

Learning and detecting abnormal speed of marine robots

International Journal of Advanced Robotic Systems
March-April 2021: 1–13
© The Author(s) 2021
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/1729881421999268
journals.sagepub.com/home/arx



Sungjin Cho¹ , Fumin Zhang²  and Catherine R Edwards³ 

Abstract

This article presents anomaly detection algorithms for marine robots based on their trajectories under the influence of unknown ocean flow. A learning algorithm identifies the flow field and estimates the through-water speed of a marine robot. By comparing the through-water speed with a nominal speed range, the algorithm is able to detect anomalies causing unusual speed changes. The identified ocean flow field is used to eliminate false alarms, where an abnormal trajectory may be caused by unexpected flow. The convergence of the algorithms is justified through the theory of adaptive control. The proposed strategy is robust to speed constraints and inaccurate flow modeling. Experimental results are collected on an indoor testbed formed by the Georgia Tech Miniature Autonomous Blimp and Georgia Tech Wind Measuring Robot, while simulation study is performed for ocean flow field. Data collected in both studies confirm the effectiveness of the algorithms in identifying the through-water speed and the detection of speed anomalies while avoiding false alarms.

Keywords

Anomaly detection, adaptive learning, controlled Lagrangian particle tracking, flow mapping, autonomous blimp, wind sensor, omnidirectional robot

Date received: 24 October 2020; accepted: 27 January 2021

Topic Area: Field Robotics

Topic Editor: YangQuan Chen

Associate Editor: Angelos Amanatiadis

Introduction

Anomaly detection for marine robots is an important practical problem because marine robots are often used in distant and hostile environments such as the deep sea and the polar oceans. During long-range or long-period missions, marine creatures and biofouling may harm robot sensors and thrusters.¹ Monitoring sensors have been installed to evaluate components vulnerable to faults.² For example, damaged propellers impair propulsive efficiency to control vehicle speed. These faults could be detected with rotational speed sensors installed at the propellers; however, this approach requires increased hardware complexity and cost,³ and it may not detect unexpected external disturbances (e.g. white shark attack).

We propose anomaly detection algorithms for marine robots based on their trajectory data. Given a trajectory,

we develop a learning algorithm that estimates the through-water speed of the robot as well as the ambient flow velocity. The robot speed estimate is then used to determine whether or not robot motion is abnormal. Anomaly occurs when the robot speed estimate is out of the range in normal operation. This approach complements the

¹Department of Guidance and Control, Agency for Defense Development, Daejeon, South Korea

²School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA

³Skidway Institute of Oceanography, Savannah, GA, USA

Corresponding author:

Fumin Zhang, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.

Email: fumin@gatech.edu



existing methods that detect faults of individual components based on sensor measurements. The use of trajectory data in our work follows similar motivations as previous works in surveillance applications (e.g. see the literature⁴). Our work uses underwater trajectory data of marine robots while the previous work^{5,6} uses car trajectory data and surface trajectory data of marine vessels. The major difference is the type of speed information used for anomaly detection; the former is through-water speed and the latter is ground speed.

Related work

Recent studies on ground and marine robots proposed approaches to identify abnormal robot motion.^{7,8} An anomaly or fault is defined as an unacceptable deviation of at least one characteristic property of a variable from an acceptable behavior.⁹ We assume that normal behaviors occur far more frequently than abnormal behaviors for the data collected. If this assumption is not true, then such techniques can produce incorrect detection results or false alarms.¹⁰

For marine robots, most fault detection algorithms have been dealing with abnormal behaviors of the system components that are the most vulnerable to faults.¹¹⁻¹³ Blocked propellers, leaking thrusters, and rotor failure are documented as frequently occurring faults.¹⁴ In 2015, a software configuration error in an underwater glider disabled an internal mass shifter that adjusts the vehicle's trim, leading the vehicle to sink to the seafloor and resulting in a temporary loss.⁸ Instead of detecting faults in an individual component, robot motion can be used for anomaly detection. Raanan et al.¹⁵ use a threshold technique to prevent an underwater glider from hitting the seafloor. The deviation from expected robot motion in the vertical plane is detected by monitoring stern plane angle, pitch angle, and depth rate.

Serious performance degradation can result when ocean flow speed is comparable to or exceeds the maximum through-water speed of marine robots, as is the case for underwater gliders.¹⁶ However, the abnormal motion caused by ocean flow should not be misclassified as an anomaly for the marine robot. Therefore, it is important to identify the ambient ocean flow while estimating the through-water speed. Measuring through-water speed of marine robots is substantially difficult because of limited hardware capability. The inertial measurement unit (IMU) and the Doppler velocity log (DVL) can be combined to measure through-water speed.¹⁷ IMU error can be corrected by the DVL, which measures both vehicle velocity with respect to ground (bottom-tracking) and flow velocity with respect to the vehicle (water-tracking).¹⁸ However, DVL accuracy can be poor when the marine robot does not maintain constant altitude and heading angle. Alternatively, acoustic positioning systems can estimate robot speed using a short baseline or a long baseline systems,

which utilize multiple beacons located at the seafloor or the hull of a ship.¹⁹ However, this measured velocity is not through-water velocity, but rather the ground velocity that combines both through-water velocity and flow velocity, which need to be further processed to determine whether the vehicle is having a fault.

Controlled Lagrangian particle tracking (CLPT) is a theoretical framework to analyze the interaction between ocean flow and marine robot control.²⁰ In contrast to *passive* Lagrangian methods, a marine robot is viewed as a *controlled* Lagrangian particle in the sense that marine robots are not freely advected by ocean flow. In the framework of CLPT, the net motion of controlled Lagrangian particles is determined by flow velocity and controlled speed. Our work leverages the CLPT framework for anomaly detection because it helps us to identify both the through-water speed of marine robots and the ambient flow. Using these information allows us to detect abnormal motion while avoiding false alarms caused by unexpected flow.

Major contributions

We present a learning algorithm that can process the data streams of the trajectory of a marine robot and the real-time heading information. The algorithm will produce a real-time estimate of the through-water speed of the robot and the ambient flow field. Using the theory of adaptive control and CLPT, we can prove theoretically that the algorithm will drive the difference between the robot trajectory and the learned trajectory to zero. In addition, the estimation error of the through-water speed and the ambient flow field also converges when a persistent exciting condition is satisfied. Furthermore, the learning algorithm guarantees bounded estimation error under bounded disturbances. Therefore, the estimated through-water speed can be used for anomaly detection, and the estimated ambient flow field can be used to reduce false alarms.

It is generally very difficult to validate the proposed algorithm on a marine robot that operates in the ocean due to the high risk and high cost of losing a marine robot. To address this challenge, we have developed an indoor testbed that uses robotic blimps to emulate marine robots and uses wind field to emulate ocean flow field. Similar to a marine robot, the motion of a robotic blimp is influenced by a wind field. The robotic blimps are carefully designed so that faults that affect its through-air speed can be easily generated. Our algorithm can then be evaluated on these blimps. Furthermore, the wind field can be autonomously measured by a mobile robot. The measured wind field is then used to compare with the estimated wind field produced by our algorithm. It would be very difficult (if not impossible) to perform these experiments in the ocean. Experimental data have justified that the anomaly detection approach works well on the testbed. Our experiments also indicate that the algorithms can be applied to other

applications such as unmanned aerial vehicles and autonomous ground vehicles.

Organization

The remainder of this article has been organized into the following sections: in the second and the third sections, we present vehicle motion model and controlled Lagrangian localization error of marine robots. In the fourth and the fifth sections, we describe an adaptive learning algorithm and an anomaly detection algorithm for marine robots. In the sixth and the seventh sections, we demonstrate mathematical simulation and experimental results by developing Georgia Tech Miniature Autonomous Blimp (GT-MAB) and Georgia Tech Wind Measuring Robot (GT WMR). In the eighth section, we provide conclusions and future work.

Models and problem setup

Let $\mathbf{F}: D \times [0, \infty] \rightarrow \mathbb{R}^2$ represents a spatially distributed vector field for the ambient flow velocity, where $D \in \mathbb{R}^2$ is the domain of interest. Furthermore, let $\Psi_c(t) = [\cos\psi_c(t), \sin\psi_c(t)]^\top$ be a unit vector that represents the direction of movements of the robot, where $\psi_c(t)$ is the heading angle of the robot. Let V_R be the through-water speed of the robot. Then, the marine robot motion model is approximated by

$$\dot{\mathbf{x}} = \mathbf{F}_R(\mathbf{x}, t) + V_R \Psi_c(t) \quad (1)$$

where the subscript R for the flow \mathbf{F} denotes an actual flow. \mathbf{F}_R and \mathbf{v}_R are assumed to be locally Lipschitz in $\mathbf{x} = [x_1, x_2]^\top \in D$, where \mathbf{x} is the true position of the robots.

In practice situations, the through-water velocity can be saturated because of control power constraints of robots. We modify equation (1) as follows

$$\dot{\mathbf{x}} = \mathbf{F}_R(\mathbf{x}, t) + u \Psi_c(t) \quad (2)$$

where

$$u = \begin{cases} V_R & \text{if } V_R \leq u_0 \\ u_0 & \text{if } V_R > u_0 \end{cases} \quad (3)$$

The maximum through-water speed u_0 is determined by the hardware configuration of marine robots.

Assumption 1. During normal operation, V_R in equation (1) is a constant.

The flow field can be represented by spatial and temporal basis functions.²¹ We consider that spatial and temporal basis functions are to be the combination of Gaussian radial basis functions and tidal basis functions, respectively. Let N be a positive integer, and $\theta \in \mathbb{R}^{2 \times N}$ be the unknown parameters, respectively. Let $\phi: D \times [0, \infty] \rightarrow \mathbb{R}^N$ be $[\phi^1(\mathbf{x}, t), \dots, \phi^N(\mathbf{x}, t)]^\top$

$$\mathbf{F}_R(\mathbf{x}, t) = \theta \phi(\mathbf{x}, t) \quad (4)$$

where

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \theta_1^1 & \dots & \theta_1^N \\ \theta_2^1 & \dots & \theta_2^N \end{bmatrix} \quad (5)$$

The combined basis functions are

$$\phi^i(\mathbf{x}, t) = \exp\left\{-\frac{\|\mathbf{x}-\mathbf{c}_i\|}{2\sigma_i}\right\} \cos(\omega_i t + v_i), \quad i = 1, \dots, N \quad (6)$$

where \mathbf{c}_i is the center, σ_i is the width, ω_i is the tidal frequency, and v_i tidal phase. Here, we assume that the flow only contains tidal flow and biased flow, which represents the combination of high and low frequency components of flow. If ω_i equals zero, flow only has spatial variability.

Assumption 2. We assume that the heading $\psi_c(t)$ is known for all time t , and the vehicle trajectory $\mathbf{x}(t)$ can be measured or estimated for all time t .

Remark 1. While the robot is moving in the flow field, its true location may only be known occasionally. We can estimate the trajectory of the robot through localization algorithms. The localization algorithms incorporate the known locations and the heading angle command as input and produce estimated trajectories.

Given the knowledge of the trajectory $\mathbf{x}(t)$, our first goal is to estimate the parameter θ that models the flow field and the through-water speed V_R . Mathematically, let

$$\xi(t) = \begin{bmatrix} \xi_1(t) \\ \xi_2(t) \end{bmatrix} = \begin{bmatrix} \xi_1^1(t) & \dots & \xi_1^N(t) \\ \xi_2^1(t) & \dots & \xi_2^N(t) \end{bmatrix}$$

be our estimate of the parameter θ and $V_L(t)$ be our estimate for V_R , we will design a learning algorithm to generate $\xi(t)$ and $V_L(t)$ such that $\xi(t) \rightarrow \theta$ and $V_L(t) \rightarrow V_R$ as $t \rightarrow \infty$. Our second goal is to use the information of $\xi(t)$ and $V_L(t)$ to determine whether the vehicle is working properly or not.

Flow and speed estimation

Now, knowing $\xi(t)$ and $V_L(t)$, we can compute an estimate $\mathbf{z}(t)$ for the robot trajectory by integrating the following equation

$$\dot{\mathbf{z}} = \xi(t) \phi(\mathbf{z}, t) + V_L(t) \Psi_c + \beta(t) \quad (7)$$

where $\beta(t) \in \mathbb{R}^2$ is introduced as a learning injection parameter. We call $\mathbf{e} = \mathbf{x} - \mathbf{z}$ as the controlled Lagrangian localization error (CLLE). The CLLE models how much the estimated trajectory is deviated from the identified trajectory. A learning algorithm will then compute $\beta(t)$, $\xi(t)$, and $V_L(t)$ so that the CLLE can be reduced.

We first derive the CLLE dynamics by subtracting equation (7) from equation (1) to obtain

$$\dot{\mathbf{e}} = \dot{\mathbf{x}} - \dot{\mathbf{z}} = \theta \phi(\mathbf{x}, t) - \xi(t) \phi(\mathbf{z}, t) + (V_R - V_L(t)) \Psi_c - \beta(t) \quad (8)$$

Note that the parameter θ is unknown, so we cannot use $\beta(t) = \theta\phi(\mathbf{x}, t) - \xi(t)\phi(\mathbf{z}, t)$ directly. Instead, we design the learning parameter injection as follows

$$\beta(t) = \xi(t)\Phi(\mathbf{x}, t) - \xi(t)\phi(\mathbf{z}, t) + K\mathbf{e} \quad (9)$$

When we plug equation (9) into equation (8), CLLE dynamics becomes

$$\dot{\mathbf{e}} = -K\mathbf{e} + (\theta - \xi(t))\phi(\mathbf{x}, t) + (V_R - V_L(t))\Psi_c \quad (10)$$

We derive CLLE dynamics under control input constraints. Let $\delta u = u - V_R$ be the saturation term of through-water velocity. We subtract equation (7) from the combination of equations (2) and (3). CLLE dynamics becomes

$$\dot{\mathbf{e}} = -K\mathbf{e} + (\theta - \xi(t))\phi(\mathbf{x}, t) + (V_R - V_L(t))\Psi_c + \delta u\Psi_c \quad (11)$$

where δu can be viewed as an additional disturbance to CLLE dynamics. We assume that δu is bounded by δu_{\max} . Then, since $\|\Psi_c\| = 1$, the term $\delta u\Psi_c$ is bounded above as $\|\delta u\Psi_c\| \leq \delta u_{\max}$. Equation (11) is used for the proposed updating rules that ensure CLLE to be ultimately bounded in spite of saturation of through-water velocity.

The learning algorithm updates parameters $\xi(t)$ and $V_L(t)$ using CLLE dynamics so that CLLE converges to zero. Let $\bar{\xi}$, $\bar{\theta}$, and $\mathbf{e} \otimes \phi \in \mathbb{R}^{2N}$ be column vectors. That is, $\bar{\xi}(t) = [\xi_1^1(t), \dots, \xi_1^N(t), \xi_2^1(t), \dots, \xi_2^N(t)]^\top$, $\bar{\theta}(t) = [\theta_1^1(t), \dots, \theta_1^N(t), \theta_2^1(t), \dots, \theta_2^N(t)]^\top$, and $\mathbf{e} \otimes \phi = [e_1\phi^1, \dots, e_1\phi^N, e_2\phi^1, \dots, e_2\phi^N]^\top$, where \otimes is the Kronecker product. We design the updating rules for estimating the parameters as follows

$$\dot{\bar{\xi}}(t) = \bar{\gamma}\mathbf{e} \otimes \phi(\mathbf{x}, t) \quad (12)$$

$$\dot{V}_L(t) = \bar{\gamma}\mathbf{e}^\top\Psi_c \quad (13)$$

where the parameter $\bar{\gamma}$ is the parameter for learning rate, which is a positive constant.

Anomaly detection

With the updating rules represented by equations (12) and (13), we obtained the estimated flow velocity and through-water speed simultaneously. The through-water speed estimate is used for a critical measure that decides whether or not abnormal vehicle motion occurs.

Assumption 3. We assume that the maximum V_{\max} and minimum V_{\min} through-water speed are known beforehand.

If the through-water speed estimate is within the range between the maximum and the minimum through-water speeds, we determine that the robot is normally operated without abnormal motion. However, when the through-water speed estimate is out of the normal range, we need to check whether the flow estimate is accurate or not to avoid false alarm.

We introduce the flow estimate $\mathbf{F}_M(t)$ as the estimated flow when the vehicle is working in normal conditions. Since we assume that anomaly only occurs occasionally, we can let $\mathbf{F}_M(t) = \mathbf{F}_L(t)$ if the estimated vehicle speed is within the normal range. When the estimated vehicle speed is out of the normal range, $\mathbf{F}_M(t)$ can be generated from the prediction based on the parameters obtained in normal conditions. In practice, there might be prior information, such as ocean models, or measurements from other sensors, available to help us generate the $\mathbf{F}_M(t)$ needed.

The prediction $\mathbf{F}_M(t)$ will be compared with the $\mathbf{F}_L(t)$ to judge whether flow estimation is accurate or not. Thus, we define the flow estimation error $\|\mathbf{F}_M(t) - \mathbf{F}_L(t)\|$ as the difference between estimated flow velocity and modeled flow velocity generated from available flow models (e.g. literature^{22,23}). Let $\hat{F}_{L_{\max}} = \max(\|\mathbf{F}_L(\tau)\|_{\tau \in [0,t]})$ be the maximum value of estimated flow speed until time t . Let $\hat{F}_{M_{\max}} = \max(\|\mathbf{F}_M(\tau)\|_{\tau \in [0,t]})$ be the maximum value of modeled flow speed until time t . We can compute a measure for the flow estimation error as

$$p_E = \frac{\|\mathbf{F}_M(t) - \mathbf{F}_L(t)\|}{2\max(\hat{F}_{L_{\max}}, \hat{F}_{M_{\max}})} \quad (14)$$

When we compare the maximum value of estimated flow speed until time t to that of modeled flow speed until time t , we select the larger value between the two maximum values to avoid numerator near zero. The value 2 in the denominator is a scale factor that makes the measure be 1 when the difference between estimated and modeled flows is maximum.

Algorithm 2 uses p_E and the through-water speed to decide whether the robot is in normal condition or not. A threshold γ_f is selected to determine whether the error measure p_E is too large. If p_E is above the threshold, then the through-water speed estimate of the robot should not be trusted. If p_E is below the threshold, then the algorithm checks whether the through-water speed is with the normal range or not.

Theoretical justification

The algorithm given by equations (9), (12), and (13) can be justified theoretically using adaptive control theory.²⁴ In particular, we can prove that the algorithm achieves error convergence, parameter convergence, and robustness against bounded uncertainties. Error convergence indicates that the identified trajectory converges to the estimated trajectory. Parameter convergence indicates that the estimated flow field and through-water speed converge to the assumed true values. Robustness indicates that the inaccuracy in these estimates is bounded if the uncertainties are bounded. The related theorems and lemmas needed for the proofs are provided in ‘‘Review of adaptive control’’ section.

Persistent excitation assumption

Let $\tilde{\phi}_1 = \begin{bmatrix} \phi_1^1 & \cdots & \phi_1^N \\ 0 & \cdots & 0 \end{bmatrix}$ and $\tilde{\phi}_2 = \begin{bmatrix} 0 & \cdots & 0 \\ \phi_2^1 & \cdots & \phi_2^N \end{bmatrix}$ be in $\mathbb{R}^{2 \times N}$. Let $w = [\tilde{\phi}_1, \tilde{\phi}_2, \Psi_c]^\top \in \mathbb{R}^{(2N+1) \times 2}$. We can construct a matrix $W(t) \in \mathbb{R}^{(2N+1) \times (2N+1)}$ as follows

$$W(t) = \int_t^{t+T} \begin{bmatrix} \phi_1^1 \phi_1^1 & \cdots & \phi_1^1 \phi_1^N & 0 & \cdots & 0 & \phi_1^1 \cos \psi_c \\ \phi_1^2 \phi_1^1 & \cdots & \phi_1^2 \phi_1^N & 0 & \cdots & 0 & \phi_1^2 \cos \psi_c \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_1^N \phi_1^1 & \cdots & \phi_1^N \phi_1^N & 0 & \cdots & 0 & \phi_1^N \cos \psi_c \\ 0 & \cdots & 0 & \phi_2^1 \phi_2^1 & \cdots & \phi_2^1 \phi_2^N & \phi_2^1 \sin \psi_c \\ 0 & \cdots & 0 & \phi_2^2 \phi_2^1 & \cdots & \phi_2^2 \phi_2^N & \phi_2^2 \sin \psi_c \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \phi_2^N \phi_2^1 & \cdots & \phi_2^N \phi_2^N & \phi_2^N \sin \psi_c \\ \cos \psi_c \phi_1^1 & \cdots & \cos \psi_c \phi_1^N & \sin \psi_c \phi_2^1 & \cdots & \sin \psi_c \phi_2^N & 1 \end{bmatrix} d\tau \quad (15)$$

where $\phi_j^i(\mathbf{x}, \tau) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{c}_j\|}{2\sigma_j}\right) \cos(\omega_j \tau + v_j)$, $i = 1 \cdots N$, $j = 1, 2$, and $T > 0$.

For parameter convergence, we need an assumption on w as follows:

Assumption 4. The w is persistently exciting. In other words, there exist positive κ_1 , κ_2 , and T such that $W(t)$

$$\text{satisfies } \kappa_2 I \geq \int_t^{t+T} W(\tau) W^\top(\tau) d\tau \geq \kappa_1 I \forall t.$$

This persistent excitation assumption is critical to prove the convergence of parameters.²⁵ When $W(t)$ is singular, the estimation errors of parameters may not converge to zero. The persistent excitation condition requires that the trajectories are traveled by the robot to spread over the area of operation. This may not be easily satisfied in practice. Hence, the parameters may not be accurately identified. We will further address this condition in the simulation and experiment sections.

Convergence proof

The convergence of CLLE using Lemma 4 is proved as follows.

Theorem 1. Using equations (12) and (13), CLLE converges to zero when time goes to infinity, that is, $\mathbf{e}(t) \rightarrow \vec{0}$ as $t \rightarrow \infty$.

Proof. Consider a candidate Lyapunov function

$$\mathcal{V} = \frac{1}{2} \left(\mathbf{e}^\top \mathbf{e} + \frac{1}{\bar{\gamma}} (\bar{\theta} - \bar{\xi}(t))^\top (\bar{\theta} - \bar{\xi}(t)) + \frac{1}{\bar{\gamma}} (V_R - V_L(t))^2 \right) \quad (16)$$

The derivative of \mathcal{V} is

$$\begin{aligned} \dot{\mathcal{V}} &= -\mathbf{e}^\top K \mathbf{e} + \mathbf{e}^\top (\theta - \xi(t)) \phi(\mathbf{x}, t) \\ &\quad + (V_R - V_L(t)) \left(\mathbf{e}^\top \Psi_c - \frac{1}{\bar{\gamma}} \dot{V}_L(t) \right) - \frac{1}{\bar{\gamma}} (\bar{\theta} - \bar{\xi}(t))^\top \dot{\bar{\xi}}(t) \end{aligned} \quad (17)$$

We know $\mathbf{e}^\top (\theta - \xi(t)) \phi(\mathbf{x}, t) = (\bar{\theta} - \bar{\xi}(t))^\top \mathbf{e} \otimes \phi(\mathbf{x}, t)$. Then, using equations (12) and (13)

$$\dot{\mathcal{V}} = -\mathbf{e}^\top K \mathbf{e} \leq 0 \quad (18)$$

where $\dot{\mathcal{V}}$ is negative semidefinite and this implies that \mathbf{e} , $\xi(t)$, and $V_L(t)$ are bounded. In addition, the second-order time derivative of \mathcal{V} satisfies

$$\begin{aligned} \ddot{\mathcal{V}} &= -2\mathbf{e}^\top K \dot{\mathbf{e}} \\ &= -2\mathbf{e}^\top K \{ (\theta - \xi(t)) \phi(\mathbf{x}, t) + (V_R - V_L(t)) \Psi_c - K \mathbf{e} \} \end{aligned} \quad (19)$$

Because Ψ_c is bounded, $\ddot{\mathcal{V}}$ is bounded, and hence, $\dot{\mathcal{V}}$ is uniformly continuous. By Lemma 4, $\lim_{t \rightarrow \infty} \dot{\mathcal{V}}(t) = 0$. Since K is the diagonal matrix, $\mathbf{e}(t) \rightarrow \vec{0}$ as $t \rightarrow \infty$.

Even if CLLE convergence is shown, the learning algorithm may not identify actual flow because multiple parameters that represent flow are identified from one type of information, which is the estimated trajectory. Thus, we prove parameter convergence to declare that the vehicle motion is accurately identified.

Theorem 2. Under the same setting of Theorem 1, $\xi(t)$ and $V_L(t)$ converge to θ and V_R , respectively, that is, $\xi(t) \rightarrow \theta$, and $V_L(t) \rightarrow V_R$ as $t \rightarrow \infty$.

Proof. Let η_1 , η_2 , and η_3 be $(\theta_1 - \xi_1(t))$, $(\theta_2 - \xi_2(t))$, and $(V_R - V_L(t))$, respectively. We rewrite equation (10) using notations η_1 , η_2 , and η_3 as follows

$$\dot{\mathbf{e}} = \tilde{\phi}_1(\mathbf{x}, t)\eta_1 + \tilde{\phi}_2(\mathbf{x}, t)\eta_2 + \Psi_c\eta_3 - K\mathbf{e} \quad (20)$$

We augment \mathbf{e} , η_1 , η_2 , and η_3 to new state variable X . Then

$$\dot{X} = A(t)X, \quad Y = CX, \quad (21)$$

$$A(t) = \begin{bmatrix} -K & \tilde{\phi}_1 & \tilde{\phi}_2 & \Psi_c \\ -\tilde{\gamma}\tilde{\phi}_1 & 0 & 0 & 0 \\ -\tilde{\gamma}\tilde{\phi}_2 & 0 & 0 & 0 \\ -\tilde{\gamma}\Psi_c^\top & 0 & 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

where 0 is the zero matrix with proper dimensions according to the components of the first row matrix of A . Our goal is to show that the origin of $\dot{X} = A(t)X$ is uniformly asymptotically stable, which implies that $\bar{\xi}(t)$ converges to $\bar{\theta}$, and $V_L(t)$ converges to V_R when time goes to infinity. By Theorem 5, we need to show that P exists and (C, A) is uniformly completely observable. Let

$$P = \begin{bmatrix} \frac{1}{2}K^{-1} & 0 & 0 & 0 \\ 0 & \frac{1}{2\tilde{\gamma}}K^{-1} & 0 & 0 \\ 0 & 0 & \frac{1}{2\tilde{\gamma}}K^{-1} & 0 \\ 0 & 0 & 0 & \frac{1}{2\tilde{\gamma}}K^{-1} \end{bmatrix}. \quad (22)$$

Let \mathcal{V}_1 be $X^\top P X$. Then, $\dot{\mathcal{V}}_1 = X^\top (A^\top P + P^\top A + \dot{P})X \leq -\nu X^\top C^\top C X = -\nu \|Y\|^2$, where $\dot{P} = 0$. Thus, there exists a symmetric matrix $P(t)$ and positive constants c_1 and c_2 such that $c_1 I \leq P(t) \leq c_2 I$ and $A(t)^\top P(t) + P(t)A(t) + \dot{P}(t) + \nu C(t)^\top C(t) \leq 0$. Now, we will prove (C, A) is a uniformly completely observable. Because it is difficult to prove the observability of time varying system matrix A , we will instead show $(C, A + LC)$ is uniformly completely observable with some bounded matrix L called output injection

by Lemma 1. Let $L = \begin{bmatrix} K & 0 & 0 & 0 \\ \tilde{\gamma}\tilde{\phi}_1 & 0 & 0 & 0 \\ \tilde{\gamma}\tilde{\phi}_2 & 0 & 0 & 0 \\ \tilde{\gamma}\Psi_c^\top & 0 & 0 & 0 \end{bmatrix}$. Since Ψ_c is

bounded, and $\tilde{\phi}$ is a sinusoidal function with exponential magnitude, L is bounded. Then, $A + LC =$

$$\begin{bmatrix} 0 & \tilde{\phi}_1 & \tilde{\phi}_2 & \Psi_c \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad \text{Thus}$$

$$\begin{aligned} \dot{X} &= AX = (A + LC)X - LY \\ Y &= CX \end{aligned} \quad (23)$$

Let $\boldsymbol{\eta} = [\eta_1, \eta_2, \eta_3]^\top$. We have the following equation corresponding to equation (23)

$$\begin{aligned} \dot{\mathbf{e}} &= -K\mathbf{e} + w^\top \boldsymbol{\eta} \\ \dot{\boldsymbol{\eta}} &= 0 \\ Y &= \mathbf{e} \end{aligned} \quad (24)$$

By Assumption 4, w is persistently exciting. Let $\Phi(\tau) = \int_t^{\tau} \exp^{-K(\tau-\sigma)} w(\sigma) d\sigma$ be the output of equation (24) given the input w . By Lemma 2, $\Phi(\tau)$ satisfies persistently exciting conditions because $w(\sigma)$ is persistently exciting, and the transfer function of equation (24), $(sI_{2 \times 2} + K)^{-1}$, is a stable, minimum phase, proper rational transfer function. Therefore, there exists constant $\rho_1, \rho_2, T_0 > 0$ such that $\rho_2 I \geq \frac{1}{T_0} \int_t^{t+T_0} \Phi(\tau)\Phi^\top(\tau) d\tau \geq \rho_1 I \quad \forall t \geq 0$. By applying Lemma 1 to the system of equation (24), $(C, A + LC)$ is uniformly completely observable; hence, the system of equation (21) is uniformly completely observable. Therefore, the origin of $\dot{X} = A(t)X$ is uniformly asymptotically stable; that is, $X \rightarrow \vec{0}$ as $t \rightarrow \infty$. This means that η_1, η_2 , and η_3 go to zeros individually. Thus, $\bar{\xi}(t)$ and $V_L(t)$ converge to $\bar{\theta}$ and V_R , respectively.

Input constraints

The marine robots have limited power to control their motions. The control power is saturated by the maximum capacity of hardware such as motors and thrusters. This induces constraints to robot control in the ocean. Since equation (11) includes one saturated term represented by $\delta u \Psi_c$, which shows the discrepancy between input and output of the saturator, we reject additional disturbance $\delta u \Psi_c$ from saturation in equation (11) using a scheme in the literature.²⁶ We generate additional signal \mathbf{e}_δ governed by a differential equation as follows

$$\dot{\mathbf{e}}_\delta = -K\mathbf{e}_\delta + \delta u \Psi_c \quad (25)$$

Let $\boldsymbol{\varepsilon} = \mathbf{e} - \mathbf{e}_\delta$ be the difference between CLLE and the additional signal. When subtracting equation (25) from equation (11), we have

$$\begin{aligned} \dot{\boldsymbol{\varepsilon}} &= \dot{\mathbf{e}} - \dot{\mathbf{e}}_\delta \\ &= (\theta - \xi(t))\phi(\mathbf{x}, t) + (V_R - V_L(t))\Psi_c - K\boldsymbol{\varepsilon} \end{aligned} \quad (26)$$

To make $\boldsymbol{\varepsilon}$ go to zero, we replace the updating rules for time-varying parameters $\bar{\xi}$ and V_L in equations (12) and (13) by the following two equations

$$\dot{\bar{\xi}} = \tilde{\gamma}\boldsymbol{\varepsilon} \otimes \phi(\mathbf{x}, t) \quad (27)$$

$$\dot{V}_L = \tilde{\gamma}\boldsymbol{\varepsilon}^\top \Psi_c \quad (28)$$

Algorithm 1. Flow and speed estimation algorithm

Input: Vehicle trajectory $\mathbf{x}(t)$, estimated position $\mathbf{z}(t)$, heading angle $\psi_c(t)$, learning rate $\bar{\gamma}$, initial estimated parameter $\bar{\xi}(0)$, initial estimated speed $V_L(0)$, constant sampling time T_s , a diagonal matrix with positive constant components K

Output: Estimated parameter $\bar{\xi}(t+1)$, estimated flow $\bar{\xi}(t+1)\phi(\mathbf{z}(t+1), t+1)$, estimated speed $V_L(t+1)$

```

1 repeat
2    $\mathbf{e}(t) = \mathbf{x}(t) - \mathbf{z}(t)$ 
3    $\beta(t) = \bar{\xi}(t)\phi(\mathbf{x}(t), t) - \bar{\xi}(t)\phi(\mathbf{z}(t), t) + K\mathbf{e}(t)$ 
4    $\bar{\xi}(t+1) = \bar{\xi}(t) + \bar{\gamma}\mathbf{e}(t) \otimes \phi(\mathbf{x}(t), t)T_s$ 
5    $\mathbf{z}(t+1) =$ 
6      $\mathbf{z}(t) + (\bar{\xi}(t)\phi(\mathbf{z}(t), t) + V_L(t)\Psi_c(t) + \beta(t))T_s$ 
7    $V_L(t+1) = V_L(t) + \bar{\gamma}\mathbf{e}(t)^\top\Psi_c(t)T_s$ 
7 until  $t = \text{Ending Time is met}$ 

```

Algorithm 2. Anomaly detection algorithm

Input: Flow velocity estimate $F_L(t)$, modeled flow estimate $F_M(t)$, false alarm factor γ_f , controlled speed estimate $V_L(t)$, maximum speed V_{\max} , minimum speed V_{\min}

Output: Anomaly detection flag

```

1  $\hat{F}_{L_{\max}} = \max(F_L(\tau)_{\tau \in [0, t]})$ 
2  $\hat{F}_{M_{\max}} = \max(F_M(\tau)_{\tau \in [0, t]})$ 
3  $p_E = \frac{\|F_L(t) - F_M(t)\|}{2\max(\hat{F}_{L_{\max}}, \hat{F}_{M_{\max}})}$ 
4 if  $p_E > \gamma_f$  then
5   | flag = 2  $\triangleright$  False Alarm
6 else if  $V_L(t) > V_{\max}$  or  $V_L(t) < V_{\min}$  then
7   | flag = 1  $\triangleright$  Anomaly Detected
8 else
9   | flag = 0  $\triangleright$  No Anomaly Detected
10 end

```

The corresponding equations (lines 3 and 4) in Algorithm 1 will also be changed to use the error signal \mathbf{e} and \mathbf{e}_δ

Theorem 3. Under the update rules (27) and (28), CLLE is ultimately bounded

$$\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| \leq \frac{\delta u_{\max}}{q} \quad (29)$$

where the positive constant $q < 1$.

Proof. Let $\mathcal{V}_2 = \frac{1}{2}\mathbf{e}_\delta^\top K^{-1}\mathbf{e}_\delta$. The derivative of \mathcal{V}_2 is $\dot{\mathcal{V}}_2 = -\mathbf{e}_\delta^\top \mathbf{e}_\delta + \mathbf{e}_\delta^\top \delta u \Psi_c$. Then, $\dot{\mathcal{V}}_2 \leq -(1-q)\|\mathbf{e}_\delta\|^2 - q\|\mathbf{e}_\delta\|^2 + \|\mathbf{e}_\delta\|\delta u_{\max}$. When $\|\mathbf{e}_\delta\| \geq \frac{\delta u_{\max}}{q}$ given positive constant $q < 1$, $\dot{\mathcal{V}}_2 \leq -(1-q)\|\mathbf{e}_\delta\|^2$. This means $\dot{\mathcal{V}}_2$ is not positive. Thus, $\|\mathbf{e}_\delta\| \leq \frac{\delta u_{\max}}{q}$.

We will show \mathbf{e} goes to zero when time goes infinity by the following candidate Lyapunov function

$$\mathcal{V}_3 = \frac{1}{2} \left(\mathbf{e}^\top \mathbf{e} + \frac{1}{\bar{\gamma}} (\bar{\theta} - \bar{\xi})^\top (\bar{\theta} - \bar{\xi}) + \frac{1}{\bar{\gamma}} (V_R - V_L)^2 \right) \quad (30)$$

By using equations (27) and (28), $\dot{\mathcal{V}}_3 = -\mathbf{e}^\top K \mathbf{e} \leq 0$. $\dot{\mathcal{V}}_3$ is negative semidefinite and this implies that \mathbf{e} , $\bar{\xi}$, and V_L are bounded. In addition, $\dot{\mathcal{V}}_3 = -2\mathbf{e}^\top K \dot{\mathbf{e}} = -2\mathbf{e}^\top K \{(\bar{\theta} - \bar{\xi})\phi(\mathbf{x}, t) + (V_R - V_L)\Psi_c\}$. Since $\mathbf{e} = \mathbf{e}_\delta + \mathbf{e}$, \mathbf{e} is bounded. This implies that \mathbf{x} is bounded. In addition, $\bar{\xi}$ and V_L are bounded. Thus, $\dot{\mathcal{V}}_3$ is bounded, and then, $\dot{\mathcal{V}}_3$ is uniformly continuous. By Lemma 4, $\lim_{t \rightarrow \infty} \dot{\mathcal{V}}_3(t) = 0$. Since K is the diagonal matrix, $\mathbf{e} \rightarrow \bar{\mathbf{0}}$ when $t \rightarrow \infty$; $\mathbf{e} \rightarrow \mathbf{e}_\delta$ when $t \rightarrow \infty$. Thus, CLLE is ultimately bounded.

Inaccuracy in flow modeling

Although the spatial basis function well captures the spatial variability of actual flows in a specific region, the function still includes deterministic errors induced by the variability out of the region. In this section, we address the robustness of the proposed adaptive learning algorithm.

To show that the proposed algorithm is robust to disturbance in the flows, we prove the boundedness of CLLE when the actual flow model has unknown disturbances. We assume $\mathbf{F}_R(\mathbf{x}, t) = \theta\phi(\mathbf{x}, t) + \Delta$, where $\|\Delta\|$ is bounded by $\Delta_{\max} \in \mathbb{R}$. Then

$$\dot{\mathbf{e}} = (\bar{\theta} - \bar{\xi}(t))\phi(\mathbf{x}, t) + (V_R - V_L(t))\Psi_c - K\mathbf{e} + \Delta \quad (31)$$

The theorem of robustness is proved below.

Theorem 4. Under the same setting of Theorem 1 and $\mathbf{F}_R(\mathbf{x}, t) = \theta\phi(\mathbf{x}, t) + \Delta$, CLLE is ultimately bounded

$$\lim_{t \rightarrow \infty} \|\mathbf{e}(t)\| \leq \frac{1}{v} \Delta_{\max} \quad (32)$$

where the positive constant $v < \lambda_{\min}(K)$ and $\|\Delta\|$ is bounded by Δ_{\max} .

Proof. Let \mathcal{V}_4 be the Lyapunov function represented by equation (16). Using equation (31), the derivate of \mathcal{V}_4 is

$$\begin{aligned} \dot{\mathcal{V}}_4 = & -\mathbf{e}^\top K \mathbf{e} + \mathbf{e}^\top \Delta + (V_R - V_L(t))^\top \left(\frac{1}{\bar{\gamma}} \dot{V}_L(t) - \mathbf{e}^\top \Psi_c \right) \\ & + \frac{1}{\bar{\gamma}} (\bar{\theta} - \bar{\xi}(t)) \left(\frac{1}{\bar{\gamma}} \dot{\bar{\xi}}(t) - \mathbf{e} \otimes \phi(\mathbf{x}, t) \right) \end{aligned} \quad (33)$$

We plug the updating rules represented by equations (12) and (13) into equation (33). Then

$$\begin{aligned} \dot{\mathcal{V}}_4 = & -\mathbf{e}^\top K \mathbf{e} + \mathbf{e}^\top \Delta \\ & \leq -\lambda_{\min}(K)\mathbf{e}^\top \mathbf{e} + \mathbf{e}^\top \Delta \leq -\lambda_{\min}(K)\|\mathbf{e}\|^2 + \|\mathbf{e}\|\|\Delta\| \\ & \leq -(\lambda_{\min}(K) - v)\|\mathbf{e}\|^2 + \|\mathbf{e}\|\|\Delta\| - v\|\mathbf{e}\|^2 \end{aligned}$$

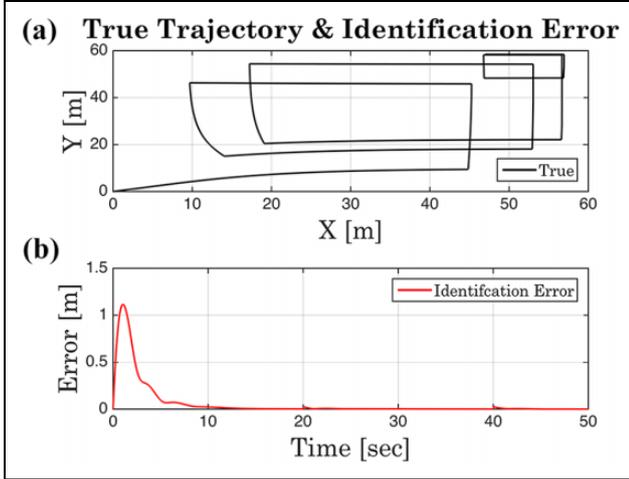


Figure 1. (a) A trajectory under true flow is shown, and (b) the identification error for a simulated trajectory using flow identification is shown. The error is measured by the root-mean-square error between true and simulated trajectories. The identification error converges to 20 s and approaches close to 0 after 100 s.

When $\|\mathbf{e}\| \geq \frac{1}{v} \|\Delta\|$ given positive constant $v < \lambda_{\min}(K)$, $\dot{\mathcal{V}}_4 \leq -(\lambda_{\min}(K) - v) \|\mathbf{e}\|^2$, which means $\dot{\mathcal{V}}_4$ is negative definite. Thus, CLLE is ultimately bounded. The bound of CLLE is $\|\mathbf{e}\| \leq \frac{1}{v} \|\Delta\|_{\max}$.

Simulations

This section describes simulation results for the anomaly detection algorithms in the section of anomaly detection.

For the presentation of 2D ocean flow, θ_1^i is selected as the true flow parameter along the horizontal direction and θ_2^i is the true flow parameter along the vertical direction, where $i = 1, 2, 3$. The three combined basis functions are composed of center c_i , width σ_i , harmonic frequency ω_i , and harmonic phase ν_i , where $i = 1, 2, 3$. Those harmonic periods are chosen by major tidal components M_2 , N_2 , and S_2 . Harmonic phases are zeros. For the anomaly detection algorithm, positive constant K in the learning parameter injection term is the identity matrix. The learning rate $\bar{\gamma}$ is 0.8. In the prevention scheme of false alarm, the factor γ_f is 0.07.

Figure 1 shows trajectories of a marine robot when the direction of the marine robot in the horizontal plane is controlled by heading angle command $\Psi_c = \frac{\pi}{2} \frac{t}{20}$. Every 20 s, the heading angle will increase 90° . In Figure 1, the simulated true trajectory represented by the black line would follow one square if there was no flow. However, because flow with spatial and temporal variabilities affects vehicle motion, the true trajectory has multiple squares. The vehicle trajectory satisfies the persistent excitation condition.

The flow estimate \mathbf{F}_L , which is $\xi(t)\phi(\mathbf{z}(t), t)$, is generated by Algorithm 1. When \mathbf{x} and \mathbf{x} are all (0,0) at $t = 0$ in Figure 1, Algorithm 1 starts to update the flow estimate with initial conditions of $\xi(t)$, as given in Table 1.

Table 1. Parameters for simulations.

Parameters	$i = 1$	$i = 2$	$i = 3$
True flow θ_1^i	0.9	0.5	0.7
True flow θ_2^i	0.8	0.5	0.9
Learned flow ξ_1^i at $t = 0$	0.17	0.2	0.05
Learned flow ξ_2^i at $t = 0$	0.07	0.1	0.1
Centers c_i	$[0, 0]^T$	$[10, 10]^T$	$[5, 5]^T$
Width σ_i	5	5	5
Harmonic frequency ω_i	12.42 h	12.66 h	12 h
Harmonic phase ν_i	0	0	0

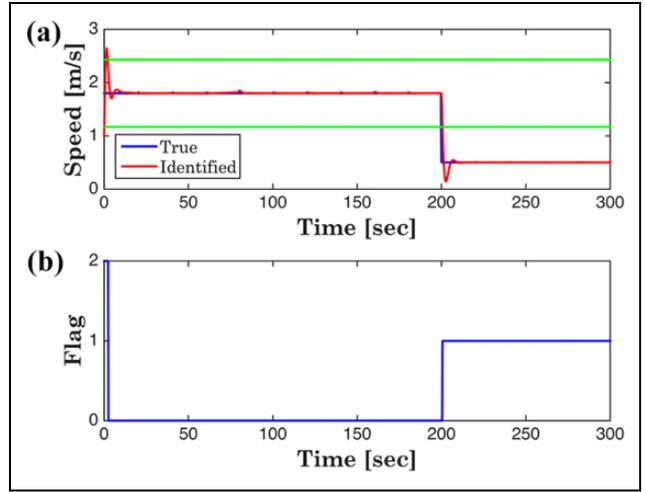


Figure 2. (a) True through-water speed and (b) flag.

Figure 2 shows simulation results of through-water speed and anomaly detection. In Figure 2(a), two green lines represent the upper and lower bound of normal through-water speed, respectively. When actual through-water speed is reduced to 0.5 m/s after 200 s due to abnormal motion, the learning algorithm keeps tracking actual through-water speed until 300 s.

The anomaly detection algorithm changes the value of flag to indicate the status of the simulated robot. The flag value is 2 within the first 10 s, which indicates that a false alarm may happen due to the inaccuracy of identified flow in the initial transient period. The flag value changes to 0 after 10 s because the flow identification error has decreased significantly, as shown in Figure 3, and the identified through-water speed is within the normal range, as shown in Figure 2(a). The flag value switches from 0 to 1 at 200 s because the identified through-water speed is out of the normal range, while the flow identification error is small in Figure 3.

Experiments and results

The verification of anomaly detection algorithms on marine robots operating in the ocean typically requires significant effort and resource. In addition, generating faults

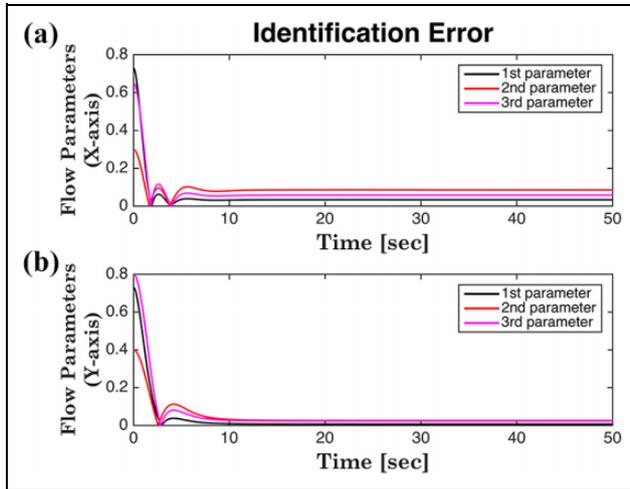


Figure 3. Identification error of flow parameters converges after 20 sec. After 100 s, the identification error converges close to zero. (a) The identification error for the parameters modeling the X-axis component of the flow is shown, and (b) the identification error for the parameters modeling the Y-axis component of the flow is shown.

on operational marine robots will induce risk of vehicle loss. Therefore, there is a need for an indoor testbed so that experiments can be carried under controlled conditions with reduced cost and risk. For this purpose, we develop the GT-MAB and the GT-WMR, which are used to collect experimental data on anomaly detection proposed in this article.

Indoor testbed development

The GT-MAB²⁷ has dynamics that are similar to the dynamics of underwater vehicles.²⁸⁻³¹ The lighter gas in the GT-MAB induces buoyancy, which plays the same role in restoring force and moment of the GT-MAB as underwater vehicles. The GT-MAB is subjected to significant air dynamic influences, which can be leveraged to emulate flow influences on underwater vehicles.

To generate flow that affects the motion of the GT-MAB, a Dyson fan is used as an artificial wind source. The Dyson fan creates more consistent flows along the direction of blowing wind than rotating fans that produce inconsistent flows. Utilizing the GT-MAB and the Dyson fan, we establish an indoor testbed shown in Figure 4.

Measuring flow generated from the Dyson fan is necessary for algorithm verification. We develop a wind measuring robot (WMR) to measure the wind field of the indoor environment. The GT-WMR collects low wind speed measurements ranged from 0 m/s to 4 m/s in all directions at each location autonomously. As shown in Figure 5, the GT-WMR integrates three wind speed sensors, each with different range and accuracy, on an omnidirectional robot. The GT-WMR autonomously moves along predetermined

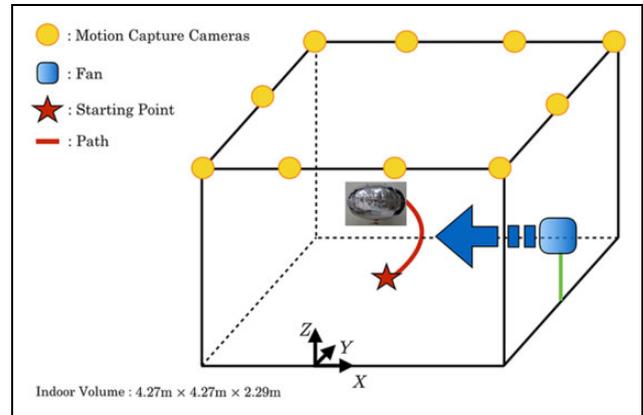


Figure 4. Indoor testbed: The yellow bulbs represent infrared motion capture cameras. The blue square represents the Dyson fan. The star represents the starting point of the GT-MAB. The red line represents the trajectory of the GT-MAB. When the GT-MAB is flying at the starting point, the GT-MAB motion is disturbed by flow generated from the Dyson fan. Then, the motion capture cameras collect the attitudes and trajectory of the GT-MAB. GT-MAB: Georgia Tech Miniature Autonomous Blimp.

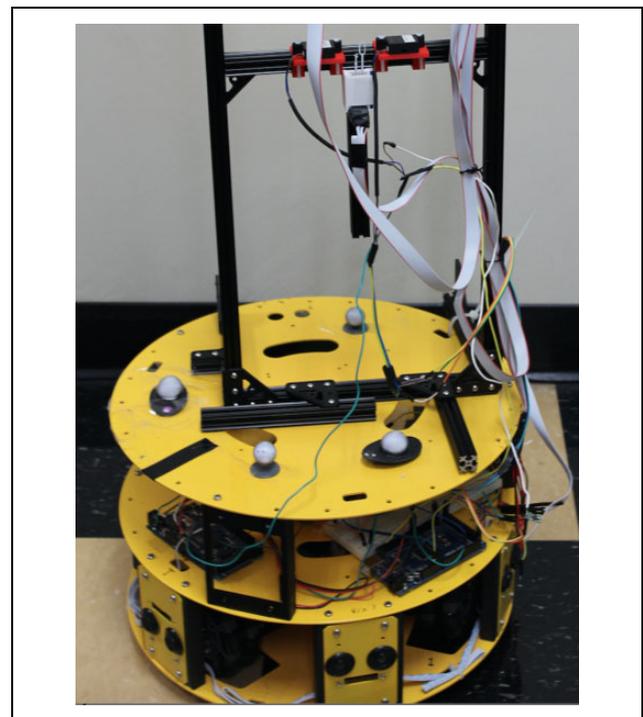


Figure 5. The GT-WMR contains two main components: an omnidirectional robot called omnibot and three wind sensors. The three wind sensors on a horizontal black frame are connected to the omnibot. The black frame can be moved vertically to measure wind speed at different heights. GT-WMR: Georgia Tech Wind Measuring Robot.

waypoints, and then, the GT-WMR collects wind measurements at an altitude, where wind sensors are fixed on the omnidirectional robot. These wind measurements enable us to identify the flow field of the Dyson fan.

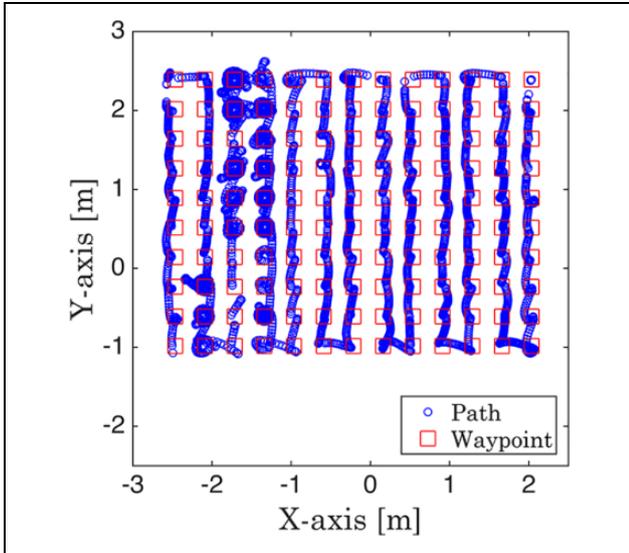


Figure 6. The path (blue) and waypoints (red) of the GT-WMR. GT-WMR: Georgia Tech Wind Measuring Robot.

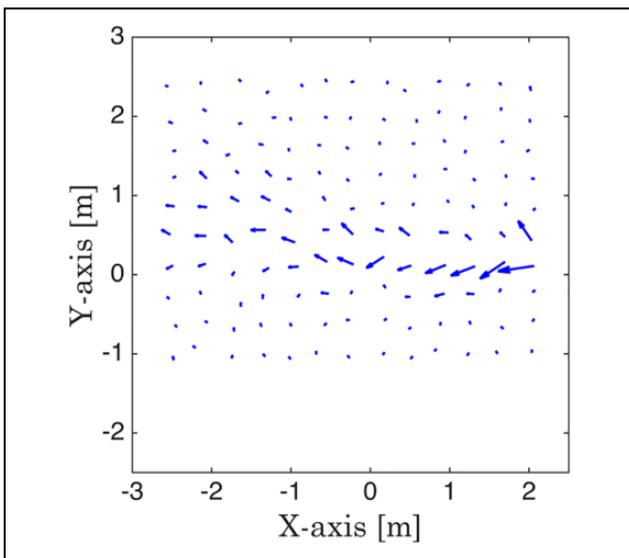


Figure 7. Measured flow velocity at each waypoint.

GT-WMR data

To densely collect wind measurements in the indoor testbed, we determine multiple waypoints that can generate the GT-WMR path having a type of lawnmower pattern. Figure 6 shows the waypoints and the GT-WMR's path, and Figure 7 shows measured wind velocity at each waypoint.

When the GT-WMR controlled by a waypoint controller accurately reaches one waypoint shown in Figure 6, the GT-WMR stays at the waypoint and changes its orientation with 30° increment starting from 0° to 360° . The wind speed sensors will collect measurements for each of the orientations. The direction where the wind speed is the maximum is chosen as the direction of the wind. Then,

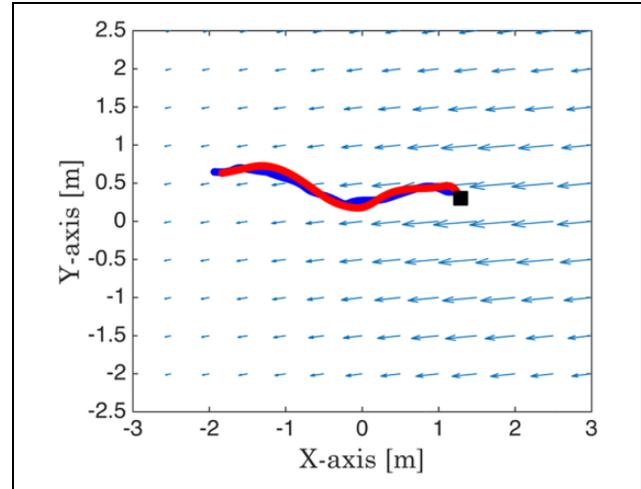


Figure 8. Estimated (blue) and identified (red) trajectories starting at no. 3 (black).

the GT-WMR moves to the next waypoint and repeats the measurement process. The measured wind field is plotted in Figure 7. The largest measurement of wind speed is acquired near the wind source. In addition, the width of the area with strong wind is around 0.2 m. This value is similar to the diameter of the Dyson fan, which is 0.254 m. These measurements are consistent with the Dyson fan.

GT-MAB data

We can measure the through-air speed of the GT-MAB, where there is no wind, as 0.0185 m/s. When the Dyson Fan is on, we use the adaptive learning algorithm to identify airflow along the GT-MAB trajectory using the adaptive learning algorithm. Comparing the wind speed measurements with the GR-WMR data will validate the learning algorithm.

For the adaptive learning algorithm, we design four spatial basis functions composed of center c_i and width σ_i , where $i = 1, 2, 3, 4$. c_1, c_2, c_3 , and c_4 are $[1.5594, 0.3]^T$, $[2.0594, 0.3]^T$, $[2.5594, 0.3]^T$, and $[1.5594, -1.5]^T$, respectively. $\sigma_1, \sigma_2, \sigma_3$, and σ_4 are all equal to 1. Time-varying basis functions with harmonic frequencies and phases are removed because of consistent flow from the Dyson fan.

When GT-MAB is operating in normal conditions, the adaptive learning algorithm is applied to identify the flow field. Figure 8 shows measured and identified trajectories of the GT-MAB. At the individual starting point represented by one black square, we select one waypoint along the Y-axis for the forward or backward motion of the GT-MAB. The blue arrows, which represent wind identified by the adaptive learning algorithm, move away from the wind source. The direction of an identified flow corresponds to the wind direction generated from the wind source. Figures 9 and 10 show measured and identified trajectories of the GT-MAB from different starting positions. The magnitude of the blue arrows at the place closing

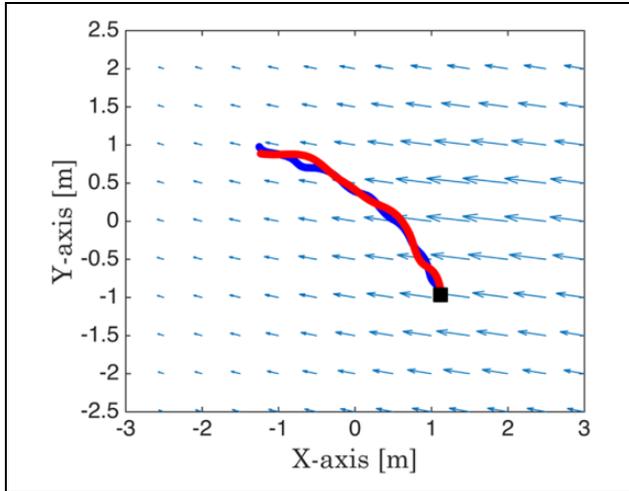


Figure 9. Estimated (blue) and identified (red) trajectories starting at no. 12 (black).

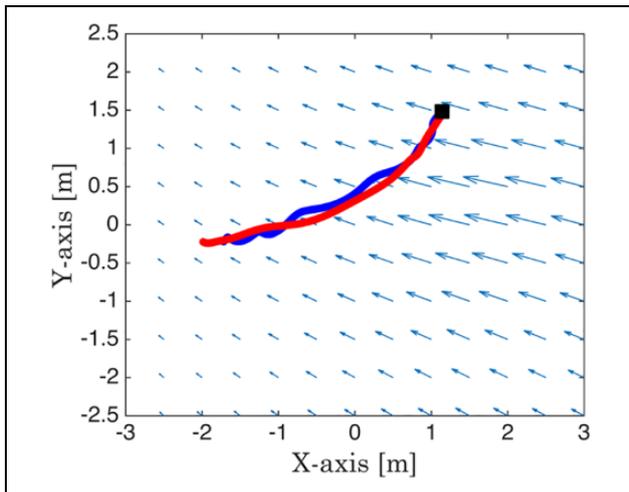


Figure 10. Estimated (blue) and identified (red) trajectories starting at no. 15 (black).

to the wind source is larger than at other places; this tendency corresponds to the fact that strong wind is generated at positions close to the wind source.

Verifying the adaptive learning algorithm is significantly difficult by simply comparing Figures 8 to 10 with Figure 7. Although wind velocities are identified in the entire space of the testbed, as shown in Figures 8 to 10, the accuracy of the wind velocities is not consistent. Because using constant heading angle command and constant spatial basis functions does not satisfy persistent excitation condition in equation (15), flow parameters for the identified flow may be inaccurately identified. But, nevertheless, this identification is necessary for anomaly detection.

Anomaly detection

The flow estimate \mathbf{F}_L , which is $\xi(t)\phi(\mathbf{z}(t), t)$, is generated by Algorithm 1. This algorithm updates $\xi(t)$ with its zero

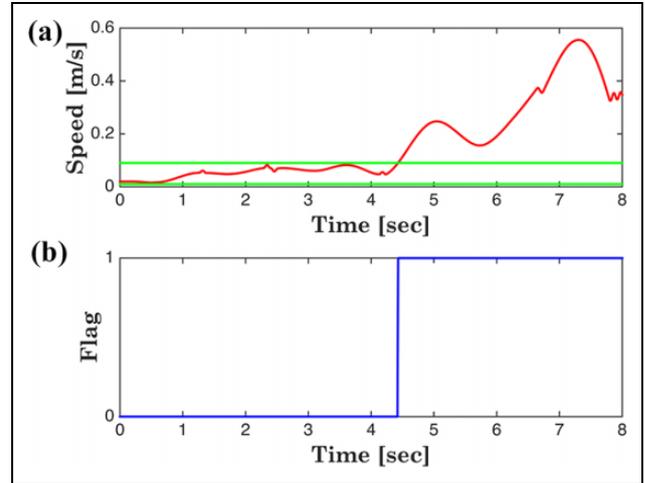


Figure 11. (a, b) Identified through-air speed along the GT-MAB trajectories. GT-MAB: Georgia Tech Miniature Autonomous Blimp.

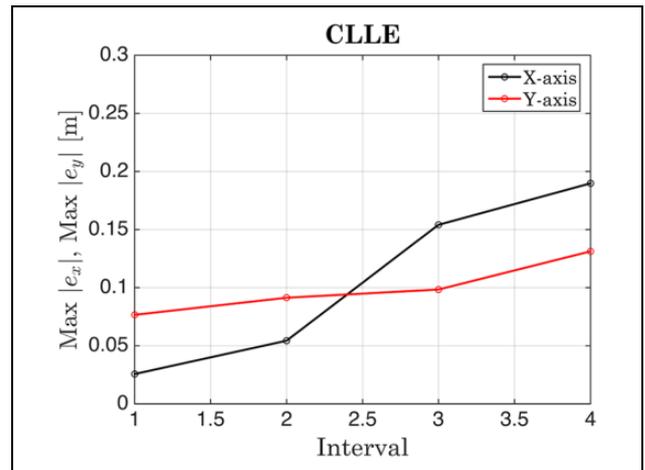


Figure 12. CLLE along the X-axis (black), and CLLE along the Y-axis (red). CLLE: controlled Lagrangian localization error.

initial condition. In addition, we use the flow field measured by the GT-WMR to generate $\mathbf{F}_M(t)$ for anomaly detection algorithms. We intentionally introduced errors in $\mathbf{F}_M(t)$ to hide the ground truth from the algorithm to test its robustness.

To generate the anomaly, we intentionally disabled a thruster on the GT-MAB to generate a fault and applied the anomaly detection algorithms. The flow parameters are identified by the learning algorithm at the same time. The fault occurs at time $t = 4$ s in Figure 11, which plots the through-air speed (Figure 11(a)) and the flags generated by the algorithm (Figure 11(b)). The normal speed range is represented by two green lines. We see that after $t = 4.43$ s, the through-air speed estimate escapes from the range. Flag 0 is maintained until 4.43 s showing no anomaly is detected. Flag 1 indicates that anomaly occurs because the through-air speed estimate is out of the

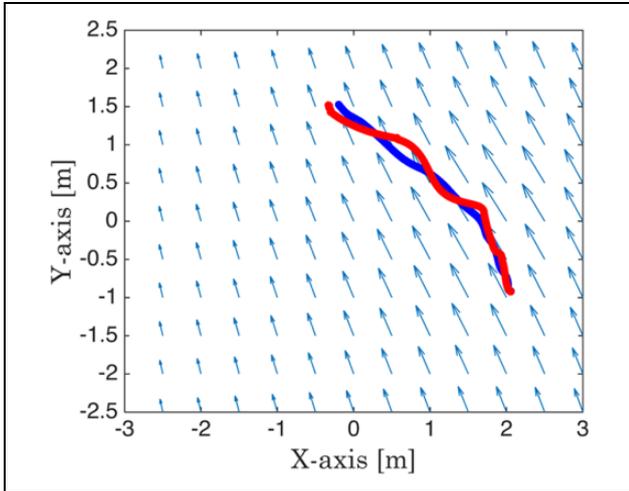


Figure 13. Estimated (blue) and identified (red) trajectories with the identified flow (blue arrow).

range. Figures 12 and 13 show CLLE and the GT-MAB trajectory, respectively.

Conclusion

The main contribution of this article is two algorithms that detect anomaly of marine robots without sensors monitoring vehicle components: adaptive learning and anomaly detection algorithms. Only using trajectory information, the proposed strategy detects abnormal vehicle motion under unknown ocean flow. It has the potential for mitigating abnormal vehicle motion with path-planning and controller design of marine robots. The experimental results of the GT-MAB and GT-WMR in an indoor testbed verify the proposed strategy. Future work will extend the proposed algorithms to the 3D space for reflecting upwelling flow dynamics. Furthermore, we will incorporate the estimated trajectory from acoustic localization into the proposed algorithms.

Review of adaptive control

We review theorems and lemmas from the theory of adaptive control²⁴ that are needed for the proofs.

Let $A(t) \in \mathbb{R}^{n \times n}$, $C(t), L(t) \in \mathbb{R}^{n \times l}$, $X(t) \in \mathbb{R}^{n \times 1}$, and $Y(t) \in \mathbb{R}^{l \times 1}$ be matrices that satisfy the following equation

$$\dot{X}(t) = A(t)X(t), \quad Y(t) = C^T(t)X(t) \quad (34)$$

Theorem 5. A necessary and sufficient condition for the uniformly asymptotically stability of the equilibrium of $\dot{X}(t) = A(t)X(t)$ is that there exists a symmetric matrix $P(t)$ such that both $c_1 I \leq P(t) \leq c_2 I$ and $A(t)^T P(t) + P(t)A(t) + \dot{P}(t) + \nu C(t)^T C(t) \leq 0$ are satisfied $\forall t$ and some constant $\nu > 0$, where $c_1 > 0$, and $c_2 > 0$ and $C(t)$ is such that $(C(t), A(t))$ is uniformly completely observable.

Definition 1.^{32,33} A vector signal u is persistently exciting if there exist positive constants κ_1, κ_2 , and T such that

$$\kappa_2 I \geq \int_t^{t+T} u(\tau)u^T(\tau)d\tau \geq \kappa_1 I \forall t.$$

Lemma 1. Assume that there exist constants $\nu > 0$, $k_\nu \geq 0$ such that for all $t_0 \geq 0$, $L(t)$ satisfies the inequality $\int_{t_0}^{t_0+\nu} \|L(\tau)\|^2 d\tau \leq k_\nu$. Then, system $(C(t), A(t))$ is a uniformly completely observable if and only if system $(C(t), A(t) + L(t)C(t)^T)$ is a uniformly completely observable.

Lemma 2. If $u : \mathbb{R}^+ \mapsto \mathbb{R}^n$ is persistently exciting, $u \in L_\infty$, $\dot{u} \in L_\infty$, and $H(s)$ is a stable, minimum phase, proper rational transfer function, then $u' = H(s)u$ is persistently exciting.

Lemma 3. Consider system $\dot{Y}_1 = A_c Y_1 - B_c \phi^T Y_2$, $\dot{Y}_2 = 0$, and $y_0 = C_c^T Y_1$, where A_c is a stable matrix, (C_c, A_c) is observable, and $\phi \in L_\infty$. If ϕ_f defined as $\phi_f \triangleq C_c^T (sI - A_c)^{-1} B_c \phi$ satisfies $\alpha_1 I \leq \frac{1}{T_0} \int_t^{t+T_0} \phi_f(\tau)\phi_f^T(\tau) \leq \alpha_2 I, \forall t \geq 0$ for constants $\alpha_1, \alpha_2, T_0 > 0$, then, the system is uniformly completely observable.

Lemma 4.^{34,35} If g is a real function of real variable t , defined and uniformly continuous for $t \geq 0$, and if the limit of the integral $\int_0^t g(s)ds$ as t tends to infinity exists and is a finite number, then $\lim_{t \rightarrow \infty} g(t) = 0$.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by ONR grants N00014-19-1-2556, N00014-19-1-2266 and N00014-16-1-2667; NSF grants OCE-1559475, CNS-1828678, and S&AS-1849228; NRL grants N00173-17-1-G001 and N00173-19-P-1412; and NOAA grant NA16NOS0120028.

ORCID iDs

Sungjin Cho  <https://orcid.org/0000-0002-5694-310X>

Fumin Zhang  <https://orcid.org/0000-0003-0053-4224>

Catherine R Edwards  <https://orcid.org/0000-0001-6486-0948>

References

1. Stanway MJ, Kieft B, Hoover T, et al. *White shark strike on a long-range AUV in Monterey bay*. In: *Oceans 2015 MTS/IEEE*, Genova, Italy, 18–21 May 2015.
2. Gertler JJ. *Fault detection and diagnosis in engineering systems*. Boca Raton, FL: CRC Press, 1999.
3. Chen J and Patton RJ. *Robust model-based fault diagnosis for dynamic systems*. Berlin: Springer, 1998.

4. Morris BT and Trivedi MM. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Trans Circuit Syst Video Technol* 2008; 18(8): 1114–1127.
5. Lei PR. A framework for anomaly detection in maritime trajectory behavior. *Knowl Inform Syst* 2016; 47: 189–214.
6. Rosén O and Medvedev A. An on-line algorithm for anomaly detection in trajectory data. In: *American control conference*, Montreal, QC, Canada, 27–29 June 2012, pp. 1117–1122.
7. Park D, Erickson Z, Bhattacharjee T, et al. Multimodal execution monitoring for anomaly detection during robot manipulation. In: *2016 IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp. 407–414.
8. Raanan BY, Bellingham JG, Zhang Y, et al. Automatic fault diagnosis for autonomous underwater vehicles using online topic models. In: *Oceans 2016 MTS/IEEE*, Monterey, CA, USA, 19–23 September 2016.
9. Isermann R. Model-based fault-detection and diagnosis – status and applications. *Annu Rev Control* 2005; 29: 71–85.
10. Chandola V, Banerjee A, and Kumar V. Anomaly detection: a survey. *ACM Comput Surveys (CSUR)* 2009; 41(3): 15.
11. Fagogenis G, Carolis VD, and Lane DM. Online fault detection and model adaptation for underwater vehicles in the case of thruster failures. In: *2016 IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp. 2625–2630.
12. Sun Y, Ran X, Li Y, et al. Thruster fault diagnosis method based on gaussian particle filter for autonomous underwater vehicles. *Int J Nav Archit Ocean Eng* 2016; 8: 243–251.
13. Caiti A, Corato FD, Fabiani F, et al. Enhancing autonomy: fault detection, identification and optimal reaction for over-actuated AUVs. In: *Oceans 2015 MTS/IEEE*, Genova, Italy, 18–21 May 2015.
14. Bono R, Bruzzone G, and Caccia M. Experiences on actuator fault detection, diagnosis and accommodation for ROVs. In: *International symposium of unmanned untethered submersible technology*, Durham: University of New Hampshire, Marine Systems Engineering Laboratory, 2001, p. 11.
15. Raanan BY, Bellingham J, Zhang Y, et al. A real-time vertical plane flight anomaly detection system for a long range autonomous underwater vehicle. In: *Oceans 2015 MTS/IEEE*, Washington, DC, USA, 19–22 October 2015.
16. Zhang F, Fratantoni DM, Paley DA, et al. Control of coordinated patterns for ocean sampling. *Int J Control* 2007; 80(7): 1186–1199.
17. Yuh J and West M. Underwater robotics. *Adv Robot* 2001; 15(5): 609–639.
18. Snyder J. Doppler velocity log (DVL) navigation for observation-class ROVs. In: *Oceans 2010 MTS/IEEE*, Seattle, Washington, USA, 20–23 September 2010.
19. Paull L, Saeedi S, Seto M, et al. AUV navigation and localization: a review. *IEEE J Ocean Eng* 2014; 39(1): 131–149.
20. Szwaykowska K and Zhang F. Controlled Lagrangian particle tracking: error growth under feedback control. *IEEE Trans Control Syst Technol* 2018; 26(3): 874–889.
21. Liang X, Wu W, Chang D, et al. Real-time modelling of tidal current for navigating underwater glider sensing networks. *Procedia Comput Sci* 2012; 10: 1121–1126.
22. Luettich RAJ, Westerink JJ, and Scheffner NW. ADCIRC: an advanced three-dimensional circulation model for shelves, coasts, and estuaries. Report 1. Theory and methodology of ADCIRC-2DDI and ADCIRC-3DL three-dimensional circulation model for shelves, coasts, and estuaries. Technical Report NRL/FR/7322–00-9962, Naval Research Lab, 1992.
23. Martin PJ. Description of the navy coastal ocean model version 1.0. Technical Report NRL/FR/7322–00-9962, Naval Research Lab, 2000.
24. Ioannou P and Sun J. *Robust adaptive control*. Upper Saddle River, NJ: Prentice Hall, 1996.
25. Narendra KS and Annaswamy AM. *Stable adaptive systems*. Englewood Cliffs, NJ: Prentice Hall, 1989.
26. Karason SP and Annaswamy AM. Adaptive control in the presence of input constraints. *IEEE Trans Autom Control* 1994; 39(11): 2325–2330.
27. Cho S, Mishra V, Tao Q, et al. Autopilot design for a class of miniature autonomous blimps. In: *2017 IEEE international conference on control technology and applications (CCTA)*, Mauna Lani, Hawaii, USA, 27–30 August 2017, pp. 841–846.
28. Fossen TI. *Guidance and control of ocean vehicles*. New York: Wiley, 1993.
29. Wang C, Zhang F, and Schaefer D. Dynamic modeling of an autonomous underwater vehicle. *J Mar Sci Technol* 2014; 20: 199–212.
30. Gomes SBV and Ramos JJG. Airship dynamic modeling for autonomous operation. In: *Proceedings of 1998 IEEE international conference on robotics and automation (ICRA)*, Leuven, Belgium, 20–20 May 1998, pp. 3462–3467.
31. Zwann S, Bernardino A, and Santos-Victor J. Vision based station keeping and docking for an aerial blimp. In: *Proceedings of the 2000 IEEE/RSJ international conference on intelligent robots and systems*, Takamatsu, Japan, 31 October–5 November 2000, pp. 614–619.
32. Khalil HK. *Nonlinear systems*. 2nd ed. Upper Saddle River: Prentice Hall, 1996.
33. Sastry SS and Bodson M. *Adaptive control*. Upper Saddle River: Prentice Hall, 1989.
34. Astrom KJ and Wittenmark B. *Adaptive control*. Second ed. Reading: Addison-Wesley, 1995.
35. Khalil HK. *Nonlinear systems*. 3rd ed. Upper Saddle River: Prentice Hall, 2002.