Research Article
AUV-Based Plume Tracking: A Simulation Study

Awantha Jayasiri,[1] Raymond G. Gosine,[1] George K. I. Mann,[1] and Peter McGuire[2]

[1]*Faculty of Engineering and Applied Science, Memorial University of Newfoundland, St. John's, NL, Canada A1B3X5*
[2]*C-CORE, Captain Robert A. Bartlett Building, Morrissey Road, St. John's, NL, Canada A1B3X5*

Correspondence should be addressed to Awantha Jayasiri; awanthas@yahoo.com

This paper presents a simulation study of an autonomous underwater vehicle (AUV) navigation system operating in a GPS-denied environment. The AUV navigation method makes use of underwater transponder positioning and requires only one transponder. A multirate unscented Kalman filter is used to determine the AUV orientation and position by fusing high-rate sensor data and low-rate information. The paper also proposes a gradient-based, efficient, and adaptive novel algorithm for plume boundary tracking missions. The algorithm follows a centralized approach and it includes path optimization features based on gradient information. The proposed algorithm is implemented in simulation on the AUV-based navigation system and successful boundary tracking results are obtained.

1. Introduction

Most plume tracking and detection systems reported in the literature are based on surface dynamic oceanographic features. These are detected by satellites and their images are preprocessed for selecting regions of interest to generate optimal tracking sequences [1, 2]. As automated data collection is becoming more prevalent, optimum path planning and trajectory designs for autonomous underwater vehicles (AUVs) are becoming more important, since those planning approaches are required to navigate the AUV for collecting information. In order to track evolving features of interest in the ocean using predictive ocean models, several waypoint selection algorithms are developed and experimentally tested in [3]. Design and control of trajectories for AUVs to obtain optimal data collection are presented in [4]. The AUVs rely on GPS data for accurate position fixes. However, due to the dielectric contrast and high dielectric loss factor of seawater compared to air, most of the strength of the GPS signal is reflected back or attenuated and the AUV has to surface occasionally for position update. Therefore, in situations like under-ice oil spills and deep-sea exploration, the detection and tracking tasks need to be performed without GPS support and rely on a very low frequency acoustic communication channel.

A review of the state-of-the-art AUV navigation techniques is presented in [5, 6], along with a brief comparison of their mission-based suitability. The methods discussed include inertial, acoustic, and geophysical AUV navigation. To this extent, obtaining GPS-based surface fixes as well as utilizing a Long Base Line (LBL), Short Base Line (SBL), or Ultra Short Base Line (USBL) system has been the standard practice for AUV navigation [7]. These systems have similar deployment and transponder positioning challenges. Inverted USBL configuration, where the USBL array is located on the vehicle, interrogates transponders placed in known positions [7]. The study reported in [8] discusses underwater transponder positioning (UTP), which requires only one transducer due to the tight coupling with the vehicle's Inertial Navigation System (INS), and serves as an alternative approach.

In this paper, we use the inverted USBL configuration, which enables AUV positioning to be single-referenced and hence greatly reduces the complex operational logistics. A GPS-capable transponder, which is suspended from surface ice or a platform, is considered. The transponder transmits its position with each ping to the AUV and provides a means to position update in a GPS-denied undersea environment. It is also assumed that several receivers are placed along the body of the AUV on a noncoplanar configuration, keeping

FIGURE 1: REMUS AUV (image credit: Kongsberg Maritime).

TABLE 1: Sensor characteristics.

Sensor type	Measurand	Frequency	Noise, σ
Accelerometer	Specific force	100 Hz	0.1 m/s^2
Gyroscope	Angular velocity	100 Hz	0.005 rad/s
Attitude sensors 1 and 2	Roll & pitch	10 Hz	0.05 rad
Attitude sensor 3	Yaw	10 Hz	0.2 rad
Pressure sensor	Pressure	5 Hz	200 Pa
DVL	Linear velocity	5 Hz	0.05 m/s
Acoustic receivers	Time delay	500 kHz	$1/\sqrt{3}\ \mu$s

a sufficient space between each of the receivers. The transponder and AUV's clocks are synchronized and the transponder broadcasts a unique signal with a known delay [9, 10]. Upon arrival of the transmitted signal to the AUV, a single trip travel time for each of the receivers is recorded. The recorded travel times are then used for the measurement update in state estimation. In between transponder broadcasts, short-term dead-reckoning is utilized based on high-rate INS data.

In operation conditions such as rough bathymetry or limited sensor range, where DVL bottom-track data is unavailable, the AUV's dynamic model-based position information is incorporated for position estimation [11–13]. A Linear Quadratic Regulator (LQR) is implemented based on the estimated states and further improved by adding a Proportional-Integral (PI) Controller for rudder control. The proposed approach is implemented in the nonlinear dynamic model of the REMUS AUV (Figure 1) presented in [14].

The paper has the following contributions:

(1) a multirate unscented Kalman filter is employed for sensor fusion in AUV localization;

(2) a novel efficient adaptive plume boundary tracking algorithm is developed using gradient information;

(3) numerous simulation results are presented to verify the approach.

The rest of the paper is organized as follows. In Section 2, an introduction on sensor package is given and the AUV navigation algorithm is presented. In Section 3, simulated results of AUV navigation are discussed. In Section 4, a plume boundary tracking algorithm is presented with its simulation results. The paper concludes by reporting the future research directions. A preliminary version of this paper appears in proceedings of the 27th Canadian Conference on Electrical and Computer Engineering (CCECE) [15]. In this version we focused on providing a detailed description of the work and more simulation results.

2. Sensor and System Modeling

2.1. Preliminaries and Notation. According to [16], $n(=6)$ Degrees Of Freedom (DOF) model of AUV dynamics and kinematics can be derived as

$$F = B(v)\,u_{\text{in}} - (C(v)\,v + D(v)\,v + g(q)) = M\dot{v} \quad (1)$$

$$\dot{\eta} = J(q)\,v, \quad (2)$$

where $M_{n \times n}$ matrix represents the inertia of the vehicle and hydrodynamic added-mass; $C_{n \times n}$ matrix includes rigid body Coriolis and centrifugal components as well as added-mass derivatives corresponding to the velocity coupling;

$D_{n \times n}$ matrix includes energy dissipative terms due to relative motion between vehicle and surrounding fluid; $g_{n \times 1}$ combines gravitational and buoyancy forces; and $B_{n \times m}$ is the thruster control matrix, where m is the number of thrusters. Furthermore, $v = [u^T\ \omega^T]^T \in \mathbb{R}^6$, where $u = [u_1\ u_2\ u_3]^T$ and $\omega = [\omega_1\ \omega_2\ \omega_3]^T$ are body-fixed linear and angular velocities. The total force matrix is represented as F. Also, $\eta = [p^T\ q^T]^T$, where $p = [x\ y\ z]^T$ is the position vector relative to the inertial reference frame origin and $q = [\phi\ \theta\ \psi]^T$ is the vector of Euler angles, which are roll, pitch, and yaw, respectively. $u_{\text{in}} \in \mathbb{R}^m$ is the control input of the thrusters. The Jacobian $J = \begin{bmatrix} J_1(q) & 0_{3\times3} \\ 0_{3\times3} & J_2(q) \end{bmatrix}$, where J_1 is the coordinate transform matrix from u to $\dot{p}$ and J_2 relates ω to Euler rate vector $\dot{q}$. These are computed as follows:

$$
\begin{aligned}
J_1 &= \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \\[2mm]
J_2 &= \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \dfrac{s\phi}{c\theta} & \dfrac{c\phi}{c\theta} \end{bmatrix},
\end{aligned}
\quad (3)
$$

where $s \Rightarrow \sin$, $c \Rightarrow \cos$, and $t \Rightarrow \tan$, with $0 \le \phi < 2\pi$, $-\pi/2 < \theta < \pi/2$, and $0 \le \psi < 2\pi$. We assume that $\theta \ne \pm\pi/2$ as otherwise J_2 approaches a singularity. If such an operation is required, quaternion-based attitude representation can be adopted as in [17]. The coefficient matrices we used in this work are based on the REMUS AUV dynamic model [14]. Our AUV has a typical set of sensors as mentioned in Table 1. The noise in each acoustic receiver is the quantization error introduced by the analog to digital converter of the system [18]. The transponder emits signals at 1 Hz and the control loop runs at 100 Hz. The AUV navigation system including USBL array is depicted in Figure 2.

2.2. Sensor Modeling

2.2.1. IMU Measurements. The Inertial Measurement Unit (IMU) sensor consists of accelerometers and gyroscopes (gyros), which measure specific forces and angular rates in
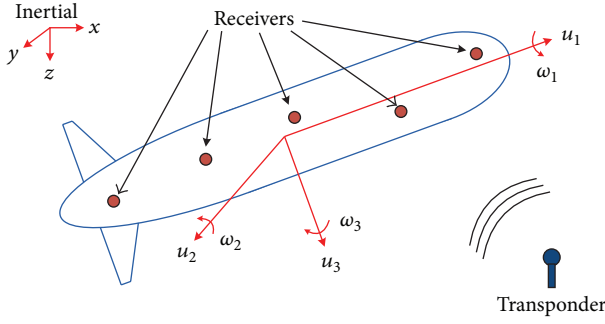
FIGURE 2: AUV navigation system adopted from [19].

the body-fixed coordinate system. The accelerometer reading Z_a can be modeled as [20]

$$Z_a = \dot{u} + [\omega\times]u + J_1^T \vec{g} + b_a + n_a, \tag{4}$$

where $[\omega\times]$ is the skew symmetric matrix cross product form of the vector ω, which is given as

$$[\omega\times] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \tag{5}$$

where $\vec{g} = [0\ 0\ 9.81\ \text{m/s}^2]^T$ is the gravity vector and $b_a = 0.05\ \text{m/s}^2$ is the bias of the reading with $\dot{b}_a = 0$. The measurement noise is distributed as $n_a \sim \mathcal{N}(0, \sigma_a^2 I_3)$, with σ_a^2 as the variance for each direction. The gyro reading Z_g is modeled as [20]

$$Z_g = \omega + b_g + n_g, \tag{6}$$

where $b_g = 0.01\ \text{rad/s}$ is the bias of the reading with $\dot{b}_g = 0$. The noise of the reading is distributed as $n_g \sim \mathcal{N}(0, \sigma_g^2 I_3)$, with σ_g^2 as the variance on each axis.

2.2.2. Attitude Measurements. Attitude sensors composed of magnetometers and compasses measure roll, pitch, and yaw angles. These sensors are modeled as $Z_{\phi\theta\psi}$ [20]:

$$Z_{\phi\theta\psi} = q + n_{\phi\theta\psi}, \tag{7}$$

where the sensor noise is distributed as $n_{\phi\theta\psi} \sim \mathcal{N}(0, Q_{\phi\theta\psi})$ with $Q_{\phi\theta\psi} = \text{diag}(\sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2)$ having $\sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2$ as the variances of ϕ, θ, ψ measurements.

2.2.3. Pressure Sensor Reading. A model for the pressure sensor is given as [20]

$$Z_{pr} = \rho_{sw}\vec{g}^T(p + J_1 l_{pr}) + n_{pr}, \tag{8}$$

where Z_{pr} is the sensor reading, ρ_{sw} is the seawater density, and l_{pr} is the sensor location in body frame. The sensor noise is distributed as $n_{pr} \sim \mathcal{N}(0, \sigma_{pr}^2)$ with σ_{pr}^2 as the variance.

2.2.4. DVL Sensor Reading. The DVL update (in processed form) is modeled as

$$Z_{dvl} = u + n_{dvl}, \tag{9}$$

where Z_{dvl} is the DVL reading and the sensor noise is distributed as $n_{dvl} \sim \mathcal{N}(0, Q_{dvl})$ with $Q_{dvl} = \sigma_{dvl}^2 I_3$ having σ_{dvl}^2 as the variance in each direction.

2.2.5. Velocity Estimation Using Dynamic Model. When DVL bottom-track data is unavailable, the body-fixed linear velocities are calculated using AUV's nonlinear dynamic model and used in the measurement update process:

$$Z_{mod} = u + n_{mod}, \tag{10}$$

where Z_{mod} is an estimate for u calculated using (1) and $n_{mod} \sim \mathcal{N}(0, Q_{mod})$. The noise matrix Q_{mod} for this estimate can be calculated recursively as

$$Q_{mod} = G_t Q_{mod} G_t^T + V_t M_t V_t^T, \tag{11}$$

where $G_t = I_3 + M^{-1}[\partial F/\partial u]dt$ and $V_t = [\partial u/\partial u_{in}]$. The matrix $M_t = 0.01 I_3$ represents the noise in control space, which includes the noises in rudder and elevator angle control and the noise in thruster force.

Therefore, the linear velocity measurement $Z_u = Z_{dvl}$ or Z_{mod} and its noise covariance $Q_u = Q_{dvl}$ or Q_{mod} accordingly.

2.2.6. Time of Arrival Measured Using USBL. The time of arrival (TOA) of the acoustic wave from the transponder to the ith receiver r_i ($i = 1, \ldots, 5$) is modeled as follows:

$$Z_{t_i} = \frac{\lfloor f \lVert (p + J_1 l_{r_i}) - \text{Tr}_{xyz} \rVert / V_s \rfloor}{f} + n_{r_i}, \tag{12}$$

where Z_{t_i} is the counted reading, f is the sampling frequency, l_{r_i} is the position of r_i on the AUV body, Tr_{xyz} is the transponder position, V_s is the speed of sound in the seawater, and n_{r_i} is the quantization noise. Note that $n_{r_i} \sim \mathcal{U}(-1/(2f), 1/(2f))$. In this work we assume $V_s = 1500\ \text{m/s}$. However, in practical implementations V_s can be computed using Conductivity Temperature Depth (CTD) sensor readings [21]. Furthermore, we assume that TOA is disturbed by a noise of $n_{t_i} \sim \mathcal{N}(0, Q_t)$ with $Q_t = \sigma_{TOA}^2 I_5$ and $\sigma_{TOA} = 1/\sqrt{3}\ \mu$s, having σ_{TOA}^2 as the noise variance.

A tightly coupled approach was adopted to infer the range and angle of arrival information. This is performed by feeding an array of TOA measurements to the estimation program. Let Z_t be the vector that includes all TOA measurements as

$$Z_t = \left[Z_{t_1}, \ldots, Z_{t_5} \right]^T. \tag{13}$$

2.3. System Modeling. Incorporating the kinematics and measurements, the system equations can be written as follows [20]:

$$\dot{q} = J_2\left(Z_g - b_g\right),$$
$$\dot{b}_g = 0,$$
$$\dot{p} = J_1 u, \tag{14}$$
$$\dot{u} = Z_a - [\omega\times]\,u - J_1^T\vec{g} - b_a,$$
$$\dot{b}_a = 0.$$

The state vector consists of 15 states:

$$y = \begin{bmatrix} q^T & b_g^T & p^T & u^T & b_a^T \end{bmatrix}^T. \tag{15}$$

The process noise covariance matrix is given as

$$R = \begin{bmatrix} J_2\left(\sigma_g^2 I_3\right)J_2^T dt^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & J_1\left(\sigma_u^2\right)J_1^T dt^2 & 0 \\ 0 & \left(\sigma_a^2 I_3\right)dt^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{16}$$

The measurement model is written as $Z = h + Q$, where Z is the measurement vector, h is the measurement function, and Q is the measurement noise covariance. They are given as follows:

$$Z = \begin{bmatrix} Z_{\mathrm{pr}} & Z_t & Z_{\phi\theta\psi} & Z_u \end{bmatrix}^T,$$

$$h = \begin{bmatrix} \left. \begin{array}{c} \rho_{\mathrm{sw}}\vec{g}^T\left(p + J_1 l_{\mathrm{pr}}\right) \\ \dfrac{\lfloor f\,\|\left(p + J_1 l_{r_i}\right) - \mathrm{Tr}_{xyz}\|/V_s\rfloor}{f},\ i \in [1,5] \\ q\,(= h_2) \\ u\,(= h_3) \end{array} \right\}\,(= h_1) \end{bmatrix}, \tag{17}$$

$$Q = \begin{bmatrix} \sigma_{\mathrm{pr}}^2 & 0 & 0 & 0 \\ 0 & Q_t & 0 & 0 \\ 0 & 0 & Q_{\phi\theta\psi} & 0 \\ 0 & 0 & 0 & Q_u \end{bmatrix}.$$

3. State Estimation, Control, and Navigation

Based on (14), the state vector is propagated through time and a multirate unscented Kalman filter (UKF) is implemented for state estimation. The filter update process runs asynchronously as a response to the measurement readings. Once the transmitted signal is received by all of the receivers, the system performs the position update. If the difference in time of arrival of the signal between the first and last receiver is very small, then the AUV movement at that time is neglected.

3.1. Controller Implementation. The navigation system is decoupled for ease of control implementation, assuming negligible coupling effects between vertical and horizontal planes. AUV movement in these two planes is governed by the control of its elevator angle (δ_e) and rudder angle (δ_r), respectively. Two controllers are developed based on estimated states and the linearised versions of state equations to control δ_e and δ_r. Mechanical constraints on these angles are such that $-\pi/3 \le \delta_e, \delta_r \le \pi/3$.

δ_e is controlled by C_{δ_e}:

$$\begin{aligned} C_{\delta_e} &= \mathrm{LQR}_e \\ &= ek_1 u_3 + ek_2\omega_2 + ek_3\theta + ek_4 z + ek_5\int z_{\mathrm{err}}, \end{aligned} \tag{18}$$

where LQR_e is the Linear Quadratic Regulator (LQR) controller for elevator angle, $ek_1,\ldots,ek_5$ are LQR gains, and $z_{\mathrm{err}} = z - \mathrm{WP}_z$ (i.e., the difference between z and current waypoint z coordinate).

δ_r is controlled by C_{δ_r}, which combines LQR and Proportional-Integral (PI) Controllers:

$$\begin{aligned} C_{\delta_r} &= \mathrm{LQR}_r + \mathrm{PI}_r, \\ \mathrm{LQR}_r &= rk_1 u_2 + rk_2\omega_3 + rk_3\psi_{\mathrm{err}}, \\ \mathrm{PI}_r &= k_p d_\perp + k_i\int d_\perp, \end{aligned} \tag{19}$$

where rk_1, rk_2, rk_3 are LQR gains and k_p, k_i are proportional and integral gains. Also, $\psi_{\mathrm{err}} = \psi - \psi_d$, where

$$\psi_d = \tan^{-1}\left(\frac{\mathrm{WP}_y - y}{\mathrm{WP}_x - x}\right), \tag{20}$$

and $d_\perp$ is the perpendicular distance from the AUV position to the assigned path x, y plane. After several iterations to reduce settling time, overshoot, and steady state errors, the optimized controller gains were found to be

$$\begin{aligned} ek_1 &= 0.5110, \\ ek_2 &= -4.1867, \\ ek_3 &= -6.6161, \\ ek_4 &= 3.1698, \\ ek_5 &= 0.0035. \\ rk_1 &= 0.3137, \\ rk_2 &= -0.9071, \\ rk_3 &= -1.1180, \\ k_p &= 0.01, \\ k_i &= 0.00001. \end{aligned} \tag{21}$$

AUV reaching the waypoint is determined by

$$\|\mathrm{WP}_{xyz} - p\| < 1\ \mathrm{m}. \tag{22}$$

3.2. An Algorithm for Localization and Autonomous Navigation.

Incorporating the state estimation and control strategies, an algorithm is developed for AUV localization and navigation and is shown in Algorithm 1.

One advantage of this approach is that gyroscope readings are not treated as measurements and hence ω is not included in the state vector. Consequently, the dimension of the state vector is reduced. In fact, ω is used for the state-based controller implementation. However, since the gyros have very low noise (σ = 0.005 rad/s), we assume that eliminating the bias error from gyro readings will provide ω with sufficient accuracy.

3.3. Simulation Setup.

Navigation simulations are performed assuming a 2400 m × 1000 m × 10 m three-dimensional space. The ocean current velocity is assumed as V_C (m/s) = $[0.2 \quad 0.4 \quad 0.3]^T$ in x, y, and z directions, respectively. The mission is to perform a lawnmower-type navigation in the environment under five different cases:

(1) navigation without acoustic-based position fixes,

(2) navigation with position aiding and DVL dropout,

(3) dynamic model-aided navigation when DVL data is unavailable,

(4) navigation with transponder dropout and no velocity aiding,

(5) navigation with incremental transponder dropouts.

The start and end positions of the mission are given as 1200 m, 0 m, and −10 m and −1200 m, 1000 m, and −10 m in inertial x, y, z directions, respectively. While navigating between these two positions, a 10 m step change occurs in vertical (z) direction.

3.4. Bias Estimation.

The predefined bias values of the accelerometers and gyroscopes are estimated. This is depicted in Figure 3. Figures 3(a)–3(c) show the estimates of accelerometer bias in directions u_1–u_3, respectively. Figures 3(d)–3(f) show gyro bias estimates, respectively, in ω_1–ω_3. All the bias estimates converged quickly within less than 30 s and remained constants thereafter.

3.5. Navigation without Acoustic-Based Position Fixes.

Simulation is performed without transponder aiding for x, y position fixes throughout the run. However, velocity aiding is provided by DVL measurements. The results are shown in Figure 4. Figures 4(b)–4(d) show the error in AUV position estimates of inertial x, y, z directions, respectively. The errors in x, y positions are still low due to the accurate localization based on low noise DVL measurements. The error in z is still very low as the estimation of z is updated based on pressure sensor readings.

3.6. Navigation with DVL Dropout.

Navigation simulation is performed with a DVL data dropout over a 5000 s period, but transponder aiding is provided throughout the run. During the DVL dropout period, (9) and (10) are not used and the

```
(1)   Initialize with: y₀ and its covariance P₀
(2)   for i ∈ (1,…,number of WP) do
(3)       for k ∈ (1,…,∞) do
(4)           Read IMU measurements
(5)           Calculate sigma points
(6)           Compute y_k⁻, P_k⁻ using y_{k−1}⁺, P_{k−1}⁺ and (14)
(7)           Redraw sigma points
(8)           if Attitude measurements are available then
(9)               Read Z_{ϕθψ}
(10)          if Pressure measurement is available then
(11)              Read Z_pr
(12)          if DVL bottom-track is available then
(13)              Z_u = Z_DVL, Q_u = Q_DVL
(14)          else
(15)              Z_u = Z_mod, Q_u = Q_mod
(16)          if Tr signal is received by all receivers then
(17)              Construct Z_t
(18)          Compute y_k⁺, P_k⁺
(19)          Implement C_{δ_s} and C_{δ_r} using (18) and (19)
(20)          Check the condition in (22)
(21)          if WP is achieved then
(22)              Exit and start from the next WP
```

ALGORITHM 1: AUV localization and navigation algorithm.

lines (12)–(15) in Algorithm 1 are not implemented in the controller. The results are shown in Figure 5. In Figure 5(a), the blue path shows the navigation in normal conditions where DVL data is available. The red path from A to B shows the time where DVL data is unavailable.

3.7. Model-Aided Navigation under DVL Dropout.

In this case, the AUV's dynamic model-based information is incorporated in the DVL dropout period. Figures 6(a) and 6(b) depict the 2D and 3D plots of the trajectory, respectively. Note that from A to B the algorithm uses model-based data and successful navigation is performed. Figures 7(a)–7(c) show the errors associated with this model-aided navigation which occurs between 0.5×10^4 s and 1×10^4 s. Figures 8(a)–8(c) show the errors in Euler angles under model-aided navigation from A to B shown in Figure 6. The errors in ϕ and θ are very low and under the range of ± 0.02 rad. However, error in ψ is under ± 0.05 rad. The computation of Euler angles is not affected by the DVL dropout.

3.8. Navigation with Transponder Dropout and No Velocity Aiding.

Navigation simulation is performed with the transponder dropout and no velocity aiding for 5000 s period. The 2D plot of the trajectory is shown in Figure 9(a) and errors in x, y, and z are shown in Figures 9(b), 9(c), and 9(d), respectively. Note that, with no velocity aiding, the AUV quickly diverges from the assigned path resulting in high errors in x and y. Once the velocity aiding is established again (in this case the DVL reading) the AUV converges and errors in x, y are decreased.

(a) Accelerometer bias in u_1

(b) Accelerometer bias in u_2

(c) Accelerometer bias in u_3

(d) Gyro bias in ω_1

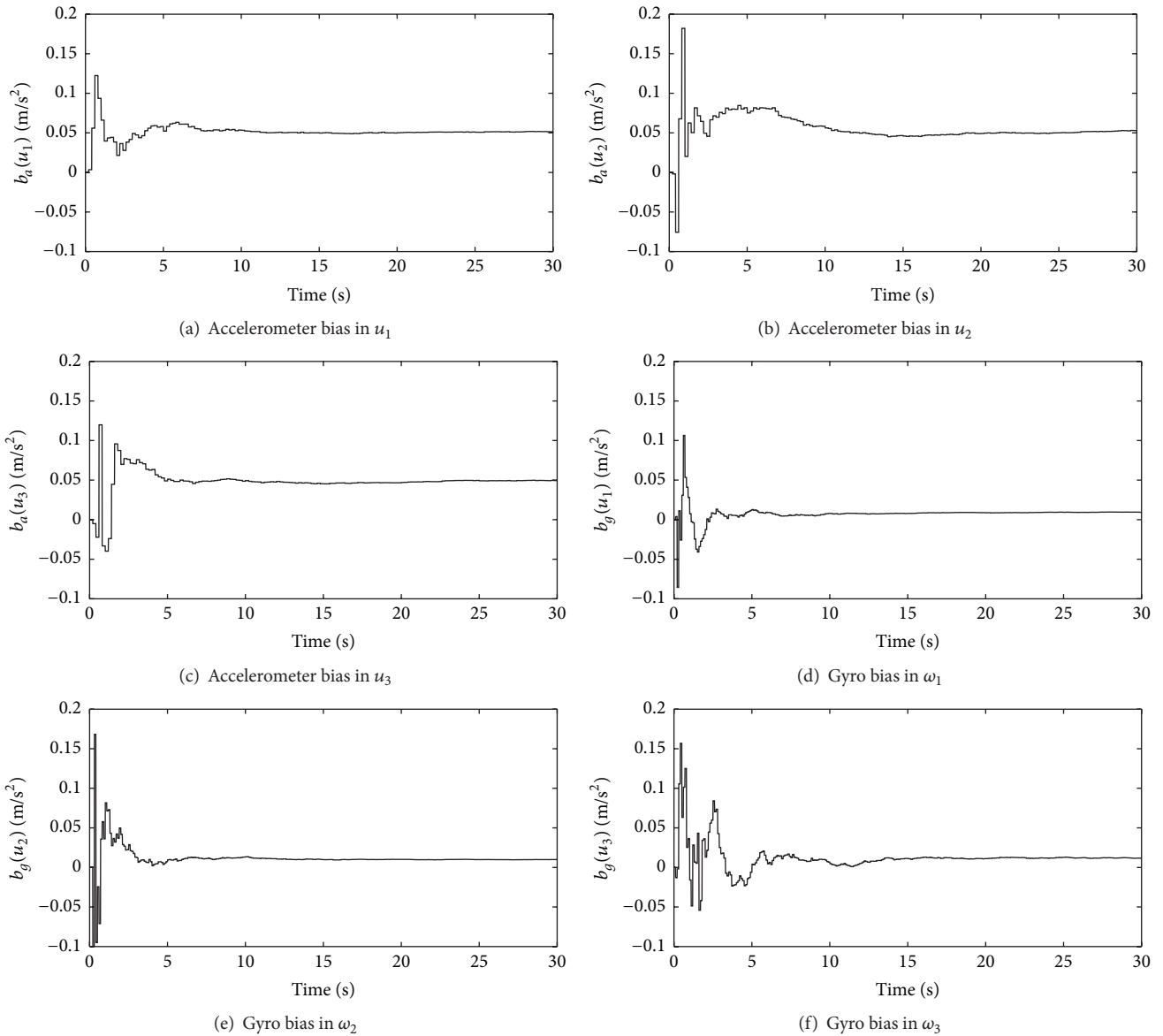(e) Gyro bias in ω_2

(f) Gyro bias in ω_3

FIGURE 3: Estimated accelerometer and gyro biases.

3.9. Navigation with Incremental Transponder Dropouts. In this case, we consider a scenario where transponder dropout occurs at every 50 s and consequent loss of x, y position fixes. The simulation run is limited to 10000 s. Figures 10(a) and 10(b) show the errors in range and azimuth from the AUV to the transponder position, respectively. Figures 10(c) and 10(d) show the errors in inertial x and y directions due to the transponder dropouts. It can be observed that, even with incremental loss of x, y position, the estimated horizontal position is maintained with low error bounds.

3.10. Discussion on the AUV Navigation Results. We have performed numerous simulations using the AUV localization and navigation algorithm. The AUV localization is performed by employing an unscented Kalman filter, where the true nonlinear system is used to capture the correct mean and

covariance to the 3rd order, providing better performance than standard extended Kalman filter based approaches [22]. The localization scheme yields satisfactory performance even with the transponder dropouts.

In Figures 5(b) and 5(c) increased errors in x, y positions can be observed when DVL dropout occurs. At this period the AUV failed to perform a smooth navigation. Also, in Figure 5(d) there is a small reduction of the error in inertial z at the DVL dropout period. At the DVL drop the observability matrix is not full rank. Hence, to preserve the observability the dimensions are reduced. As a result, the cross-covariance (which is introduced by DVL measurements) of filter covariance matrix is eliminated and a better representation of the uncertainty can be obtained in inertial z direction, which may cause the error reduction. However, in practice the error in z is mainly caused by the depth controller, which is mostly
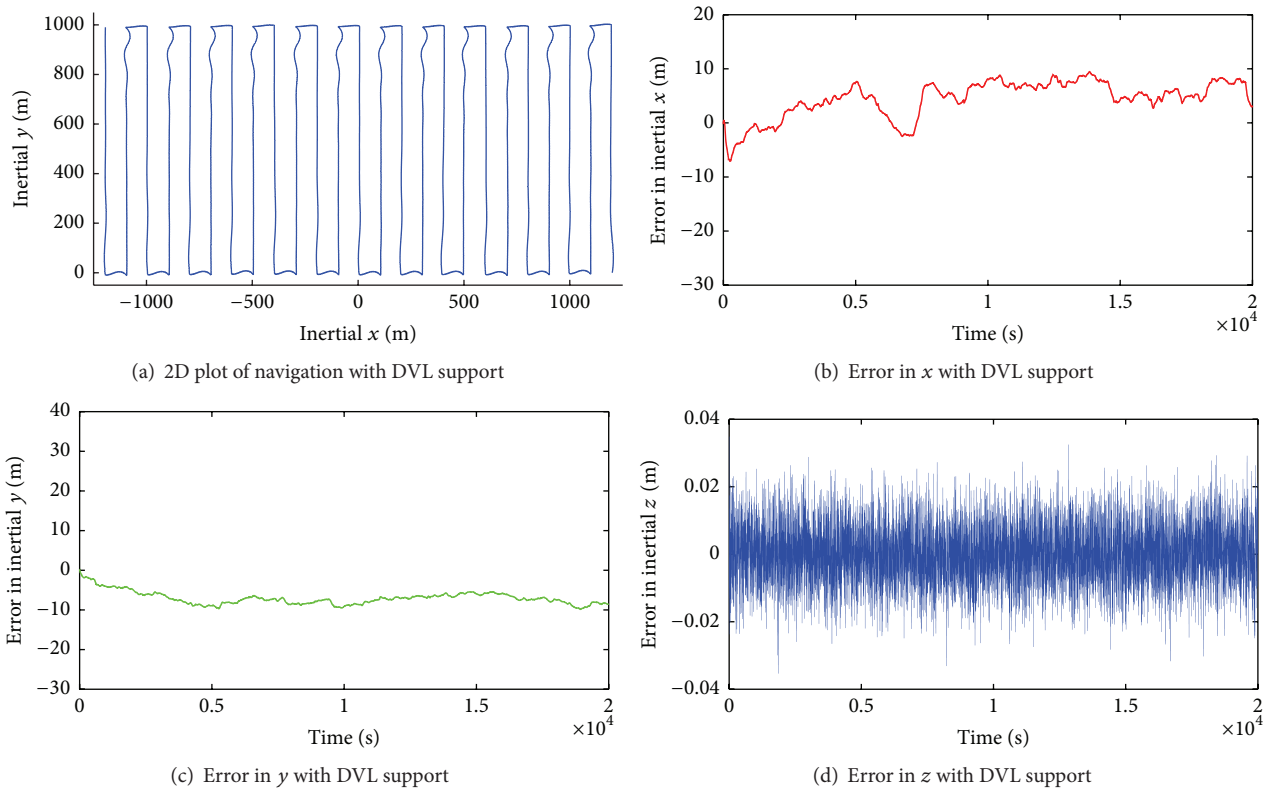
(a) 2D plot of navigation with DVL support

(b) Error in x with DVL support

(c) Error in y with DVL support

(d) Error in z with DVL support

FIGURE 4: Navigation with DVL support and without x, y position fixes.

(a) 2D plot of navigation with DVL dropout from A to B

(b) Error in x with DVL dropout for 5000 s

(c) Error in y with DVL dropout for 5000 s

(d) Error in z with DVL dropout for 5000 s
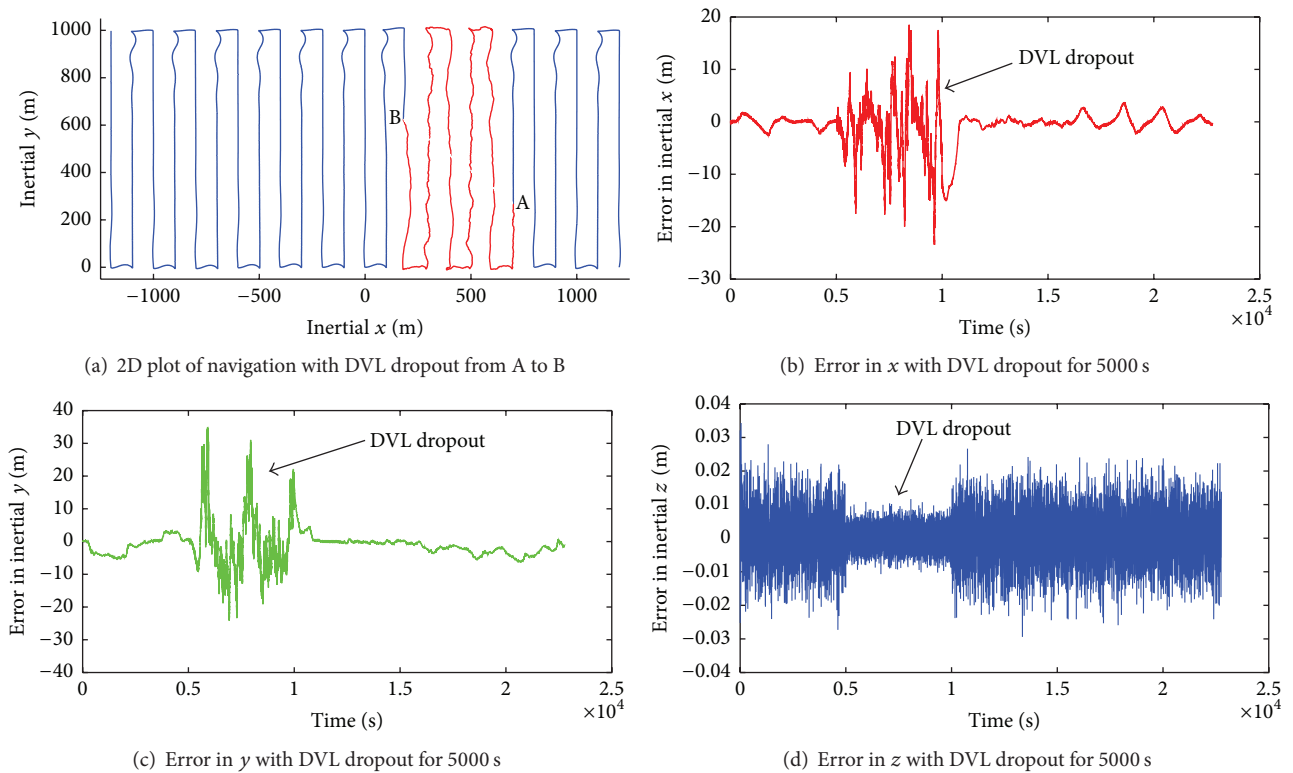
FIGURE 5: Navigation with DVL dropout for 5000 s.

(a) 2D plot

(b) 3D plot

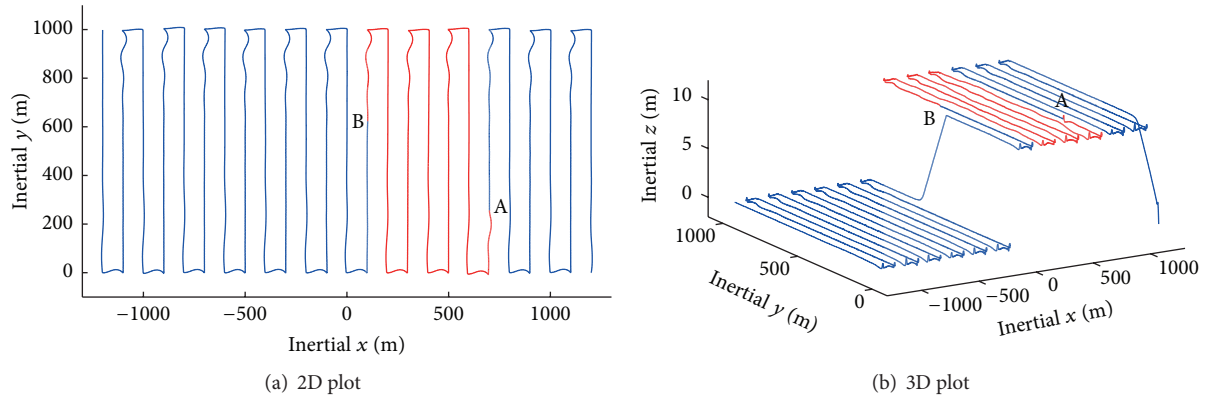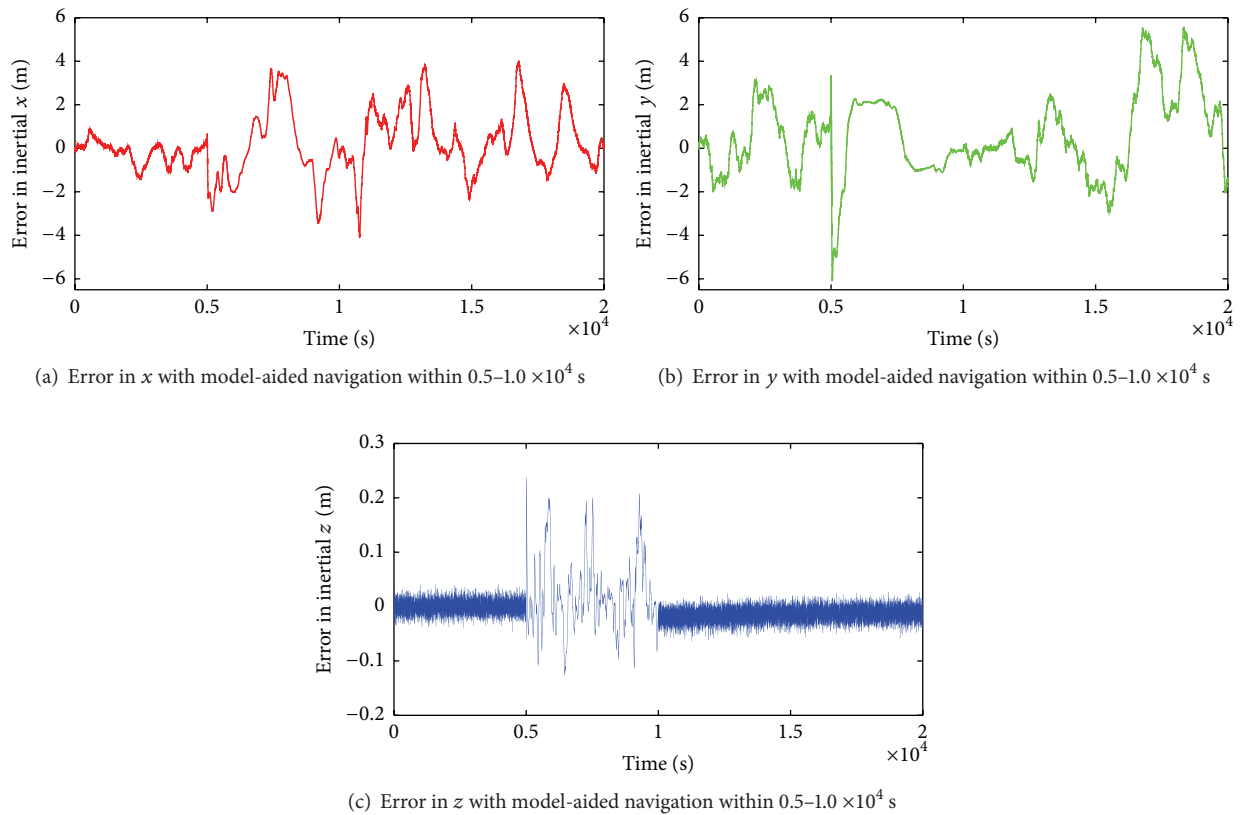FIGURE 6: 2D and 3D plots under model-aided navigation from A to B.

(a) Error in x with model-aided navigation within 0.5–1.0 $\times 10^4$ s

(b) Error in y with model-aided navigation within 0.5–1.0 $\times 10^4$ s

(c) Error in z with model-aided navigation within 0.5–1.0 $\times 10^4$ s

FIGURE 7: Errors of x, y, z in model-aided navigation within 0.5–1.0 $\times 10^4$ s (A to B) in Figure 6.

decoupled from the horizontal motion and makes decisions based readings from only the pressure sensor.

In the model-aided navigation under DVL dropout, an increment of the error in inertial z direction can be observed at the DVL dropout period on Figure 7(c). This can be due to the fact that the new cross-covariance, which is introduced by using the model-aided information in the DVL dropout period, degrades the filter performance, hence leading to higher error in z.

The rudder angle control is achieved by combining LQR with PI controller. As a result, the AUV was able to withstand the strong currents (V_C (m/s) $= [0.2 \quad 0.4 \quad 0.3]^T$) and successful performances are shown in all cases.

4. Adaptive Plume Tracking

According to Algorithm 2, AUV navigation is performed and information is collected about the plume boundary based on *in situ* fluorometer readings. Incorporating that information, an estimate on dispersion of the plume can be achieved. Table 2 depicts the methods and limitations of the state-of-the-art plume tracking strategies used in the

---

input: Fluorometer readings Fr_i $(i = 1, \ldots, n)$, threshold (T_f)
output: Adaptive boundary tracking algorithm
(1) **while** $Fr_i < T_f$ **do**
(2)   Navigate the AUV through pre-defined path.
(3)   **if** AUV *reaches to last* WP **then**
(4)   Could not locate the plume...! exit
(5) **while** *boundary track is not complete* **do**
(6)   **if** $Fr_i > T_f$ **then**
(7)   Plume boundary is detected: $\log(x_t, y_t)$
(8)   Move AUV until $i = k_1$ $(\Delta a \propto k_1)$
(9)   Calculate the gradient g_t using (x_t, y_t) and (x_{t-1}, y_{t-1})
(10)   Set $\psi = g_t$
(11)   Calculate section length $\Delta b \propto 1/\|g_t - g_{t-1}\|$ and move AUV
(12)   Set $\psi = \pi/2$ and move
(13)   **if** $Fr_i < T_f$ **then**
(14)   Repeat from line (8) to (12)
(15)   **if** $i > k_2$ **then**
(16)   Plume cannot be located...!
(17)   Calculate the center point (x_c, y_c) using $x_1, y_1, \ldots, x_t, y_t$
(18)   Set ψ to (x_c, y_c) and move
(19)   Repeat from line (6)

ALGORITHM 2: Plume boundary tracking algorithm.

literature. In this work, we developed an efficient adaptive plume tracking algorithm to track the plume boundary and investigate its dispersion, with path optimization. An initial bounding box is assigned based on some prior knowledge (such as remote sensing data) and it is assumed that the oil plume is located inside the bounding box. The tracking algorithm navigates the AUV through the predefined search path and fluorometer readings are incorporated for detecting the plume boundary. Once a fluorometer reading Fr_i exceeds a predefined threshold T_f, a plume can be detected with a higher concentration. At this point, the AUV reaches the first boundary point and the algorithm switches to the boundary detecting mode.

The proposed approach is depicted in Figure 11 and the steps followed are described in Algorithm 2. Let k_1 be a predefined number of consecutive fluorometer readings to ensure the AUV is inside the plume boundary. Also, let k_2 be another predefined number of consecutive fluorometer readings, which are lower than T_f, to ensure that the plume cannot be located. The transect Δa is proportional to k_1. The same can be applied when the AUV is outside the boundary. Furthermore, Δb is used for rapid plume mapping and it helps to improve the efficiency of the algorithm. To calculate Δb, the system needs the two latest consecutive gradient values (as stated in line (11) of Algorithm 2). These consecutive gradient values are obtained from the four consecutive logged (x, y) positions. The calculation of Δb is mentioned in line (9) of Algorithm 2. When the plume boundary is relatively smooth, the two consecutive gradients are nearly equal and the section length Δb can be long. This is better for smooth sections of

the plume boundary. Also when the plume boundary seems irregular, those consecutive gradients will be different and Δb will be short comparatively. As a result, the system can map the complex, irregular-shaped section of the boundary.

Lines (15)–(18) in Algorithm 2 ensure that the plume is tracked all the time. Here, k_2 controls the distance the AUV moves before it turns to the center point direction. The condition in line (5) is evaluated based on a predefined distance between start and end points (points A and B in Figure 11) and terminates the plume detection mode.

One limitation of the approach is that the accuracy of plume boundary coverage entirely depends on the sensitivity of the fluorometer and the predefined k_1, k_2 values. The performance on "noisy" plume boundaries may be improved by adaptively changing the k_1, k_2 values and adding an array of fluorometers.

4.1. Plume with an Ellipse-Shaped Boundary.

A horizontal plume dispersion with a simple ellipse shape is considered. The major and minor axis lengths of the ellipse are 100 m and 150 m. The angle shift from the major axis is $\pi/3$. Figure 12(a) shows the true and detected plume boundaries under DVL-aided and dropout (dynamic model-based) navigation. Figures 12(b) and 12(c) show the 3D view of the navigation and the error in plume detection in x, y, z directions, respectively. Point A is the place where the AUV looses the DVL data. Although instant high errors can be observed in inertial x, y, z, those errors slowly converge due to the help of model-aided navigation.

TABLE 2: Different plume tracking strategies in literature.

Article	Number of agents	Properties	Limitations
[23]	Multiple	Uses CUSUM algorithm, presents the boundary estimation problem as a HMM, and recast as an optimization problem	Considered only ellipse-shaped, no path optimization, simulation only, intervehicle communication issues not addressed, and AUV dynamics not considered
[24]	Multiple	Decentralized, gradient-free algorithm, convergent and stable	AUV dynamics not considered, high computation cost, intervehicle communication issues not addressed, and only very simple plume shapes considered
[3]	Single, multiple	Cooperative, generating polygons to follow based on ocean model predictions, simulation, and practical implementation	Trajectory based on (roughly) approximated polygons, temporal constraints not considered, unable to react to the fast moving features, and ignoring the dynamics of the glider
[25]	Single	Behavior-based approach for plume mapping, subsumption architecture, showing experimental results	Limitations of behavior-based approach, no description on adaptive mapping, and implementing simple preplanned lawn-mower strategy
[26]	Single	Uses colored dissolved organic matter (CDOM) sensor for planed missions, adaptive planning using *in situ* current, and temperature measurements, and gets the 3D track of the AUV, practical implementation	No information on path optimization and no adaptive tracking
[27]	Single	Based on peak-capture algorithm, it generates a sawtooth trajectory and uses depth information and practical implementation	No information on path optimization
[28]	Multiple	Adaptive behavior-based system, acoustic communication within AUVs, representing the plume using Fourier orders when reconstructing	No information on path optimization and communication overhead
[29]	Single	Uses a plume indicator function and real-time implementation and uses adaptive transects; transect length depends on number of consecutive samples; distance between transects is a percentage of previous samples	Less path optimization, not using an AUV, no information about the convergence, and coverage of the used algorithm
[30]	Single	Uses remote sensing data to detect hotspots, uses surface current to project plumes spatiotemporally, and runs in a lawnmower type pattern, practical implementation	No path optimization, only using predefined pattern, and no adaptive tracking
Proposed	Single	Path optimization based on gradient information, adaptive plume tracking, and centralized approach	Only simulation results, no comparison data available, relying on remote sensing data for locating the plume region initially, low performance to noisy plume boundaries, and using only one fluorometer

4.2. Plume Boundary Modeled Using Fourier Orders. A rough estimate of the horizontal dispersion of a real plume can be obtained using Fourier orders of the form [28]

$$R_k = \sum_{i=0}^{k} A_i \cos(i\theta + \phi_i) + R_u, \qquad (23)$$

where R_k is the radial distance to the plume boundary, $k(= 10)$ is the highest Fourier order of the series, $R_u(= 50\,\text{m})$ is the undisturbed plume radius, $\theta(= [0, 2\pi))$ is the angles about the center of the plume evaluated for each degree, $A_i(= (+25R_u/2k, -25R_u/2k))$ is the radial amplitude perturbation, and $\phi_i(= [-\pi, +\pi])$ is the phase shift of ith order. Figures 13(a)
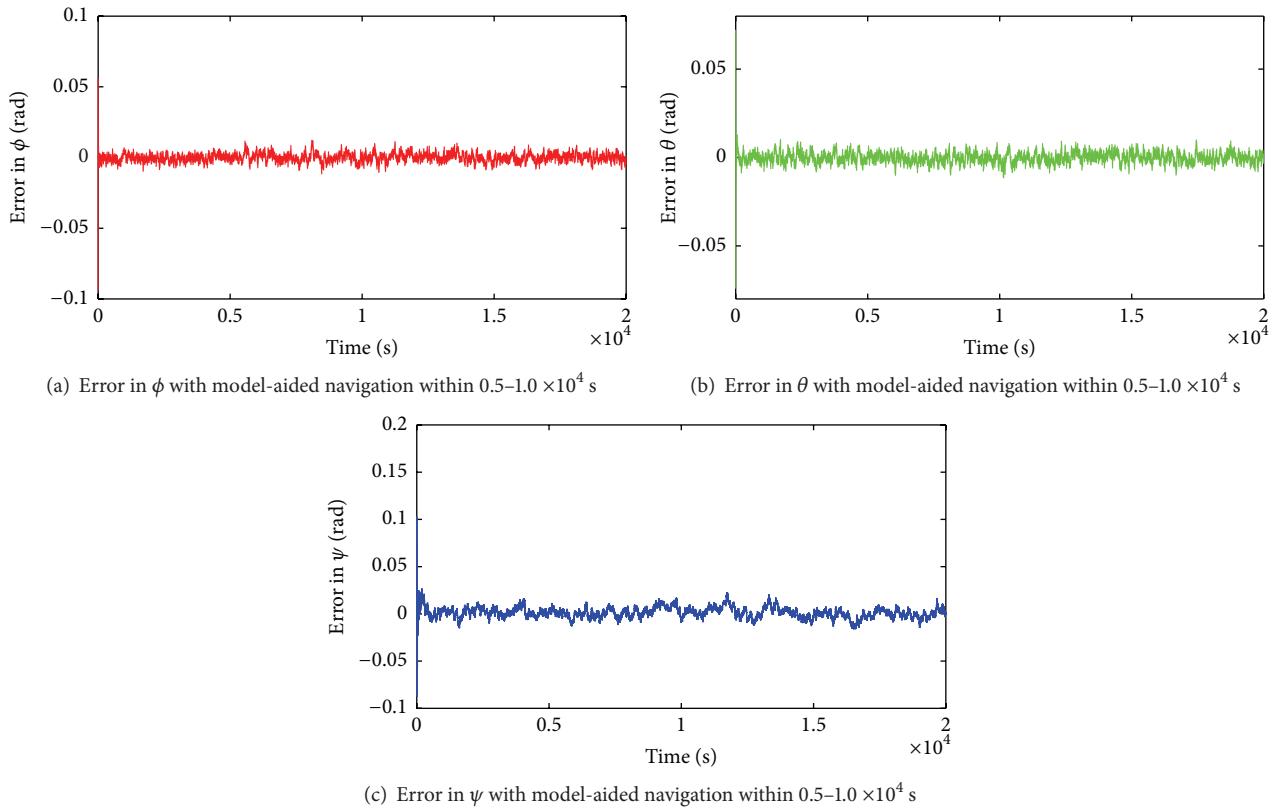
(a) Error in ϕ with model-aided navigation within 0.5–1.0 $\times 10^4$ s

(b) Error in θ with model-aided navigation within 0.5–1.0 $\times 10^4$ s

(c) Error in ψ with model-aided navigation within 0.5–1.0 $\times 10^4$ s

FIGURE 8: Errors of Euler angles in model-aided navigation within 0.5–1.0 $\times 10^4$ s (A to B) in Figure 6.

(a) x, y plot of navigation with transponder dropout for 5000 s

(b) Error in x for navigation with transponder dropout for 5000 s

(c) Error in y for navigation with transponder dropout for 5000 s

(d) Error in z for navigation with transponder dropout for 5000 s
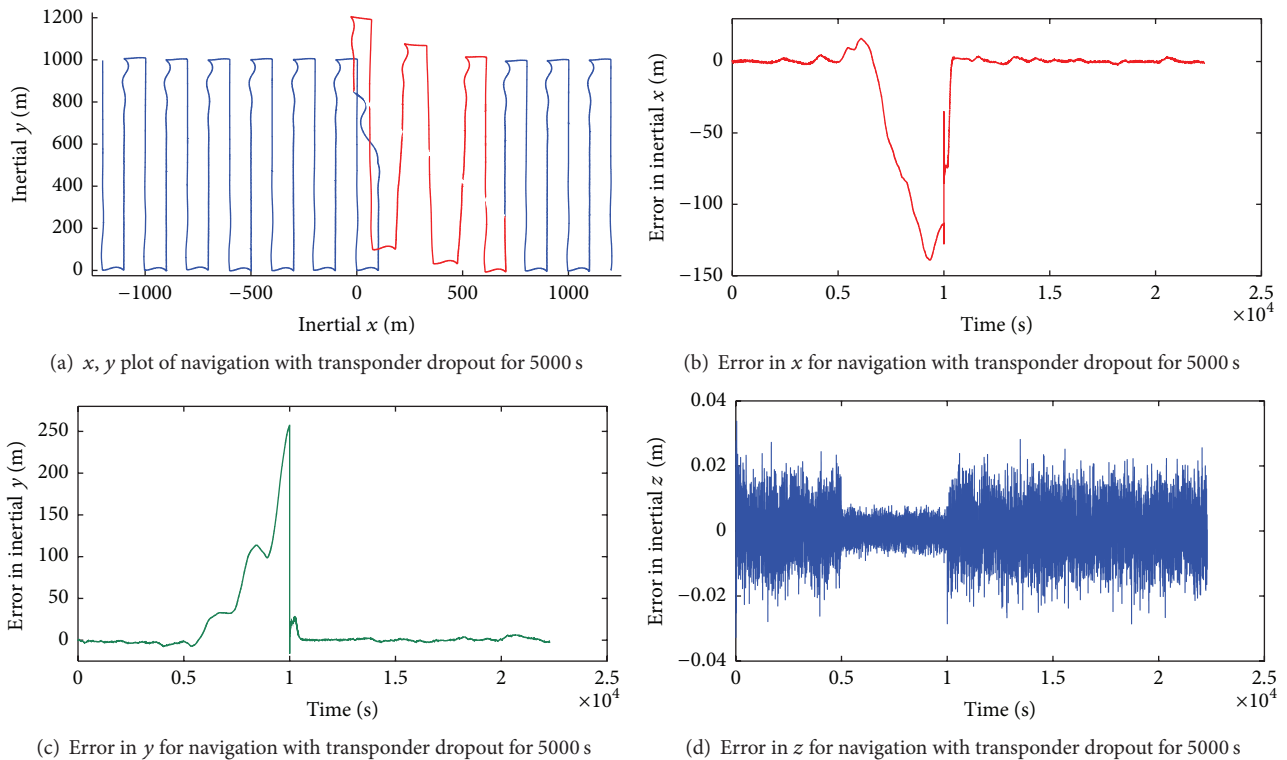
FIGURE 9: Navigation with a transponder dropout and no velocity aiding for 5000 s.

(a) Range error for incremental transponder dropouts at every 50 s

(b) Azimuth error for incremental transponder dropouts at every 50 s

(c) Error in x for incremental transponder dropouts at every 50 s

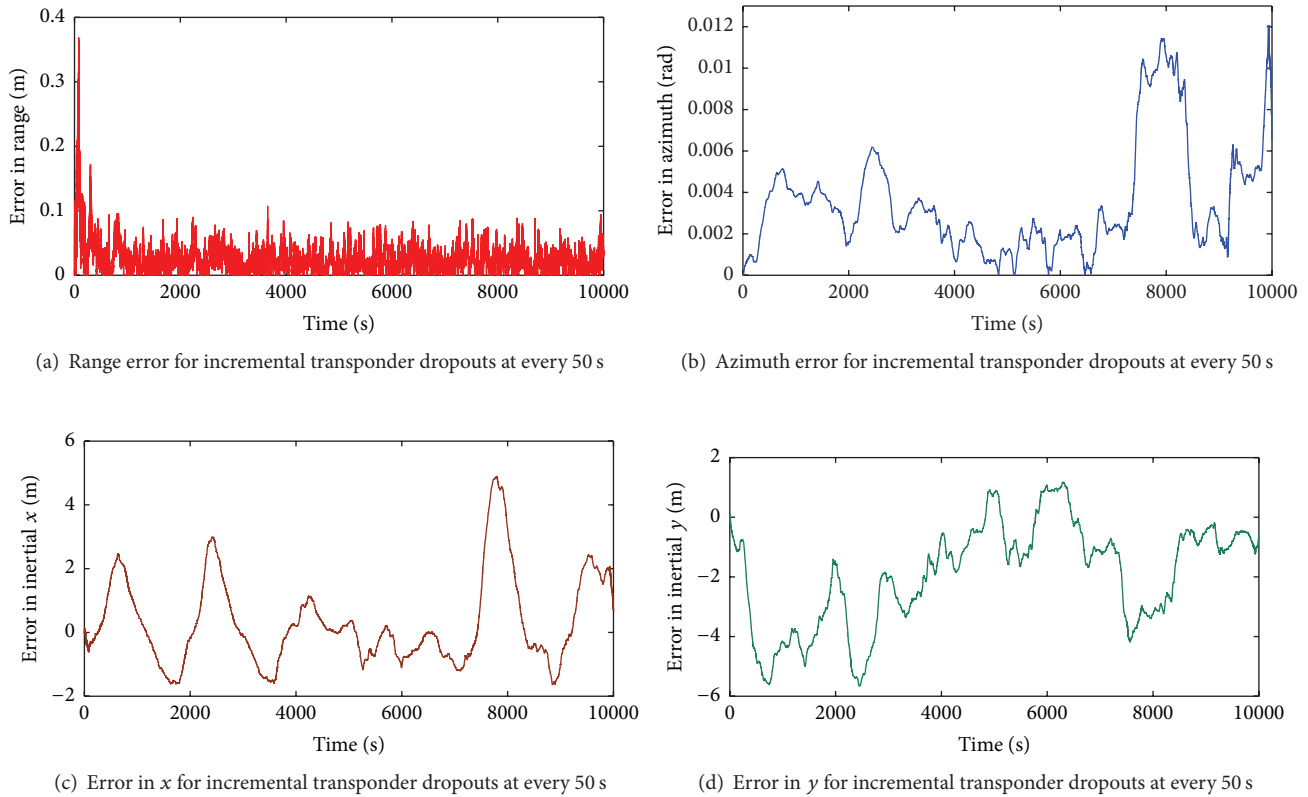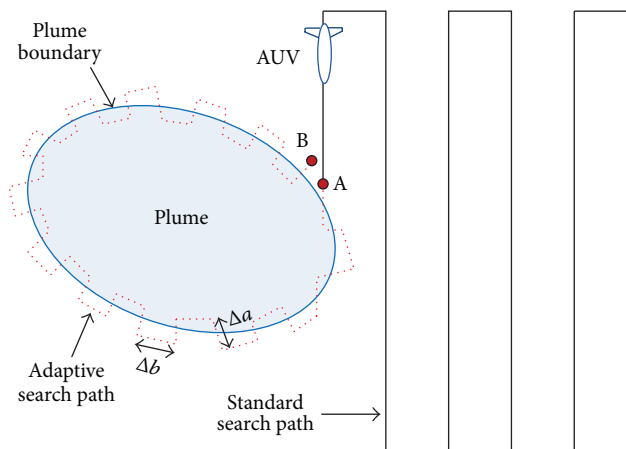(d) Error in y for incremental transponder dropouts at every 50 s

FIGURE 10: Navigation with incremental transponder dropouts at every 50 s.

FIGURE 11: Plume boundary tracking.

and 13(b) show the dispersion of the plume and the detected plume boundary in 2D and 3D view. Figure 13(c) shows the error in plume detection in x, y, z directions. Note that the section length Δb is reduced due to the irregularity of the plume dispersion.

4.3. Discussion on the Developed Plume Tracking Algorithm. The proposed approach was able to successfully track a more realistic plume boundary demonstrating its path

optimization and adaptive features. This can be observed in Figure 13(a). From sections A-B where the irregularity of the plume boundary is lower, the AUV crosses the plume more loosely. Also at the places where the irregularity seems higher (sections C-D or E-F) AUV crosses the plume tightly. Also, the AUV navigation plan is not predefined and adaptive according to the shape of the plume boundary.

Certainly there are lots of facts that could influence the limitations of the algorithm. The measurement uncertainty of

(a) Dispersion in 2D

(b) Dispersion in 3D

(c) Error in x, y, z. At 5000 s DVL drop occurs and model-aided navigation starts
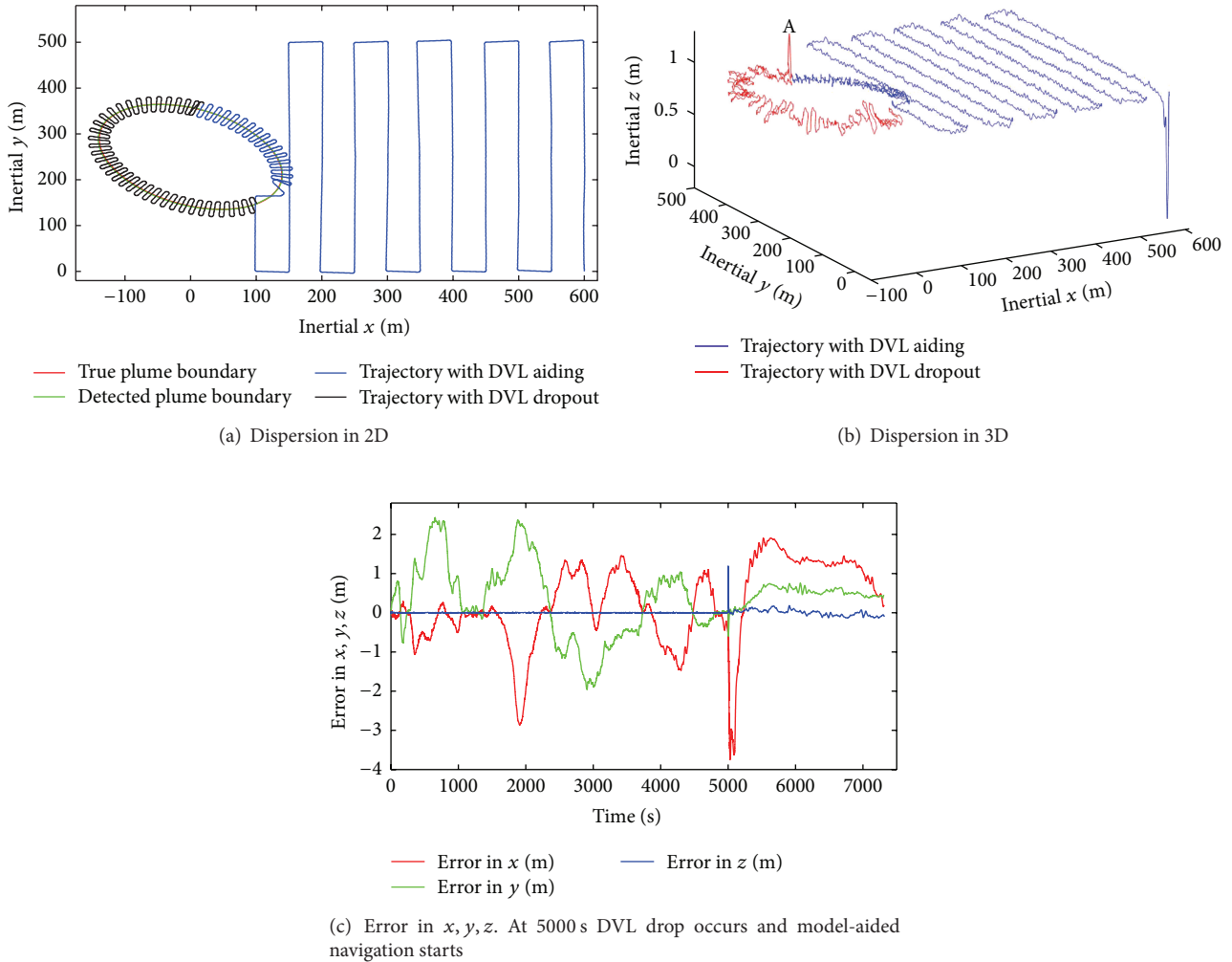
FIGURE 12: Plume with an ellipse-shaped boundary.

the oil sensor is not considered and the operation is assumed as an ON/OFF type. Consequently, those uncertainties can affect the accuracy of the detection/tracking of the plume boundary. The accuracy of the plume tracking system can be improved by adding an array of fluorometers to better estimate the oil concentration. Another challenge is the noisy plume boundaries where the gradient information based on fluorometer readings cannot be established accurately enough. Consequently, the AUV could fail to generate a reliable boundary tracking. Furthermore, the accuracy of the boundary tracking specially with irregular gradients is limited by the physical AUV dynamics and actuator constraints. Moreover, due to the medium density changes in a plume area the behavior of the acoustic could be complicated. This can introduce screening effects that severely impair the acoustic system and the AUV may fail to estimate x, y positions accurately. However, at this work we did not model the medium density changes and its effect in acoustics. We assumed that the acoustic channel behaves the same both in and out of the plume. We believe it can be an interesting future work.

5. Conclusion

In this work we have developed a multirate UKF algorithm for AUV localization in a GPS-denied undersea environment. Furthermore, an adaptive plume detection and tracking system is developed. The proposed tracking algorithm uses gradient information. The algorithms are implemented in simulations and successful results are obtained.

The adaptive plume tracking system developed in this work is not limited to tracking oil plumes and it can be also used to track other types of plumes such as biological and chemical. As a future work we expect extending the proposed system to track 3D dynamic plumes and implementing it in real time to investigate most of the issues present in physical scenarios, which are not examined in detail with the current simulation work. Moreover, extending this system for multi-AUV navigation missions such as implementing a cooperative AUV network for under-ice oil plume detection and tracking would be an interesting research study. This will decrease the overall time for feature tracking and improve the usage and sharing of information,
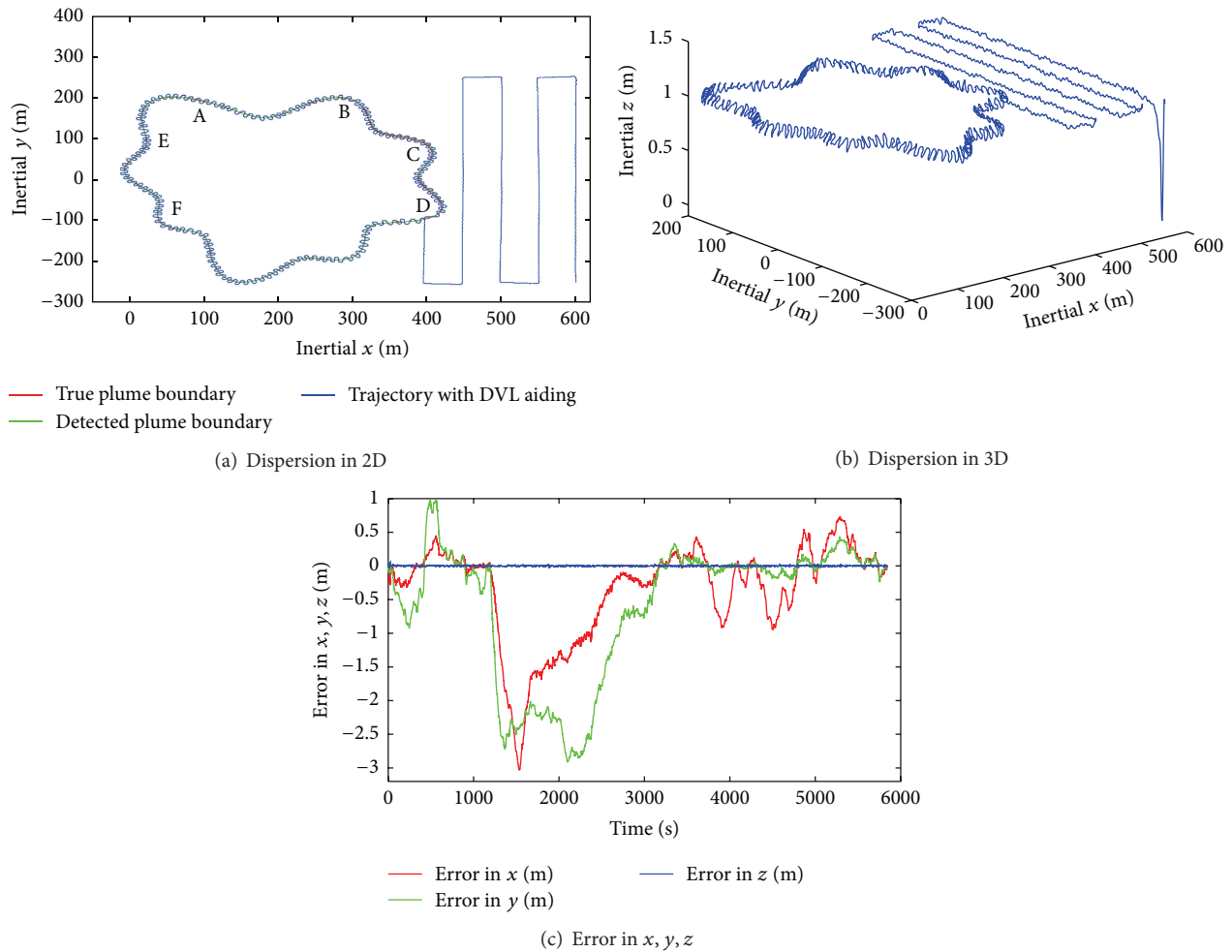
(a) Dispersion in 2D

(b) Dispersion in 3D

(c) Error in x, y, z

FIGURE 13: Plume boundary modeled using Fourier orders.

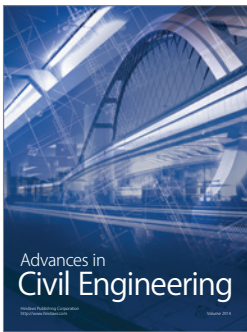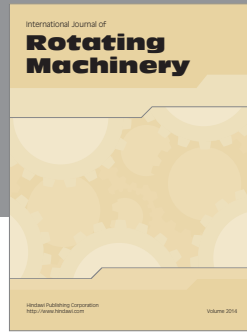which lead to a better representation of the plume boundary.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] J. Das, F. Py, T. Maughan et al., "Simultaneous tracking and sampling of dynamic oceanographic features with autonomous underwater vehicles and lagrangian drifters," in *Experimental Robotics: The 12th International Symposium on Experimental Robotics*, vol. 79 of *Springer Tracts in Advanced Robotics*, pp. 541–555, Springer, Berlin, Germany, 2014.

[2] J. Das, K. Rajan, S. Frolov et al., "Towards marine bloom trajectory prediction for AUV mission planning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 4784–4790, IEEE, Anchorage, Alaska, USA, May 2010.

[3] R. N. Smith, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *International Journal of Robotics Research*, vol. 29, no. 12, pp. 1475–1497, 2010.

[4] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis, "Collective motion, sensor networks, and ocean sampling," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 48–74, 2007.

[5] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: a review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.

[6] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, "Navigation technologies for autonomous underwater vehicles," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 38, no. 4, pp. 581–589, 2008.

[7] M. Morgado, P. Oliveira, C. Silvestre, and J. F. Vasconcelos, "Embedded vehicle dynamics aiding for USBL/INS underwater navigation system," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 1, pp. 322–330, 2014.

[8] O. Hegrenas, K. Gade, O. K. Hagen, and P. E. Hagen, "Underwater transponder positioning and navigation of autonomous underwater vehicles," in *Proceedings of the MTS/IEEE Biloxi—Marine Technology for Our Future: Global and Local Challenges (OCEANS '09)*, pp. 1–7, IEEE, Biloxi, Miss, USA, October 2009.

[9] R. M. Eustice, L. L. Whitcomb, H. Singh, and M. Grund, "Experimental results in synchronous-clock one-way-travel-time acoustic navigation for autonomous underwater vehicles," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 4257–4264, Roma, Italy, April 2007.

[10] M. F. Fallon, M. Kaess, H. Johannsson, and J. J. Leonard, "Efficient AUV navigation fusing acoustic ranging and side-scan sonar," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 2398–2405, IEEE, Shanghai, China, May 2011.

[11] A. V. Inzartsev, *Underwater Vehicles*, InTech, Rijeka, Croatia, 2009.

[12] Ø. Hegrenæs, O. Hallingstad, and K. Gade, "Towards model-aided navigation of underwater vehicles," *Modeling, Identification and Control*, vol. 28, no. 4, pp. 113–123, 2007.

[13] Ø. Hegrenaes and O. Hallingstad, "Model-aided INS with sea current estimation for robust underwater navigation," *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 316–337, 2011.

[14] T. Prestero, *Verification of a six-degree of freedom simulation mode for the REMUS autonomous underwater vehicle [Master's Thesis]*, Massachusetts Institute of Technology, Cambridge, Mass, USA, 2001.

[15] A. Jayasiri, R. G. Gosine, G. K. I. Mann, and P. McGuire, "Simulation of aided AUV navigation and adaptive plume tracking," in *Proceedings of the IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE '14)*, pp. 1–6, IEEE, Toronto, Canada, May 2014.

[16] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, John Wiley & Sons, 1994.

[17] J. Diebel, "Representing attitude: euler angles, unit quaternions, and rotation vectors," Tech. Rep., Stanford University, Stanford, Calif, USA, 2006.

[18] M. K. L. Zaworski, D. Chaberski, and M. Zielinski, "Quantization error in time-to-digital converters," *Metrology and Measurement Systems*, vol. 19, no. 1, pp. 115–122, 2012.

[19] M. Morgado, P. Oliveira, C. Silvestre, and J. Vasconcelos, "Improving aiding techniques for usbl tightly-coupled inertial navigation system," in *Proceedings of the 17th IFAC World Coggress*, pp. 15 973–15 978, Seoul, South Korea, July 2008.

[20] P. A. Miller, J. A. Farrell, Y. Zhao, and V. Djapic, "Autonomous underwater vehicle navigation," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 3, pp. 663–678, 2010.

[21] G. Grenon, P. E. An, S. M. Smith, and A. J. Healey, "Enhancement of the inertial navigation system for the morpheus autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 548–560, 2001.

[22] E. Wan and R. Van der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proceedings of the Adaptive Systems for Signal Processing, Communications, and Control Symposium (AS-SPCC '00)*, pp. 153–158, IEEE, Alberta, Canada, 2000.

[23] Z. Jin and A. L. Bertozzi, "Environmental boundary tracking and estimation using multiple autonomous vehicles," in *Proceedings of the 46th IEEE Conference on Decision and Control (CDC '07)*, pp. 4918–4923, IEEE, New Orleans, La, USA, December 2007.

[24] M. Kemp, A. Bertozzi, and D. Marthaler, "Multi-UUV perimeter surveillance," in *Proceedings of the Autonomous Underwater Vehicles (IEEE/OES '14)*, pp. 102–107, Sebasco, Me, USA, June 2004.

[25] Y. Tian, W. Li, A. Zhang, and J. Yu, "Behavior-based control of an autonomous underwater vehicle for adaptive plume mapping," in *Proceedings of the 2nd International Conference on Intelligent Control and Information Processing (ICICIP '11)*, vol. 2, pp. 719–724, IEEE, Harbin, China, July 2011.

[26] P. Rogowski, E. Terrill, M. Otero, L. Hazard, and W. Middleton, "Mapping ocean outfall plumes and their mixing using autonomous underwater vehicles," *Journal of Geophysical Research-Oceans*, vol. 117, no. 7, Article ID C07016, pp. 1–12, 2012.

[27] Y. Zhang, R. S. McEwen, J. P. Ryan et al., "A peak-capture algorithm used on an autonomous underwater vehicle in the 2010 Gulf of Mexico oil spill response scientific survey," *Journal of Field Robotics*, vol. 28, no. 4, pp. 484–496, 2011.

[28] S. Petillo, H. Schmidt, and A. Balasuriya, "Constructing a distributed AUV network for underwater plume-tracking operations," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 191235, 12 pages, 2012.

[29] C. J. Cannell, A. S. Gadre, and D. J. Stilwell, "Boundary tracking and rapid mapping of a thermal plume using an autonomous vehicle," in *Proceedings of the OCEANS*, pp. 1–6, IEEE, Boston, Mass, USA, September 2006.

[30] J. Das, K. Rajan, S. Frolov et al., "Towards marine bloom trajectory prediction for AUV mission planning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 4784–4790, Anchorage, Alaska, USA, May 2010.