

Unmanned Surface Vehicle Energy Consumption Modelling under Various Realistic Disturbances Integrated into Simulation Environment

Walid Touzout, Yahia Benmoussa, Djamel Benazzouz, Erwan Moreac, Jean-Philippe Diguet

▶ **To cite this version:**

Walid Touzout, Yahia Benmoussa, Djamel Benazzouz, Erwan Moreac, Jean-Philippe Diguet. Unmanned Surface Vehicle Energy Consumption Modelling under Various Realistic Disturbances Integrated into Simulation Environment. Ocean Engineering, 2021, 222, pp.108560. 10.1016/j.oceaneng.2020.108560 . hal-03450393

HAL Id: hal-03450393
https://cnrs.hal.science/hal-03450393

Submitted on 26 Nov 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Unmanned Surface Vehicle Energy Consumption Modelling under Various Realistic Disturbances Integrated into Simulation Environment

Walid Touzout[1], Yahia Benmoussa[1], Djamel Benazzouz[1], Erwan Moreac[2], and Jean-Phillippe Diguet[2]

[1]Solid Mechanics and Systems Laboratory-LMSS/University M'Hamed Bougara Boumerdes, Algeria
[2]Lab-STICC-CNRS/UBS University, Lorient, France
[1]{*w.touzout@univ-boumerdes.dz, y.benmoussa@univ-boumerdes.dz, d.benazzouz@univ-boumerdes.dz, erwan.moreac@univ-ubs.fr, jean-philippe.diguet@univ-ubs.fr*

Abstract

Energy consumption estimation and management of the maritime Unmanned Surface Vehicles (USV) is an important issue to deal with energy minimization techniques such as path planning, tasks scheduling etc. In this paper, we introduce the energy consumption parameter in USV simulation through three contributions: 1) An analytic USV's energy consumption model is developed based on the three-degrees-of-freedom dynamic model of surface vessels. 2) A reverse engineering approach is proposed allowing the identification of the previously used dynamic model coefficients based on a set of scenarios executed within the simulation environment discussed in [1]. 3) The simulator engine is enriched with the consumption modelling tools such that the power absorbed by the USV is instantaneously calculated and returned; so, the required energy of any predefined scenario is available as a new simulation result.

Key words–Energy consumption, Power modelling, Autonomous Robot, Unmanned Surface Vehicles, Robot Operating System, Simulation Environment.

1 Introduction

Autonomous surface drones or Unmanned Surface Vehicles (USVs) are a promising solution for different marine applications such as port navigation, rescue, environment control, military missions, ocean maps generation etc. The main advantage of the USVs is their ability to evolve in environments where humans are not able to intervene safely in addition to their cost and continuous activity. Actually, USVs operate generally in difficult environment conditions needing precision, reliability and autonomy. To provide solutions to these critical requirements, scientific community is more and more focusing their research in the USVs field and their applications [2].

One of the most challenging issues faced when dealing with USVs is the autonomy problem. Like any autonomous robot, the USVs capabilities to execute their missions are limited by the capacity of their batteries. However, this issue is more critical in case of USV since they can operate in environment where frequent battery discharges may impact considerably the good achievement of the executed mission especially that their autonomy depends on two complex and variable environment parameters: wind and water current. Thus, this energy constraint

makes the consumption optimization one of the most important challenges for mobile robots' research community. The energy limitation is therefore becoming a significant constraint to either solve path planning problems, to schedule tasks or to design an energy-based optimal controller. The energy consumption estimation of the USVs can be achieved by measuring instantaneously the real power absorbed by the vehicle using on-board sensors (mainly current or power sensors). However, this may require additional hardware which may affect the drones' behaviour due to the additional weights, as well as this approach is usually infeasible and would be very costly in case of too many scenarios or huge ones and not possible of course during the USV design phase. Another approach consists to use of simulation environment to test and validate USVs autonomy and their consumption at early phase of mission planning which allow saving cost and time. The drawback of the existing solutions is the use of simple and non-realistic energy models which are not representative of real-life scenarios. Hence, our proposed solution begins by developing a more realistic power model of the USVs based on their dynamics and to integrate it into one of the robust USV simulators that exist. There are many simulation environments for USVs such as the one presented in [1] which is an open source, recent and very robust simulator that simulates different USV types under realistic disturbances. However, none of the available simulators considers the energy consumption parameter in their engines.

In this paper, we introduce the energy consumption parameter in USVs simulation. Our methodology is general but implemented on the open-source simulation environment [1]. This last one is a recent Robot Operating System (ROS) and Gazebo-based software package that simulates virtual USVs in realistic environment and disturbances. Actually, we introduce the energy consumption parameter into the simulator through the following steps: First, an analytical USVs energy consumption model is developed based on the three-degrees-of-freedom (3-DOF) dynamic model given in [3] by ignoring the wave effect. Then, we explain how to identify the constant coefficients of the used dynamic model based on a set of scenarios executed within the simulation environment using a reverse engineering approach [4]. We apply this methodology particularly for Lutra-Prop boat, a representative differential drive USV available in the used simulation environment. Thereafter, the energy model obtained from the two previous steps is integrated into the simulator engine. Finally, we run different simulation scenarios and analyse the obtained estimated energy consumption results in a short time and without any hardware requirement.

The rest of the paper sections is organized as follows: section 2 contains the related work; section 3 involves the USV dynamics with and without disturbances; power consumption modelling is given in section 4; section 5 illustrates the model parameters identification for the Lutra-prop differential drive USV using the reverse-engineering approach; section 6 which demonstrates the integration of the obtained power model into its corresponding virtual package within the USV simulator, section 7 includes the simulation results of seven different realistic scenarios. Finally The paper is ended with a general conclusion and references.

2 Related Works

Many recent works have been published aiming to model, estimate and minimize the energy consumption of different mobile robots [5, 6, 7, 8, 9, 10, 11]. Most of researchers are considering the energy efficiency to be the key on autonomous robots' performances since they are constrained by their batteries' capacity limitation. From this literature, we notice that most of related work were about designing power models for ground robots to estimate their energy consumption. Among the contributions presented in this paper is the development of the energy consumption model of a different unmanned vehicle type which is the surface drone (or marine drone). The consumed power of a USV is that absorbed by: the on-board computer, sensors, control systems, thruster's losses, and mainly thrusters' useful power generating the mechanical energy.

Usually, The power absorbed by the on-board computer and electrical losses and components (sensors, control system, etc.) is a small quantity compared to the thruster useful power consumption [12] and can be approximated by a constant value or more accurate model. Thus, the main power consumption is due to thrusters (80-90%) transferred to the mechanical power which should be modelled according to the USV speed and environment as shown in Fig.1

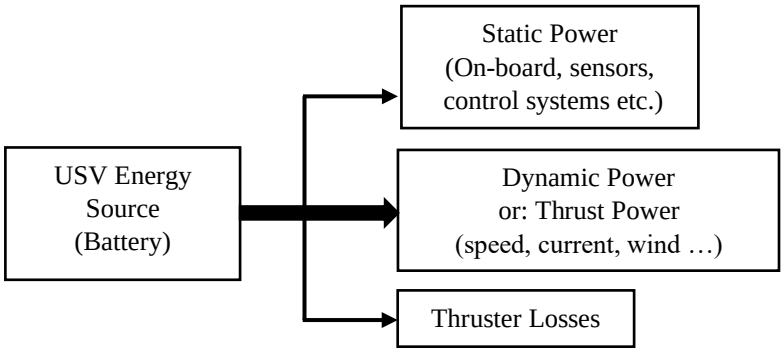


Figure 1: Energy transmission of the USV.

The development of the consumption model of a surface drone as function of its environment can be achieved with its dynamic model expressing the relationship between the velocity and a set of forces applied on the boat's body as given later in this paper; thus, many works have been taken into consideration to obtain the power model of the USVs and to identify its parameters: Muske K. R. et al [13] have given a solution to identify the parameters of a 3-DOF non-linear dynamic surface vehicle model and validated their approach by carrying out on an experimental model. Li. C et al [14] have presented also a 3-DOF dynamic model of a rudderless double thruster USV and they identified the parameters of a $1.5m$ long, $50Kg$ USV through system identification approach, the accuracy of their modelling and identification approaches was verified by an experimental testing. Niu. H et al [15] have studied the energy efficient path planning algorithms for USV and they studied in [16] the energy-based efficient path planning for USV in spatially and temporally variant environment; however, they considered a simple power model as the cost function based on 2-DOF USV dynamics to calculate the consumption weight of a given path. Mu. D et al [17] have presented a model and identification approach of the propulsion vector of a USV based on its 3-DOF dynamics and response model, and compared the simulated turning and zigzag tests with a physical $7.02m \times 2.6m$ USV. Jin. J et al [18] have designed a non-linear controller for the heading and the velocity of a USV based on 3-DOF model and identified its parameters. Sonnenburg C. R et al [19] have also described in their paper USV modelling, identification and control. Wirtensohn. S et al [20] have presented a model of a twin hull-based unmanned surface craft and identified the model parameters via a weighted least square approach. From the given literature review, we notice that energy estimation; modelling; and identification approaches require complicated tasks and eventually much hardware set-up, as well as the obtained data are not very precise due to the presence of disturbances especially in case of huge scenarios. This can be solved by establishing a suitable and more realistic power model of USVs consumption based on their dynamic behaviours and then by the integration of the obtained model after identifying its parameters into a robust simulator engine that simulates the USV performances in realistic way under realistic disturbances.

Realistic simulation and prototyping of autonomous vehicles generally and surface drones especially play a significant role to reduce the hardware set-up and the amount of time spent in developing a given mobile robot application such as the estimation of energy consumption. Moreover, simulation and estimation tools allow designers and researchers to focus on the interesting parts of their applications to achieve better results. However, before simulating a given

robotic tasks or scenarios, it is necessary to choose a suitable simulation environment since different simulators offer different performances. In addition to the above research works: Kramer. J et al [21] have presented a survey that addresses autonomous mobile Robot Development Environment (RDE) by comparing a set of open-source freely available RDEs from different points of view. Zlajpah. L [22] has given an overview of simulation in robotics by showing how simulation makes things easier and has presented some advantages and drawbacks of simulation in robotics. Pitonakova. L et al [23] provide comparison study between the most used robot simulators involving Gazebo, V-REP and ARGoS. Consequently, many open source software platforms are available in order to simulate robotic behaviours in realistic environment such as Gazebo and ROS which stands for Robot Operating System and Gazebo. Gazebo software was developed in 2002 at the University of Southern California [24]. The idea was about creating a high-fidelity simulation environment that provides the ability to simulate robots in outdoor and indoor environments under different conditions, so Gazebo is a 3D robot simulator while ROS serves as the robot's interface.

ROS [25] is a meta-operating system that provides communication between users; computers' OS; and other external equipment such as cameras, sensors as well as robots using executable codes representing ROS nodes. ROS nodes communicate with each other by sending and receiving messages organized into specific categories named topics [26]. By convention, when ROS node sends messages to a topic, it means the node is publishing to that topic. Alternatively, when the node receives messages from a topic, means the node is subscribing to that topic. Furthermore, ROS relates to many existing robot simulation frameworks such as Gazebo, a brief overview of some application software which uses ROS is given in [25]. There are many available robot prototypes and models that have been realized based on Gazebo simulator, either for ground robots, aerial drones, surface drones or underwater drones. The prototyping is done using Unified Robot Description Format (URDF) files that describe the physical shapes of the robot. The virtual robot can behave realistically by implementing Gazebo's plug-ins that is used to apply forces and torques on the robot' body based on their actuators' control and environment disturbances. These prototypes can be enriched as well as with different sensors based on Gazebo such as odometry sensors that publishes the position and the velocity vectors to a particular topic. such that Gazebo node publishes the position of the robot and its velocity vector as well as the linear and angular velocities information messages to odometry topic to be subscribed or used by other ROS nodes and processes.

Fortunately, ROS and Gazebo-based robot simulators are free and open source platforms, which help researchers to simulate their applications especially those for unmanned robots. Among the most robust and high-fidelity Gazebo-based simulators that exist we find the Unmanned Ground Vehicle simulator [27], Unmanned Underwater Vehicles UUV simulator [28], Unmanned Aerial Vehicle UAV simulator [29] and finally the Unmanned Surface Vehicle USV simulator [1]. The latter is our case of study simulator since it is a recent and robust ROS and Gazebo-based package [30]. The software simulates realistically the effect of waves, buoyancy, water current, wind current, underwater behaviours etc. It is provided with four different USV types (differential boat, air boat, sail boat and ruder boat) that can be controlled under different environment conditions and disturbances in different modes using Gazebo plug-ins to apply different forces and torques on the boat shape such as the hydrodynamic force, the hydro-static force, the wind force, the wave force, and the propulsion forces. Based on the simulated forces, the virtual boats can behave in the virtual space like the physical one; however, the energy parameter is not introduced within the simulator package which motivates us to enrich it with power management and estimation tools by integrating a developed power model into its package. this contribution is very useful to estimate the required energy of any predefined scenario.

In other words, the novelties of our contribution are given below:

1. The energy consumption model of the differential drive USVs is established as a function

of the environment (wind, speed, water current) based on a simplified 3-DOF dynamic model and fluid dynamics.

2. Using the reverse-engineering approach to identify the model parameters of an autonomous differential drive boat named: Lutra-prop USV from its virtual version available on the USV simulator [1].

3. Integration of the obtained USV consumption model into the simulator engine so that the simulator is enriched with power management and estimation tools used to estimate and verify the required power and energy for any predefined scenario on the simulator without any hardware requirement which is a significant contribution that can be used to solve many autonomy problems such as path planning, tasks scheduling etc.

3 The 3-DOF USV dynamics

In real scenarios, a marine drone can move longitudinally, laterally, and vertically, it can also move rotationally around each axis producing roll, pitch and yaw movements. This type of motion system is known as: six-degrees-of-freedom (6-DOF) system. In order to simplify our analysis, we neglect the marine wave effect; thus, the vertical motion, roll rotation and pitch rotation are ignored because the energy consumed in these dimensions is much lower. Therefore, the system is reduced to a 3-DOF motion system by considering only the surge u (longitudinal motion), sway v (sideways motion) and yaw rate r (rotation around the vertical axis). The motion velocity vector (u, v, r) can be represented in two reference frames: $\{e\}$ and $\{b\}$ as shown in Fig.2; where $\{e\}$ is the earth-fixed reference frame and $\{b\}$ is the body-fixed reference frame. In this section, we define first the 3-DOF USV dynamic model with and without disturbances (mainly: wind and water current) used to develop the power consumption model. Then we present in the next section the power modelling approach based on the given USV dynamic model.

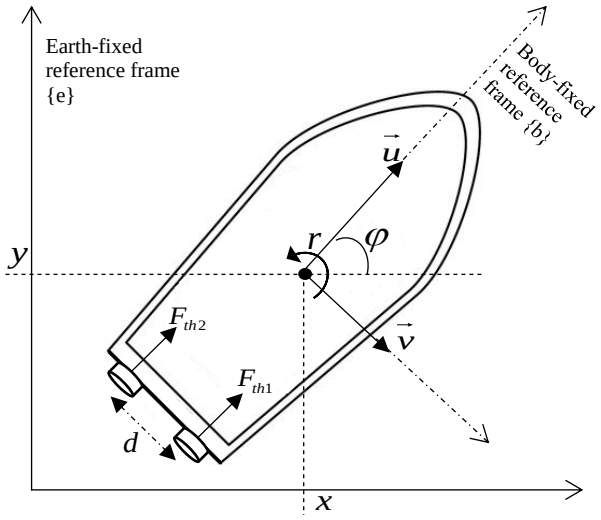


Figure 2: Differential Drive USV dynamics representation in $\{e\}$ and $\{b\}$ frames.

3.1 Three-DOF USV dynamics without disturbances

In order to exploit the physical properties of the sea-keeping and manoeuvring models, the equations of motions are represented in a vector form to reduce the number of coefficients and

to simplify calculations. Thus, the complete 3-DOF dynamic model of the USVs considered in this paper uses the representation in $\{b\}$ frame given in Eq. (1)[3]

$$M.\dot{V} + C(V).V + D(V).V = \tau_{thrust} + \tau_{disturbance} \tag{1}$$

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & -m.y_g \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ -my_g & mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix}$$

$$C(V) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) + Y_{\dot{v}} v + \frac{Y_{\dot{r}} + N_{\dot{v}}}{2} r \\ 0 & 0 & (m - X_{\dot{u}})u \\ m(x_g r + v) - Y_{\dot{v}} v - \frac{Y_{\dot{r}} + N_{\dot{v}}}{2} r & -(m - X_{\dot{u}})u & 0 \end{bmatrix}$$

$$D(V) = D + D_n(V) = -\begin{bmatrix} X_u & 0 & 0 \\ 0 & Y_v & Y_r \\ 0 & N_v & N_r \end{bmatrix} - \begin{bmatrix} X_{u|u|}|u| & 0 & 0 \\ 0 & Y_{v|v|}|v| + Y_{v|r|}|r| & Y_{r|v|}|v| + Y_{r|r|}|r| \\ 0 & N_{v|v|}|v| + N_{v|r|}|r| & N_{r|v|}|v| + N_{r|r|}|r| \end{bmatrix}$$

where; $V = [u, v, r]^T$ is the velocity vector with respect to $\{b\}$ frame given in Fig.2. The components (u, v, r) are the coordinates of the velocity vector V representing the surge velocity, sway velocity and yaw angular velocity respectively. M represents the inertia matrix including the added mass parameters. The vector components (x_g, y_g) represent the coordinates of the gravity centre point of the USV in $\{b\}$ frame. C is the Coriolis and centripetal matrix. D is hydrodynamic drag matrix which contains the linear drag term D and the non-linear drag term $D_n(v)$. τ_{thrust}, $\tau_{disturbance}$ are the thruster and disturbance (wind, wave) forces respectively applied on the USV body expressed in $\{b\}$ frame. $X_{\dot{u}}$, $Y_{\dot{v}}$, $Y_{\dot{r}}$, $N_{\dot{v}}$, $N_{\dot{r}}$, $I_{\dot{z}}$,X_u, Y_v, Y_r, N_v, N_r are the corresponding hydrodynamic coefficients respectively.

In this paper, we supposed that the USV is operated as a differential drive boat having two propellers (thrusters) placed on the rear portion of the boat (see Fig. 2) to differentially control the boat's speed and direction. Thus, the resultant thruster force vector for the differential drive USV is given as follows: [17, 18]

$$\tau_{thrust} = \begin{bmatrix} \tau_u \\ 0 \\ \tau_r \end{bmatrix} = \begin{bmatrix} F_{Th1} + F_{Th2} \\ 0 \\ (F_{Th1} - F_{Th2}).\frac{d}{2} \end{bmatrix} \tag{2}$$

where; τ_u is the longitudinal thrust, τ_r is the yaw moment. F_{Th1} and F_{Th2} are the left and right thrusters' forces (thrust) respectively along $x - axis$ in $\{b\}$ frame, d is the distance between the centrelines of each thruster as given in Fig.2. To develop the proposed dynamical model, we consider the following hypotheses [14, 17]:

1. The maximum speed of the USV does not exceed $1.5m/s$, hence, the effect of non-linear drag term of the hydrodynamic drag matrix can be neglected i.e. $D_n(v) = O$.

2. The off-diagonal terms are much smaller than the diagonal terms of the matrices M and D, so the off-diagonal terms of both matrices can be ignored.

3. The coincident center of the added mass and gravity N_v can be replaced by Y_r. A combination of approximate fore-aft symmetry and light draft suggests that the sway force arising from yaw rotation and the yaw moment induced by the acceleration in the sway direction are much smaller than the inertial and added mass terms. Therefore, we assume in the Coriolis and centripetal matrix $C(v)$ that $N_v = Y_r = 0$.

As a result, the simplified 3-DOF dynamic model of the USV is simplified as given in Eq.(3).

$$M.\dot{V} + C(V).V + D(V).V = \tau_{thrust} + \tau_{disturbance} \tag{3}$$

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 \\ 0 & 0 & I_z - N_{\dot{r}} \end{bmatrix} = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix}$$

$$C(v) = \begin{bmatrix} 0 & 0 & -(m - Y_{\dot{v}})v \\ 0 & 0 & (m - X_{\dot{u}})u \\ (m - Y_{\dot{v}})v & -(m - X_{\dot{u}})u & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -m_{22}v \\ 0 & 0 & m_{11}u \\ m_{22}v & -m_{11}u & 0 \end{bmatrix}$$

$$D(v) = -\begin{bmatrix} X_u & 0 & 0 \\ 0 & Y_v & 0 \\ 0 & 0 & N_r \end{bmatrix} = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix}$$

Since the given model is expressed in $\{b\}$ reference frame; hence, some appropriate kinematic transformations between the two references $\{e\}$ and $\{b\}$ should be derived. Let's consider that η and $\dot{\eta}$ are position vector and velocity vector respectively in $\{e\}$ reference frame. The velocity vector represented in $\{e\}$ frame is given by Eq.(4)[14, 17].

$$\dot{\eta} = [\dot{x}, \dot{y}, \dot{\varphi}]^T = J(\eta).v \tag{4}$$

Where the vector η is the position vector in $\{e\}$ frame given by Eq.(5). x and y are the USV's position coordinates in $x - axis$ and $y - axis$ respectively in $\{e\}$ frame and φ represents its yaw angle (or angle of attack of the USV). The term $J(\eta)$ is the transformation matrix [17, 19] given by Eq.(6). Thus, the velocity vector in $\{e\}$ frame is given by Eq.(7).

$$\eta = [x, y, \varphi]^T \tag{5}$$

$$J(\eta) = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ \sin\varphi & -\cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$\dot{\eta} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} u\cos\varphi + v\sin\varphi \\ u\sin\varphi - v\cos\varphi \\ r \end{bmatrix} \tag{7}$$

The identification and verification of the 3-DOF dynamic model of the case of study surface drone is done using an expanded form given in Eq(8,9,10,11) extracted from Eq.(3) and Eq.(7). The details of the model parameters identification approach is given later in section 5.

$$\begin{cases} \dot{x} = u\cos\varphi + v\sin\varphi \\ \dot{y} = u\sin\varphi - v\cos\varphi \\ \dot{\varphi} = r \end{cases} \tag{8}$$

$$(m - X_{\dot{u}})\dot{u} - (m - Y_{\dot{v}})vr + X_u u = F_{Th} = F_{Th1} + F_{Th2} \tag{9}$$

$$(m - Y_{\dot{v}})\dot{v} + (m - X_{\dot{u}})ur + Y_v v = 0 \tag{10}$$

$$(I_z - N_{\dot{r}})\dot{r} + (X_{\dot{u}} - Y_{\dot{v}})uv + N_r r = F_N\frac{d}{2} = (F_{Th1} - F_{Th2})\frac{d}{2} \tag{11}$$

3.2 USV Dynamics with water current disturbances

Let's assume that the USV is moving on an environment having a considerable water current disturbance. The water current can be either helpful or obstructive to USV motion and therefore, it has a significant impact on the USV behaviour and consumption. The analytic modelling of the water current disturbances is done by differentiating the actual USV velocity vector and the water current velocity vector to get the USV's relative speed as given by Eq.(12) [3].

$$\dot{\eta}_r = \dot{\eta} - V_c \tag{12}$$

Where $\dot{\eta}_r$ is the relative velocity vector of the USV represented in $\{e\}$ frame, V_c is the water current vector represented in $\{e\}$ frame as well such that:$V_c = [V_{cx}, V_{cy}, 0]^T$.

$$\dot{\eta}_r = \begin{bmatrix} u\cos\varphi + v\sin\varphi - V_{cx} \\ u\sin\varphi - v\cos\varphi - V_{cy} \\ r \end{bmatrix} \tag{13}$$

In order to find the relative velocity vector in $\{b\}$ reference frame, we use Eq.(4) and the inverse of the transformation matrix as follows:

$$\begin{aligned} \dot{\eta}_r &= J(\eta).v_r \\ J^{-1}(\eta).\dot{\eta}_r &= J^{-1}(\eta).J(\eta).V_r \\ V_r &= J^{-1}(\eta).\dot{\eta}_r \end{aligned} \tag{14}$$

where: V_r is the USV's relative velocity in $\{b\}$ frame such that: $V_r = [u_r, v_r, r_r]^T$ and $J^{-1}(\eta)$ is the inverse transformation matrix given in Eq.(15).

$$J^{-1}(\eta) = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ \sin\varphi & -\cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ \sin\varphi & -\cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{15}$$

Thus, the relative velocity vector represented in $\{b\}$ frame is given in Eq.(16).

$$\begin{aligned} V_r = \begin{bmatrix} u_r \\ v_r \\ r \end{bmatrix} &= \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ \sin\varphi & -\cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} u\cos\varphi + v\sin\varphi - V_{cx} \\ u\sin\varphi - v\cos\varphi - V_{cy} \\ r \end{bmatrix} \\ &= \begin{bmatrix} u - V_{cx}\cos\varphi - V_{cy}\sin\varphi \\ v - V_{cx}\sin\varphi + V_{cy}\cos\varphi \\ r \end{bmatrix} \end{aligned} \tag{16}$$

Based on Eq.(3), the thrusters' force vector can be expressed now as function of the relative velocity as given in Eq(17). The wind force modelling is established in the next subsection.

$$\tau_{thrust} = M.\dot{V}_r + C(V_r).V_r + D(V_r).V_r - \tau_{wind} \tag{17}$$

3.3 USV Dynamics with wind disturbances

Like water current, wind disturbances can be either helpful or obstructive according to the direction of the motion and that of the wind. To study the wind effect on the USV energy consumption, we need to model both lateral and frontal wind forces applied on the lateral and frontal areas of the non-submerged part of the boat respectively. Let $V_w = [V_{wx}, V_{wy}]^T$ be the non-rotational wind velocity vector expressed in $\{e\}$ frame, and$V_w = [V_w, \gamma_w]^T$ represents its vector in $\{b\}$ frame as shown in Fig.3.

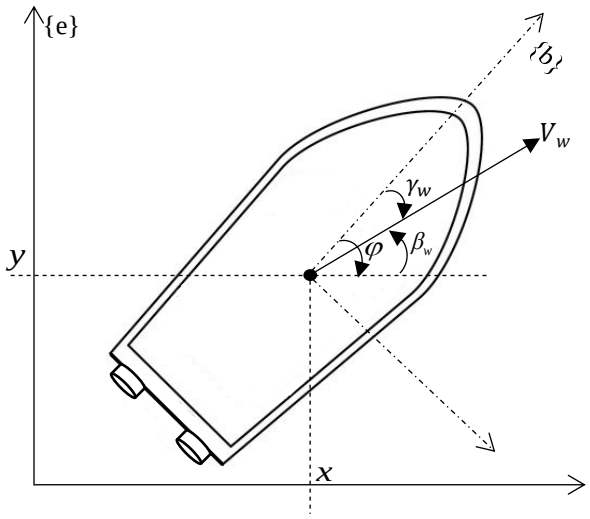


Figure 3: Wind speed vector representation in $\{e\}$ and $\{b\}$ frames.

In the 3-DOF system, the wind disturbance creates wind force $\tau_{wind} = [X_{wind}, Y_{wind}, N_{wind}]^T$ on the centre of the boat as given below [3]

$$\tau_{wind} = \begin{bmatrix} X_{wind} \\ Y_{wind} \\ 0 \end{bmatrix} = \begin{bmatrix} q.C_X(\gamma_w).A_{Fw} \\ q.C_y(\gamma_w).A_{Lw} \\ 0 \end{bmatrix} \tag{18}$$

Where; γ_w is the angle of attack of wind with respect to the USV in $\{b\}$ frame such that: $\gamma_w = \varphi - \beta_w$. q is the dynamic pressure of the apparent wind given by:

$$q = \frac{1}{2}\rho_a V_w^2$$

β_w is the wind direction given in $\{e\}$ reference frame. A_{Fw} and A_{Lw} are the frontal and lateral projected areas respectively. ρ_a is the air density which approximately equals to $1.184Kg/m^3$ when $T = [10°C, 25°C]$. For ships that are symmetrical with respect to xz plane, the wind coefficients $C_x(\gamma_w)$ and $C_y(\gamma_w)$ for horizontal plane motions can be approximated as given in [3] such that: $C_x(\gamma_w) = c_x \cos(\gamma_w)$, $C_y(\gamma_w) = c_y sin(\gamma_w)$. Where: c_x and c_y are some constant values that depend on the physical shape of the vessel. Thus, the wind force applied on USV center in $\{b\}$ is expressed as follows:

$$\begin{aligned} \tau_{wind} &= \begin{bmatrix} X_{wind} \\ Y_{wind} \\ N_{wind} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\rho_a.A_{Fw}.c_x.V_w^2 \cos(\gamma_w) \\ \frac{1}{2}\rho_a.A_{Lw}.c_y.V_w^2.\sin(\gamma_w) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2}\rho_a.A_{Fw}.c_x.V_w^2.\cos(\varphi - \beta_w) \\ \frac{1}{2}\rho_a.A_{Lw}.c_y.V_w^2.\sin(\varphi - \beta_w) \\ 0 \end{bmatrix} \end{aligned} \tag{19}$$

4 USV power consumption model

As previously mentioned, the consumed power of a USV is divided into two parts: the static power due to static consumption (on-board computer and electrical losses) and the dynamic power involving the thrust power. Thus, the total consumption of a given scenario is obtained

by integrating the instantaneous power in the scenario's duration interval. In other word, the total energy consumption of the USV can be modelled by Eq.(20).

$$E_{USV} = \int P_{USV}.dt \tag{20}$$

Where, P_{USV} is the total power absorbed by the USV given by Eq.(21).

$$P_{USV} = P_{thrusters} + P_{static} = P_{Thrust} + P_{Losses} + P_{static} \tag{21}$$

Such that, P_{static} is the static power absorbed by electrical devices, $P_{thrusters}$ is the total power absorbed by the thrusters (DC motors) involving; P_{thrust} which is the useful power converted to mechanical power; P_{losses} is the power due to the electrical losses of the thrusters. The thrust power P_{thrust} can to be modelled as function of the environment of the boat (wind, current, speed etc.) using the USV dynamics by making the scalar product of the relative velocity vector of the USV and the thruster force given in Eq.(17). The thrust power is obtained and presented in Eq.(22).

$$P_{thrust} = \tau_{thrust}.V_r = M.\dot{V}_r.V_r + C(V_r).V_r.V_r + D(V_r).V_r.V_r - \tau_{wind}.V_r \tag{22}$$

Thus,

$$\begin{aligned} P_{thrust} = & \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix} . \begin{bmatrix} \dot{u}_r \\ \dot{v}_r \\ \dot{r}_r \end{bmatrix} \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & + \begin{bmatrix} 0 & 0 & -m_{22}v_r \\ 0 & 0 & m_{11}u_r \\ m_{22}v_r & -m_{11}u_r & 0 \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & + \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} - \tau_{wind}. \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \end{aligned} \tag{23}$$

By putting Eq.(19) in Eq.(23), the global power model of the USVs is obtained as function of its environment (speed, water current and wind disturbances) as given in Eq.(24).

$$\begin{aligned} P_{USV} = & \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix} . \begin{bmatrix} \dot{u}_r \\ \dot{v}_r \\ \dot{r}_r \end{bmatrix} \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & + \begin{bmatrix} 0 & 0 & -m_{22}v_r \\ 0 & 0 & m_{11}u_r \\ m_{22}v_r & -m_{11}u_r & 0 \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & + \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & - \begin{bmatrix} \frac{1}{2}\rho_a.A_{Fw}.c_x.V_w^2.\cos(\varphi - \beta_w) \\ \frac{1}{2}\rho_a.A_{Lw}.c_y.V_w^2.\sin(\varphi - \beta_w) \\ 0 \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} + P_{Losses} + P_{static} \end{aligned} \tag{24}$$

With:

$$V_r = \begin{bmatrix} u_r \\ v_r \\ r \end{bmatrix} = \begin{bmatrix} u - V_{cx}\cos\varphi - V_{cy}\sin\varphi \\ v - V_{cx}\sin\varphi + V_{cy}\cos\varphi \\ r \end{bmatrix}$$

Consequently, the energy consumption model is obtained by Eq.(25).

$$E_{USV} = \int P_{USV}.dt = \sum P_{USV}.\Delta t \tag{25}$$

The terms m_{11},m_{22} and m_{33} represent the added mass parameters, and the terms d_{11},d_{22} and d_{33} represent the dynamic coefficients. Both added masses and dynamic coefficients are the power model parameters to be identified. The developed model is general and can be applied to any differential drive USV as long as the considered assumptions are true. In the next section we illustrate the approaches used to identify the power model parameters of the case of study differential drive boat USV named Lutra-prop given in Fig.4.

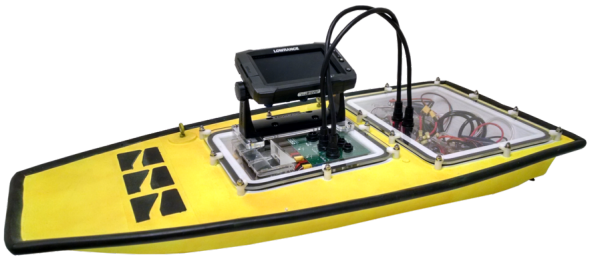


Figure 4: Real Lutra-prop USV

5 Model Parameters Identification approach applied to Lutra-prop Boat

This section details the approach used to identify the added mass parameters m_{11},m_{22},m_{33} and the dynamic coefficient d_{11},d_{22},d_{33} of a case of study drone to obtain its power model given in Eq.(24). The added mass parameters are identified based on their estimated expressions given in [13] and [14]. Whereas a reserve engineering approach [4] is used to identify the dynamic coefficients from a recent USV simulator [1] that includes a virtual version of the case study USV. In our work, the considered drone is the differential drive Lutra-prop boat (see Fig. 4) having the characteristics given in Table 1.

Table 1: Real Lutra-prop USV parameters.

Parameters	Numerical values
Length (L)	$1.06m$
Width (W)	$0.48m$
Height (H)	$0.15m$
Hull volume (V_H)	$0.02m^3$
Weight (m)	$9.7Kg$
Maximum thruster force (F_{Th})	$23N(F_{Th1} = F_{Th2} = 11.5)$
Maximum surge velocity (u)	$1.35m/s$
Moment of inertia (I_z)	$1.094Kg.m^2$
Submerged depth (D)	$0.02m$
Distance between the thrusters (d)	$0.16m$

5.1 Added mass parameters identification

The mass parameters m_{ii} include added mass contribution that represent hydraulic pressure forces and torque due to force harmonic motion of the vessel which are proportional to acceleration [13]. Using Table 1 and the estimate of the added mass terms, the added mass parameters of the Lutra-prop differential drive boat could be obtained as follows [13, 14]:

$$m_{11} = m - X_{\dot{u}} \simeq m + 0.05 = 9.75Kg \tag{26}$$

$$m_{22} = m - Y_{\dot{v}} = m + 0.5(\rho\pi D^2 L) = 10.364Kg \tag{27}$$

$$m_{33} = I_Z - N_{\dot{r}} = \frac{m(L^2 + W^2) + 0.5(0.1md^2 + \rho\pi D^2 L^3)}{12} \simeq 1.158Kg \tag{28}$$

where ρ is the water density ($\approx 1025Kg/m^3$ for salt water at moderate temperature)

5.2 Dynamic coefficients identification

The USV simulator [1] is used to identify the hydrodynamic coefficient parameters d_{ii} of the virtual Lutra-prop differential drive USV (see Fig. 5) using the reverse-engineering approach. In this section we explain three considered scenarios used to identify each parameter: Scenario (a) (linear motion) which allows us to identify the parameter d_{11} in the steady state phase using Eq.(9). Scenario (b) (circular motion) which provides the identification of d_{22} using Eq.(10) in the steady state mode. And finally Scenario (c) (rotation around the center point) which enables the identification of d_{33} in the steady state mode as well using Eq.(11). Since the USV simulator is a ROS-based software, a ROS node named */plot* is implemented to subscribe to */state* topic to repeatedly receive the odometry information of the boat as odometry message type involving the position of the boat in $\{e\}$ frame and its velocity vector in $\{b\}$ frame. The odometry information is published by Gazebo ROS node via position and velocity sensors provided within the Gazebo simulator. The ROS node */plot* plots different graphs needed to identify each parameter from each scenario as described in this section.

5.2.1 Scenario (a) - Linear motion

The USV is configured to move longitudinally with maximum thrust $F_{Th1} = F_{Th2} = 11.5N$) by neglecting all disturbance. The USV moves through a linear trajectory given in Fig.6-a. The surge velocity for this scenario was recorded and plotted by the RPS node process */plot* and given in Fig.6-b. In the steady state of scenario (a), the sway velocity and yaw rate are null, as well as the surge acceleration. i.e. $\dot{u} = r = v = 0$, and the boat moves at the maximum speed i.e. $u = u_{max} = 1.35m/s$. Thus, by using Eq.(9) the first dynamic coefficient d_{11} is obtained as given in Eq.(29).

$$d_{11} = X_u = \frac{F_{Th}}{u_{\max}} \simeq 16.296Ns/m \tag{29}$$

5.2.2 Scenario (b) -Rotation around about the vertical axis

In this scenario, the USV thrusters were configured to thrust in opposite directions i.e. $F_{Th1} = -F_{Th2} = 11.5N$ making the USV rotating around the vertical axis as illustrated in Fig.7-a. The yaw rate was recorded and plotted by the process of the ROS node */plot* and given in Fig.7-b. In the steady state of this scenario, the surge and sway velocities are null values as well as the yaw rate acceleration i.e. $\dot{r} = u = v = 0$, which implies that the d_{33} coefficient can be obtained from Eq.(11) as follows:

$$d_{33} = N_r = \frac{F_N.d}{2r_{\max}} \simeq 4.63Nms/rad \tag{30}$$

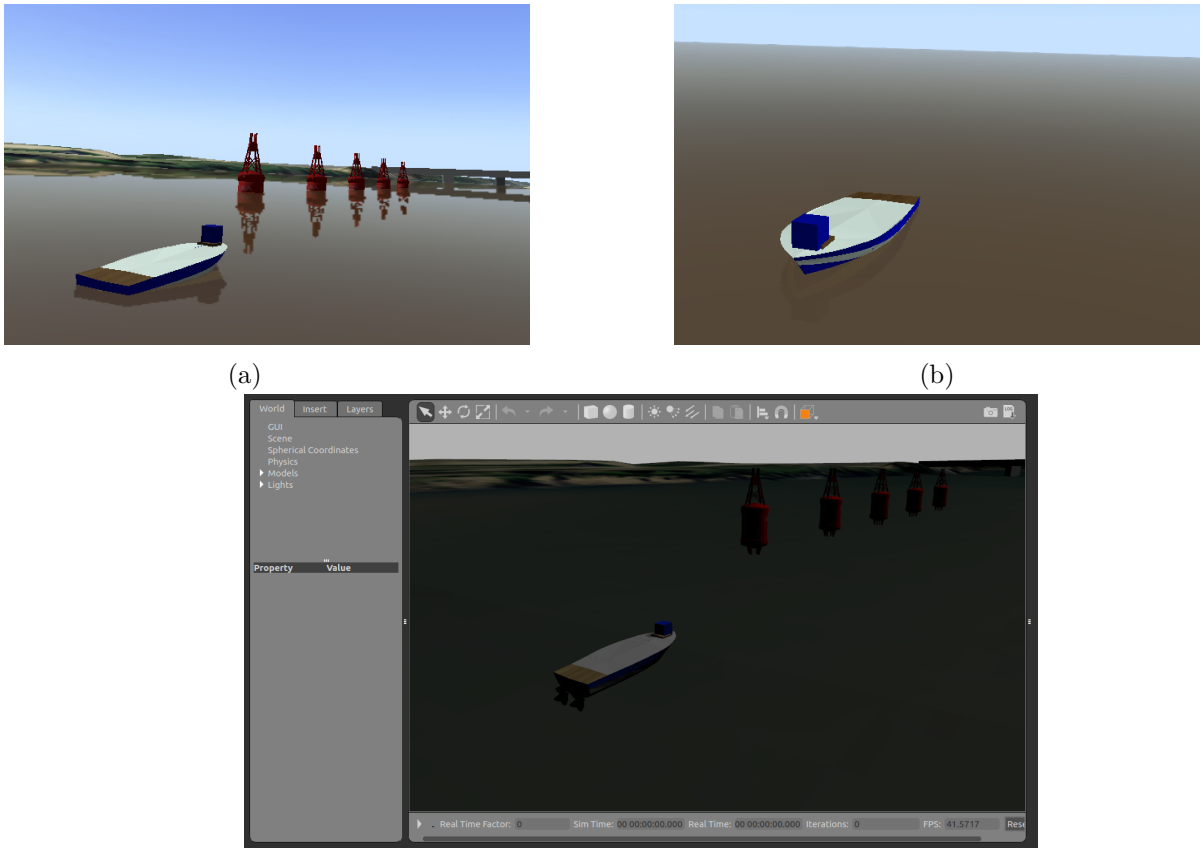


Figure 5: Lutra-prop boat on the under Water simulator window (a,b) and on Gazebo window (c)

5.2.3 Scenario (c) -Circular motion

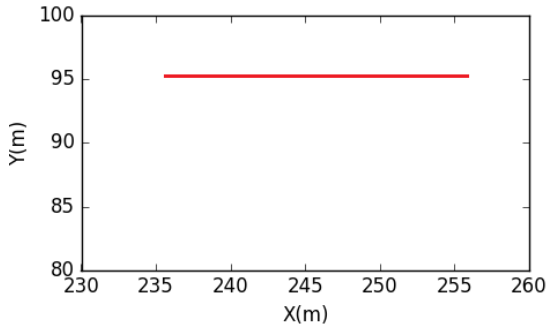
The USV is configured in this scenario to be controlled only by one thruster at maximum thrust value i.e. $F_{Th1} = 11.5N$, $F_{Th2} = 0N$. The boat follows a circular trajectory shown in Fig.8-a. The instantaneous values of surge sway and yaw rate u, v, r are recorded through this scenario and plotted by the same ROS node process and presented in Fig.8-b to Fig.8-d. In the steady state mode, the sway acceleration is a null value i.e. $\dot{v} = 0$. Thus, by using Eq.(10), we obtain the d_{22} parameter as given in Eq.(31).

$$d_{22} = Y_v = \frac{-m_{11}ur}{v} = \frac{-6.78 \times 0.92 \times (-0.25)}{0.22} \simeq 10.193 Kg.rad/s \tag{31}$$

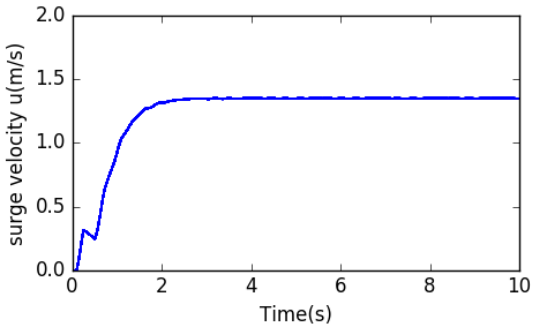
5.3 Power model expression of the case of study USV

After identifying the parameters of the Lutra-prop USV model, the complete 3-DOF dynamic model parameters are summarized in Table 2. The obtained results have been validated by verifying Eq.(9), Eq.(10), and Eq.(11) onto the simulator which show the correctness of obtained parameters and the effectiveness of the identification approach.

The USV simulator is provided with Gazebo model plug-in that applies wind force on the centre point of the USV as function of wind velocity and some parameters given in Eq.(19). In addition to the dynamic model parameters of the Lutra-prop boat given in Table 2, the wind force parameters used by the plug-in are extracted and given in Table 3.

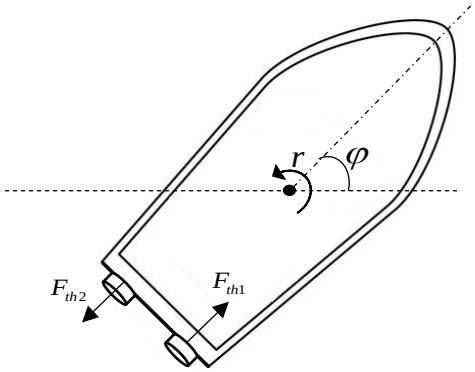


(a) Trajectory line

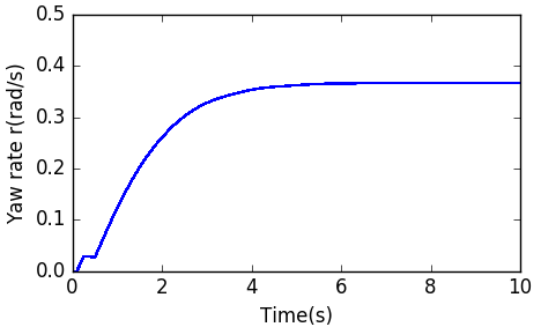


(b) Surge velocity $u(t)$

Figure 6: Trajectory and the linear speed representation for scenario a



(a) Thrust force representation



(b) Yaw rate $r(t)$

Figure 7: Thrust forces and the angular speed representations for scenario b

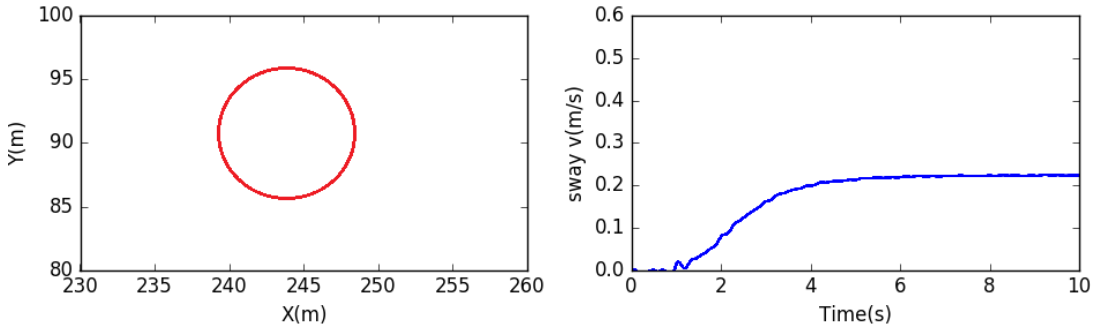
Finally, the Lutra-prop differential boat USV energy consumption model is established as function of the environment and given in Eq.(32). As a remark, Coriolis centripetal matrix has been omitted by calculation hence, it has no effect in the energy consumption of the surface drones.

$$\begin{aligned} P_{USV} = & \begin{bmatrix} 6.78 & 0 & 0 \\ 0 & 1.06 & 0 \\ 0 & 0 & 6.03 \end{bmatrix} . \begin{bmatrix} \dot{u}_r \\ \dot{v}_r \\ \dot{r}_r \end{bmatrix} \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & + \begin{bmatrix} 16.30 & 0 & 0 \\ 0 & 7.09 & 0 \\ 0 & 0 & 4.63 \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} \\ & - \begin{bmatrix} 0.032 V_w^2 . \cos(\varphi - \beta_w) \\ 0.105 V_w^2 . \sin(\varphi - \beta_w) \\ 0 \end{bmatrix} . \begin{bmatrix} u_r \\ v_r \\ r_r \end{bmatrix} + P_{static} \end{aligned} \tag{32}$$

With:

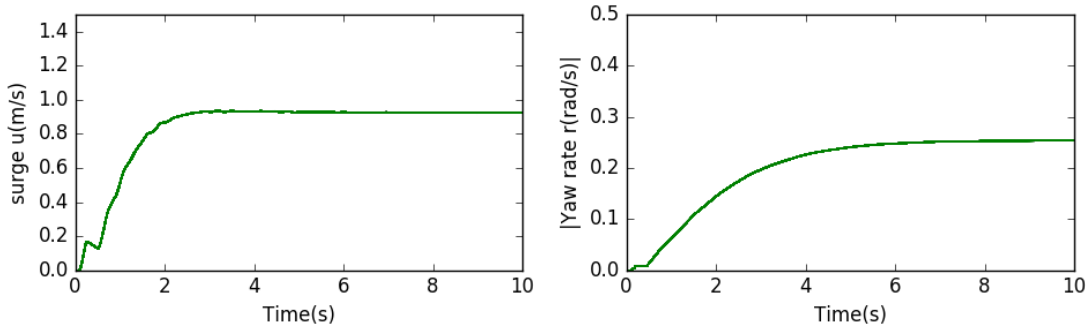
$$\begin{bmatrix} u_r \\ v_r \\ r \end{bmatrix} = \begin{bmatrix} u - V_{cx} \cos\varphi - V_{cy} \sin\varphi \\ v - V_{cx} \sin\varphi + V_{cy} \cos\varphi \\ r \end{bmatrix} \tag{33}$$

Where: $[u_r, v_r, r]^T$ is the USV's relative velocity vector, $[u, v, r]^T$ is the actual USV velocity vector, $[V_{cx}, V_{cy}]^T$ is the water current speed vector given in $\{e\}$ frame, $[V_w, \beta_w]^T$ is the polar



(a) The circular trajectory taken by the USV

(b) Sway velocity $v(t)$



(c) Surge velocity $u(t)$

(d) Yaw rate $r(t)$

Figure 8: Trajectory and velocity representation for scenario c

vector of the wind speed in $\{e\}$ frame as well. ϕ is the angle of attack of the USV. As well as P_{static} is the static power absorbed by the electrical components the USV.

The energy consumption model of the differential drive Lutra-boat is well established and presented in this section. However, in order to verify and test the model, we propose to integrate it into the simulator engine by implementing additional processes that calculate instantaneously the power absorbed by the USV when it is operating. Therefore, the simulator is enriched with power management and estimation tools that can be used for further energy-based applications such as path-planing and mission management. In the next section we give the details on the power model integration approach.

6 Energy Consumption Mode Integration into USV Simulation Environment

This section illustrates the approach used to integrate the power model into the simulation environment. The simulator is a ROS and Gazebo based software that models different virtual boats with realistic behaviours and environment disturbances. So, after establishing the theoretical model of a surface drone and identifying its parameters using the reverse engineering approach, the obtained model has to be verified. And since the simulator is an open source software, we propose to enrich it with power monitoring tools to calculate and plot the absorbed dynamic power instantaneously for a variety of scenarios. Since the simulator is a ROS-based package, it uses ROS nodes for navigation and control that communicate with each other through different message types such as odometry messages involving the boat's position and its velocity vectors to be published or subscribed to different topics. As given in the flowchart of Fig.9, The energy model integration of the USV is done by creating a process as a ROS node say: */power_calculator* that subscribes to */diffboat/state* topic to receive odometry information involving the actual USV's velocity components and position coordinates accord-

Table 2: Lutra-prop boat dynamic model parameters

Parameter	Numerical value
$d_{11} = -X_u$	$16.296Ns/m$
$X_{\dot{u}}$	$-0.050Kg$
$d_{22} = -Y_v$	$10.193Kgrad/s$
$Y_{\dot{v}}$	$-0.664Kg$
$d_{33} = -N_r$	$4.630Nms/rad$
$N_{\dot{r}}$	$-0.064Kg.m^2$
m_{11}	$9.750Kg$
m_{22}	$10.364Kg$
m_{33}	$1.158Kg.m^2$

Table 3: Wind force parameters

Parameter	Numerical value
ρ_a	$1.184Kg/m^3$
A_{Fw}	$0.08m^2$
A_{Lw}	$0.18m^2$
c_x	0.68
c_y	1.11

ing to the capabilities of Gazebo and its available sensors. The received information is used to calculate the USV's relative speed vector components and accelerations. In addition to the odometry information, the node repeatedly subscribes to the */gazebo/current* topic to read the water current status. The ROS node uses these data and by fixing the wind speed to calculate instantaneously the absorbed dynamic power by the USV during any given scenario based on Eq.(32). The interconnected processes of the simulator are represented in the directed graph of Fig.10. The given graph is named the RQT-graph [26] which is obtained using the *rqt_graph* command in the Linux terminal window. The vertices of the graph represent the active ROS nodes while its edges represent the published/subscribed topics. From the given graph, we see that the differential boat includes nine active ROS node involving the power nodes subscribing to the water current and boat's state topics to repeatedly receive the required information.

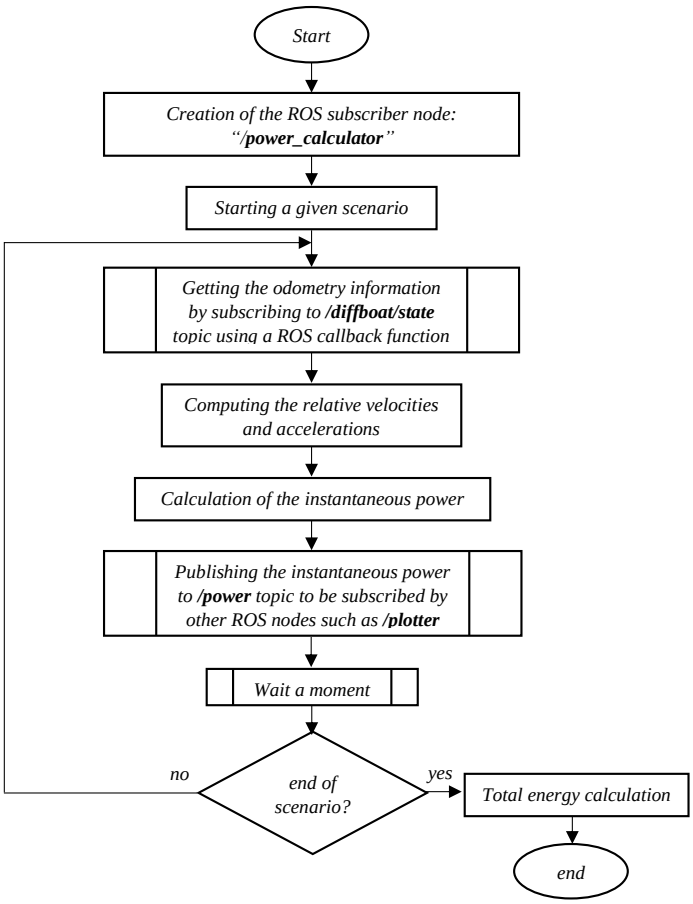


Figure 9: Flowchart of the power calculation process

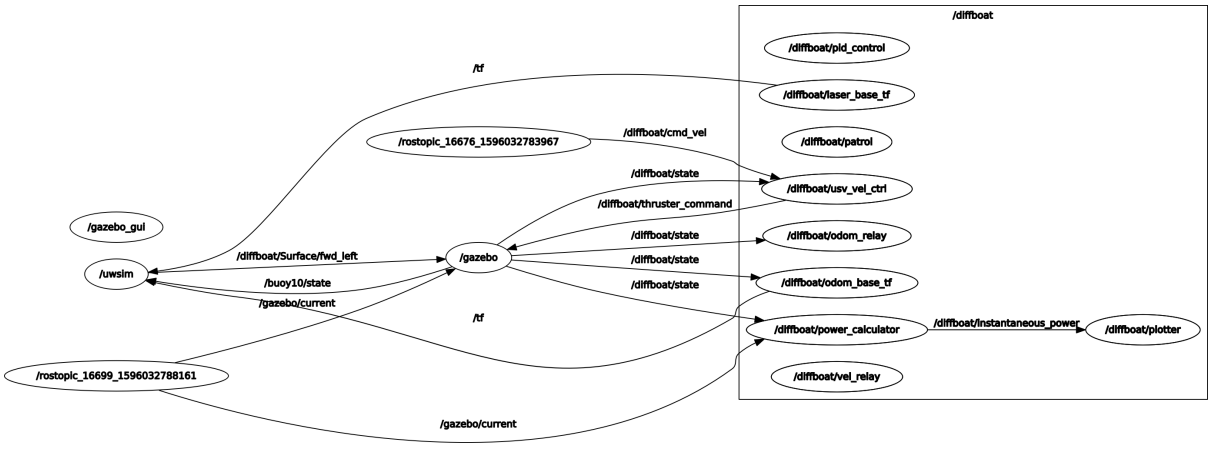


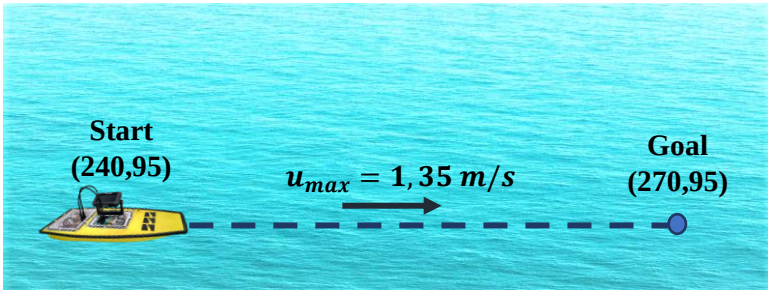
Figure 10: The RQT-graph showing the interconnected ROS processes

7 Simulation and Results

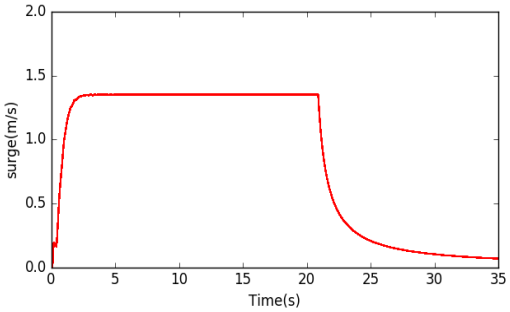
In order to show the effectiveness of the energy modelling, identification and integration into the simulation environment approaches, different scenarios are conducted with and without the presence of wind and water current disturbances. In this section we go through seven different and independent scenarios, and we record the surge velocity $u(t)$ and the calculated dynamic power absorbed by the USV instantaneously for each scenario by neglecting the static power. These scenarios have been chosen in such a way to be as realistic as possible to real situation that can be happen. For each scenario, the implemented ROS processes calculate and return instantaneously the absorbed power based on the power model given in Eq.(32) and the published odometry data as previously explained. We present at the end of this section the effect of the speed of the USV on its power consumption.

7.1 Scenario 1

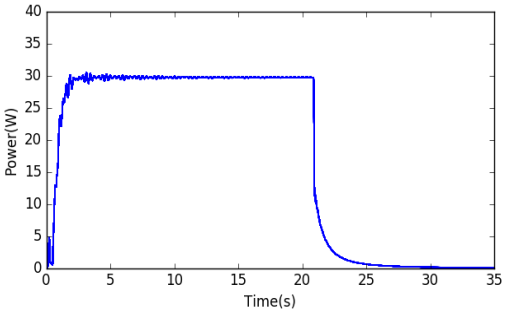
In this scenario, the boat is configured to move from a starting point $(240; 95)$ to a target point $(270; 965)$ represented in $\{e\}$ reference frame at a maximum speed ($u = 1.34m/s$) without any disturbances (see Fig.11-a). The surge velocity $u(t)$ and the calculated power are recorded and plotted as given in Fig.11-b and Fig.11-c. The total consumption returned by the implemented power processes at the end of the scenario using Eq.(25,32) is $607.87J$.



(a) Scenario 1



(b) Surge velocity $u(t)$

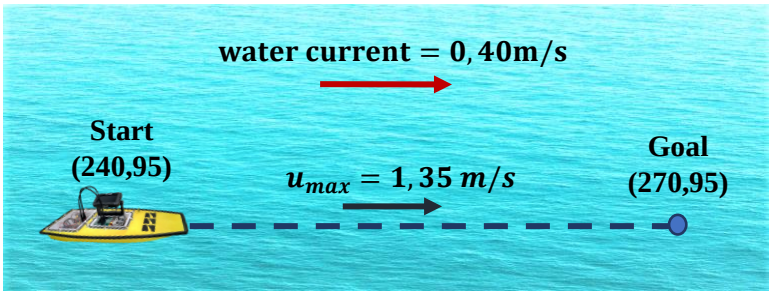


(c) Power consumption $p(t)$

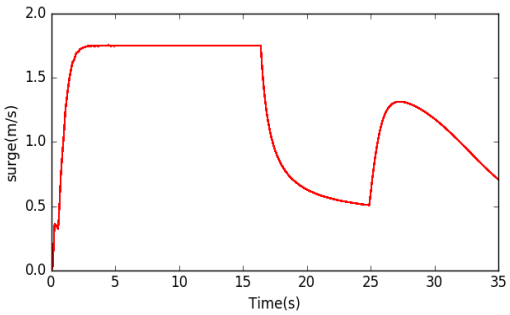
Figure 11: Representation of scenario 1

7.2 Scenario 2

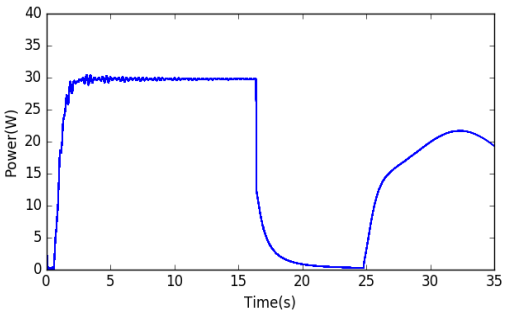
The second scenario is similar to the first one except that the water current is configured now to run in the same direction of the drone's motion at $0.40m/s$ (see Fig.12-a). The recorded surge velocity and power consumption variations are given in Fig.12-b and Fig.12-c. From this figure we see that the second scenario is faster than the first which took only 16.5 seconds to finish and consumed about $461.81J$. The drone's willingness to stay at the target point resisting the water current flux results in additional power consumption after the end of the scenario.



(a) Scenario 2



(b) Surge velocity $u(t)$



(c) Power consumption $p(t)$

Figure 12: Representation of scenario 2

7.3 Scenario 3

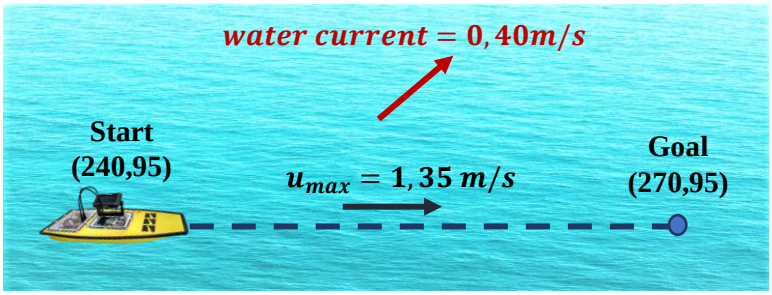
As given in Fig.13-a, the third scenario is configured such that the water flows in the direction of 45° at a speed of $0.40m/s$. The surge velocity of the boat and its power consumption are plotted and represented in Fig.13-b and Fig.13-c. We notice that this scenario consumes much power than the previous scenario due to the water current flow direction constraint. The total energy consumed at the end if this scenario is $573.78J$.

7.4 Scenario 4

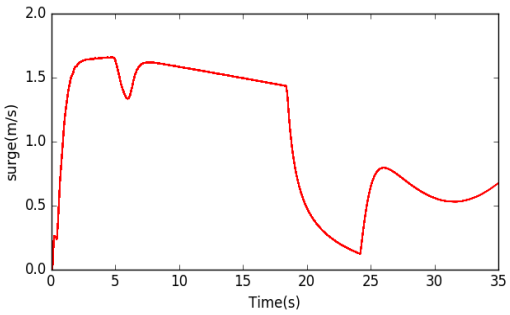
This scenario is similar to the third one except that the drone comes back from the target point to its original point (see Fig.14-a). The surge velocity component and power variation are represented in Fig.14-c. We notice that this scenario is longer than the previous one and consumes much power ($825.49J$). We notice also in Fig.14-b that the drone's velocity can not reach the maximum speed since it is moving against the water flow.

7.5 Scenario 5

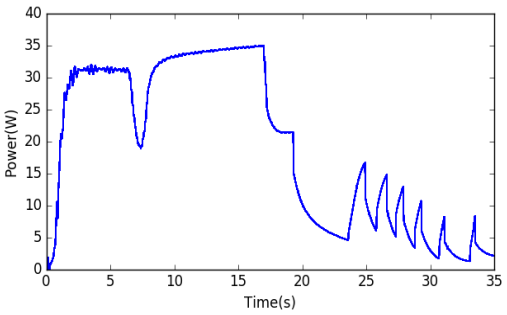
The boat is configured in this scenario to move from the starting point to the target point at a maximum speed of $1m/s$ (see Fig.15-a). The surge velocity and power consumption variations are plotted and represented in Fig.15-b and Fig.15-c. The total energy required to complete the scenario is returned by the power process which equals to $460.47J$. The overshoots shown in the two graphs are obtained due to default controller used within the USV simulator which demonstrate that the optimal controller design plays a significant impact on the power consumption; thus, our power simulator is a very useful tool to design and tune the controller.



(a) Scenario 3

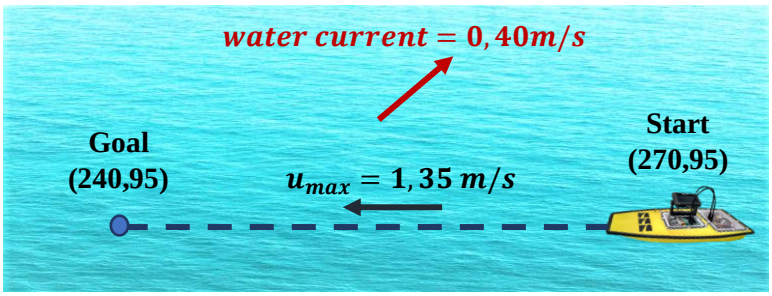


(b) Surge velocity $u(t)$

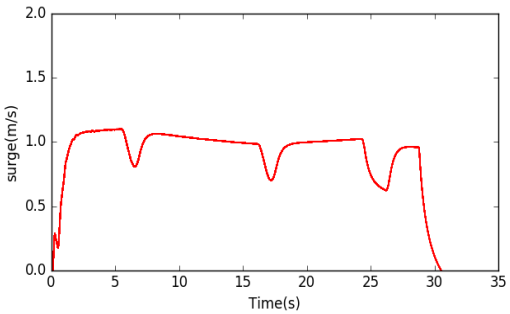


(c) Power consumption $p(t)$

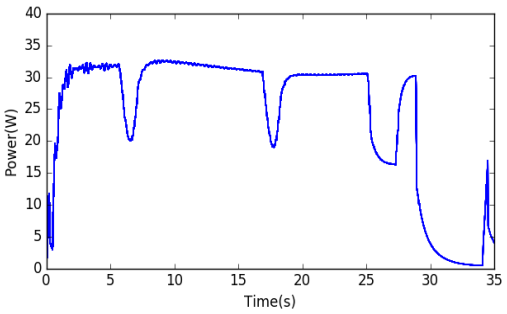
Figure 13: Representation of scenario 3



(a) Scenario 4



(b) Surge velocity $u(t)$

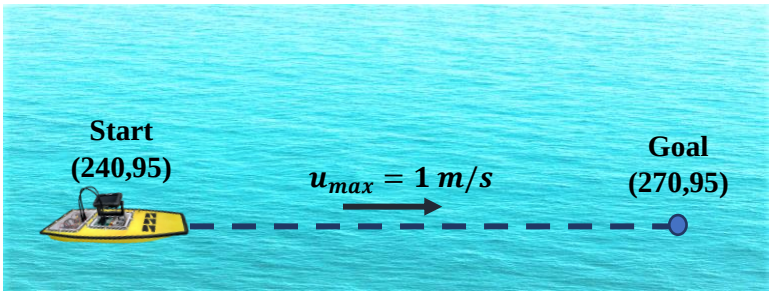


(c) Power consumption $p(t)$

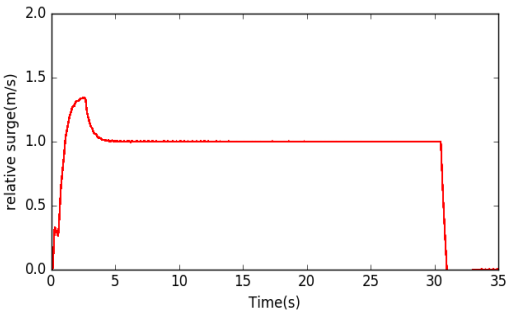
Figure 14: Representation of scenario 4

7.6 Scenario 6

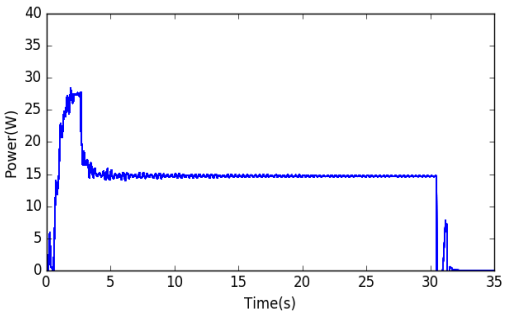
In this scenario, the vehicle is configured to move from the starting point to the target point at a maximum speed of $1m/s$ but with the presence of water current disturbance $(0.30m/s, 45°)$ and wind disturbance $(0.20m/s, 30°)$ as represented in Fig.16-a. The relative velocity given



(a) Scenario 5



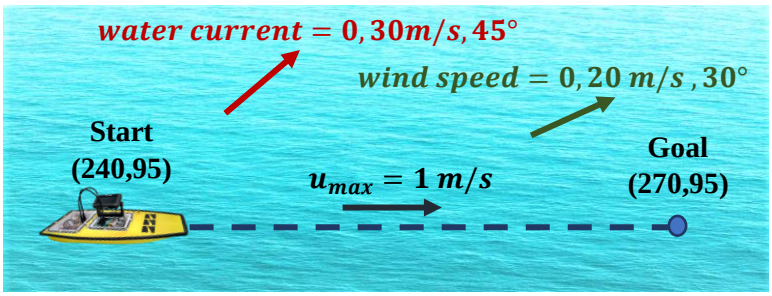
(b) Relative surge velocity $u(t)$



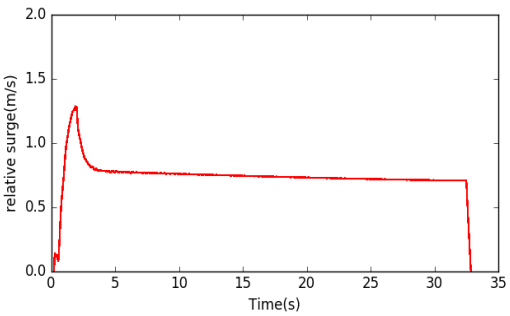
(c) Power consumption $p(t)$

Figure 15: Representation of scenario 5

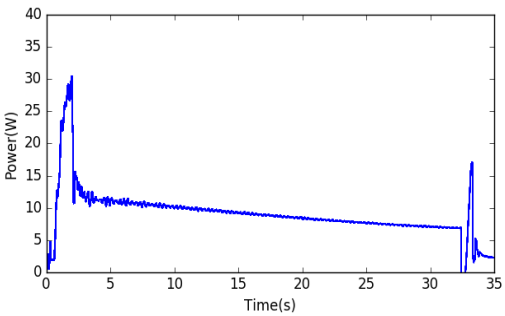
in Fig.16-b is smaller than the USV's speed because the disturbances are not obstructing the boat's motion. The total consumption of this scenarios given by the implemented power process is about $305.73J$.



(a) Scenario 6



(b) Relative surge velocity $u(t)$

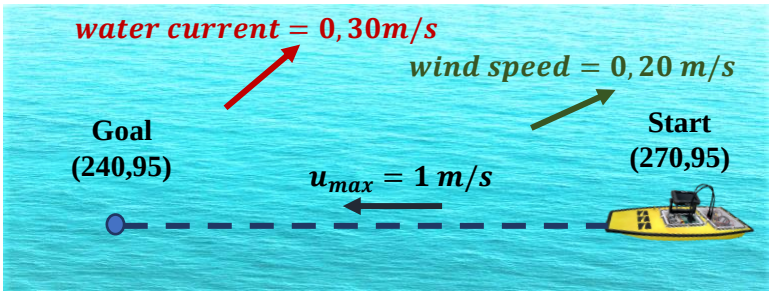


(c) Power consumption $p(t)$

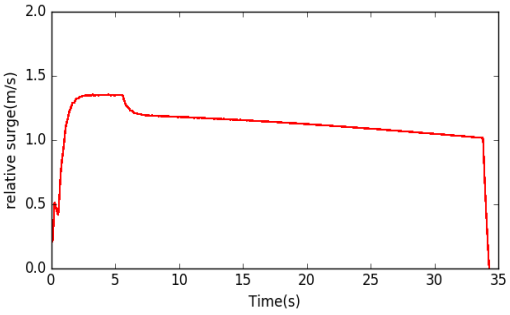
Figure 16: Representation of scenario 6

7.7 Scenario 7

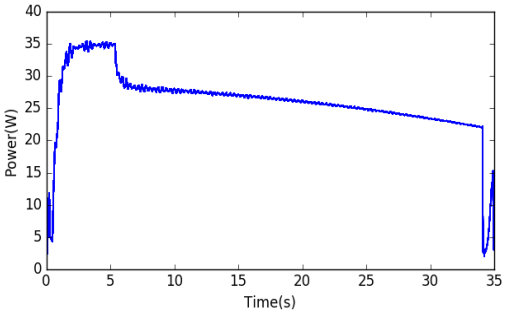
This scenario is similar to scenario 6 except that the drone should come back from the target point to the original point under the same disturbances (see Fig.17-a). From Fig.17-b and Fig.17-c, we notice that this scenario consumes much power than the previous one since the disturbances are obstructing the drone's motion; thus, the total energy required to complete this scenario is evaluated by the power process which equals to $898.44J$.



(a) Scenario 7



(b) Relative surge velocity $u(t)$



(c) Power consumption $p(t)$

Figure 17: Representation of scenario 5

The obtained duration and consumption values for each of the above described scenarios are summarized in Table 4. By comparing the first four scenarios, we deduce that the drone took different duration with different consumption due the disturbances variations. Furthermore, by comparing scenario 1 and scenario 5, we observe that the velocity variation of the surface drones has a significant effect of its consumption. This effect is well introduced in the next sub section.

Table 4: Results summary

Scenario number	Total time (s)	Total energy ($joules$)
1	21.0	607.87
2	16.5	441.81
3	18.5	573.78
4	29.0	825.49
5	30.5	460.47
6	32.5	305.73
7	33.5	898.44

7.8 The effect of the USV velocity on the energy consumption

This section shows the effect of the USV speed on its power consummation. The virtual USV was configured to complete different scenarios such that in each scenario, the USV is supposed to move from one point to another at a given linear speed by skipping wind and water current disturbances. After that, the steady state power consumption for each scenario is calculated by the implemented power processes and recorded in the graph of Fig.18 which illustrates the absorbed power versus the surge velocity variation given by the simulator. The given graph indicates that the power consumption varies exponentially with respect to the USV's speed in the case study speed interval. As already hypothesized in section 3.1, the non-linear drag coefficients should not be ignored if the surface drone speed exceeds a given value (mainly $> 1.5m/s$) and obviously this will lead to a more exponential power and energy variation.

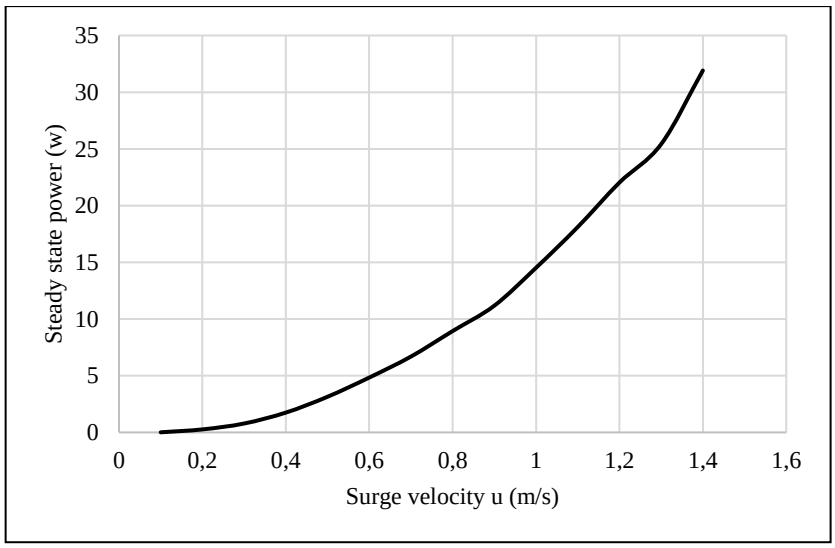


Figure 18: The effect of the linear velocity on the power consumption

Conclusion

The energy consumption model of the Unmanned Surface Vehicles was established first and presented as a function of their environments (speed, wind, water current etc.) based on the 3-DOF dynamic model of surface vessels. A reverse-engineering approach was applied on a recent and robust USV simulation environment to identify the dynamic model parameters of the Lutra-prop USV, hence, its complete energy consumption model is deduced. The wind dynamic parameters were obtained from the Gazebo's fluid dynamics plug-in integrated within the simulator package. Finally, the energy consumption model was well integrated into the simulation environment and verified through different scenarios and thus, the simulator is enriched with power management and estimation tools. The provided results show the variation of the USV consumption as environment conditions change and allow us also to evaluate the relationship between the USV's speed and its consumption. The presented work shows the importance of simulator-based power management and estimation which plays a significant role and help to solve many energy-based problems such as optimal path planning investigations; optimal tasks scheduling; autonomy problems; as well as designing an optimal energy-based control systems. The presented work motivates as to think about identifying much realistic consumption model by considering the wave effect and the non-linear drag impact to be represented in further papers. Our approach can be exploited with different unmanned vehicles type such as aerial drones

or under water drones by integrating the power model into their corresponding simulators given in [27, 28, 29].

References

[1] Marcelo Paravisi et al. "Unmanned surface vehicle simulator with realistic environmental disturbances". In: *Sensors* 19.5 (2019), p. 1068.

[2] Vitor AM Jorge et al. "A survey on unmanned surface vehicles for disaster robotics: Main challenges and directions". In: *Sensors* 19.3 (2019), p. 702.

[3] Thor I Fossen. *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[4] Gerardo Canfora, Massimiliano Di Penta, and Luigi Cerulo. "Achievements and challenges in software reverse engineering". In: *Communications of the ACM* 54.4 (2011), pp. 142–151.

[5] Mudasser Wahab, Fernando Rios-Gutierrez, and Adel El Shahat. *Energy modeling of differential drive robots*. IEEE, 2015.

[6] Linfei Hou, Liang Zhang, and Jongwon Kim. "Energy modeling and power measurement for mobile robots". In: *Energies* 12.1 (2019), p. 27.

[7] Linfei Hou, Liang Zhang, and Jongwon Kim. "Energy Modeling and Power Measurement for Three-Wheeled Omnidirectional Mobile Robots for Path Planning". In: *Electronics* 8.8 (2019), p. 843.

[8] Stephen L Canfield, Tristan W Hill, and Stephen G Zuccaro. "Prediction and experimental validation of power consumption of skid-steer mobile robots in manufacturing environments". In: *Journal of Intelligent & Robotic Systems* 94.3-4 (2019), pp. 825–839.

[9] Mauricio F Jaramillo-Morales et al. "Predictive power estimation for a differential drive mobile robot based on motor and robot dynamic models". In: *2019 Third IEEE International Conference on Robotic Computing (IRC)*. IEEE. 2019, pp. 301–307.

[10] Mauricio F Jaramillo-Morales et al. "Energy estimation for differential drive mobile robots on straight and rotational trajectories". In: *International Journal of Advanced Robotic Systems* 17.2 (2020), p. 1729881420909654.

[11] Liang Zhang, Jongwon Kim, and Jie Sun. "Energy Modeling and Experimental Validation of Four-Wheel Mecanum Mobile Robots for Energy-Optimal Motion Control". In: *Symmetry* 11.11 (2019), p. 1372.

[12] Chanmin Yoon et al. "Accurate power modeling of modern mobile application processors". In: *Journal of Systems Architecture* 81 (2017), pp. 17–31.

[13] Kenneth R Muske et al. "Identification of a control oriented nonlinear dynamic USV model". In: *2008 American Control Conference*. IEEE. 2008, pp. 562–567.

[14] Chunyue Li et al. "Modeling and experimental testing of an unmanned surface vehicle with rudderless double thrusters". In: *Sensors* 19.9 (2019), p. 2051.

[15] Hanlin Niu et al. "An energy-efficient path planning algorithm for unmanned surface vehicles". In: *Ocean Engineering* 161 (2018), pp. 308–321.

[16] Hanlin Niu et al. "Efficient path planning algorithms for unmanned surface vehicle". In: *IFAC-PapersOnLine* 49.23 (2016), pp. 121–126.

[17] Dongdong Mu et al. "Modeling and identification for vector propulsion of an unmanned surface vehicle: Three degrees of freedom model and response model". In: *Sensors* 18.6 (2018), p. 1889.

[18] Jiucai Jin, Jie Zhang, and Deqing Liu. “Design and verification of heading and velocity coupled nonlinear controller for unmanned surface vehicle”. In: *Sensors* 18.10 (2018), p. 3427.

[19] Christian R Sonnenburg and Craig A Woolsey. “Modeling, identification, and control of an unmanned surface vehicle”. In: *Journal of Field Robotics* 30.3 (2013), pp. 371–398.

[20] Stefan Wirtensohn et al. “Modelling and identification of a twin hull-based autonomous surface craft”. In: *2013 18th International Conference on Methods & Models in Automation & Robotics (MMAR)*. IEEE. 2013, pp. 121–126.

[21] James Kramer and Matthias Scheutz. “Development environments for autonomous mobile robots: A survey”. In: *Autonomous Robots* 22.2 (2007), pp. 101–132.

[22] Leon Žlajpah. “Simulation in robotics”. In: *Mathematics and Computers in Simulation* 79.4 (2008), pp. 879–897.

[23] Lenka Pitonakova et al. “Feature and performance comparison of the V-REP, Gazebo and ARGoS robot simulators”. In: *Annual Conference Towards Autonomous Robotic Systems*. Springer. 2018, pp. 357–368.

[24] Nathan Koenig and Andrew Howard. “Design and use paradigms for gazebo, an open-source multi-robot simulator”. In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*. Vol. 3. IEEE. 2004, pp. 2149–2154.

[25] Morgan Quigley et al. “ROS: an open-source Robot Operating System”. In: *ICRA workshop on open source software*. Vol. 3. 3.2. Kobe, Japan. 2009, p. 5.

[26] Carol Fairchild and Thomas L Harman. *ROS Robotics By Example: Learning to control wheeled, limbed, and flying robots using ROS Kinetic Kame*. Packt Publishing Ltd, 2017.

[27] Zandra B Rivera, Marco C De Simone, and Domenico Guida. “Unmanned ground vehicle modelling in Gazebo/ROS-based environments”. In: *Machines* 7.2 (2019), p. 42.

[28] Musa Morena Marcusso Manhães et al. “UUV simulator: A gazebo-based package for underwater intervention and multi-robot simulation”. In: *OCEANS 2016 MTS/IEEE Monterey*. IEEE. 2016, pp. 1–8.

[29] Mengmi Zhang et al. “A high fidelity simulator for a quadrotor uav using ros and gazebo”. In: *IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society*. IEEE. 2015, pp. 002846–002851.

[30] Marceloparavisi. *disaster-robotics-proalertas/usv-sim-lsa*. 2019. URL: `https://github.com/disaster-robotics-proalertas/usv_sim_lsa` (visited on 09/01/2019).