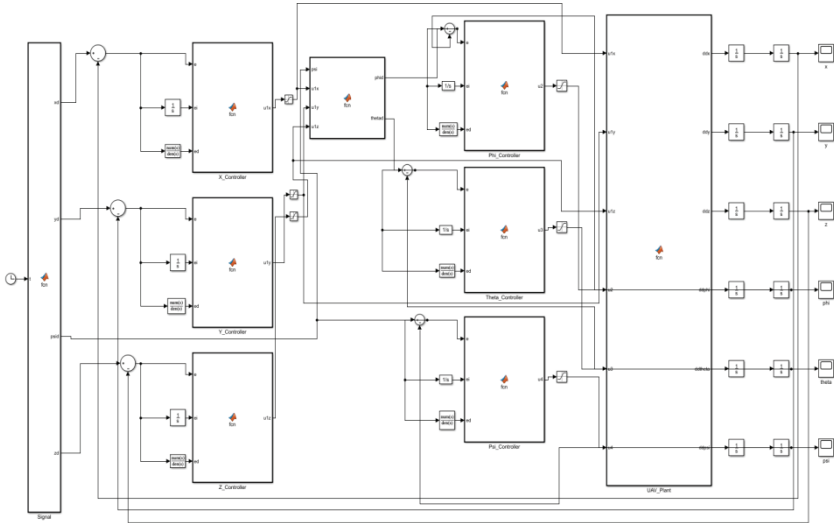


FIGURE 11. PID controller simulink

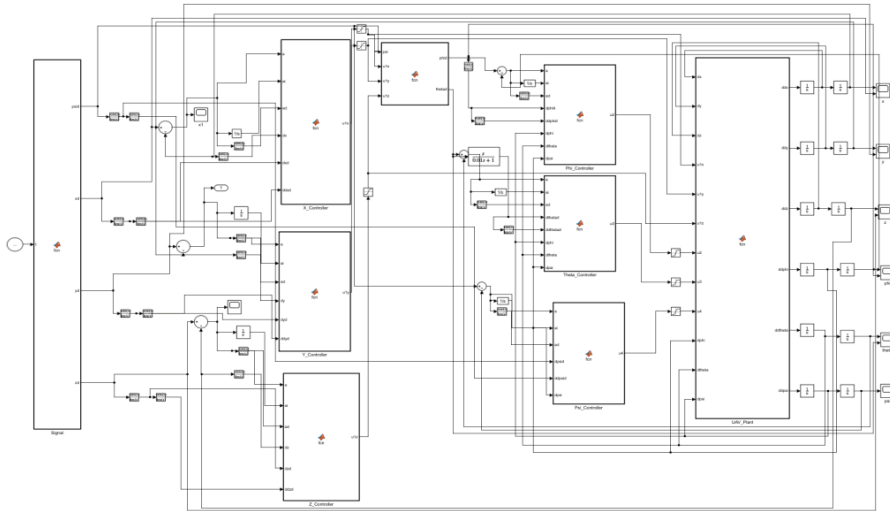


FIGURE 12. MTN controller simulink

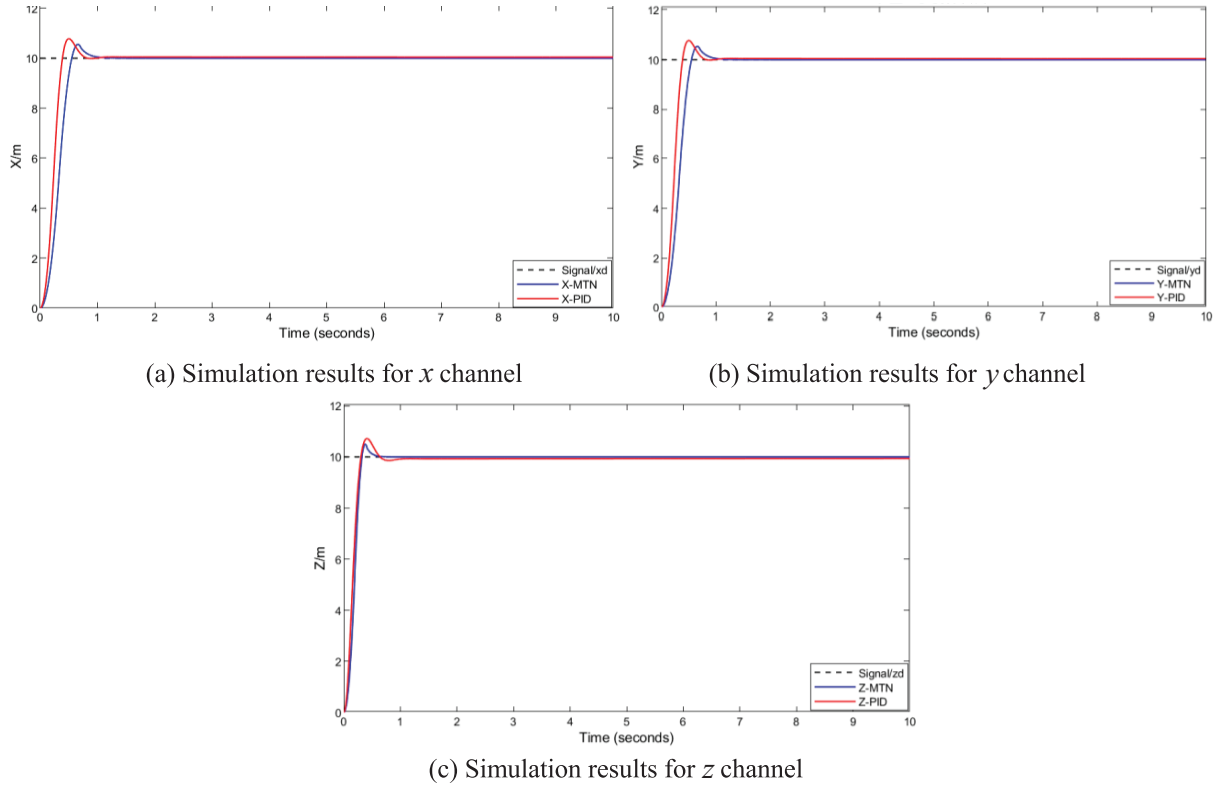


FIGURE 13. Comparisons of PID controller and our MTN controller

the consequent simulation results are plotted in Figure 13, where their comparisons are combined together from different channels, i.e., $\{x, y, z\}$. Observing Figure 13(a), the adjustment time and overshoot for the PID controller are 1.44 s and 7.8%, respectively, which are slightly higher than adjustment time 1.33 s and 5.5% overshoot for the MTN controller. Meanwhile, from the adjustment time, the adjustment speed of the MTN controller is 7.6%, being faster than the PID controller, and the overshoot of the MTN controller is 30.77%, being lower than the PID controller. Also, there are no steady state errors in the MTN controller, but there is a static error 0.04m in the PID controller. Later, from Figure 13(b) and Figure 13(c), simulation results for y channel and z channel for the MTN controller are all better than the PID controller, due to shorter adjustment time, smaller overshoot and no steady state error.

6. Conclusion. Direct data driven control is more widely studied all over academia and industry, due to its advantage of avoiding the complex system modeling process. The combination of nonlinearity and adaptation into direct data driven control strategy is formulated in detail to achieve both control missions, i.e., perfect tracking and disturbance rejection. The proposed controller can be efficiently designed by virtue of learning optimization algorithm, and still benefits from the addition of data science that embodies the network structure. In this way, direct data driven control is to design controller directly only through data, and adaptation means estimate controller parameters recursively, while multidimensional Taylor network remains nonlinear structure and adjusts network weights to approach well through learning algorithm. As a line for future research, the application of the proposed approach to nonlinear systems and more general practical engineering will be considered, e.g., nonlinear differential geometric control, or nonlinear formation flight system and their intersections.

REFERENCES

- [1] S. K. Chiluka et al., A novel robust virtual reference feedback tuning approach for minimum and non-minimum phase systems, *ISA Transactions*, vol.115, no.9, pp.163-191, 2021.
- [2] B. Lakshminarayanan, F. Dettu et al., Inverse supervised learning of controller tuning rules, *Automatica*, vol.178, no.2, pp.52-61, 2025.
- [3] D. Linaro, F. Bizzarri, D. Giudice et al., Continuous estimation of power system inertia using convolutional neural networks, *Nature Communication*, vol.14, no.1, pp.24-35, 2023.
- [4] J. Wang, R. A. Ramirez-Mendoza and Y. Xu, Nonlinear direct data driven control for UAV formation flight system, *Journal of Systems Engineering and Electronics*, vol.34, no.6, pp.1409-1418, 2023.
- [5] S. Proadhan and S. Yuksel, Near optimality of Lipschitz and smooth policies in controlled diffusions, *Systems & Control Letters*, vol.193, 2024.
- [6] A. Köhler, P. Marijan and P. Zhang, Enforcement of current state opacity in signal interpreted Petri nets, *IEEE Transactions on Automatic Control*, vol.69, no.11, pp.8104-8111, 2024.
- [7] A. Chandrasekor, K. Baghdassarian et al., Image based modeling and control for batch processes, *Journal of Process Control*, vol.143, no.11, pp.56-66, 2024.
- [8] C. T. Yilmaz et al., Adaptive model-free disturbance rejection for continuum robots, *Automatica*, vol.171, no.11, pp.74-85, 2025.
- [9] H. Yan and G. Wang, Adaptive tracking control for stochastic nonlinear systems with time varying delays using multidimensional Taylor network, *ISA Transactions*, vol.132, no.3, pp.246-257, 2023.
- [10] A. Kang and H. Yan, Asymptotic tracking and dynamic regulation of MIMI nonaffine nonlinear system with actuator saturation via multidimensional Taylor network controller, *IEEE Transactions on Systems, Man and Cybernetics Systems*, vol.52, no.8, pp.4937-4949, 2022.
- [11] G. Wang, H. Yan and X. Zheng, Contraction based stochastic model predictive control for nonlinear systems with input delay using multidimensional Taylor network, *IEEE Transactions on Automatic Control*, vol.68, no.12, pp.7498-7513, 2023.
- [12] R. Wen, Adaptive safe data driven control strategy for closed loop system, *IEEE Access*, vol.13, no.1, pp.95876-95887, 2025.

Author Biography



Xiao Luo received the Master degree of Engineering from Waseda University, Japan, in 2011 and is presently a Ph.D. student in Waseda University. His current research interests include model predictive control and system identification.



Shigeyuki Tateno received the Ph.D. degree from the Department of Chemical Engineering, Kyushu University, Japan, in 2002. In 2003, he was an associate professor with the Graduate School of Information, Production and Systems, Waseda University, Japan. Since April 2021, he has been a professor at Waseda University. His research interests include fault diagnosis, monitoring systems, wireless communication, and artificial intelligence.



Jianhong Wang received the Master degree in Engineering Cybernetics from Yunnan University, China, in 2007. In 2011, he received the Dr.Sc. degree in College of Automation Engineering from Nanjing University of Aeronautics and Astronautics, China. From 2013 to 2015, he was a postdoctoral fellow in Informazione Politecnico di Milano, Italy. From 2016 to 2018, he was a professor in University of Seville, Spain. From 2019 until now, he has been a professor in Jiangxi University of Science and Technology, China. His current research interests include real-time distributed control, nonlinear control and differential geometry control.