Adaptive Neural Network Control for a Quadrotor Landing on a Moving Vehicle

Ze Qing[1], Ming Zhu[2],Zhe Wu[3]

1. School of Aeronautic Science and Engineering, Beihang University, Beijing, 100191, P. R. China
E-mail: buaaqingze@buaa.edu.cn

2. School of Aeronautic Science and Engineering, Beihang University, Beijing, 100191, P. R. China
E-mail: zhuming@buaa.edu.cn

3. School of Aeronautic Science and Engineering, Beihang University, Beijing, 100191, P. R. China
E-mail: wuzhe@buaa.edu.cn

Abstract: An autonomous vehicle landing control algorithm of a quadrotor is investigated for the situation when the quadrotor hovers above the vehicle in this paper. To facilitate the controller design, the problem of autonomous landing is converted from general trajectory tracking problem of a quadrotor to a stabilization problem of relative motion. A four-degrees-of-freedom (4-DOF) nonlinear relative motion model with four control inputs is estimated. An adaptive radial basis function neural network (RBFNN) is developed to estimate the unknown disturbance and is applied to design the controller through a backstepping technique. It is proved that all the states in the closed-loop system are uniformly ultimately bounded and the error converges to a small neighborhood of origin. Numerical simulation results illustrate the good performance of the proposed controller.

Key Words: Autonomous Vehicle Landing, Adaptive Control, Neural Network

1 INTRODUCTION

During the past few years, some growing considerable researches on the design and operation of quadrotors have been performed [1]. Compared with the conventional aircraft, the quadrotor has many advantages including hovering capacity, vertical take-off and landing ability [2]. These advantages allow the quadrotor to cope with many military and civil scopes, such as search, rescue, site surveillance, parcel delivery and aerial imaging [3] [4]. However, the quadrotor is challenging to applied in above missions on its own because of its relatively short battery life and short range. Deploying quadrotor from moving vehicles can alleviate this issue [5]. For example, the synergy between quadrotor and vehicle can save time in parcel delivery, search and rescue operation. Therefore, it is of great significance to study the synergistic relationship between the quadrotor and vehicle.

One of the major problem of the synergistic relationship is the autonomous vehicle landing of the quadrotor, which can be realized by trajectory tracking algorithm or relative motion modeling method. Some research groups have considered the problem of landing a quadrotor on a moving vehicle by using trajectory tracking technique. A nonlinear 2D trajectory tracking controller was employed to land a quadrotor on top of a moving vehicle by decomposing the problem in initially tracking and finally descending on the platform in [6]. In reference [7], the autonomous landing problem of a quadrotor on a fixed inclined surface was addressed using lasers to estimate the distance from the quadrotor and inclination of the vehicle, which were used for the generation of a hybrid trajectory. While in another work [8], the case of landing a quadrotor on vertical surface was demonstrated as one of the applications of a method for designing trajectories and controllers for aggressive maneuvers. A time-optimal trajectory generation algorithm for landing a 3-DOF quadrotor model onto a moving vehicle is proposed in [9]. However, the researches about relative motion modeling are small in scale. Relative motion modeling method is generally used in spacecraft docking system to facilitate the controller design [10] [11], inspired by which we can apply relative motion modeling to convert the general trajectory tracking problem of a quadrotor to a stabilization problem of relative motion.

Recently, the learning-based adaptive control methodology using neural network has drawn a lot of attentions for their strong approximate capacity. The adaptive neural network method has been applied in the nonlinear system and performs well [12] [13]. Therefore, the neural network can be constructed to estimate the uncertainty (including the disturbance) [14], in which the parameter update laws all evolved from Lyapunov-based approaches.

The whole autonomous vehicle landing of a quadrotor usually consists of two steps: the approaching stage and the landing stage. In the first step, the quadrotor is controlled to hover above the vehicle. In the second step, the quadrotor chooses appropriate time to land vertically on the vehicle. The disturbance from airflow causes a complex landing environment for the landing stage, so the secend step is the key to the autonomous landing process.

This work was supported in part by the National Key R&D Program of China under grant No. 2016YFB1200100.

978-1-5386-1243-9/18/$31.00

In this paper, we present a controller for autonomous vehicle landing of a quadrotor using relative motion model for the situation when the quadrotor hovers above the vehicle. Main contributions in this paper are summarized as following aspects: 1) A 4-DOF nonlinear relative motion model is established. 2) Based on the proposed model, the backstepping technique is introduced for the adaptive RBFNN control with disturbance. 3) All the states in the closed-loop system are uniformly ultimately bounded based on Lyapunov stability analysis.

The rest structure of this paper is arranged as follows: Problem statement is proposed in Section 2. Section 3 is devoted to designing the landing control algorithm. In Section 4, numerical simulations are performed to verify the designed control approach. All of these are following by conclusion in Section 5.

2 PROBLEM STATEMENT

The relative motion of the quadrotor and the moving vehicle is illustrated in Figure 1. As shown in Figure 1, we define the inertial frame and body-fixed frames for the quadrotor and the vehicle respectively. The inertial frame $E = \{O_e - x_e y_e z_e\}$ is fixed on the Earth. The frame $B_1 = \{O_b^1 - x_b^1 y_b^1 z_b^1\}$ is taken as the body-fixed frame of the quadrotor, whose coordinate origin O_b^1 is the geometric center point of the quadrotor, the $O_b^1 x_b^1$ axis points to the head of the quadrotor, the $O_b^1 y_b^1$ axis is perpendicular to the $O_b^1 z_b^1$ axis and points to right, whereas $O_b^1 z_b^1$ is perpendicular to the plane $O_b^1 x_b^1 y_b^1$ and follow the right-hand rule. Let $B_2 = \{O_b^2 - x_b^2 y_b^2 z_b^2\}$ be the body-fixed frame of the vehicle, which is defined similarly as B_1.

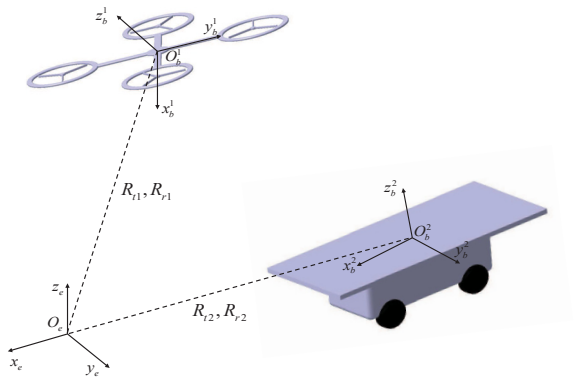

Figure 1: Relative motion of the quadrotor and the vehicle

2.1 Models of the Quadrotor and the Vehicle

In this paper, the controller is designed for the situation when the quadrotor hovers above the vehicle, so that we only consider the problem of relative altitude and attitude, without controlling the position in Oxy plane.

The kinematics of the quadrotor can be expressed as [15]

$$\begin{cases} \dot{z_1} = \boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{V_1} \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{1}} = \boldsymbol{R_{r1}}\boldsymbol{\Omega_1} \end{cases} \tag{1}$$

where z_1 and $\boldsymbol{\eta_1} = [\phi_1, \theta_1, \psi_1]^T$ denote respectively the altitude and the attitude of the quadrotor in E. The attitude angles consist of the roll angle ϕ_1, pitch angle θ_1 and yaw angle ψ_1. Three translation velocities and three rotation velocities in B_1 are expressed as $\boldsymbol{V_1} = [u_1, v_1, w_1]^T$ and $\boldsymbol{\Omega_1} = [p_1, q_1, r_1]^T$ respectively.

$$\boldsymbol{R_{r1}} = \begin{bmatrix} 1 & \tan\theta_1 \sin\phi_1 & \tan\theta_1 \cos\phi_1 \\ 0 & \cos\phi_1 & -\sin\phi_1 \\ 0 & \sec\theta_1 \sin\phi_1 & \sec\theta_1 \cos\phi_1 \end{bmatrix}$$

$\boldsymbol{J}(\boldsymbol{\eta_1}) = [-\sin\theta_1 \quad \cos\theta_1 \sin\phi_1 \quad \cos\theta_1 \cos\phi_1]$ is the last row of the rotation matrix from B_1 to E. Additionly, we can obtain $\dot{\boldsymbol{J}}(\boldsymbol{\eta_1}) = \boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{S}(\boldsymbol{\Omega_1})$, where $\boldsymbol{S}(\boldsymbol{\Omega_1})$ denotes the skew-symmetric matric defined as

$$\boldsymbol{S}(\boldsymbol{\Omega_1}) = \begin{bmatrix} 0 & -r_1 & q_1 \\ r_1 & 0 & -p_1 \\ -q_1 & p_1 & 0 \end{bmatrix}$$

The dynamic model of the quadrotor can be presented by using Newton's laws [15].

$$\begin{cases} m_1\dot{\boldsymbol{V}}_{\boldsymbol{1}} + \boldsymbol{S}(\boldsymbol{\Omega_1})m_1\boldsymbol{V_1} = \boldsymbol{F} - \boldsymbol{F_{aero}} - \boldsymbol{F_{grav}} + \boldsymbol{d_F} \\ \boldsymbol{I_1}\dot{\boldsymbol{\Omega}}_{\boldsymbol{1}} + \boldsymbol{S}(\boldsymbol{\Omega_1})\boldsymbol{I_1}\boldsymbol{\Omega_1} = \boldsymbol{T} - \boldsymbol{T_{aero}} + \boldsymbol{d_M} \end{cases} \tag{2}$$

where m_1 is the mass of the quadrotor, $\boldsymbol{I_1} = \text{diag}\{I_x, I_y, I_z\}$ is the total inertial matrix of the quadrotor. $\boldsymbol{F}$, $\boldsymbol{F_{aero}}$, $\boldsymbol{F_{grav}}$ and $\boldsymbol{T}$, $\boldsymbol{T_{aero}}$ are respectively propeller thrust, aerodynamic force, gravity, propeller moment and aerodynamic moment, which are explained as $\boldsymbol{F} = [0, 0, F_{total}]^T$, $\boldsymbol{F_{aero}} = \boldsymbol{K_t}\boldsymbol{V_1}$, $\boldsymbol{F_{grav}} = m_1\boldsymbol{R_{t1}^T}\boldsymbol{g}$, $\boldsymbol{T} = [M_x, M_y, M_z]^T$, $\boldsymbol{T_{aero}} = \boldsymbol{K_r}\boldsymbol{\Omega_1}$, with $F_{total} = \sum_{i=1}^{4} F_i$, $M_x = d(F_2 - F_4)$, $M_y = d(F_3 - F_1)$, $M_z = c\sum_{i=1}^{4}(-1)^{i+1}F_i$. The four forces $F_i (i = 1, 2, 3, 4)$ are the thrusts produced by each propeller of the quadrotor. $\boldsymbol{K_t} = \text{diag}\{K_{t1}, K_{t2}, K_{t3}\}$ and $\boldsymbol{K_r} = \text{diag}\{K_{r1}, K_{r2}, K_{r3}\}$ are two aerodynamic friction matrices, $\boldsymbol{g} = [0, 0, g_s]^T (g_s = 9.8m/s)$ is the gravity vector, d is the distance from the epicenter of the quadrotor to the rotor axes and $c > 0$ is the drag factor. $\boldsymbol{d_F} = [d_{fx}, d_{fy}, d_{fz}]^T$ and $\boldsymbol{d_M} = [d_{mx}, d_{my}, d_{mz}]^T$ are the disturbance force and torque.

The kinematics of the vehicle can be described in the same way as the quadrotor's [16]

$$\begin{cases} \dot{z}_2 = \boldsymbol{J}(\boldsymbol{\eta_2})\boldsymbol{V_2} \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{2}} = \boldsymbol{R_{r2}}\boldsymbol{\Omega_2} \end{cases} \tag{3}$$

where z_2 and $\boldsymbol{\eta_2} = [\phi_2, \theta_2, \psi_2]^T$ are the altitude and attitude of the vessel represented in E respectively. The velocity $\boldsymbol{V_2} = [u_2, v_2, w_2]^T$ and angular velocity $\boldsymbol{\Omega_2} = [p_2, q_2, r_2]^T$ are defined in B_1. $\boldsymbol{J}(\boldsymbol{\eta_2})$ and $\boldsymbol{R_{r2}}$ have the same forms as the quadrotor with $\boldsymbol{\eta_2}$ respectively.

In order to simplify the analysis, we neglect the motions of vehicle in heave, pitch and roll, that is the rotation variables $\phi_2, \theta_2, p_2, q_2$ are always zero. So we have $\boldsymbol{J}(\boldsymbol{\eta_2})$ and $\boldsymbol{R_{r2}}$ are simplified as $\boldsymbol{J}(\boldsymbol{\eta_2}) = [0 \quad 0 \quad 1]$ and $\boldsymbol{R_{r2}} = \boldsymbol{I_3}$.

The dynamics equation can be generally given as [16]

$$\begin{cases} \boldsymbol{M_1}\dot{\boldsymbol{V}}_{\boldsymbol{2}} + \boldsymbol{C_1}\boldsymbol{\Omega_2} + \boldsymbol{D_1}\boldsymbol{V_2} = \boldsymbol{\Upsilon_1} \\ \boldsymbol{M_2}\dot{\boldsymbol{\Omega}}_{\boldsymbol{2}} + \boldsymbol{C_2}\boldsymbol{V_2} + \boldsymbol{D_2}\boldsymbol{\Omega_2} = \boldsymbol{\Upsilon_2} \end{cases} \tag{4}$$

where $\boldsymbol{M_1}$ and $\boldsymbol{M_2}$ are symmetric and positive definite matrices. $\boldsymbol{C_1}$ and $\boldsymbol{C_2}$ are the matrices of Coriolis and centripetal terms. $\boldsymbol{D_1}$ and $\boldsymbol{D_2}$ are the damping matrices. $\boldsymbol{\Upsilon_1} = [\tau_x, \tau_y, 0]^T$ and $\boldsymbol{\Upsilon_2} = [0, 0, \tau_z]^T$ are external torque and force inputs. The details of these matrices are given as follows $\boldsymbol{M_1} = \text{diag}\{m_{11}, m_{22}, 0\}$, $\boldsymbol{M_2} = \text{diag}\{0, 0, m_{33}\}$, $\boldsymbol{D_1} = \text{diag}\{d_{11}, d_{22}, 0\}$, $\boldsymbol{D_2} = \text{diag}\{0, 0, d_{33}\}$,

$$\boldsymbol{C_1} = \begin{bmatrix} 0 & 0 & -m_{22}v_2 \\ 0 & 0 & m_{11}u_2 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{C_2} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ m_{22}v_2 & -m_{11}u_2 & 0 \end{bmatrix}.$$

2.2 4-DOF Relative Motion Model

According to the models proposed in subsection 2.1, the relative kinematics are presented as

$$\begin{cases} \dot{z}_e = \boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{V_1} - \boldsymbol{J}(\boldsymbol{\eta_2})\boldsymbol{V_2} = \boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{V_1} - w_2 \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{e}} = \boldsymbol{R_{r1}}\boldsymbol{\Omega_e} \end{cases} \tag{5}$$

where $z_e = z_1 - z_2$ is the relative altitude in E, $\boldsymbol{\eta_e} = \boldsymbol{\eta_1} - \boldsymbol{\eta_2}$ is the relative attitude in E, $\boldsymbol{\Omega_e} = \boldsymbol{\Omega_1} - \boldsymbol{R_{r_{12}}}\boldsymbol{\Omega_2}$ is the relative angular velocity in B_1, $\boldsymbol{\Omega_2} = [0, 0, r_2]^T$ and $\boldsymbol{R_{r_{12}}} = \boldsymbol{R_{r_1}^{-1}}\boldsymbol{R_{r_2}} = \boldsymbol{R_{r_1}^{-1}}$ are transformation matrices from B_2 to B_1.

The relative dynamics can be expressed as

$$\begin{cases} \ddot{z}_e = f_1 + u_1 + d_1 \\ \dot{\Omega}_e = \boldsymbol{f_2} + \boldsymbol{u_2} + \boldsymbol{d_2} \end{cases} \tag{6}$$

where

$$f_1 = -\frac{\boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{K_t}\boldsymbol{V_1}}{m_1} - g_s$$

$$u_1 = \frac{\cos\theta_1 \cos\phi_1}{m_1} F_{total}$$

$$d_1 = \frac{\boldsymbol{J}(\boldsymbol{\eta_1})}{m_1}\boldsymbol{d_F}$$

$$\begin{aligned} \boldsymbol{f_2} = &- \begin{bmatrix} \frac{K_{r1}p_1}{I_x} & \frac{K_{r2}q_1}{I_y} & \frac{K_{r3}r_1}{I_z} \end{bmatrix}^T \\ &- \begin{bmatrix} \frac{(I_z-I_y)q_1r_1}{I_x} & \frac{(I_x-I_z)p_1r_1}{I_y} & \frac{(I_y-I_x)p_1q_1}{I_z} \end{bmatrix}^T \\ &- \dot{\boldsymbol{R}}_{\boldsymbol{r_{12}}} \begin{bmatrix} 0 & 0 & r_2 \end{bmatrix}^T \\ &- \boldsymbol{R_{r_{12}}} \begin{bmatrix} 0 & 0 & \frac{(\tau_z - d_{33}r_2 + (m_{11}-m_{22})u_2v_2)}{m_{33}} \end{bmatrix}^T \end{aligned}$$

$$\boldsymbol{u_2} = \begin{bmatrix} \frac{M_x}{I_x} & \frac{M_y}{I_y} & \frac{M_z}{I_z} \end{bmatrix}^T$$

$$\boldsymbol{d_2} = \begin{bmatrix} \frac{d_{mx}}{I_x} & \frac{d_{my}}{I_y} & \frac{d_{mz}}{I_z} \end{bmatrix}^T$$

To facilitate analysis, we denote the system states $\boldsymbol{e_1} = [z_e, \boldsymbol{\eta_e^T}]^T$ and $\boldsymbol{e_2} = [\dot{z}_e, \boldsymbol{\Omega_e^T}]^T$, then the relative equation (5) and (6) can be rewritten as

$$\begin{cases} \dot{\boldsymbol{e}}_{\boldsymbol{1}} = \boldsymbol{\Pi} \\ \dot{\boldsymbol{e}}_{\boldsymbol{2}} = \boldsymbol{f} + \boldsymbol{u} + \boldsymbol{d} \end{cases} \tag{7}$$

where $\boldsymbol{\Pi} = [\boldsymbol{J}(\boldsymbol{\eta_1})\boldsymbol{V_1} - w_2, (\boldsymbol{R}_{r_1}\boldsymbol{\Omega_e})^T]^T$, $\boldsymbol{f} = [f_1, \boldsymbol{f_2}^T]^T$, $\boldsymbol{u} = [u_1, \boldsymbol{u_2}^T]^T$, $\boldsymbol{d} = [d_1, \boldsymbol{d_2}^T]^T$.

2.3 Control Objective

For the subsequent development of the control laws, the following assumptions are employed in this paper.

Assumption 1: The quadotor and the vehicle are both rigid bodies and the quadrotor is symmetrical with respect to the axes $O_b^1x_b^1$, $O_b^1y_b^1$, $O_b^1z_b^1$.

刚体

Assumption 2: The quadrotor hovers above the vessel in initial time before the action of designed controller, and the pitch angle of the quadrotor is bounded as $-\frac{\pi}{2} \leq \theta_1 \leq \frac{\pi}{2}$.

Assumption 3: All the states of the quadrotor and the vehicle are measurable accurately. The unknown disturbances d_F and d_M are both bounded and have the bounded first and second time derivatives. All states of the vehicle are also bounded under the effect of the vehicle's input.

有界干扰

The control objective is to design control law $\boldsymbol{u}$ in (7) with unknown disturbance so that the quadrotor can land on the vehicle steadlily, while it is guaranteed that the tracking error $\boldsymbol{e_1}$ is restricted to small values and all signals of the closed-loop system remain bounded.

3 CONTROL DESIGN AND STABILITY ANALYSIS

3.1 RBFNN Approximation

Suppose $f(\boldsymbol{x}) : R^m \to R$ is an unknown smooth nonlinear function, which can be approximated over a compact set $\boldsymbol{\Omega} \in R^m$ with the RBFNN:

$$f(\boldsymbol{x}) = \boldsymbol{\omega}^{*T}\boldsymbol{\Psi}(\boldsymbol{x}) + \varepsilon \tag{8}$$

where the node number of NN is l, and ε is the approximation error that is bounded over $\boldsymbol{\Omega}$, namely, $|\varepsilon| \leq \bar{\varepsilon}$, where $\bar{\varepsilon}$ is an unknown constant. $\boldsymbol{\omega^*} \in R^l$ is the optimal weight vector defined as

$$\boldsymbol{\omega^*} = \arg\min_{\hat{\omega}} \left\{ \sup_{\boldsymbol{x}\in\boldsymbol{\Omega}} |f(\boldsymbol{x}) - \hat{\boldsymbol{\omega}}^T\boldsymbol{\Psi}(\boldsymbol{x})| \right\}$$

where $\hat{\boldsymbol{\omega}}$ is the estimation of $\boldsymbol{\omega^*}$. $\boldsymbol{\Psi}(\boldsymbol{x}) = [\psi_1(\boldsymbol{x}), \cdots, \psi_l(\boldsymbol{x})]^T : \boldsymbol{\Omega} \to R^l$ represents the radial basis function vector, the element of which is expressed as

$$\psi_i(\boldsymbol{x}) = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{\mu_i}||^2}{\epsilon_i^2} \right), i = 1, \cdots, l$$

where $\boldsymbol{\mu_i} \in R^m$ and $\epsilon_i \in R$ are the center and spread.

3.2 Backstepping Controller Design

First define two error vectors as

$$\begin{cases} \boldsymbol{z_1} = \boldsymbol{e_1} - \boldsymbol{e_{1d}} \\ \boldsymbol{z_2} = \boldsymbol{e_2} - \boldsymbol{\alpha} \end{cases} \tag{9}$$

where $\boldsymbol{e_{1d}} = [z_{ed}, \boldsymbol{\eta_{ed}^T}]$ is the desired value, $\boldsymbol{\alpha}$ is a virtual control to be designed later.

The time derivative of $\boldsymbol{e_1}$ is given as

$$\dot{\boldsymbol{e}}_{\boldsymbol{1}} = \begin{bmatrix} \dot{z}_e \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{e}} \end{bmatrix} = \begin{bmatrix} \dot{z}_e \\ \boldsymbol{R_{r1}}\boldsymbol{\Omega_e} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \boldsymbol{R_{r1}} \end{bmatrix} \begin{bmatrix} \dot{z}_e \\ \boldsymbol{\Omega_e} \end{bmatrix} = \boldsymbol{R}e_2 \tag{10}$$

where $\boldsymbol{R} = \text{diag}\{1, \boldsymbol{R_{r1}}\}$.

Then the time derivative of $\boldsymbol{z_1}$ is calculated as

$$\dot{\boldsymbol{z}}_{\boldsymbol{1}} = \dot{\boldsymbol{e}}_{\boldsymbol{1}} - \dot{\boldsymbol{e}}_{\boldsymbol{1d}} = \boldsymbol{R}(\boldsymbol{z_2} + \boldsymbol{\alpha}) - \dot{\boldsymbol{e}}_{\boldsymbol{1d}} \tag{11}$$

The Lyapunov function is denoted as follows

$$\boldsymbol{V_1} = \frac{1}{2}\boldsymbol{z_1}^T\boldsymbol{z_1} \tag{12}$$

We can calculate the time derivative of $\boldsymbol{V_1}$ along (11) as

$$\dot{\boldsymbol{V}}_{\boldsymbol{1}} = \boldsymbol{z_1}^T\dot{\boldsymbol{z}}_{\boldsymbol{1}} = \boldsymbol{z_1}^T(\boldsymbol{R\alpha} - \dot{\boldsymbol{e}}_{\boldsymbol{1d}}) + \boldsymbol{z_1^T R z_2} \tag{13}$$

Design the virtual control $\boldsymbol{\alpha}$ to stabilize $\boldsymbol{z_1}$

$$\boldsymbol{\alpha} = \boldsymbol{R^{-1}}(-\boldsymbol{k_1 z_1} + \dot{\boldsymbol{e}}_{\boldsymbol{1d}}) \tag{14}$$

where $\boldsymbol{k_1} \in R^{4\times4}$ is a positive definite symmetric matrix. Substituting (14) into (13) yields

$$\dot{\boldsymbol{V}}_{\boldsymbol{1}} = -\boldsymbol{z_1}^T\boldsymbol{k_1 z_1} + \boldsymbol{z_1}^T\boldsymbol{R z_2} \tag{15}$$

The coupling term $\boldsymbol{z_1^T R z_2}$ will be canceled in the next step.

We employ an RBFNN with l nodes to approximate $\boldsymbol{\Delta} = \boldsymbol{f} + \boldsymbol{d}$

$$\boldsymbol{\Delta} = \boldsymbol{W}^{*T}\boldsymbol{\Psi}(\boldsymbol{\vartheta}) + \boldsymbol{\varepsilon} \tag{16}$$

where $\boldsymbol{W^*} \in R^{l\times4}$ is the weight matrix with $\hat{\boldsymbol{W}}$ and $\tilde{\boldsymbol{W}} = \hat{\boldsymbol{W}} - \boldsymbol{W^*}$ as its estimation and estimation error. $\boldsymbol{\Psi}(\boldsymbol{\vartheta}) \in R^l$ denotes the radial basis function vector with the input vector $\boldsymbol{\vartheta} = [w_e, p_e, q_e, r_e, w_e p_e, w_e q_e, w_e r_e, p_e q_e, p_e r_e, q_e r_e]^T$, where $(\cdot)_e = (\cdot)_1 - (\cdot)_2$. $\boldsymbol{\varepsilon} \in R^4$ is the error vector with $|\varepsilon_i| \le \bar{\varepsilon}_i, i = 1,2,3,4$ and $\bar{\varepsilon}_i, i = 1,2,3,4$ is an unknown constant vector.

Definite $\boldsymbol{\zeta} = \bar{\boldsymbol{f}} + \bar{\boldsymbol{d}} + \bar{\boldsymbol{\varepsilon}}$ with its estimation $\hat{\boldsymbol{\zeta}} \in R^4$ and estimation error $\tilde{\boldsymbol{\zeta}} = \hat{\boldsymbol{\zeta}} - \boldsymbol{\zeta}$.

Consider the Lyapunov function candidate as follows

$$\boldsymbol{V_2} = \frac{1}{2}\boldsymbol{z_2}^T\boldsymbol{z_2} + \frac{1}{2}\mathrm{tr}(\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}\tilde{\boldsymbol{W}}) + \frac{1}{2}\tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}\tilde{\boldsymbol{\zeta}} \tag{17}$$

where $\boldsymbol{\Gamma_i} = \boldsymbol{\Gamma_i}^T, i = 1,2$ is positive definite matrix.

The time derivative of $\boldsymbol{\alpha}$ is along (9) and (14)

$$\dot{\boldsymbol{\alpha}} = \dot{\boldsymbol{R}}^{-1}(-\boldsymbol{k_1 z_1} + \dot{\boldsymbol{e}}_{\boldsymbol{1d}}) + \boldsymbol{R}^{-1}[-\boldsymbol{k_1}(\dot{\boldsymbol{e}}_{\boldsymbol{1}} - \dot{\boldsymbol{e}}_{\boldsymbol{1d}}) + \ddot{\boldsymbol{e}}_{\boldsymbol{1d}}] \tag{18}$$

Then the time derivative of $\boldsymbol{V_2}$ satisfies

$$\begin{aligned}\dot{\boldsymbol{V}}_{\boldsymbol{2}} =& \boldsymbol{z_2}^T(\boldsymbol{u} + \boldsymbol{W}^{*T}\boldsymbol{\Psi}(\boldsymbol{\vartheta}) + \boldsymbol{\varepsilon} - \dot{\boldsymbol{\alpha}}) + \mathrm{tr}(\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}\dot{\hat{\boldsymbol{W}}}) \\ &+ \tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}\dot{\hat{\boldsymbol{\zeta}}}\end{aligned} \tag{19}$$

We design the following control input command

$$\boldsymbol{u} = -\boldsymbol{k_2 z_2} - \boldsymbol{R}^T\boldsymbol{z_1} - \hat{\boldsymbol{W}}^T\boldsymbol{\Psi}(\boldsymbol{\vartheta}) - \tanh(\frac{\boldsymbol{z_2}}{\varrho})\hat{\boldsymbol{\zeta}} + \dot{\boldsymbol{\alpha}} \tag{20}$$

where $\boldsymbol{k_2} \in R^{4\times4}$ is a positive definite symmetric matrix. $\tanh(x) : R \to R$ is the hyperbolic tangent function defined as $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$, and the inequality $0 \le |x| - x\tanh(\frac{x}{\varrho}) \le \kappa\varrho$ holds for any $\varrho \in R^+$, where $x \in R$ and $\kappa = 0.2785$ satisfying $\kappa = e^{-(\kappa+1)}$.

Substituting (20) into (19), we can obtain

$$\begin{aligned}\dot{\boldsymbol{V}}_{\boldsymbol{2}} =& -\boldsymbol{z_2}^T\boldsymbol{k_2 z_2} - \boldsymbol{z_2}^T\boldsymbol{R}^T\boldsymbol{z_1} + \mathrm{tr}(\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}\dot{\hat{\boldsymbol{W}}}) + \tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}\dot{\hat{\boldsymbol{\zeta}}} \\ &+ \boldsymbol{z_2}^T(-\hat{\boldsymbol{W}}^T\boldsymbol{\Psi}(\boldsymbol{\vartheta}) - \tanh(\frac{\boldsymbol{z_2}}{\varrho})\hat{\boldsymbol{\zeta}} + \boldsymbol{W}^{*T}\boldsymbol{\Psi}(\boldsymbol{\vartheta}) + \boldsymbol{\varepsilon}) \\ \le& -\boldsymbol{z_2}^T\boldsymbol{k_2 z_2} - \boldsymbol{z_2}^T\boldsymbol{R}^T\boldsymbol{z_1} + \mathrm{tr}(\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}\dot{\hat{\boldsymbol{W}}}) + \tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}\dot{\hat{\boldsymbol{\zeta}}} \\ &- \boldsymbol{z_2}^T\hat{\boldsymbol{W}}^T\boldsymbol{\Psi}(\boldsymbol{\vartheta}) - \boldsymbol{z_2}^T\tanh(\frac{\boldsymbol{z_2}}{\varrho})\hat{\boldsymbol{\zeta}} + |z|_2^T\boldsymbol{\zeta} \\ \le& -\boldsymbol{z_2}^T\boldsymbol{k_2 z_2} - \boldsymbol{z_2}^T\boldsymbol{R}^T\boldsymbol{z_1} + \tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}(\dot{\hat{\boldsymbol{W}}} - \boldsymbol{\Gamma_1}\boldsymbol{\Psi}(\boldsymbol{\vartheta})\boldsymbol{z_2}) \\ &+ \tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}(\dot{\hat{\boldsymbol{\zeta}}} - \boldsymbol{\Gamma_2}\tanh(\frac{\boldsymbol{z_2}}{\varrho})\boldsymbol{z_2}) + \kappa\varrho\boldsymbol{\zeta}\end{aligned} \tag{21}$$

where $\boldsymbol{\varepsilon} \le \bar{\boldsymbol{\varepsilon}} \le \boldsymbol{\zeta}$.

Consider the following adaptive law

$$\begin{cases}\dot{\hat{\boldsymbol{W}}} = \boldsymbol{\Gamma_1}(\boldsymbol{\Psi}(\boldsymbol{\vartheta})\boldsymbol{z_2} - k_w\hat{\boldsymbol{W}}) \\ \dot{\hat{\boldsymbol{\zeta}}} = \boldsymbol{\Gamma_2}(\tanh(\frac{\boldsymbol{z_2}}{\varrho})\boldsymbol{z_2} - k_\zeta\hat{\boldsymbol{\zeta}})\end{cases} \tag{22}$$

where $k_w > 0$ and $k_\zeta > 0$ are two small positive constants. By completion of squares, we can get

$$\begin{cases}-k_w\tilde{\boldsymbol{W}}^T\hat{\boldsymbol{W}} \le -\dfrac{k_w||\tilde{\boldsymbol{W}}||^2}{2} + \dfrac{k_w||\boldsymbol{W^*}||^2}{2} \\ -k_\zeta\tilde{\boldsymbol{\zeta}}^T\hat{\boldsymbol{\zeta}} \le -\dfrac{k_\zeta||\tilde{\boldsymbol{\zeta}}||^2}{2} + \dfrac{k_\zeta||\tilde{\boldsymbol{\zeta}}^*||^2}{2}\end{cases} \tag{23}$$

Substituting (22) and (23) into (21), we have

$$\begin{aligned}\dot{\boldsymbol{V}}_{\boldsymbol{2}} \le& -\boldsymbol{z_2}^T\boldsymbol{k_2 z_2} - \boldsymbol{z_2}^T\boldsymbol{R}^T\boldsymbol{z_1} + \kappa\varrho\boldsymbol{\zeta} - \frac{k_w||\tilde{\boldsymbol{W}}||^2}{2} \\ &+ \frac{k_w||\boldsymbol{W^*}||^2}{2} - \frac{k_\zeta||\tilde{\boldsymbol{\zeta}}||^2}{2} + \frac{k_\zeta||\tilde{\boldsymbol{\zeta}}^*||^2}{2}\end{aligned} \tag{24}$$

The main result of the control performance is studied in the following theorem based on Lemma 1.

Lemma 1: For bounded initial conditions, if there exits a C^1 continuous and positive defined Lyapunov function $V(x)$ satisfying $k_1(||x||) \le V(x) \le k_2(||x||)$, such that $V(x) \le -\gamma V(x) + \nu$, where $k_1, k_2 : R^n \to R$ are class κ functions and ν is a positive constant, then the solution $x(t)$ is uniformly bounded.

Theorem 1. Consider the closed-loop system consisting of the relative model (7) under Assumptions 1-3, the controller (20) with adaptive laws (22). For any bounded initial conditions satisfying $\boldsymbol{\Omega_v} = \{[\boldsymbol{z_1}^T, \boldsymbol{z_2}^T, \hat{\boldsymbol{W}}^T, \hat{\boldsymbol{\zeta}}^T]^T : V_1 + V_2 \le A\}$, where A is a positive constant, there exist appropriate design parameters such that all states in the closed-loop control system remain ultimately bounded and the relative errors $\boldsymbol{e_1}$ converges to a small neighborhood of zero.

Proof. Consider the Lyapunov function candidate as

$$\begin{aligned}\boldsymbol{V} &= \boldsymbol{V_1} + \boldsymbol{V_2} \\ &= \frac{1}{2}\boldsymbol{z_1}^T\boldsymbol{z_1} + \frac{1}{2}\boldsymbol{z_2}^T\boldsymbol{z_2} + \frac{1}{2}\mathrm{tr}(\tilde{\boldsymbol{W}}^T\boldsymbol{\Gamma_1}^{-1}\tilde{\boldsymbol{W}}) + \frac{1}{2}\tilde{\boldsymbol{\zeta}}^T\boldsymbol{\Gamma_2}^{-1}\tilde{\boldsymbol{\zeta}} \\ &\ge 0\end{aligned} \tag{25}$$

Recalling (15) and (24) we can obtain

$$\dot{\boldsymbol{V}} \leq -\alpha \boldsymbol{V} + \beta \tag{26}$$

with

$$\alpha = \min\{2\lambda_m(\boldsymbol{k_1}), 2\lambda_m(\boldsymbol{k_2}), k_w\lambda_m(\boldsymbol{\Gamma_1}), k_\zeta\lambda_m(\boldsymbol{\Gamma_2})\}$$

and

$$\beta = \frac{k_w||\boldsymbol{W}^*||^2}{2} + \frac{k_\zeta||\tilde{\boldsymbol{\zeta}}^*||^2}{2} + \kappa\varrho\boldsymbol{\zeta}$$

where $\lambda_m(\cdot)$ represents the minimum eigenvalue of a matrix.
According to (26), we have

$$0 \leq \boldsymbol{V} \leq \frac{\beta}{\alpha} + [\boldsymbol{V}(\boldsymbol{0}) - \frac{\beta}{\alpha}]e^{-\alpha t} \tag{27}$$

Furthermore, from (25) and (26), we can obtain

$$||\boldsymbol{z_1}|| \leq \sqrt{\frac{2\beta}{\alpha} + 2[\boldsymbol{V}(\boldsymbol{0}) - \frac{\beta}{\alpha}]e^{-\alpha t}} \tag{28}$$

By virtue of (26), (27) and (28), it follows that $\boldsymbol{z_1}$, $\boldsymbol{z_2}$, $\hat{\boldsymbol{W}}$, $\hat{\boldsymbol{\zeta}}$ and $\boldsymbol{V}$ are uniformly ultimately bounded, and the tracking error $\boldsymbol{z_1}$ will converge eventually to the set $\boldsymbol{\Omega_z} = \{\boldsymbol{z_1} : ||\boldsymbol{z_1}|| \leq \sqrt{2\beta/\alpha}\}$, whose size can be set arbitrarily small by appropriately choosing design parameters. Therefore, with the employed RBFNN being constructed on large enough approximation regions, a stable adaptive neural controller can be developed so that bounded initial conditions guarantee that all the states in the closed-loop system are uniformly ultimately bounded.

4 NUMERICAL SIMULATIONS

In this section, some simulation results are carried out to verify the effectiveness and the efficiency of the proposed autonomous landing approach.
The inertial, geometric, aerodynamic, hydrodynamic parameters of the quadrotor and vehicle are respectively listed in Table 1.

Table 1: Parameters for the quadrotor and the vehicle

Parameter	Value	Paremeter	Value
m_1	2kg	m_{11}	100kg
I_x	1.2416N.ms^2/rad	m_{22}	100kg
I_y	1.2416N.ms^2/rad	m_{33}	80kg.m^2
I_z	2.4832N.ms^2/rad	d_{11}	100kg/s
K_{t_1}	0.01N.s/m	d_{22}	100kg/s
K_{t_2}	0.01N.s/m	d_{33}	80kg.m^2/s
K_{t_3}	0.01N.s/m	K_{r_1}	0.001Nm.s/rad
K_{r_2}	0.001Nm.s/rad	K_{r_3}	0.001Nm.s/rad

The initial conditions are selected as $z_1 = 1.5$(m), $\boldsymbol{\eta_1} = [30, 30, 30]^T$(deg), $\boldsymbol{\Omega_1} = [5.73, 5.73, 5.73]^T$(deg/s), $\boldsymbol{V_1} = [1, 1, 0.1]^T$(m/s), $z_2 = 0$(m), $\boldsymbol{\eta_2} = [0, 0, 15]^T$(rad), $\boldsymbol{\Omega_2} = [0, 0, 0]^T$(rad/s), $\boldsymbol{V_2} = [1, 1, 0]^T$(m/s). The inputs of the vehicle are set as $\boldsymbol{\Upsilon_1} = [15, 10, 0]^T$ and $\boldsymbol{\Upsilon_2} = [0, 0, 5]^T$. The control parameters are selected as $\boldsymbol{k_1} = \text{diag}\{3, 5, 10, 10\}$, $\boldsymbol{k_2} = \text{diag}\{2, 4, 10, 10\}$. The external disturbance is specified as $\boldsymbol{d_F} = [0.5, 0.5, 0.5]^T + 0.5\sin(0.2t) + 0.05\text{rand}(3, 1)$(N), $\boldsymbol{d_M} = [0.005, 0.005, 0.005]^T + 0.005\sin(0.01t) + 0.005\text{rand}(3, 1)$(Nm), rand$(3, 1)$ denotes the vector of random Gaussian white noise signal with size 3×1. In addition, the update parameters are $l = 100$, $\boldsymbol{\Gamma_1} = 0.05\boldsymbol{I_l}$, $\boldsymbol{\Gamma_2} = 0.1\boldsymbol{I_4}$, $k_w = 5$, $k_\zeta = 0.1$, $\varrho = 0.1$, where $\boldsymbol{I_l}$ is the unit matrix with l dimension.
The simulation results are shown in Figures 2-5. As shown in Figure 2, the relative attitude between the quadrotor and vehicle tends to zero in 1.0s and the relative altitude is restricted to zero in about 2.5s, which means the quadrotor can fast track the altitude and attitude of the vehicle with a high precision. Figure 3 shows that the velocity of the quadrotor can also track the vehicle quickly and accurately. Figure 4 describes the control inputs of the quadrotor. It can be seen that the inputs are all in the specified region. Figure 5 expressed the uncertainties estimation with RBFNN, from which we can obtain that the adaptive RBFNN can estimate the uncertainties (including disturbances) well. Accordingly, the simulation result demonstrates that the proposed controller are capable of attaining satisfactory autonomous landing performance.

5 CONCLUSION

In this paper, a backstepping control strategy has been presented to handle the landing problem of a quadrotor on a moving vehicle. A fully actuated 4-DOF nonlinear relative motion model with four control inputs is established. An adaptive radial basis function neural network is developed to provide an estimation of the unknown disturbance and is applied to design the controller to solve the autonomous landing problem when the quadrotor hovers above the vehicle. All the states of the closed-loop system are proved to be bounded and the tracking errors are restricted to small values, from which we can find that the good landing performance is guaranteed. The effectiveness of the proposed control approach is verified by simulations. Further work will focus on solving the autonomous vehicle landing problem with the coupled 6-DOF nonlinear relative motion model.

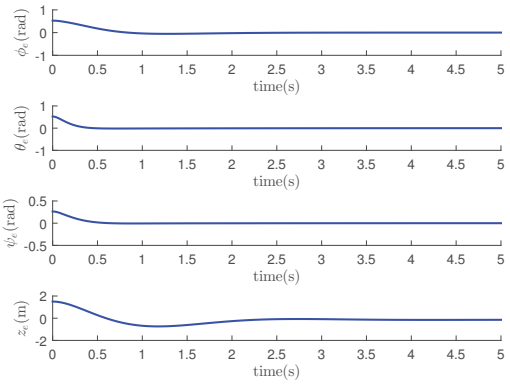

Figure 2: Relative attitude & altitude of the quadrotor and the vessel

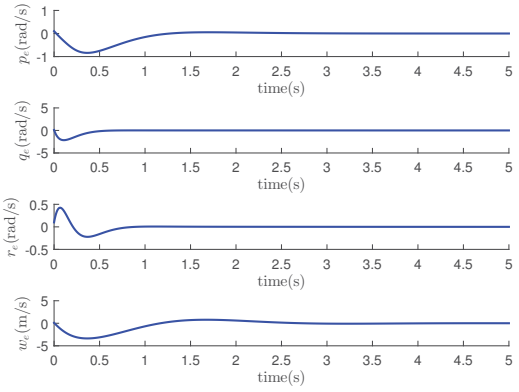

Figure 3: Velocity error of the quadrotor and the vessel

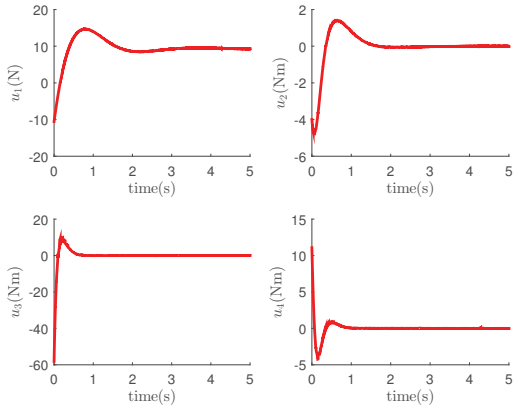

Figure 4: Control inputs of the quadrotor

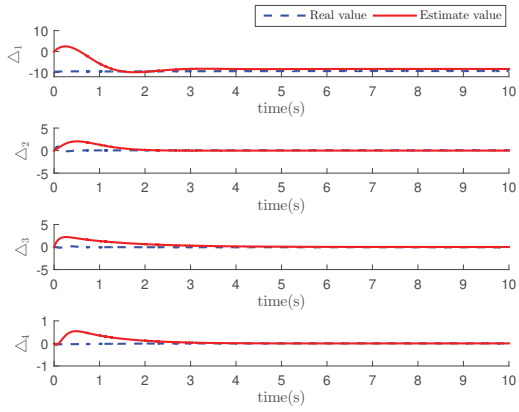

Figure 5: Uncertainties estimation with RBFNN

REFERENCES

[1] Z. Zuo and P. Ru, "Augmented unmanned aircrafts," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 3090–3101, 2014.

[2] S. Islam, P. X. Liu, and A. E. Saddik, "Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1563–1571, 2015.

[3] J.-J. Xiong and E.-H. Zheng, "Position and attitude tracking control for a quadrotor uav," *ISA Transactions*, vol. 53, no. 3, pp. 725–731, 2014.

[4] K. E. Wenzel, A. Masselli, and A. Zell, "Automatic take off, tracking and landing of a miniature uav on a moving carrier vehicle," *Journal of Intelligent and Robotic Systems*, vol. 61, no. 1-4, pp. 221–238, 2011.

[5] A. Borowczyk, D. T. Nguyen, P. V. Nguyen, Q. N. Dang, D. Saussi, and J. L. Ny, "Autonomous landing of a multi-rotor micro air vehicle on a high velocity ground vehicle," *eprint arXiv:1611.07329*, 2016.

[6] H. Voos and H. Bou-Ammar, "Nonlinear tracking and landing controller for quadrotor aerial robots," in *IEEE International Conference on Control Applications*, pp. 2136–2141, 2010.

[7] J. Dougherty, D. Lee, and T. Lee, "Laser-based guidance of a quadrotor uav for precise landing on an inclined surface," in *American Control Conference*, pp. 1210–1215, 2014.

[8] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *International Journal of Robotics Research*, vol. 31, no. 5, pp. 664–674, 2014.

[9] B. Hu and S. Mishra, "A time-optimal trajectory generation algorithm for quadrotor landing onto a moving platform," in *American Control Conference*, pp. 4183–4188, 2017.

[10] S. Liang and H. Wei, "Robust adaptive relative position tracking and attitude synchronization for spacecraft rendezvous," *Aerospace Science and Technology*, vol. 41, pp. 28–35, 2015.

[11] L. Sun and W. Huo, "Robust adaptive control of spacecraft proximity maneuvers under dynamic coupling and uncertainty," *Advances in Space Research*, vol. 56, no. 10, pp. 2206–2217, 2015.

[12] M. Chen, S. S. Ge, and B. Ren, "Adaptive tracking control of uncertain mimo nonlinear systems with input constraints," *Automatica*, vol. 47, no. 3, pp. 452–465, 2011.

[13] S. S. Ge and C. Wang, *Adaptive neural control of uncertain MIMO nonlinear systems*. IEEE Press, 2004.

[14] Z. Peng, D. Wang, Z. Chen, X. Hu, and W. Lan, "Adaptive dynamic surface control for formations of autonomous surface vehicles with uncertain dynamics," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 513–520, 2013.

[15] T. Madani and A. Benallegue, "Backstepping control for a quadrotor helicopter," in *Ieee/rsj International Conference on Intelligent Robots and Systems*, pp. 3255–3260, 2007.

[16] W. Dong and Y. Guo, "Nonlinear tracking control of underactuated surface vessel," in *American Control Conference, 2005. Proceedings of the*, pp. 4351–4356 vol. 6, 2005.