Aerospace Science and Technology 110 (2021) 106483

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Modified Gaussian process regression based adaptive control for quadrotors

Ruping Cen [a], Tao Jiang [a,*], Pan Tang [b]

[a] School of Automation, Chongqing University, Chongqing 400044, China
[b] School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China

ARTICLE INFO

Article history:
Received 16 August 2020
Received in revised form 1 December 2020
Accepted 27 December 2020
Available online 5 January 2021
Communicated by Tsourdos Antonios

Keywords:
Gaussian process regression
Integral modification term
Quadrotor control
Nonparametric representation
Data-driven control

ABSTRACT

Robustness is crucial for flight vehicle control, which is directly related to control performance and mission security. Adaptive control methods are effective for handling perturbations, but most rely on expert knowledge in choosing parametric estimator structures. These uncertainties in quadrotor dynamics own complex representation form, so the structure of the estimator is difficult to determine. Our work integrates modified Gaussian process regression (GPR) with a quaternion-based command filtered backstepping framework, such that quadrotors subjected to perturbations can rapidly and accurately track the desired trajectory. GPR-based control architecture applies Bayesian nonparametric representation to estimate the distribution of perturbations; the nonparametric estimation method requires no prior knowledge about uncertainties and can inherently manage measurement noises. The zero-mean prior of GPR is modified by using the error integral, which compensates for the estimation bias of constant and/or low-frequency disturbances. Furthermore, to estimate the rapidly changing perturbations, a time-dependent form regarding state point selection is applied to increase the influence of recent measurements. Simulation results demonstrate the effectiveness of these modifications and the superiority of the proposed control system.

1. Introduction

Unmanned quadrotors have attracted considerable interest in the research community because of their capabilities: highly agile maneuvering in strictly constrained circumstances, vertical take-off and landing, stable hovering, and so on [1][2]. Quadrotor drones are required to track the desired trajectory accurately and rapidly in many application scenes, such as defense, surveillance, supervision, etc. Recently, numerous nonlinear control techniques have been exploited to address quadrotor control problems, for example, backstepping control [3–8], model predictive control [9][10], sliding mode control [11][12], singular perturbation theory [13], dynamic inverse control [14], Lyapunov-based approach [15], etc. Among these, backstepping is the most convenient and practical framework in that it is flexible enough to adapt to different flight modes, such as position, velocity, attitude, and angular rate control mode [3].

Drones in the flight phase always encounter significant parameter uncertainties, complex nonlinear dynamics, and external perturbations [1][16][17]. Robustness is crucial for flight vehicle control, which is directly relevant to control performance and mission security. Adaptive control is an effective method for amplifying system robustness and recovering nominal performance and has been widely used in practical control systems, for example, in robotic manipulators [18], helicopters [6][19][20], and aerial flight vehicles [21]. A classical adaptive method needs to select the parametric structure for the perturbation estimator, and the parameters are updated during the control process to approximate actual disturbances [22][23]. However, an accurate model of these uncertainties is difficult to determine by classical first-principles techniques, especially for flight vehicle applications. The interaction with unstructured and a priori unknown environments further increases the uncertainty. Neural network (NN) or fuzzy forms of the adaptive estimator can attenuate this problem, but the parameters for the NN or fuzzy structure still need to be specified in advance, and the parameter selection process is time-consuming [18]. Another limitation of adaptive control relates to the convergence speed of the perturbation regression. In certain applications, fast adaptation is needed to achieve stringent tracking performance specifications in the face of high system uncertainties and abrupt changes

* Corresponding author.
E-mail address: jiangtao_1992@outlook.com (T. Jiang).

https://doi.org/10.1016/j.ast.2020.106483
1270-9638/

in system dynamics. Large adaptive gains can speed up the convergence, but they amplify the effect of measurement noise and cause high-frequency oscillations in the control response. Some recent works have tried to tackle this limitation via low-frequency learning [24], filtered regression [25][26], and multi-step learning [27].

In this work, we use Gaussian process regression (GPR), which is a data-driven learning approach. As a supervised learning technique, GPR possesses several advantages [28][29]. It requires only a minimum of prior knowledge to represent an arbitrary complex function, generalizes well even for small training datasets, and has a precise trade-off between fitting the data and filtering the noise measurement. GPR is a non-parametric Bayesian learning method for system identification that does not require a priori knowledge about the structure of the perturbations [30][31].

Recent work employs the Gaussian process (GP) in model reference adaptive control (MRAC) to address the limitation of the radial basis function network (RBFN) with preallocated centers [32]. The works [33] and [34] applied a GPR-based control (GPRC) approach to achieving trajectory tracking for a Euler-Lagrange system and strict feedback-form plant. The previous GPR models did not consider constant and/or slowly varying disturbances. The prior mean functions of these GPR models were set to zeros, and their training sets were sampled off-line. However, for general application scenes, time-varying disturbances do exist, and these disturbances reduce the estimation accuracy of GPR.

Our work focuses on modifying the classical GPR so that it can deal with time-varying disturbances online. Due to the time-varying disturbances, the time-dependent point form is used as the input state for the covariance function. The prior mean function for the GP distribution is generated from an integral approximation, which compensates for the loss of GP estimation. The integral term computed online can effectively approximate constant or low-frequency perturbations and is suitable for the role of prior mean function. The real-time update of the GP training model is executed when the estimation error is above the threshold, which is a trade-off between the fitting performance and the computational consumption. Additionally, a filter for the GPR estimation is employed to attenuate the chattering in the control responses caused by the noise measurements and fast adaptation to the time-varying perturbations. When the proposed GPR is applied, the perturbations in the process of the drones' flight can be immediately estimated and further canceled online. The contribution of our work is threefold. Firstly, the nonparametric machine learning GPR technique is applied in the control field to perform an online approximation of unknown time-varying perturbations. Secondly, the proposed GPR-based adaptive control combined with a quaternion-based backstepping framework is employed to address high-performance quadrotor control problems. Thirdly, the priors of the disturbances, given by the integration of errors, are integrated into the GPR framework to enhance estimation performance.

This work is structured as follows. Section 2 describes the problem setting and presents the preliminaries for GPR. Section 3 illustrates the proposed modified GPRC. Section 4 shows the design process for the quadrotor control system and gives the corresponding stability analysis. A numerical simulation is presented in Section 5, and Section 6 provides the conclusion of our work.

2. Preliminaries

2.1. Problem formulation

This section presents the nonlinear dynamics of quadrotors. First, two reference frames are defined as follows: earth reference frame (ERF) $\mathbb{I} = \{Oxyz\}$, which is fixed to the earth; body reference frame (BRF) $\mathbb{B} = \{O_b x_b y_b z_b\}$, whose origin is located at the quadrotor's center of gravity [8]. The dynamic model of helicopter can be described as follows:

$$\dot{\boldsymbol{P}} = \boldsymbol{V} \tag{1}$$

$$\dot{\boldsymbol{V}} = g\boldsymbol{e_3} + (1/m)\mathcal{R}(\boldsymbol{q})\boldsymbol{F} \tag{2}$$

$$\dot{\mathcal{R}}(\boldsymbol{\Theta}) = \mathcal{R}(\boldsymbol{\Theta})S(\boldsymbol{\omega}) \tag{3}$$

$$J\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times J\boldsymbol{\omega} + \boldsymbol{M} \tag{4}$$

where $\boldsymbol{P} \triangleq \begin{bmatrix} x & y & z \end{bmatrix}^T \in \mathbb{R}^3$ and $\boldsymbol{V} \triangleq \begin{bmatrix} u & v & w \end{bmatrix}^T \in \mathbb{R}^3$ refer to the helicopter's position and velocity vector in the ERF, respectively; m is the mass, and g is gravitational acceleration constant; $\boldsymbol{e_3} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \in \mathbb{R}^3$ is a unitary vector; For any vector $\boldsymbol{l} \in \mathbb{R}^3$, $S(\boldsymbol{l})$ is a skew-symmetric matrix so that $\boldsymbol{l} \times \boldsymbol{n} = S(\boldsymbol{l})\boldsymbol{n}$, where $\times$ represents the vector cross product; J represents the approximate inertia matrix

$$J \approx \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \tag{5}$$

The rotation matrix from BRF to ERF is given as follows:

$$R(\boldsymbol{\Theta}) = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix}, \tag{6}$$

where $C_{(\cdot)}$ and $S_{(\cdot)}$ are shorts for $\cos(\cdot)$ and $\sin(\cdot)$, respectively. $\boldsymbol{\Theta} \triangleq \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T \in \mathbb{R}^3$ represent the Euler angles, which include roll, pitch, and yaw angles, respectively. $\boldsymbol{\omega} \triangleq \begin{bmatrix} p & q & r \end{bmatrix}^T \in \mathbb{R}^3$ denote the angular rates in the BRF.

Here, unit quaternion representations are applied to describe the rotational motion to avoid singularity problems [2]. Let $\boldsymbol{q} = \begin{bmatrix} q_w & q_v^T \end{bmatrix}^T \in \mathbb{R}^4$ denote unit quaternion which parameterizes the rotation matrix R, where $q_w \in \mathbb{R}$ and $\boldsymbol{q_v} \in \mathbb{R}^3$ are commonly referred to as the "scalar" and "vector" parts of unit quaternion $\boldsymbol{q}$. Unit quaternion satisfies that $q_w^2 + \boldsymbol{q_v^T q_v} = 1$. The transformation from rotation matrix to unit quaternion is defined as

$$\mathcal{R}(\boldsymbol{q}) = I_3 + 2q_w S(\boldsymbol{q_v}) + 2S(\boldsymbol{q_v}) \tag{7}$$

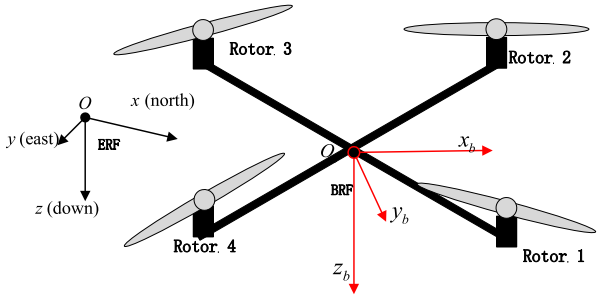

Fig. 1. "X" type quadrotor.

where $I_n \in \mathbb{R}^{n\times n}$ denotes an $n \times n$ identity matrix. Multiplication between two quaternions, $\boldsymbol{q_i} = \begin{bmatrix} q_{i,w} & \boldsymbol{q}_{i,\boldsymbol{v}}^{\boldsymbol{T}} \end{bmatrix}^T \in \mathbb{R}^4$, $i \in 1, 2$, is defined as

$$\boldsymbol{q_1} \otimes \boldsymbol{q_2} = \begin{bmatrix} q_{1,w}q_{2,w} - \boldsymbol{q}_{\boldsymbol{1},\boldsymbol{v}}^{\boldsymbol{T}}\boldsymbol{q}_{\boldsymbol{2},\boldsymbol{v}} \\ q_{1,w}\boldsymbol{q}_{\boldsymbol{2},\boldsymbol{v}} + q_{2,w}\boldsymbol{q}_{\boldsymbol{1},\boldsymbol{v}} + S(\boldsymbol{q}_{\boldsymbol{1},\boldsymbol{v}})\boldsymbol{q}_{\boldsymbol{2},\boldsymbol{v}} \end{bmatrix} \tag{8}$$

With the identity element $\boldsymbol{q_I} = \begin{bmatrix} 1 & \boldsymbol{0}_{\boldsymbol{3}\times\boldsymbol{1}}^{\boldsymbol{T}} \end{bmatrix}^T \in \mathbb{R}^4$, each $\boldsymbol{q} = \begin{bmatrix} q_w & \boldsymbol{q}_{\boldsymbol{v}}^{\boldsymbol{T}} \end{bmatrix}^T \in \mathbb{R}^4$ has an inverse $\boldsymbol{q}^{-1} = \boldsymbol{q}^T = \begin{bmatrix} q_w & -\boldsymbol{q}_{\boldsymbol{v}}^{\boldsymbol{T}} \end{bmatrix}^T \in \mathbb{R}^4$, so that $\boldsymbol{q^{-1}} \otimes \boldsymbol{q} = \boldsymbol{q} \otimes \boldsymbol{q^{-1}} = \boldsymbol{q_I}$. Considering the vector $\boldsymbol{l} \in \mathbb{R}^3$, the rotation action using quaternion is achieved with the double product,

$$v\big(\mathcal{R}(\boldsymbol{q}) \cdot \boldsymbol{l}\big) = \boldsymbol{q} \otimes v(\boldsymbol{l}) \otimes \boldsymbol{q}^{-1} \tag{9}$$

where $v(\boldsymbol{l}) := \begin{bmatrix} 0 & \boldsymbol{l}^{\boldsymbol{T}} \end{bmatrix}^T \in \mathbb{R}^4$ denotes the "lift" operator. If the angular velocity $\boldsymbol{\omega}$ in BRF is known, the derivative of $\boldsymbol{q}$ is represented as follows:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_w \\ \dot{\boldsymbol{q}}_{\boldsymbol{v}} \end{bmatrix} = \frac{1}{2}\boldsymbol{q} \otimes v(\boldsymbol{\omega}) = \frac{1}{2}\begin{bmatrix} -\boldsymbol{q}_{\boldsymbol{v}}^{\boldsymbol{T}} \\ q_w\boldsymbol{I_3} + S(\boldsymbol{q}_{\boldsymbol{v}}) \end{bmatrix}\boldsymbol{\omega} \tag{10}$$

The transformation from unit quaternion to the Euler angle is given as

$$\boldsymbol{\Theta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}(2(q_w q_{v,1} + q_{v,2}q_{v,3}), 1 - 2(q_{v,1}^2 + q_{v,2}^2)) \\ \text{asin}(2(q_w q_{v,2} - q_{v,3}q_{v,1})) \\ \text{atan2}(2(q_w q_{v,3} + q_{v,1}q_{v,2}), 1 - 2(q_{v,2}^2 + q_{v,3}^2)) \end{bmatrix} \tag{11}$$

In (2) and (4), $\boldsymbol{F} \in \mathbb{R}^3$ and $\boldsymbol{M} \in \mathbb{R}^3$ are the external force and torque exerted on fuselage in BRF, respectively, which are given as

$$\boldsymbol{F} = -\begin{bmatrix} 0 & 0 & T \end{bmatrix}^{\mathrm{T}} + \Delta\boldsymbol{F_e}, \quad M = \begin{bmatrix} M_\phi & M_\theta & M_\psi \end{bmatrix}^T + \boldsymbol{\Delta M_e} \tag{12}$$

where T, M_ϕ, M_θ, and M_ψ are the generated thrust and moment from the actuator output. $\Delta\boldsymbol{F_e} \in \mathbb{R}^3$ and $\boldsymbol{\Delta M_e} \in \mathbb{R}^3$ are the external force and moment disturbances, involving wind gust disturbance, the drag of fuselage, flight velocity, the unmodeled rotor dynamics, and so on.

Aiming at quadrotors' dynamics, the total lift force and the control torque are generated as follows:

$$\begin{bmatrix} T \\ M_\phi \\ M_\theta \\ M_\psi \end{bmatrix} = A_p A_a \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = A_p A_a \boldsymbol{U} = A_p \begin{bmatrix} u_t \\ u_\phi \\ u_\theta \\ u_\psi \end{bmatrix} \tag{13}$$

where u_i $(i = 1, 2, 3, 4)$ are the input of motors. $A_p \in \mathbb{R}^{4\times4}$ is the matrix that reflects system's characteristics, including its size, servo mounting position, motor characteristics, etc. $A_a \in \mathbb{R}^{4\times4}$ is control allocation matrix which depends on the type of aircraft. Considering "X" type quadrotor equipped with four rotors (see Fig. 1), it gives

$$A_p \approx \begin{bmatrix} k_f & 0 & 0 & 0 \\ 0 & k_f l_1 & 0 & 0 \\ 0 & 0 & k_f l_2 & 0 \\ 0 & 0 & 0 & k_q \end{bmatrix}, \quad A_a = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{14}$$

where k_f and k_q are the coefficients that reflects the relationship from motors' throttle input (Electrical Speed Controller, ESC) to thrust and torque generated by propellers, respectively; l_1 and l_2 are the constant length relative to the size of quadrotor helicopter; u_i is the bounded throttle input of the ith motor; $\begin{bmatrix} u_t & u_\phi & u_\theta & u_\psi \end{bmatrix}^T \in \mathbb{R}^4$ denote the inputs to control each individual control channel, including altitude, roll, pitch, and yaw angle channels. Combining (2), (4) with (13), one has

$$\begin{aligned} \dot{\boldsymbol{V}} &= g\boldsymbol{e_3} - R(\boldsymbol{\Theta})\begin{bmatrix} 0 \\ 0 \\ \frac{k_f}{m} \cdot u_t \end{bmatrix} + \frac{\boldsymbol{\Delta F_e}}{m} = g\boldsymbol{e_3} - R(\boldsymbol{\Theta})\begin{bmatrix} 0 \\ 0 \\ k_{t0} \cdot u_t \end{bmatrix} + \boldsymbol{d_V}(x,t) \\ \dot{\boldsymbol{\omega}} &= -J^{-1}(\boldsymbol{\omega} \times J\boldsymbol{\omega}) + J^{-1}\boldsymbol{\Delta M_e} + \begin{bmatrix} J_{xx}^{-1}k_f l_1 \cdot u_\phi \\ J_{yy}^{-1}k_f l_2 \cdot u_\theta \\ J_{zz}^{-1}k_q \cdot u_\psi \end{bmatrix} = \begin{bmatrix} k_{\phi 0} \cdot u_\phi \\ k_{\theta 0} \cdot u_\theta \\ k_{\psi 0} \cdot u_\psi \end{bmatrix} + \boldsymbol{d_\omega}(x,t) \end{aligned} \tag{15}$$

where k_{t0}, $k_{\phi 0}$, $k_{\theta 0}$, and $k_{\psi 0}$ are the approximation of control coefficients. They can be identified by the simple flight experiments. $\boldsymbol{d_V}(x,t) \in \mathbb{R}^3$ and $\boldsymbol{d_\omega}(x,t) \in \mathbb{R}^3$ denote the unknown time-varying state-dependent disturbances generated by the external force and moment, where $\boldsymbol{x} = [V^T, q_V^T, \omega^T]$ denotes the concatenate of the related quadrotor states.

By introducing the immediate variable, the dynamics (15) can be further expressed as

$$\begin{aligned} \dot{\boldsymbol{V}} &= g\boldsymbol{e_3} + \boldsymbol{U_V} + \boldsymbol{d_V}(x,t) \\ \dot{\boldsymbol{\omega}} &= \boldsymbol{U_\omega} + \boldsymbol{d_\omega}(x,t) \end{aligned} \tag{16}$$

where the control signals $\boldsymbol{U_V} = -R(\boldsymbol{\Theta})\boldsymbol{e_3}k_{t0} \cdot u_t$, $\boldsymbol{U_\omega} = K_{\omega 0}\begin{bmatrix} u_\phi & u_\theta & u_\psi \end{bmatrix}^T$, $K_{\omega 0} = diag(k_{\phi 0}, k_{\theta 0}, k_{\psi 0})$.

In this work, aiming at the quadrotor system under model uncertainties, external disturbances, the main objective is to design a nonlinear controller to steer quadrotor system states to track the reference trajectory $\boldsymbol{P_d}(\boldsymbol{t}) \in \mathbb{R}^3$ such that tracking errors converge to a small region of zeros. The following assumptions are given for the later control design.

Assumption 1. The desired trajectory $\boldsymbol{P_d}(\boldsymbol{t})$, and its k-order time derivative $\boldsymbol{P_d^{(k)}}(\boldsymbol{t})$ is continuous and bounded with $k = 1, \ldots, 4$.

Assumption 2. The disturbance function $d_i(x,t)$ and its partial derivatives with respect to $(\boldsymbol{x}, t)$ are continuous and bounded, i.e. $\|d_i(\boldsymbol{x},t)\| < b_f$, $\|\frac{\partial d_i(\boldsymbol{x},t)}{\partial t}\| \le c_f$, and $\|\frac{\partial d_i(\boldsymbol{x},t)}{\partial x}\| \le c_{fx}$, where b_f, c_f, and c_{fx} are positive constants and $i = \{\boldsymbol{V}, \boldsymbol{\omega}\}$.

Remark 1. Generally, practical system models are uncertain due to the large variety of system parameters and the time-varying external environment. In accordance with the characteristic of direct adaptive tracking control, the reference signal $\boldsymbol{P_d}(\boldsymbol{t})$ and its time derivatives $\boldsymbol{P_d^{(k)}}(\boldsymbol{t})$, $k = 1, \ldots, 4$ are assumed to be bounded in Assumption 1. Assumption 2 implies the external disturbances $\boldsymbol{d}_i(\boldsymbol{t})$, $i = \{\boldsymbol{V}, \boldsymbol{\omega}\}$ and their derivative are bounded which is reasonable in most of practical scenarios for quadrotor helicopters [5]. Additionally, these assumptions are widely used in disturbance elimination based or adaptive control to guarantee the convergence of the estimation errors. Due to the time-varying environment (the effects of wind gusts) and the existence of model bias (such as unmodeled rotor dynamics, and the dependency on system state and configuration parameters), the structures of these uncertainties are complex, so classical adaptive control with a specific approximation structure is unsuitable for estimating the uncertainty terms rapidly and precisely. This work applies the GPR method with a nonparametric model online to learn the local uncertainty structure and approximate unknown uncertainties.

2.2. Gaussian process regression

A GP, denoted as $\mathcal{GP}(m,k)$, is fully specified by the mean function $m_i : \mathbb{R}^l \mapsto \mathbb{R}$, $i \in \{1, .., n\}$ and kernel $k_i : \mathbb{R}^l \times \mathbb{R}^l \mapsto \mathbb{R}$ as a measurement of the correlation of two states (ϕ, ϕ'). A GP is a nonparametric regression tool to approximate a vector-valued and nonlinear function $\boldsymbol{f}(\phi)$ with $\boldsymbol{f} : \mathbb{R}^l \mapsto \mathbb{R}^n$ using (potentially noisy) system measurements [33]. The measurement $\boldsymbol{y} \in \mathbb{R}^n$, such that $\boldsymbol{y} = \boldsymbol{f}(\phi) + \eta$, where η denotes a σ-sub-Gaussian noise. Examples of such noise are found in Gaussian and uniform distributions.

The training data $\mathcal{D} = \{\Phi, Y\}$ consists of N function evaluations at $\Phi = [\phi^{\{1\}}, \ldots, \phi^{\{N\}}] \in \mathbb{R}^{l \times N}$, with output values $Y = [y^{\{1\}}, \ldots, y^{\{N\}}] \in \mathbb{R}^{n \times N}$. The prediction of each component of $\boldsymbol{f}$ at test input $\phi^* \in \mathbb{R}^l$ is derived from a Gaussian joint distribution. The conditional Gaussian distribution is defined as a normal distribution $\mathcal{N}(\mu_*, var_*)$ with mean and variance as follows

$$\begin{aligned} \mu_*\big(f_i \mid \phi^*, \mathcal{D}\big) &= m_i\big(\phi^*\big) + k_i\big(\phi^*, \Phi\big)^T \big(K_i + I\sigma_i^2\big)^{-1}\big(Y_{:,i} - m_i(\Phi)\big) \\ var_*\big(f_i \mid \phi^*, \mathcal{D}\big) &= k_i\big(\phi^*, \phi^*\big) - k_i\big(\phi^*, \Phi\big)^T \big(K_i + I\sigma_i^2\big)^{-1} k_i\big(\phi^*, \Phi\big) \end{aligned} \tag{17}$$

The matrix function $K_i \in \mathbb{R}^{N \times N}$ is called the covariance matrix with $[K_i]_{j,l} = k_i(\phi^{\{j\}}, \phi^{\{l\}})$, where each element represents the covariance between two points in the training set Φ. The vector-valued covariance function $k_i(\phi^*, \Phi) \in \mathbb{R}^N$ calculates the covariance between the test point ϕ^* and the training set Φ, i.e., $[k_i]_j = k_i(\phi^*, \phi^{\{j\}})$. The only restriction on the kernel function is that the covariance matrix in (3) must be positive definite [29].

For the general case, we assume that no prior system knowledge about the function is available, so the best guess for the mean function m is at the midpoint of the set of real numbers, i.e., 0. The mean function is rewritten as

$$\mu\big(f_i \mid \phi^*, \mathcal{D}\big) = k_i\big(\phi^*, \Phi\big)^T \big(K_i + I\sigma_i^2\big)^{-1} Y_{:,i} \tag{18}$$

The m components of $\boldsymbol{f} \mid \phi^*, D$ are combined into a multivariable Gaussian distribution with

$$\begin{aligned} \mu\big(\boldsymbol{f} \mid \phi^*, \mathcal{D}\big) &= \big[\mu\big(f_1 \mid \phi^*, \mathcal{D}\big), \ldots, \mu\big(f_n \mid \phi^*, \mathcal{D}\big)\big]^T \\ \Sigma\big(\boldsymbol{f} \mid \phi^*, \mathcal{D}\big) &= \mathrm{diag}\big(var\big(f_1 \mid x^*, \mathcal{D}\big), \ldots, var\big(f_n \mid x^*, \mathcal{D}\big)\big) \end{aligned} \tag{19}$$

The regression of a GP model will depend, in part, on the choice of covariance function. The reproducing kernel Hilbert space (RKHS) is a Hilbert space induced by the covariance function k_i, and it contains elements of the form $\sum_{j=1}^{\infty} \alpha_j k_i(\phi^{\{j\}}, \cdot)$, $\alpha_j \in \mathbb{R}$. The respective norm $\|\cdot\|_{k_i}$ measures smoothness relative to the kernel function k_i.

Assumption 3. The function $\boldsymbol{f}$ has a bounded RKHS norm with respect to a known kernel k_i, i.e., $\|f_i\|_{k_i} \le B_{f_i} < \infty$, $i = 1, \ldots, n$.

Because the kernel k_i encodes information about f_i, Assumption 3 implies that the kernel k_i can adequately approximate the function f_i. In this work, the often-employed squared-exponential function is chosen as the covariance function

$$k_i(\phi, \phi') = \exp\left(\frac{1}{2}(\phi - \phi')M_i^{-1}(\phi - \phi')\right) \tag{20}$$

where the diagonal matrix M_i corresponds to the rate of change of the function f_i with respect to ϕ. Squared-exponential kernels are universal kernels that possess the characteristic that the members of the corresponding RKHS can uniformly approximate continuous functions precisely on compact sets. Other examples of universal kernels include spline and Laplacian kernels. An exhaustive discussion of kernels and their respective properties is given in [28][29].

Remark 2. Actually, the task of selecting the appropriate kernel function k_i is simpler than choosing a parametric structure in nonlinear system identification methods, because kernels pose significantly fewer restrictions than parametric structures [34]. Additionally, the hyper-parameters in the above analysis, including the measurement noise σ_i and length scale $[M]_{(i,i)}$, can be learned from the observed data by solving a maximum log-likelihood problem using efficient gradient-based optimization algorithms such as conjugate gradients [29].

The convergence of the trained GP model is given by the following theorem.

Lemma 1 *([31]). Consider an uncertain system and a trained GP model that satisfies Assumption 2. The model error is bounded by*

$$P\left\{\left\|\mu\left(f_i \mid \phi^*, \mathcal{D}\right) - f_i\left(\phi^*\right)\right\| \geq \beta_i\sigma_i\right\} \leq \delta \tag{21}$$

where the state point $\phi^* \in \mathbb{R}^l$ satisfies $\|f_i(\phi^*)\|_{k_i} \leq B_{f_i}$ with $\delta \in (0,1)$. β_i, $\gamma_N^i \in \mathbb{R}$ and $\beta_i = B_{f_i} + 4\sigma_i\sqrt{\gamma_N^i + 1 + \ln(\frac{1}{\delta})}$. The maximum information gain γ_N^i is given as

$$\gamma_N^i = \max_{\widetilde{\phi}^{\{1\}},\ldots,\widetilde{\phi}^{\{N\}}} \frac{1}{2}\log\left|I + \sigma_i^{-2}\widetilde{K}_i\right| \tag{22}$$

where $[\widetilde{K}_i]_{jk} = k_i(\widetilde{\phi}^{\{j\}}, \widetilde{\phi}^{\{k\}})$ and $|\cdot|$ represent the determinant operator.

When this result is extended to a multivariable system, it yields the following lemma.

Lemma 2. *Consider the function $\boldsymbol{f} : \mathbb{R}^l \to \mathbb{R}^n$, which satisfies Assumption 3. The following then holds with a probability of at least $1 - \delta_h$,*

$$\left\|\mu\left(f_i \mid \phi^*, \mathcal{D}\right) - f_i\left(\phi^*\right)\right\| \leq \beta_i\sigma_i, \quad \forall i = 1, \ldots, n \tag{23}$$

where $\delta_h = n\delta$, $\|f_i(\phi^*)\|_{k_i} \leq B_{f_i}$, and $\beta_i = B_{f_i} + 4\sigma_i\sqrt{\gamma_N^i + 1 + \ln(\frac{1}{\delta})}$.

Proof. Due to Lemma 1, $P\{\|\mu(f_i \mid \phi^*, \mathcal{D}) - f_i(\phi^*)\| \geq \beta_i\sigma_i\} \leq \delta$, for $i = 1, \ldots, n$. Applying the union bound yields

$$P\left\{\bigcup_{i=1}^{n}\left\|\mu\left(f_i \mid \phi^*, \mathcal{D}\right) - f_i\left(\phi^*\right)\right\| \geq \beta_i\sigma_i\right\} \leq n\delta = \delta_h \tag{24}$$

It implies

$$P\left\{\bigcup_{i=1}^{n}\left\|\mu\left(f_i \mid \phi^*, \mathcal{D}\right) - f_i\left(\phi^*\right)\right\| \leq \beta_i\sigma_i\right\} \geq 1 - \delta_h \tag{25}$$

The proof is thus complete. □

A multivariable GP formulation exists and can approximate each $f(x,t)$ using a single GP model, but such a formulation is quite cumbersome and corresponds to high computational cost relative to employing multiple scalar GPs [33].

3. Modified GPR for disturbance estimation

For the plant of quadrotors in (1), the variables that need to be estimated are $\boldsymbol{f} = [\boldsymbol{d_V^T}(x,t), \boldsymbol{d_\omega^T}(x,t)]^T \in \mathbb{R}^6$. For each entry of the function $f_i \in \mathbb{R}$ where $i = 1, \ldots, 6$, a GP is trained with N data pairs $\mathcal{D} = \{\phi^{\{j\}}, y_i^{\{j\}}\}_{j=1}^N$ of the system. Due to the existence of time-varying perturbations and parameter uncertainties, the input state is selected as $\phi^{\{j\}} = (x^{\{j\}}, \alpha_c t_j) \in \mathbb{R}^{10}$, where $x^{\{j\}} = [V^{\{j\}T}, q_v^{\{j\}T}, \omega^{\{j\}T}]^T \in \mathbb{R}^9$, and α_C is a temperature parameter to weight the effect of the time-varying term. According to (16), the output measurements $y^{\{j\}}$ for the disturbances can be obtained from measuring the derivatives $\dot{x}^{\{j\}}$ by subtracting the known components, and is represented as

$$\boldsymbol{y}^{\{\boldsymbol{j}\}} = \begin{bmatrix} \dot{\boldsymbol{V}}^{\{\boldsymbol{j}\}} - \left(g\boldsymbol{e_3} + \boldsymbol{U}_{\boldsymbol{V}}^{\{\boldsymbol{j}\}}\right) \\ \dot{\boldsymbol{\omega}}^{\{\boldsymbol{j}\}} - \boldsymbol{U}_{\boldsymbol{\omega}}^{\{\boldsymbol{j}\}} \end{bmatrix} \tag{26}$$

In cases where explicit measurements for $\dot{\boldsymbol{V}}^{\{\boldsymbol{j}\}}$ and $\dot{\boldsymbol{\omega}}^{\{\boldsymbol{j}\}}$ are not available, the representations for these signals can be obtained from fixed point smoothers.

Assumption 4. The disturbance of quadrotors $\boldsymbol{f} = [\boldsymbol{d}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t), \boldsymbol{d}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t)]^T \in \mathbb{R}^6$ has a bounded RKHS norm with respect to a known kernel function k_i, i.e., $\|f_i\|_{k_i} \leq B_{f_i} < \infty$, $i = 1, \ldots, n$.

Generally, the prior of the mean function is set to zero, i.e., $m_i(\phi^*) = 0$. Consider the simple case where only a constant disturbance $d(t) = C_d$ is imposed. If there is just one training data point and the squared-exponential kernel function is employed, then it yields

$$\mu\left(f_i \mid \phi^*, \mathcal{D}\right) = k_i\left(\phi^*, \phi^{\{1\}}\right)^T \left(1 + \sigma_i^2\right)^{-1} C_d \tag{27}$$

The conditional distribution of the predicted disturbance produces an estimation of lesser magnitude than its real value ($k_i(\phi^*, \phi^{\{1\}}) < 1$). This phenomenon is caused by the bad prior of the zero-mean function, which is weighted and integrated into the function estimation. In the case where the state point ϕ^* is far from the set Φ, the GPR will lose its role due to the relatively small value of $k_i(\phi^*, \phi^{\{1\}})^T \ll 1$. Furthermore, constant or/and slow time-varying disturbances are extremely common in practical application scenarios. In our work, the constant disturbance estimator is used to compensate for the reduction in the regression, where the constant part of the disturbance f_{ci} is approximately estimated by the adaptive law

$$\dot{\hat{f}}_{ci} = \Gamma_{\mathrm{ci}} E_i \tag{28}$$

where $\Gamma_{\mathrm{ci}} \in \mathbb{R}$ is a constant learning rate. $E_i \in \mathbb{R}$ denotes the ith component of the tracking error $\boldsymbol{E} = [\boldsymbol{V}_{\boldsymbol{e}}^{\boldsymbol{T}}, \boldsymbol{\omega}_{\boldsymbol{e}}^{\boldsymbol{T}}]^T \in \mathbb{R}^6$, where $\boldsymbol{V}_{\boldsymbol{e}} \in \mathbb{R}^3$ and $\boldsymbol{\omega}_{\boldsymbol{e}} \in \mathbb{R}^3$ are the velocity and angular rate-tracking error defined in the next section.

Thus, the prior mean function is set to $m_i = \hat{f}_{ci}$, and the update of the GPR follows (17). The estimator equates to the integral adaptive control, with the constant non-parametrical bias function given as follows

$$\mu\left(f_i \mid \phi^*, \mathcal{D}\right) = \left(1 - k_i\left(\phi^*, \Phi\right)^T \left(K_{\mathrm{i}} + I\sigma_i^2\right)^{-1} 1_{N,1}\right) \hat{f}_{ci} + k_i\left(\phi^*, \Phi\right)^T \left(K_{\mathrm{i}} + I\sigma_i^2\right)^{-1} Y_{:,i} \tag{29}$$

where $1_{i,j}$ is a $i \times j$ matrix where every element is equal to 1. When the point ϕ^* is far from the training set Φ, the estimation is mainly dependent on the adaptive law.

Finally, the adaptive control term is represented as

$$\left[\hat{\boldsymbol{d}}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t), \hat{\boldsymbol{d}}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t)\right]^T = -\mu\left(\boldsymbol{f} \mid \phi^*, \mathcal{D}\right) \in \mathbb{R}^6. \tag{30}$$

Directly following the results of Lemma 2 and Assumptions 2 and 4, it gives that, with a probability of at least $1 - \delta_h$, the estimation errors satisfy the relation

$$\left\|\boldsymbol{d}_{\boldsymbol{V}}(x,t) - \hat{\boldsymbol{d}}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t)\right\| \leq B_d, \quad \left\|\boldsymbol{d}_{\boldsymbol{\omega}}(x,t) - \hat{\boldsymbol{d}}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t)\right\| \leq B_d \tag{31}$$

where B_d is the positive constant related to the probability δ_h, the noise σ, and the approximation between the kernel space and the perturbation functions.

We updated the training set for the proposed GPRC online, which reduced the effect of the time-varying perturbations. After the noise measurements are collected, the training buffer should be updated, which involves adding new data and removing one of the old elements from the set to keep the buffer size constant [30]. There are several schemes for performing the deletion operation. The simplest way is to delete the oldest element in the buffer. Another scheme removes the data with the largest Kullback-Leibler (KL) divergence between the current GP and the alternative GP that is missing this data point [30]. In our work, the state point is selected randomly from the first five data points. This is an easy scheme to implement, and it maintains diversity in the training data. Additionally, when the current approximation error is above the preset threshold, the GPR model is updated by maximizing the likelihood. This treatment provides a trade-off between regression performance and computational resource consumption.

Remark 3. To further attenuate the chattering caused by the fast approximation of perturbations and to smoothen the adaptive process, we cut off the high-frequency response using a filter for the regression result, which is given as $\dot{\hat{\boldsymbol{f}}}(t) = -\alpha_f(\hat{\boldsymbol{f}}(t) - \mu(\boldsymbol{f} \mid \phi^*, \mathcal{D}))$ with a designed filter constant $\alpha_f > 0$. Then, the adaptive control term can be rewritten as $[\hat{\boldsymbol{d}}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t), \hat{\boldsymbol{d}}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t)]^T = -\hat{\boldsymbol{f}}(t)$.

Remark 4. From the Lemma, it can be concluded that the ultimate boundary for the estimation errors is dependent on the noise level (σ_i), the capacity of the kernel function space to express the space of disturbances (B_{fi}), the confidence probability we expect (δ), and the maximum information gain for the current dataset (γ_N^i). Thus, the way to improve the accuracy of the perturbation approximation process is by reducing the measurement noise, choosing a better kernel function, and/or improving the dataset.

4. Quadrotor controller design

The quadrotor controller design is based on the command filtered backstepping framework, which consists of four control loops, including the position loop, velocity loop, angle loop, and angular rate loop. After computing the control commands, the control input allocation module, which depends on the configuration structure of the multi-rotor UAV, generates the motor inputs. Fig. 2 shows the block diagram of our proposed quadrotor control.

4.1. Step 1: position loop

Define the position error as $\boldsymbol{P}_{\boldsymbol{e}} = \boldsymbol{P} - \boldsymbol{P}_{\boldsymbol{d}}$. From (1), the position error dynamics are given as follows:

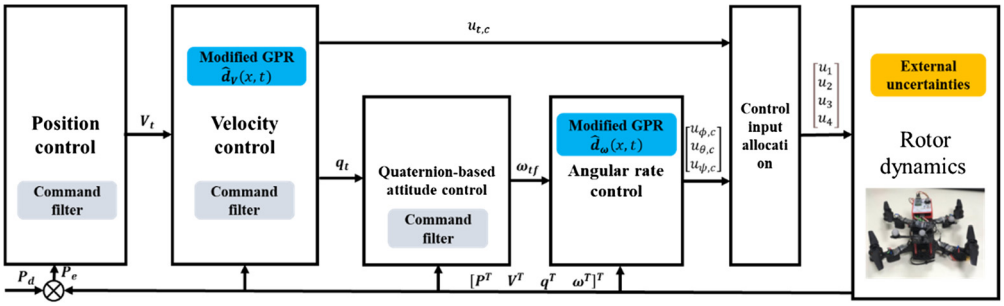

Fig. 2. Control block diagram of quadrotors.

$$\dot{\boldsymbol{P}}_{\boldsymbol{e}} = \boldsymbol{V} - \dot{\boldsymbol{P}}_{\boldsymbol{d}} \tag{32}$$

Design the virtual control

$$\boldsymbol{V}_{\boldsymbol{d}} = \dot{\boldsymbol{P}}_{\boldsymbol{d}} - \boldsymbol{P}_{\boldsymbol{e}}, \tag{33}$$

where K_p is a positive constant. Let the desired velocity command vector pass through the low-pass filter to get the target command and its derivative as follows

$$\dot{\boldsymbol{V}}_{\boldsymbol{t}} = -\boldsymbol{K}_{\boldsymbol{V}_f}(\boldsymbol{V}_{\boldsymbol{t}} - \boldsymbol{V}_{\boldsymbol{d}}) \tag{34}$$

where the constant $K_{V_f} > 0$.

4.2. Step 2: velocity loop

Let velocity error $\boldsymbol{V}_{\boldsymbol{e}} = \boldsymbol{V} - \boldsymbol{V}_{\boldsymbol{t}}$. Following (16), we have

$$\dot{\boldsymbol{V}}_{\boldsymbol{e}} = g\boldsymbol{e}_{\boldsymbol{3}} + \boldsymbol{U}_{\boldsymbol{V}} + \boldsymbol{d}_{\boldsymbol{V}}(x,t) - \dot{\boldsymbol{V}}_{\boldsymbol{t}} \tag{35}$$

The desired control is designed as follows:

$$\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}} = \dot{\boldsymbol{V}}_{\boldsymbol{t}} - g\boldsymbol{e}_{\boldsymbol{3}} - K_{v1}\boldsymbol{V}_{\boldsymbol{e}} - \hat{\boldsymbol{d}}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t) \tag{36}$$

According to the definition of $\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}$, the throttle control signal and the direction of the thrust vector in ERF are expressed as follows:

$$\begin{aligned} u_{t,c} &= \frac{\|\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}\|}{k_{t0}} \\ \boldsymbol{R}_{\boldsymbol{3d}} &= \begin{bmatrix} R_{13,d} & R_{23,d} & R_{33,d} \end{bmatrix}^T = -\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}/\|\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}\| \end{aligned} \tag{37}$$

where $\boldsymbol{R}_{\boldsymbol{3d}} \in \mathbb{R}^3$ represents the desired orientation of drones.

Define the error vector of the thrust direction $\gamma = \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}^T = R_t^T \boldsymbol{R}_{\boldsymbol{3d}}$. $R_t = \mathcal{R}(\boldsymbol{q}_{\boldsymbol{t}})$, where $\boldsymbol{q}_{\boldsymbol{t}}$ denotes the target quaternion. The following quaternion-based command filter is proposed

$$\dot{\boldsymbol{q}}_{\boldsymbol{t}} = \frac{1}{2}\boldsymbol{q}_{\boldsymbol{t}} \otimes v(\boldsymbol{\omega}_{\boldsymbol{t}}) \tag{38}$$

where the target angular rate $\boldsymbol{\omega}_{\boldsymbol{t}} = \begin{bmatrix} \omega_{1t} & \omega_{2t} & \omega_{3t} \end{bmatrix}^T$ is expressed as

$$\omega_{1t} = -K_\gamma \gamma_2, \quad \omega_{2t} = K_\gamma \gamma_1, \quad \omega_{3t} = \omega_{3d} \tag{39}$$

where the constant $K_\gamma > 0$. Considering that length of error vector γ is equal to 1, it requires another freedom to be specified. Hence, ω_{3d} are chosen to be obtained from outside, which is often produced by the real-time remote control or the prescribed trajectory.

Remark 5. Note that there exists a singularity in (37) if $\|\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}\| = 0$. This unusual case may happen when the drone sharply declines. It can be avoided through the following processing:

$$U_{v,d,3} = \text{sat}(U_{v,c,3}, -\infty, -\epsilon_T) \tag{40}$$

where the vertical channel of $\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}$ is constrained by the bound constant $\epsilon_T > 0$. $U_{v,c,3}$ is the third element of the control vector $\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}$. Then it is concluded that $\|\boldsymbol{U}_{\boldsymbol{v},\boldsymbol{d}}\| \geq \epsilon_T$.

Remark 6. The time derivative of $\boldsymbol{\gamma}$ is described as

$$\dot{\boldsymbol{\gamma}} = -S(\omega_t)\boldsymbol{\gamma} + R_t^T \dot{R}_{3d} \tag{41}$$

Define the positive Lyapunov function $V_\gamma = \frac{1}{2}\|\boldsymbol{\gamma_e}\|^2 \leq \gamma_1^2 + \gamma_2^2$, where $\boldsymbol{\gamma_e} = \boldsymbol{e_3} - \boldsymbol{\gamma}$. Its derivative is expressed as

$$\dot{V}_\gamma = -\dot{\gamma}_3 \leq \left(-K_\gamma \gamma_1^2 - K_\gamma \gamma_2^2\right) + \|\dot{\boldsymbol{R}}_{\boldsymbol{3d}}\| \tag{42}$$

It further yields

$$\dot{V}_\gamma \leq -K_{\gamma,1} V_\gamma + \|\dot{\boldsymbol{R}}_{\boldsymbol{3d}}\| \tag{43}$$

If $\|\dot{\boldsymbol{R}}_{\boldsymbol{3d}}\|$ is bounded, it concludes that the error vector γ_e can converge to an arbitrarily small region of e_3 with large enough K_γ, i.e. $\|\boldsymbol{\gamma_e}\| < B_\gamma$.

4.3. Step 3: angle loop

Define the rotation error $R_e = R_t^T \cdot R$. Its unit quaternion representation is given $\boldsymbol{q_e} = \begin{bmatrix} q_{e,w} & \boldsymbol{q_{e,v}^T} \end{bmatrix}^T = \boldsymbol{q_t^T} \otimes \boldsymbol{q}$. Using the property of unit quaternion, its time derivative is given as follows:

$$\dot{\boldsymbol{q}}_{\boldsymbol{e}} = \frac{1}{2}\boldsymbol{q_e} \otimes v\left(\boldsymbol{\omega} - R_e^T \boldsymbol{\omega_t}\right) \tag{44}$$

Design the virtual angular rate $\boldsymbol{\omega_d}$ as

$$\begin{aligned} \boldsymbol{\omega_d} &= R_e^T \boldsymbol{\omega_t} + \boldsymbol{\omega_{e,d}} \\ \boldsymbol{\omega_{e,d}} &= -K_q \cdot \mathrm{sign}(q_{e,w}) \cdot \boldsymbol{q_{e,v}} \end{aligned} \tag{45}$$

The target command is filtered as follows

$$\dot{\boldsymbol{\omega}}_{\boldsymbol{tf}} = -\boldsymbol{K}_{\boldsymbol{\omega f}}(\boldsymbol{\omega_{tf}} - \boldsymbol{\omega_d}) \tag{46}$$

where $\boldsymbol{\omega_{t,f}} \in \mathbb{R}^3$ is the state vector of the filter; $K_{\omega f} > 0$ is a constant design gain.

Remark 7. In the intermediate control law (45), the item $\mathrm{sign}(q_{e,w})$ is used to avoid the unwinding phenomenon of the quaternion-based control [35]. Compared with the attitude control based on Euler angle representation, the quaternion-base control avoids the singular phenomenon. Furthermore, the basic computation in quaternion form is easier and more efficient than the rotation matrix representation.

4.4. Step 4: angular rate loop

Let the angular rate error as $\boldsymbol{\omega_e} = \begin{bmatrix} \omega_{e,1} & \omega_{e,2} & \omega_{e,3} \end{bmatrix}^T = \boldsymbol{\omega} - \boldsymbol{\omega_{tf}}$. From (16), its derivative is calculated

$$\dot{\boldsymbol{\omega}}_{\boldsymbol{e}} = \boldsymbol{U_M} + \boldsymbol{d_\omega}(x,t) \tag{47}$$

The control inputs are obtained as follows:

$$\begin{aligned} \boldsymbol{U_M} &= -K_\omega \boldsymbol{\omega_e} - \hat{\boldsymbol{d}}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t) + \dot{\boldsymbol{\omega}}_{\boldsymbol{tf}} \\ \begin{bmatrix} u_{\phi,c} & u_{\theta,c} & u_{\psi,c} \end{bmatrix}^T &= K_{\omega 0}^{-1} \boldsymbol{U_{M,c}}, \end{aligned} \tag{48}$$

where K_ω is a positive control gain.

4.5. Proof of stability

The closed-loop error dynamics can be divided into the inner-loop part (attitude tracking) and the outer-loop part (position tracking), which are expressed as

The outer-loop part:

$$\begin{aligned} \dot{\boldsymbol{P}}_{\boldsymbol{e}} &= -K_p \boldsymbol{P_e} + \boldsymbol{V_e} + \boldsymbol{V_{ef}} \\ \dot{\boldsymbol{V}}_{\boldsymbol{e}} &= -K_V \boldsymbol{V_e} + \widetilde{\boldsymbol{d}}_{\boldsymbol{V}} - (R_t - R_d)\boldsymbol{e_3} k_{t0} u_t - (R - R_t)\boldsymbol{e_3} k_{t0} u_t \end{aligned} \tag{49}$$

The inner-loop part:

$$\begin{aligned} \dot{\boldsymbol{q}}_{\boldsymbol{e}} &= \frac{1}{2}\begin{bmatrix} -\boldsymbol{q_{e,v}^T} \\ \boldsymbol{q_{e,w}} I_3 + S(\boldsymbol{q_{e,v}}) \end{bmatrix}\left(-K_q \cdot \mathrm{sign}(\boldsymbol{q_{e,w}}) \cdot \boldsymbol{q_{e,v}} + \boldsymbol{\omega_{ef}} + \boldsymbol{\omega_e}\right) \\ \dot{\boldsymbol{\omega}}_{\boldsymbol{e}} &= -K_\omega \boldsymbol{\omega_e} + \widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}} \end{aligned} \tag{50}$$

where $\boldsymbol{V_{ef}} = \boldsymbol{V_t} - \boldsymbol{V_d}$, $\boldsymbol{\omega_{ef}} = \boldsymbol{\omega_{tf}} - \boldsymbol{\omega_d} \in \mathbb{R}^3$, $\widetilde{\boldsymbol{d}}_{\boldsymbol{V}} = \boldsymbol{d_V}(x,t) - \hat{\boldsymbol{d}}_{\boldsymbol{V}}^{\boldsymbol{T}}(x,t)$, $\widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}} = \boldsymbol{d_\omega}(x,t) - \hat{\boldsymbol{d}}_{\boldsymbol{\omega}}^{\boldsymbol{T}}(x,t)$.

The main stability theorem for our work is given as follows.

Theorem 1. *Consider the error dynamics (49) and (50) under disturbances with Assumption 1, 2 & 4, which control law* $\begin{bmatrix} u_{t,c} & u_{\phi,c} & u_{\theta,c} & u_{\psi,c} \end{bmatrix}^T$ *is given as the form in (36) and (48) with the target command filter (34), (38) and (46). If the control gains are selected properly, the origin of the error dynamics is uniformly ultimately bounded.*

Proof. Select the positive Lyapunov function

$$L = \frac{1}{2}\|\boldsymbol{P_e}\|^2 + \frac{1}{2}\|\boldsymbol{V_e}\|^2 + C_a\big(1-|q_{e,w}|\big)^2 + C_a\|\boldsymbol{q_{e,v}}\|^2 + \frac{C_a}{2}\|\boldsymbol{\omega_e}\|^2 \tag{51}$$

where $C_a = (k_{t0}u_t)^2$ is employed to balance the scales between the kinematics and angular dynamics.

For obtaining the time derivative of L, some simplifications and inequality scaling is needed.

$$\begin{aligned}\left(\frac{1}{2}\|\boldsymbol{P_e}\|^2\right)' &= \boldsymbol{P_e^T}(-K_p\boldsymbol{P_e} + \boldsymbol{V_e} + \boldsymbol{V_{ef}}) \\ &\le -(K_P - 0.5)\|\boldsymbol{P_e}\|^2 + \|\boldsymbol{V_e}\|^2 + \|\boldsymbol{V_{ef}}\|^2\end{aligned}$$

Then, following (49), it gives

$$\begin{aligned}\left(\frac{1}{2}\|\boldsymbol{V_e}\|^2\right)' &= \boldsymbol{V_e^T}\big(-K_V\boldsymbol{V_e} + \widetilde{\boldsymbol{d}}_{\boldsymbol{V}} - (R_t - R_d)\boldsymbol{e_3}k_{t0}u_t - (R - R_t)\boldsymbol{e_3}k_{t0}u_t\big) \\ &= -K_V\|\boldsymbol{V_e}\|^2 + \|\boldsymbol{V_e}\|\|\widetilde{\boldsymbol{d}}_{\boldsymbol{V}}\| + \|\boldsymbol{V_e}\|\big\|\big(\mathcal{R}(\boldsymbol{q_e}) - I_3\big)\boldsymbol{e_3}\big\|k_{t0}u_t + \|\boldsymbol{V_e}\|\|\boldsymbol{\gamma_e}\|k_{t0}u_t \\ &\le -(K_V - 0.5)\|\boldsymbol{V_e}\|^2 + \|\widetilde{\boldsymbol{d}}_{\boldsymbol{V}}\|^2 + C_a\big(4\|\boldsymbol{q_{e,v}}\|^2 + \|\boldsymbol{\gamma_e}\|^2\big)\end{aligned}$$

with the positive constants $\gamma_{V_e,2}$, $\gamma_{\xi_t,1}$, and $\gamma_{\xi_t,2}$. Following the properties of the unit quaternion, the following relation is yields

$$\begin{aligned}\big(\big(1-|q_{e,w}|\big)^2 + \|\boldsymbol{q_{e,v}}\|^2\big)' &= 2\,\mathrm{sign}(q_{e,w})\dot{q}_{e,w} \\ &\le -\boldsymbol{K_q}\|\boldsymbol{q_{e,v}}\|^{\boldsymbol{2}} + \|\boldsymbol{q_{e,v}}\|\|\boldsymbol{\omega_e}\| + \|\boldsymbol{q_{e,v}}\|\|\boldsymbol{\omega_{ef}}\| \\ &\le -(K_q - 0.5)\|\boldsymbol{q_{e,v}}\|^{\boldsymbol{2}} + \|\boldsymbol{\omega_e}\|^{\boldsymbol{2}} + \|\boldsymbol{\omega_{ef}}\|^{\boldsymbol{2}}\end{aligned}$$

Considering the error dynamics of the angular rate (50), it follows that

$$\begin{aligned}\big(\|\boldsymbol{\omega_e}\|^2\big)' &= \boldsymbol{\omega_e^T}(-K_\omega\boldsymbol{\omega_e} + \widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}}) \\ &\le -(K_\omega - 1)\|\boldsymbol{\omega_e}\|^{\boldsymbol{2}} + \|\widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}}\|^2\end{aligned}$$

Additionally, $\dot{\boldsymbol{V}}_{\boldsymbol{d}}$, $\dot{\boldsymbol{R}}_{\boldsymbol{3d}}$, and $\dot{\boldsymbol{\omega}}_{\boldsymbol{d}}$ are the function of tracking errors, the high-order derivative of desired trajectory and time derivative of perturbations. Furthermore, when the error satisfies the constraints $\{X_e \mid L < C_1\}$ with $\boldsymbol{X_e} = [\boldsymbol{P_e}, \boldsymbol{V_e}, \boldsymbol{q_{e,v}}, \boldsymbol{\omega_e}]^T \in \mathbb{R}^{12}$, combining with Assumption 1 & 2 gives the $\|\dot{\boldsymbol{V}}_{\boldsymbol{d}}\| \le \sigma_v$, $\|\dot{\boldsymbol{R}}_{\boldsymbol{3d}}\| \le \sigma_R$, $\|\dot{\boldsymbol{\omega}}_{\boldsymbol{d}}\| \le \sigma_\omega$, where σ_v, σ_R and σ_ω are the constants. Following [36] and Remark 7, given any μ_f, when $\mathcal{S} = \{X_e \mid L < C_1\}$ is satisfied after a finite time, there exist sufficiently large gains K_{V_f}, K_γ, and $K_{\omega f}$, such that $\|\boldsymbol{V_{ef}}\|^2 \le \mu_f$, $C_a\|\boldsymbol{\gamma_e}\|^2 \le \mu_f$, $C_a\|\boldsymbol{\omega_{ef}}\|^2 \le \mu_f$, $\forall t \in [0,\infty)$.

Following the property of squared-exponential kernel functions and the boundness of perturbations, one has $\|\widetilde{\boldsymbol{d}}_{\boldsymbol{V}}\| \le B_d$ and $\|\widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}}\| \le B_d$. Combining the above analysis and the relation $(1-|q_{e,w}|)^2 + \|\boldsymbol{q_{e,v}}\|^2 \le 2\|\boldsymbol{q_{e,v}}\|^2$, it concludes

$$\begin{aligned}\dot{L} &\le -(K_P - 0.5)\|P_e\|^2 - (K_V - 1.5)\|\boldsymbol{V_e}\|^2 - C_a(K_q - 4.5)\|\boldsymbol{q_{e,v}}\|^{\boldsymbol{2}} - C_a(K_\omega - 2)\|\boldsymbol{\omega_e}\|^{\boldsymbol{2}} \\ &\quad + \|\boldsymbol{V_{ef}}\|^2 + C_a\|\boldsymbol{\gamma_e}\|^2 + C_a\|\boldsymbol{\omega_{ef}}\|^{\boldsymbol{2}} + \|\widetilde{\boldsymbol{d}}_{\boldsymbol{V}}\|^2 + \|\widetilde{\boldsymbol{d}}_{\boldsymbol{\omega}}\|^2 \\ &\le -C_L L + D_{L1}\end{aligned}$$

with $C_L = 2\min\{(K_P - 0.5), (K_V - 1.5), \frac{(K_q - 4.5)}{4}, (K_\omega - 2.0)\}$ and $D_{L1} = 3\mu_f + 2B_d^2$. To ensure the stability for the closed-loop system, the control gains should be chosen such that $C_L > 0$. One obtains that

$$0 \le L \le \frac{D_{L1}}{C_L} + \left(L(0) - \frac{D_{L1}}{C_L}\right)e^{-\lambda_L t}$$

$\exists t_a > 0$, the condition $L < \frac{C_L}{D_{L1}} + \epsilon_t \le C_1$ for $t > t_a$, where ϵ_t is a small positive constant. Choosing large control gains is a common way to guarantee $\frac{D_{L1}}{C_L} < C_1 - c_\delta$ (i.e. $C_L > \frac{D_{L1}}{C_1 - c_\delta}$) with a small constant $c_\delta > 0$. Thus, it concludes that by selecting proper control gains and filter parameters, the tracking errors of the closed-loop system are uniformly ultimately bounded.

Additionally, following (31), it gives that with probability of at least $1 - \delta_h$,

$$\dot{L} \le -C_L L + D_{L2} \tag{52}$$

where $D_{L2} = 3\mu_f + 2C_{Cd}^2$.

Thus, with probability of at least $1 - \delta_{\mathrm{h}}$, the system's ultimate tracking error boundary is given by (52). By adjusting the nominal control gains, the ultimate position tracking error boundary $(\frac{2C_L}{D_{L2}})^{1/2}$ can be made arbitrarily small with high probability. Additionally, large filter gains and good estimation results may also amplify the tracking accuracy.

This concludes the proof. □

Table 1
Parameters of quadrotor helicopter.

Symbol	Description	Value
m	Mass of quadrotor helicopter	0.56 kg
g	Acceleration of gravity	9.805 m/s^2
J	The moment of inertia matrix	diag$(0.0018 \quad 0.0020 \quad 0.0030)$ kg·m^2
l_1	Arm length in y axis	0.12 m
l_2	Arm length in x axis	0.12 m
k_f	Thrust coefficient of rotor in (14)	6.25 kg·m·s^{-2}
k_q	Torque coefficient of rotor in (14)	0.625 kg·m^2·s^{-2}

Remark 8. For the proposed GPRC, the controller includes the nominal control part and the adaptive part. The selection of the nominal control gain is dependent on the characteristic of the physical plant, and determines on the response (such as the transient response, the overshoot, and so on) for the nominal plant without unknown perturbations. The parameters for regression consist of the kernel scale M_i, covariance for output noise σ_i, filter gain α_f and integral gain Γ_c. The GRP model parameters M_i and σ_i are optimized by maximizing the likelihood of the training set. We first adjust Γ_c, such that the tracking error converges to the preset region. Then, the filter gain α_f is chosen to smoothen the control input and the regression response.

5. Simulation

In this section, numerical simulation results are presented to illustrate the effectiveness of our proposed GPR based control law for quadrotors. Table 1 summarizes the corresponding model parameters of quadrotors. Note that all of these parameters in Table 1 are unknown in our controller design process.

Following [6], lumped disturbances in (12) are introduced in the simulation environment to verify the robustness of the helicopter's control system. These related perturbations are caused by model uncertainties, external disturbances (wind gusts) and the position of the center of mass (CoM).

$$\Delta F_e = \Delta F_w + \Delta F_M, \quad \Delta M_e = \Delta M_w + \Delta M_M \tag{53}$$

Wind disturbance velocity V_w is assumed to possess wind components along $\begin{bmatrix} x & y \end{bmatrix}$ in ERF. The wind is set as $V_w = \begin{bmatrix} 5 & 5 & 0 \end{bmatrix}^T$. The force and moment perturbation caused by wind is expressed as follows:

$$\Delta F_w = -0.5\rho \cdot A_b \cdot {}^B V_{hw}^2 \cdot \text{sign}({}^B V_{hw})$$
$$\Delta M_w = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$$

where ${}^B V_{hw} = R^T \cdot (V - V_w)$ denotes the helicopter's relative airspeed in BRF; ρ is the air density; $A_b = \begin{bmatrix} A_{bx} & A_{by} & A_{bz} \end{bmatrix}^T = \begin{bmatrix} 0.02 & 0.02 & 0.04 \end{bmatrix}^T (m^2)$ is a vector including the quadrotor's effective drag areas along each axis of BRF. The position of CoM in BRF is $P_C = \begin{bmatrix} P_{Cx} & P_{Cy} & 0 \end{bmatrix}^T = \begin{bmatrix} 0.03 & 0.03 & 0 \end{bmatrix}^T$. The force and torque disturbances that it causes are described as

$$\Delta F_M = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$$
$$\Delta M_M = \begin{bmatrix} 0 & k_f P_{Cy} & 0 & 0 \\ 0 & 0 & k_f P_{Cx} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot A_a U$$

The offset of CoM mainly influences the roll and pitch moment by altering the location of the rotors' thrust. The servo dynamics of rotor can be approximated as a first-order filter

$$\dot{u}_{i,a} = \frac{1}{T_r}(u_i - u_{i,a}), \quad i = 1, 2, 3, 4 \tag{54}$$

where $T_r = 0.05$ s is the time constant; $u_{i,a}$ is the actual rotor output.

Before designing the controller of drones, control coefficients of the quadrotor model are approximately given as $k_{t0} = 10$, $k_{\phi 0} = 400$, $k_{\theta 0} = 400$, $k_{\psi 0} = 300$. These parameters are the only we need to know about the quadrotor's model, which can be obtained from the crude and simple identification experiments. The measurement noise is added to further simulate real environment, in which the standard deviation for the position, velocity, attitude, and angular rate channels are selected as $\sigma_P = 0.02$ (m), $\sigma_V = 0.01$ (m/s), $\sigma_a = 0.005$ (rad), $\sigma_\omega = 0.005$ (rad/s), respectively.

The proposed GPR-based controller is given in (36) and (48), with the parameters for each part as follows: $K_P = 1$, $K_V = 3$, $K_q = 10$, $K_\omega = 20$ for the nominal tracking performance of the closed-loop system; $K_{V_f} = K_\gamma = K_{\omega f} = 15$ for the command filter part. The length of the training dataset buffer is 15. At every simulation step, the training data pair $\{y_i\}$ and $(x^{\{j\}}, \alpha_c t_j) \in \mathbb{R}^{10}$ with $\alpha_c = 1.0$ is saved to the training set. Squared exponential kernel functions are employed for the GP model, and they encode infinitely differentiable functions. The GP parameters, including the kernel scale and the noise covariance, are updated online by the maximum likelihood when the current regression error is above the threshold. Additionally, we add the measurement noise with zero mean and covariance $\sigma_1 = \sigma_2 = 10^{-2}$. The gains for the integral errors are $\Gamma_{c1} = \Gamma_{c2} = \Gamma_{c3} = 2$ (for the velocity channel) and $\Gamma_{c4} = \Gamma_{c5} = \Gamma_{c6} = 5$ (for the angular rate channel). Additionally, the filter in the Remark 4 is employed to prevent the chattering phenomenon in GPR, and the filter gains are chosen as $\alpha_f = 5$.

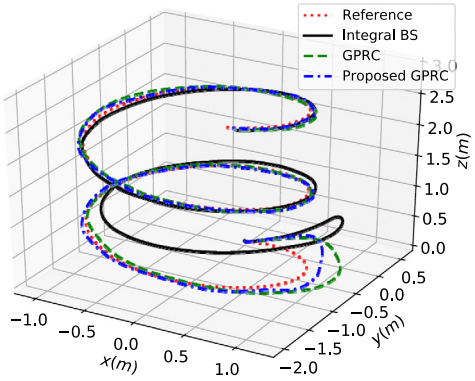

Fig. 3. The response for 3D position of the quadrotor.

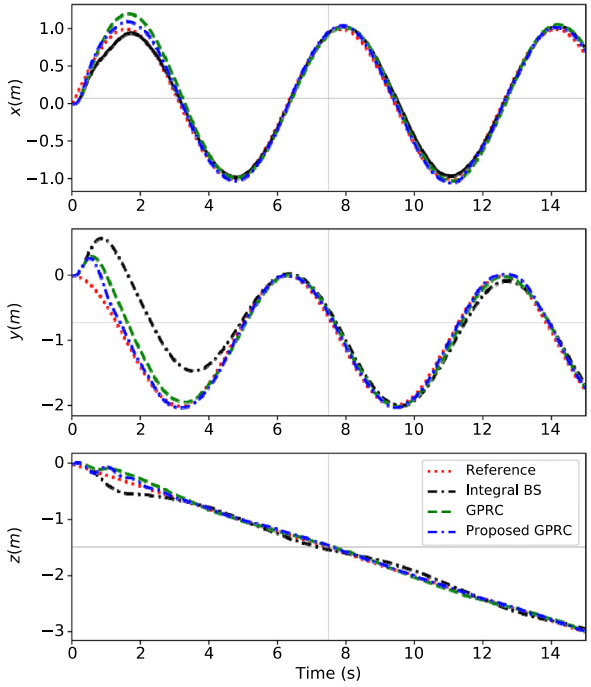

Fig. 4. The response for position tracking.

To further demonstrate the effectiveness of our modification for GPRC, several comparison methods are presented, including 1) classical integral backstepping control (named "integral BS"), 2) classical GPR control (named "GPRC"), and 3) the proposed GPRC with the modification and the filter for GPR estimation (named "Proposed GPRC"). The desired spiral reference trajectory is described as:

$$\boldsymbol{P_d}(t) = \begin{bmatrix} \sin t \\ \cos t - 1 \\ -0.2t \end{bmatrix} (m) \qquad \omega_{3d} = 0 \ (\mathrm{rad/s}). \tag{55}$$

The quadrotor is expected to follow a spiral path while continuing to lift. At this stage, the quadrotor's lift, sideslip, and forward movement are evaluated comprehensively. The simulation was built in a Python environment, which ran on a laptop computer with an Intel i7-8700 CPU. The average single-step processing time of our proposed method was approximately 2.9 ms, which satisfied the requirements for the practical on-line control task.

The simulation results are illustrated in Figs. 3–6. Fig. 3 shows the three-dimensional position response curves of the closed-loop system based on the above three methods. To show the result clearly, some quantitative control results are given in Table 2, including the average error norm for the position tracking and the average error norm for the perturbation estimations $\tilde{d}_v$ and $\tilde{d}_\omega$.

The precision of the position tracking errors follows the sequence *Integral BS* < *GPRC* < *Proposed GPRC*. The proposed modified GPR control framework employs nonparametric estimation to approximate the effects of the unknown perturbations. Compared with integral control, GPR describes more complex dynamic characteristics from the control plants. Furthermore, the proposed modified GPR, which employs the integral terms as the prior for the mean function, offers better disturbance estimation accuracy than the comparative methods. Fig. 5 shows the corresponding disturbance estimation results.

Fig. 6 depicts the corresponding attitude response. The angular responses of our proposed method exhibit a more intense oscillation, which is caused by the fast, high-precision tracking and the effects of measurement noise. Fig. 7 illustrates that the angular rate in the z-

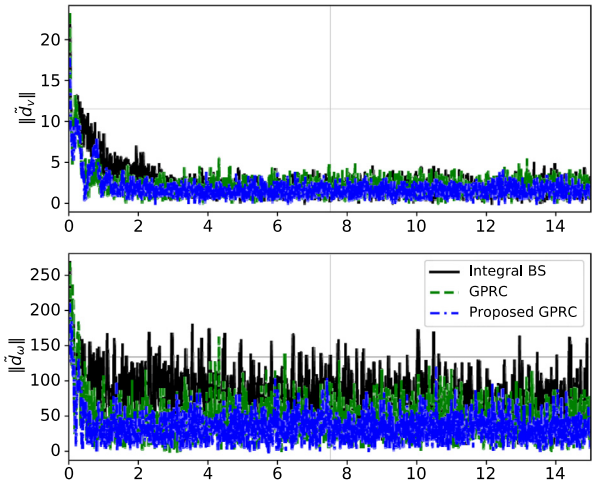

Fig. 5. The results of perturbation estimation.

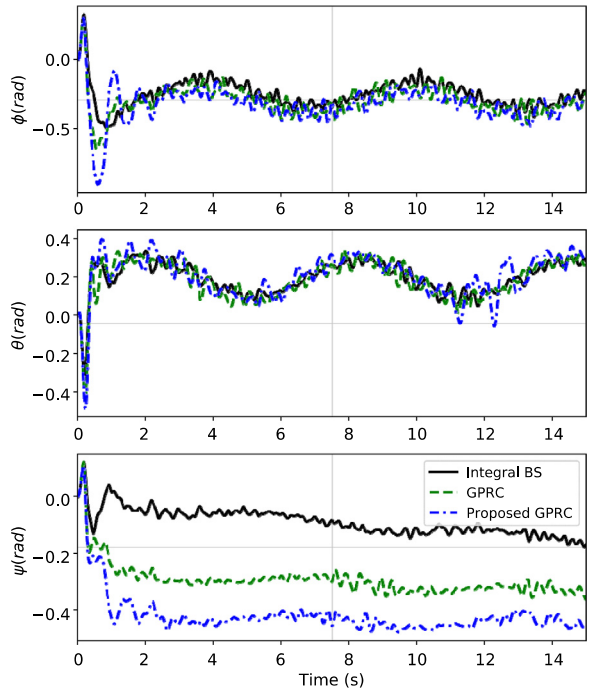

Fig. 6. The response for the attitude (Euler angle representation).

Table 2
The quantification for the simulated controllers.

Average error	Integral BS	GPRC	Proposed GPRC
$\|P_e\|$	0.051	0.032	**0.025**
$\|\tilde{d}_v\|$	3.127	2.526	**2.283**
$\|\tilde{d}_\omega\|$	103.736	66.915	**48.927**

direction can track the desired commands, which verifies the effectiveness of the proposed quaternion-based BS framework for quadrotor control. Finally, the control input commands for different controllers are shown in Fig. 8. The slight chattering phenomenon is caused by the measurement noise and the requirement for rapid perturbation elimination.

6. Conclusion

This work combines nonparametric machine learning GPR with control theory to tackle the control problem of uncertain plants. Quadrotor control is addressed by integrating GPR approximation into a command filtered backstepping framework. The uncertainties existing in quadrotor dynamics are structurally complex and time-variant. Nonparametric and local GPR is suitable for approximating these unknown terms, and some modifications are employed to make it more beneficial in practical online control scenarios. Unlike the weak assumption of the prior mean function in classical GPR, the mean prior function is modified by the online error integral estimation.

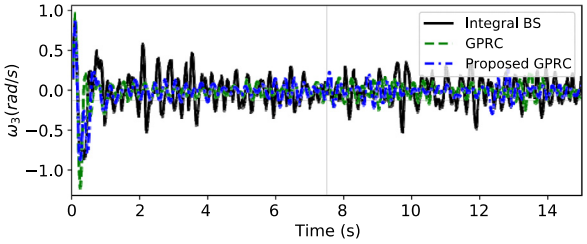

Fig. 7. The response for the angular rate of z channel.

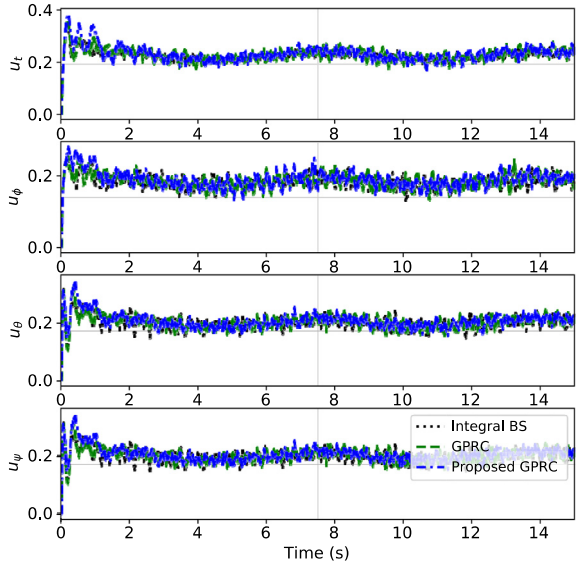

Fig. 8. The control input responses.

Furthermore, a time-dependent form for state points is employed to increase the weight of the most recent measurements. The training sets and parameters for the GPR models are updated online to quickly respond to change. The simulation results demonstrate that the proposed GPR control methodology possesses better perturbation elimination and higher tracking precision.

Notably, the covariance of the estimation gives a measurement for the quality of the regression performance. Our future work focuses on employing covariance information to adaptively adjust the control gains and further verifying GPRC in practical experiments.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. Mahony, V. Kumar, P. Corke, Multirotor aerial vehicles: modeling, estimation, and control of quadrotor, IEEE Robot. Autom. Mag. 19 (3) (2012) 20–32.
[2] Q. Quan, Introduction to Multicopter Design and Control, Springer, Singapore, 2017.
[3] Z. Zuo, Trajectory tracking control design with command-filtered compensation for a quadrotor, IET Control Theory Appl. 4 (11) (2010) 2343–2355.
[4] W. Jasim, D. Gu, Integral backstepping controller for quadrotor path tracking, in: Proceedings of the International Conference on Advanced Robotics, Istanbul, Turkey, 2015.
[5] A. Aboudonia, A. El-Badawy, R. Rashad, Active anti-disturbance control of a quadrotor unmanned aerial vehicle using the command-filtering backstepping approach, Nonlinear Dyn. 90 (1) (2017) 1–17.
[6] T. Jiang, J. Huang, B. Li, Composite adaptive finite-time control for quadrotors via prescribed performance, J. Franklin Inst. (2020).
[7] T. Madani, A. Benallegue, Backstepping control for a quadrotor helicopter, in: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 2006, pp. 3255–3260.
[8] T. Jiang, D. Lin, T. Song, Finite-time backstepping control for quadrotors with disturbances and input constraints, IEEE Access 6 (2018) 62037–62049.
[9] K. Alexis, G. Nikolakopoulos, A. Tzes, Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances, Control Eng. Pract. 19 (10) (2011) 1195–1207.
[10] A. Altan, Ö. Aslan, R. Hacıoğlu, Model predictive control of load transporting system on unmanned aerial vehicle (UAV), in: Proceedings of the Fifth International Conference on Advances in Mechanical and Robotics Engineering, 2017.
[11] T. Madani, A. Benallegue, Backstepping sliding mode control applied to a miniature quadrotor flying robot, in: Proceedings of the Industrial Electronics 2006-32nd Annual Conference, Paris, France, 2006, pp. 700–705.
[12] T. Jiang, T. Song, D. Lin, Integral sliding mode based control for quadrotors with disturbances: simulations and experiments, Int. J. Control. Autom. Syst. 17 (8) (2019) 1987–1998.
[13] S. Bertrand, N. Guénard, T. Hamel, H. Piet-Lahanier, L. Eck, A hierarchical controller for miniature vtol UAVs: design and stability analysis using singular perturbation theory, Control Eng. Pract. 19 (10) (2011) 1099–1108.
[14] A. Das, K. Subbarao, F. Lewis, Dynamic inversion with zero-dynamics stabilisation for quadrotor control, IET Control Theory Appl. 3 (3) (2009) 303–314.
[15] M.-D. Hua, T. Hamel, P. Morin, C. Samson, A control approach for thrust-propelled underactuated vehicles and its application to vtol drones, IEEE Trans. Autom. Control 54 (8) (2009) 1837–1853.

[16] S. Omari, M.D. Hua, G. Ducard, T. Hamel, Hardware and software architecture for nonlinear control of multi-rotor helicopters, IEEE/ASME Trans. Mechatron. 18 (6) (2013) 1724–1736.
[17] G.V. Raffo, M.G. Ortega, F.R. Rubio, An underactuated H_∞ control strategy for a quadrotor helicopter, in: Proceedings of the European Control Conference, Budapest, Hungary, 2009, pp. 3845–3850.
[18] F.L. Lewis, S. Jagannathan, A. Yesildirak, Neural Network Control of Robot Manipulators and Nonlinear Systems, CRC Press, Philadelphia, PA, USA, 1998.
[19] C. Nicol, C.J.B. Macnab, A. Ramirez-Serrano, Robust adaptive control of a quadrotor helicopter, Mechatronics 21 (6) (2011) 927–938.
[20] A. Altan, Ö. Aslan, R. Hacıoğlu, Real-time control based on NARX neural network of hexarotor UAV with load transporting system for path tracking, in: 2018 6th International Conference on Control Engineering & Information Technology (CEIT), IEEE, 2018, pp. 1–6.
[21] W.A. Butt, L. Yan, S.K. Amezquita, Adaptive integral dynamic surface control of a hypersonic flight vehicle, Int. J. Syst. Sci. 46 (10) (2015) 1717–1728.
[22] M. Krstic, P.V. Kokotovic, I. Kanellakopoulos, Nonlinear and Adaptive Control Design, Wiley, New York, USA, 1995.
[23] G. Tao, Adaptive Control Design and Analysis, Wiley, New York, 2003.
[24] T. Yucelen, W.M. Haddad, Low-frequency learning and fast adaptation in model reference adaptive control, IEEE Trans. Autom. Control 58 (4) (2012) 1080–1085.
[25] N. Cho, H. Shin, Y. Kim, A. Tsourdos, Composite model reference adaptive control with parameter convergence under finite excitation, IEEE Trans. Autom. Control 63 (3) (2018) 811–818.
[26] S.B. Roy, S. Bhasin, I.N. Kar, Combined MRAC for unknown MIMO LTI systems with parameter convergence, IEEE Trans. Autom. Control 63 (1) (2018) 283–290.
[27] T. Jiang, D. Lin, T. Song, Composite adaptive control with fast convergence for multilayer neural network, Int. J. Robust Nonlinear Control 29 (13) (2019) 4454–4471.
[28] C.E. Rasmussen, C.K. Williams, Gaussian Processes for Machine Learning, MIT Press, Cambridge, MA, 2006.
[29] C.M. Bishop, et al., Pattern Recognition and Machine Learning, vol. 4, Springer, New York, 2006.
[30] J. Candela, C. Rasmussen, A unifying view of sparse approximate Gaussian process regression, J. Mach. Learn. Res. 6 (2005) 1939–1959.
[31] L. Csató, M. Opper, Sparse on-line Gaussian processes, Neural Comput. 14 (3) (2002) 641–668.
[32] G. Chowdhary, H.A. Kingravi, J.P. How, et al., Bayesian nonparametric adaptive control using Gaussian processes, IEEE Trans. Neural Netw. Learn. Syst. 26 (3) (2015) 537–550.
[33] T. Beckers, D. Kulić, S. Hirche, Stable Gaussian process based tracking control of Euler–Lagrange systems, Automatica 103 (2019) 390–397.
[34] A. Capone, S. Hirche, Backstepping for partially unknown nonlinear systems using Gaussian processes, IEEE Control Syst. Lett. 3 (2) (2019) 416–421.
[35] C.G. Mayhew, R.G. Sanfelice, A.R. Teel, Quaternion-based hybrid control for robust global attitude tracking, IEEE Trans. Autom. Control 56 (11) (2011) 2555–2566.
[36] Y. Pan, H. Yu, Composite learning from adaptive dynamic surface control, IEEE Trans. Autom. Control 61 (9) (2015) 2603–2609.