2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Attitude Control of the Hydrobatic Intervention AUV Cuttlefish using Incremental Nonlinear Dynamic Inversion

Tom Slawik*, Shubham Vyas*, Leif Christensen*, Frank Kirchner*

***Abstract*— In this paper, we present an attitude control scheme for an autonomous underwater vehicle (AUV), which is based on incremental nonlinear dynamic inversion (INDI). Conventional model-based controllers depend on an exact model of the controlled system, which is difficult to find, especially for marine vehicles subject to highly nonlinear hydrodynamic effects. INDI trades off model accuracy with sensor accuracy by incorporating acceleration feedback and actuator output feedback to linearize a nonlinear system incrementally. Existing research primarily focuses on studying INDI on unmanned aerial vehicles. However, there is barely any research on controlling marine vehicles using INDI. The control task we are performing is a 90 degrees pitch-up maneuver, where the dual-arm intervention AUV *Cuttlefish* transitions from a horizontal traveling pose to a vertical intervention pose. We compare INDI to a classical model-based control scheme in the maritime test basin at DFKI RIC, Germany, and we find that INDI keeps the AUV much more steady both in the transitioning phase as well as in the station keeping phase.**

I. INTRODUCTION

With the expansion of the blue economy, there has been a notable increase in subsea infrastructures. This includes a rise in aquaculture installations and offshore wind farms, along with their corresponding foundation structures, as documented by GWEC [1]. This is leading to a growing need for automated subsea operations not only within emerging markets but also within the traditional oil and gas industry. Human diving operations face many dangers and depend on good weather and have restricted depths [2]. Remotely Operated Vehicles (ROVs) are currently used for deeper dives, but using them requires large support vessels with special equipment, making ROV operations time-consuming and expensive. The new class of so-called *intervention* AUVs (I-AUVs) [3] aims to facilitate autonomous interaction with the subsea infrastructures mentioned above. The AUV *Cuttlefish*, shown in Figure 1 and described in detail in [4], is an I-AUV equipped with two arms and the capability for hydrobatic motions (see [5] on the term *hydrobatic*). Using its eight thrusters, it can take on arbitrary orientations in the water column to interact with features of an offshore structure that are otherwise difficult to reach. The vehicle can change its center of mass and center of buoyancy to achieve the desired stability for the current task, such as manipulation or transitioning into a new pose. The attitude control of autonomous underwater vehicles is crucial for their overall maneuverability and stability [6]. In this paper, we focus on the attitude control of the hydrobatic AUV *Cuttlefish* using incremental nonlinear dynamic inversion (INDI) [7]. This advanced control method allows for precise and agile maneuvering of the AUV, making it suitable for a wide range of underwater intervention tasks.

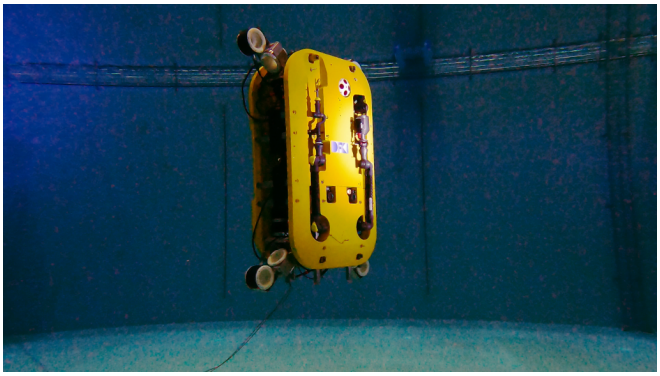

Fig. 1: AUV *Cuttlefish* in intervention pose in the test basin of DFKI RIC, Bremen, Germany.

Due to *Cuttlefish*'s ability to assume an arbitrary pose, particulary an intervention pose with a pitch angle of 90 degrees, we do not rely on Euler angles for attitude control due to the Gimbal-Lock problem [8]. Instead, we use quaternions for parameterizing the orientation. During the transition to an upright pose, the AUV leaves its stable orientation, resulting in nonlinear restoring forces which must be compensated. Since the early 1990s, there has been a significant volume of academic research in literature focused on model-based control for underwater vehicles, the main results are summarized in [9] and [10]. In [11], a quaternion-based adaptive controller for a 6-DOF AUV is proposed, where the restoring forces are identified and canceled during operation. The resulting controller is evaluated in real world experiments on the AUV ODIN. Another work has been done by [12], where the center of buoyancy is identified using an Extended Kalman Filter (EKF) based on the Intervention-AUV SAUVIM. Besides hydrostatic restoring forces, there are hydrodynamic effects acting on the vehicle body during motion. Modeling these effects is difficult, since the shape of *Cuttlefish* is highly complex. Hydrodynamic effects which cannot be modeled also result in model uncertainty, which harms the control performance. An alternative to model-based control is sensor-based control, where model accuracy is traded off by sensor

An open-source implementation of the presented controller is available at https://github.com/dfki-ric-underactuated-lab/auv_control_indi. The accompanying video can be accessed at https://youtu.be/8u8k607lpn4

The work described in this paper has received funding by the German Federal Ministry of Education and Research (grant no. 01IW22003) as well as the Federal Ministry of Economic Affairs and Climate Action (grant no. 03SX540D)

*All authors are affiliated with the German Research Center for Artificial Intelligence (DFKI GmbH), Robotics Innovation Center, Robert-Hooke-Straße 1, 28359 Bremen, Germany. (e-mail: tom.slawik@dfki.de)

accuracy. Incremental nonlinear dynamics inversion (INDI) is a sensor-based control scheme that iteratively linearizes a nonlinear system based on acceleration and input feedback. INDI has its origins in the unmanned aerial vehicle community [7], [13], [14]. However, there is barely any research on controlling marine vehicles using INDI. To date, the authors know about only one single publication [15], where the authors evaluate INDI to control an Unmanned Aerial-Underwater Vehicle, which is a quadrotor capable of diving. In this work, we present the first study of using INDI for motion control of a 6-DOF underwater vehicle. In Section II we will derive the mathematics behind INDI based on an underwater Fossen model [9]. Additionally, we will define the control laws for the model-based baseline controller, and the attitude and altitude control loops. Subsequently, in Section III we conduct real-world experiments of a control task, where the AUV *Cuttlefish* performs a 90-degrees pitch-up maneuver. Additionally, we perform a 6-DOF model identification of the AUV to compare INDI to a model-based controller. Our results show that INDI outperforms the model-based controller, especially in keeping the pose steady with minimum drift, which is crucial for intervention tasks.

II. IMPLEMENTATION

A. Vehicle Modeling

For modeling the AUV *Cuttlefish*, we use a 6-DOF Fossen model [9], where $\vec{\nu} = [u, v, w, p, q, r]^\top \in \mathbb{R}^6$ is the body-fixed velocity vector, containing three linear velocities: surge, sway, and heave, and three angular velocities: roll, pitch, and yaw. $\vec{\eta} = [x, y, z, q_x, q_y, q_z, q_w]^\top \in \mathbb{R}^7$ is the pose vector, where the orientation is encoded as a quaternion $\vec{q}$. The pose and velocity vectors are coupled through a nonlinear relationship:

$$\dot{\vec{\eta}} = \mathrm{J}(\vec{\eta})\vec{\nu} \ , \tag{1}$$

where $\mathrm{J}(\vec{\eta})$ is a Jacobian, also defined in [9]. The Fossen equations are given by:

$$\mathrm{M}\dot{\vec{\nu}} + \mathrm{C}(\vec{\nu})\vec{\nu} + \vec{d}(\vec{\nu}) + \vec{g}(\vec{\eta}) = \vec{\tau} \ , \tag{2}$$

where $\dot{\vec{\nu}} \in \mathbb{R}^6$ is the acceleration vector, $\mathrm{M} \in \mathbb{R}^{6\times6}$ is the mass-inertia matrix, and $\vec{\tau} \in \mathbb{R}^6$ is the input wrench. $\mathrm{C}(\vec{\nu}) \in \mathbb{R}^{6\times6}$ is the coriolis-centripetal matrix, which is derived from the mass matrix M. The vector-field $\vec{d}(\vec{\nu})$ models the hydrodynamic drag. Due to its favorable trade-off between accuracy and robustness, we employ a quadratic drag model with $\mathrm{D}_{\mathrm{lin}}, \mathrm{D}_{\mathrm{quad}} \in \mathbb{R}^{6\times6}$ as parameters:

$$\vec{d}(\vec{\nu}) = \mathrm{D}_{\mathrm{lin}}\nu + \mathrm{D}_{\mathrm{quad}}|\nu|\nu \in \mathbb{R}^6 \ . \tag{3}$$

The function $\vec{g}(\vec{\eta}) \in \mathbb{R}^6$ models the hydrostatic forces which are caused by gravity and buoyancy, and is defined as:

$$\vec{g}(\vec{\eta}) = \Phi_g(\vec{\eta})\vec{\theta}_g \ , \tag{4}$$

where

$$\Phi_g(\vec{\eta}) = \begin{pmatrix} \mathrm{R}(\vec{\eta})\vec{z} & 0_{3\times3} \\ \vec{0}_{3\times1} & \mathrm{S}(\mathrm{R}(\vec{\eta})\vec{z}) \end{pmatrix} \tag{5}$$

is a (6×4) matrix further described in [10]. Here, $z = [0, 0, -1]^\top$ is the axis of the gravitational field, and $\vec{\theta}_g \in \mathbb{R}^4$ is a parameter vector consisting of the net buoyancy force and three parameters for the restoring torques. $\mathrm{R}(\vec{\eta})$ is the rotation matrix from world frame to body frame, and $S(\cdot)$ is the skew-symmetric cross-product operator.

The actuation $\vec{\tau}$ is modeled as a resulting wrench caused by the thrusters. It is calculated using the thruster allocation matrix $\mathrm{T} \in \mathbb{R}^{6\times8}$ and the thruster RPM vector $u \in \mathbb{R}^8$:

$$\vec{\tau}(\vec{u}) = \mathrm{T}f_{\mathrm{thr}}(\vec{u}) \ . \tag{6}$$

The relationship between the rotational speed u_i and its corresponding propulsive force is modeled as a quadratic curve $f_{\mathrm{thr}}(u_i) = \kappa u_i|u_i|$. The coefficient κ was measured beforehand. For this actuation model, we ignore the moment of inertia caused by each individual thruster to simplify calculations.

B. Model Identification (Least-Squares)

We can express the dynamic equations given by (2) as a matrix-vector product:

$$\vec{\tau}_\theta(\vec{\nu}, \vec{\eta}) = \Phi(\vec{\nu}, \vec{\eta})\vec{\theta} \ , \tag{7}$$

where

$$\Phi(\vec{\nu}, \vec{\eta}) = \left[\dot{\vec{\nu}}, \vec{\nu}, \vec{\nu}|\vec{\nu}|, \cdots, \Phi_g(\vec{\eta})\right] \in \mathbb{R}^{6\times p} \ . \tag{8}$$

$\vec{\theta}$ is a parameter vector containing 36 mass-matrix entries, 72 damping matrix entries, and 4 hydrostatic parameters:

$$\vec{\theta} = \left[X_{\dot{u}}, X_{\dot{v}}, \cdots, X_u, X_{|u|u}, \cdots, Y_{\dot{u}}, \cdots\right]^\top \in \mathbb{R}^p \ . \tag{9}$$

In order to identify the parameter vector $\vec{\theta}$, we collect $n+1$ samples from experiments and construct the regression matrix using (8):

$$\mathrm{A} = \begin{bmatrix} \dot{\vec{\nu}}_0, \vec{\nu}_0, \vec{\nu}_0|\vec{\nu}_0|, \cdots, \Phi_g(\vec{\eta}_0) \\ \vdots \\ \dot{\vec{\nu}}_n, \vec{\nu}_n, \vec{\nu}_n|\vec{\nu}_n|, \cdots, \Phi_g(\vec{\eta}_n) \end{bmatrix} \in \mathbb{R}^{6n\times p} \ , \tag{10}$$

and the output vector

$$\vec{y} = [\vec{\tau}_0, \vec{\tau}_1, \cdots, \vec{\tau}_n]^\top \in \mathbb{R}^{6n} \ . \tag{11}$$

We want to minimize the prediction error using the sum of squared errors $J(\vec{\theta}) = \|\vec{y} - \vec{\tau}_\theta(\vec{\nu}, \vec{\eta})\|_2$. A solution is given by the Moore-Penrose pseudoinverse [16]:

$$\vec{\theta} = (\mathrm{A}^\top\mathrm{A})^{-1}\mathrm{A}^\top\vec{y} \ . \tag{12}$$

C. Feedback Linearization (FBL)

State feedback linearization (also known as „Nonlinear Dynamic Inversion“ (NDI) in aerospace control systems) is a technique to transform a nonlinear system into a linear system through a feedback loop [17]. Recall the dynamics equation presented in (2). We rewrite this equation as follows:

$$\mathrm{M}\dot{\vec{\nu}} + \vec{f}(\vec{\nu}, \vec{\eta}) = \vec{\tau} \ , \tag{13}$$

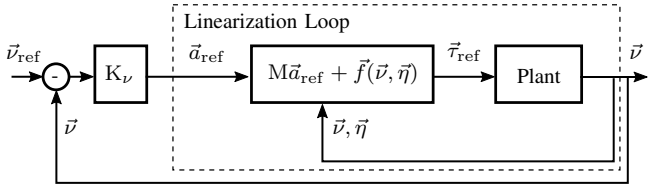

Fig. 2: Block diagram of the FBL-based velocity control loop

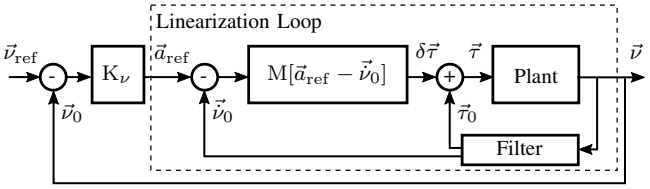

Fig. 3: Block diagram of the INDI-based velocity controller

where $\vec{f}(\vec{\nu},\vec{\eta})$ is a smooth vector field describing the nonlinear system dynamics:

$$\vec{f}(\vec{\nu},\vec{\eta}) = \mathrm{C}(\vec{\nu})\vec{\nu} + \vec{d}(\vec{\nu}) + \vec{g}(\vec{\eta}) \ . \tag{14}$$

Now, we introduce a new control variable $\dot{\vec{\nu}} = \vec{a}_{\mathrm{ref}}$, which acts as the input vector for the linearized system:

$$\vec{\tau}_{\mathrm{ref}} = \mathrm{M}\vec{a}_{\mathrm{ref}} + \vec{f}(\vec{\nu},\vec{\eta}) \ , \tag{15}$$

where $\vec{\tau}_{\mathrm{ref}}$ is the commanded wrench. Feedback linearization not only cancels the modeled nonlinearities, it also decouples each input $\vec{a}_{\mathrm{ref,i}}$ to only affect its corresponding output $\vec{\nu}_i$. However, the approach follows the assumptions that a) an exact model of the system dynamics is given, and b), that the state vector is exactly known at each time step.

A block diagram showing feedback linearization in conjunction with the velocity controller is shown in Figure 2.

D. Incremental Nonlinear Dynamic Inversion (INDI)

Incremental Nonlinear Dynamic Inversion (INDI) is an extension of Nonlinear Dynamic Inversion (NDI) / Feedback Linearization. The idea is to incrementally update the control command $\vec{\tau}_{\mathrm{ref}}$ in order to linearize the system. At each time step, the increment is calculated from accelerometer readings and measurements of the actual thruster rates. This is opposed to NDI, where the control command is calculated directly from the model.

To derive the control law for INDI, we first rearrange the Fossen equation (13) to an implicit form:

$$\dot{\vec{\nu}} = \mathrm{M}^{-1}\left[\vec{\tau} - \vec{f}(\vec{\nu},\vec{\eta})\right] \ . \tag{16}$$

Now, we develop the first-order Taylor expansion of the previous equation, where $\vec{\nu}_0, \vec{\eta}_0$ and $\vec{\tau}_0$ correspond to the measured state and input vectors:

$$\begin{aligned}\dot{\vec{\nu}} \approx\ & \mathrm{M}^{-1}\left[\vec{\tau}_0 - \vec{f}(\vec{\nu}_0,\vec{\eta}_0)\right] \\ & + \frac{\partial}{\partial\vec{\tau}}\left[\mathrm{M}^{-1}\vec{\tau}\right]_{\vec{\tau}=\vec{\tau}_0}(\vec{\tau}-\vec{\tau}_0) \\ & + \frac{\partial}{\partial\vec{\nu}}\left[\mathrm{M}^{-1}\vec{f}(\vec{\nu},\vec{\eta})\right]_{\vec{\nu}=\vec{\nu}_0}(\vec{\nu}-\vec{\nu}_0) \\ & + \frac{\partial}{\partial\vec{\eta}}\left[\mathrm{M}^{-1}\vec{f}(\vec{\nu},\vec{\eta})\right]_{\vec{\eta}=\vec{\eta}_0}(\vec{\eta}-\vec{\eta}_0) \ .\end{aligned} \tag{17}$$

We make the assumption that the thruster dynamics are much faster than the dynamics of the AUV body:

$$|\vec{f}(\vec{\nu},\vec{\eta}) - \vec{f}(\vec{\nu}_0,\vec{\eta}_0)| << |\vec{\tau} - \vec{\tau}_0| \ . \tag{18}$$

With this assumption, we simplify the dynamics to:

$$\begin{aligned}\dot{\vec{\nu}} \approx\ & \mathrm{M}^{-1}\left[\vec{\tau}_0 - \vec{f}(\vec{\nu}_0,\vec{\eta}_0)\right] \\ & + \mathrm{M}^{-1}\left(\vec{\tau} - \vec{\tau}_0\right) \ .\end{aligned} \tag{19}$$

Since $\dot{\vec{\nu}}_0 = \mathrm{M}^{-1}[\vec{\tau}_0 - \vec{f}(\vec{\nu}_0,\vec{\eta}_0)]$, we replace the dynamic terms of the previous equation with the measured acceleration $\dot{\vec{\nu}}_0$:

$$\dot{\vec{\nu}} \approx \dot{\vec{\nu}}_0 + \mathrm{M}^{-1}\left(\vec{\tau} - \vec{\tau}_0\right) \ . \tag{20}$$

Inverting this equation leads to

$$\delta\tau = \vec{\tau} - \vec{\tau}_0 = \mathrm{M}\left[\dot{\vec{\nu}} - \dot{\vec{\nu}}_0\right] \ , \tag{21}$$

which is the increment of the control wrench for one time step. If we introduce $\dot{\vec{\nu}} = \vec{a}_{\mathrm{ref}}$ to be our new control variable (as in feedback linearization), we finally get the incremental control law:

$$\vec{\tau}_{\mathrm{ref}} = \mathrm{M}\left[\vec{a}_{\mathrm{ref}} - \dot{\vec{\nu}}_0\right] + \vec{\tau}_0 \ . \tag{22}$$

Note that $\vec{\tau}_0$ is the actual input wrench, which is obtained by the actual thruster's rpm readings and the actuation model described in (6). As the angular acceleration $\dot{\vec{\omega}} \in \mathbb{R}^3$ cannot be measured directly, we estimate it using finite differences: $\dot{\vec{\omega}} \approx (\vec{\omega}_{t+1} - \vec{\omega}_t)/dt$, where $dt = 0.01\,\mathrm{s}$ is the sampling interval. To reduce noise, we apply a first-order Butterworth low-pass filter with a cutoff frequency of $4.0\,\mathrm{Hz}$ to the angular velocity readings before differentiation. The same low-pass filter is also applied to the thruster rate readings $\vec{u}_0$ and the linear acceleration, since the phase must be synchronized in order to avoid oscillations. An overview of the INDI linearization loop is shown in Figure 3.

E. Velocity Controller (Inner Loop)

The linearized system can now be controlled using linear control theory. For controlling the linear and angular velocities we use a proportional control law:

$$\vec{a}_{\mathrm{ref}} = \mathrm{K}_\nu\left(\vec{\nu}_{\mathrm{ref}} - \vec{\nu}\right) \ , \tag{23}$$

where $\vec{\nu}_{\mathrm{ref}} \in \mathbb{R}^6$ is the desired velocity vector. To find the optimal gain parameters, we use trial-and-error to find a diagonal gain matrix which is optimized such that the step response of the maximum velocity control commands does not exceed the force/torque limits of $300\,\mathrm{N}$ / $300\,\mathrm{Nm}$ (which are given due to internal safety and power regulations), while maintaining an optimal tracking performance:

$$\mathrm{K}_\nu = \mathrm{diag}(0.3, 0.3, 0.3, 1.0, 0.7, 0.7) \ . \tag{24}$$

F. Attitude Controller (Outer Loop)

Euler angles are not suitable for describing the orientation of an AUV that is designed for variable attitude maneuvers, since certain configurations cause a loss of one degree of freedom. This problem is also called the „Gimbal-Lock“ problem [8]. For this reason, we use quaternions to describe the orientation of the AUV. To control the orientation, we implement an attitude controller which is derived from the work of Chaturvedi et al. [8]:

$$\omega_{\text{ref}}(\mathrm{R}, \mathrm{R}_d) = \mathrm{K}_\Omega \sum_{i=1}^{3} e_i \times (\mathrm{R}_d^\top \mathrm{R}^\top e_i) \ . \tag{25}$$

The attitude controller generates an angular velocity command $\omega_{\text{ref}} \in \mathbb{R}^3$ based on rotation matrices $\mathrm{R} \in \mathbb{R}^{3\times3}$ (current orientation), and $\mathrm{R}_d \in \mathbb{R}^{3\times3}$ (desired orientation). Both matrices can be directly calculated from the current and desired quaternions. $[e_1\ e_2\ e_3]$ is the identity matrix. K_Ω is a proportional gain matrix, whose elements are chosen during real-world experiments such that there is minimum overshoot during a pitch-up maneuver:

$$\mathrm{K}_\Omega = \mathrm{diag}(0.15, 0.15, 0.15) \ . \tag{26}$$

To avoid too large angular velocities, $\vec{\omega}_{\text{ref}}$ is bounded to $\|\vec{\omega}_{\text{ref}}\|_{\max} < 0.2\,\mathrm{rad/s}$.

G. Depth Controller

In order to keep the depth of the AUV steady, a proportional-gain depth controller is implemented:

$$\vec{v}_{\text{ref}} = \mathrm{K}_{\text{pos}} \mathrm{R}_B^I \left[[0, 0, z_{\text{ref}}]^\top - [0, 0, z]^\top \right] \ , \tag{27}$$

where z_{ref} is the commanded depth, and z is the actual, measured depth. $\vec{v}_{\text{ref}} \in \mathbb{R}^3$ is the commanded linear velocity, and R_B^I is a rotation matrix from world frame to body frame. K_Ω is a proportional gain matrix, whose elements are chosen during real-world experiments to result in minimum overshoot:

$$\mathrm{K}_{\text{pos}} = \mathrm{diag}(0.5, 0.5, 0.5) \ . \tag{28}$$

The maximum linear velocity command $\vec{v}_{\text{ref}}$ is bounded to $\pm[0.5\,\mathrm{m/s}, 0.5\,\mathrm{m/s}, 0.2\,\mathrm{m/s}]$.

III. EVALUATION

A. Experimental Setup

TABLE I: *Cuttlefish* Specifications [4]

Specification	Value
Length / Width / Height	2.8 m / 2.0 m / 0.8 m
Weight	1200 kg
Inertial Navigation System	iXblue Phins Compact C3
DVL	Rowe SeaPilot
Pressure Sensor	Keller PAA-33x
Rated Driving Power	21.6 kW (8x 2.7 kW)

The experiments are conducted in the maritime test basin of DFKI RIC, Bremen, Germany [1]. The target vehicle is the dual-arm intervention AUV *Cuttlefish* [4], which is based on a steel frame construction equipped with 13 pressure housings for electronics and PVC-based foam, to make the vehicle positive buoyant. The two robotic arms are not used for our experiments and are fixed to the body. To control the 1200 kg heavy vehicle, eight 2.7 kW rim-driven thrusters are installed (four vertical ones and four horizontal ones). The thrusters were developed by Wittenstein cyber motor GmbH specifically for *Cuttlefish*, and are designed to be precisely controllable, even at very low RPM. The measured time constant of the thruster acceleration is $T_{\text{thr}} = 0.672\,s$. For estimating the attitude, velocities and linear accelerations, we use the iXblue Phins Compact C3 inertial navigation system. The Phins C3 is based on a fiber-optic gyroscope, offering high-precision rotation rate measurements with a resolution of 0.001 deg/s. Additionally, the Phins C3 is capable of estimating the orientation in roll and pitch with an accuracy of 0.05 deg RMS. The accuracy of the linear velocity estimate is 0.1 kn = 0.0514 m/s. The Phins C3 uses its own Kalman filter implementation to fuse data with various external sensors, in our setup a Rowe SeaPilot DVL to measure the speed over ground, and the Keller PAA-33x pressure sensor. A summary of the basic specifications is given in Table I. *Cuttlefish* follows a centralized computer architecture, where all software modules are running on a single x86-based computer, equipped with an Intel(R) Core(TM) i7-8700 CPU and 32 GB of RAM. The software stack is composed of Ubuntu 18.04 as the host system, running a ROS Noetic environment in a Docker container. Our control stack is split into two ROS-nodes, where one ROS-node is responsible for high-level tracking of depth and attitude, while the other ROS-node embeds the low-level FBL- and INDI - based velocity controllers. The update rate of the control stack amounts to 100 Hz.

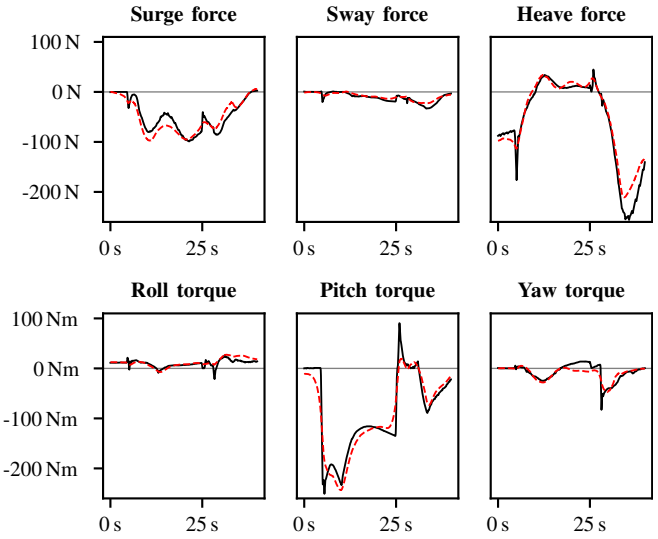

Fig. 4: Model identification test (dashed line: prediction)

[1] https://robotik.dfki-bremen.de/en/research/research-facilities-labs/maritime-infrastructure

TABLE II: Model Identification RMSE

DOF	Linear Model	Quadratic Model
Surge	11.31 N	9.38 N
Sway	6.62 N	5.01 N
Heave	16.92 N	18.94 N
Roll	6.39 Nm	6.86 Nm
Pitch	17.91 Nm	15.83 Nm
Yaw	13.74 Nm	13.40 Nm

B. Model Identification

In order to identify the motion model, we conducted a series of test drives where we recorded 57 minutes of training data in total. During the first test drive, we manually carried out random motions with the AUV, incorporating coupled maneuvers in different axes and orientations. We then identified a preliminary motion model, which was used in a second test drive. During the second test drive we did some pitch-up maneuvers using the FBL-based controller, in order to capture some training data representing the actual task we are aiming for. After collecting all training data, we identified the final motion model and did a test drive to collect a two-minute dataset solely used for testing the model identification performance, where we did three successive pitch-up maneuvers. The root mean square error of the predicted wrench is shown in Table II for a linear drag model and for a quadratic drag model. Figure 4 shows one full period of the test sequence, where the vehicle transitions into a upright pose and back into a horizontal pose. The dashed curve represents the predicted force/torque using the quadratic damping model, while the solid curve represents the actual force/torque. It can be seen that the model captures the coarse motion of the AUV accurately. However, for example the yaw prediction has an inaccuracy at 20s, and the peak in the heave DOF is not correctly predicted at 10s.

C. Experimental Results

During our real-world experiments we compare the INDI-based controller to a linear-drag and a quadratic-drag feedback-linearized controller.

1) Pitch-up maneuver: In our first experiment we evaluate the transition from horizontal into upright pose by measuring the step response from $0°$ pitch to $-90°$ pitch for 30 seconds. Subsequently, we perform a transition back to the stable horizontal pose, again for 30 seconds. The whole process is repeated for three cycles. Figure 5 shows the resulting plots. The first row shows the pitch velocity for each controller, and the second row shows the actual pitch torques, as calculated from the thruster outputs. The third row shows the actual pitch angles, while the desired pitch angle is outlined by a dashed curve. Looking at the plots we notice that the performance of all three controllers perform similarly, with a slight overshoot at the upright phase and a larger overshoot at the stabilization phase. The mean overshoot during the pitch-up maneuver is $1.36°$ for INDI, $1.33°$ for linear FBL, and $1.59°$ for quadratic FBL. During stabilization, the mean overshoot is $1.71°$ for INDI, $2.63°$ for linear FBL, and $2.10°$ for quadratic FBL. The mean energy consumption

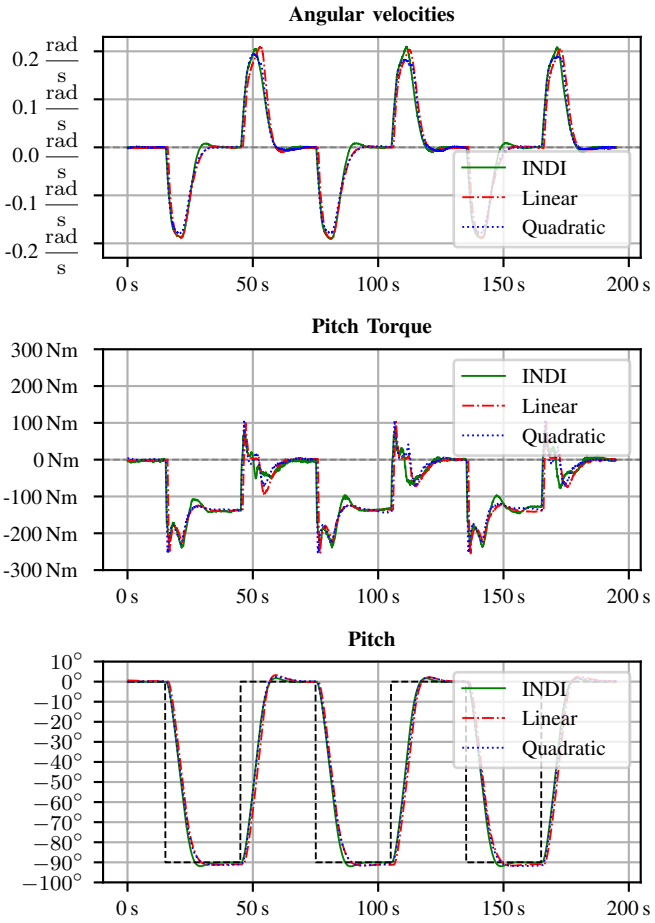

Fig. 5: Three pitch-up maneuvers

of INDI is 33.24 Wh per cycle, for linear FBL it is 33.84 Wh, and for quadratic FBL it is 34.19 Wh. We notice that both feedback-linearized controllers result in a steady-state orientation error in upright pose. To study the steady-state error in more detail, we maintain the upright pose for 300 seconds and then measure the error. For INDI, we measure an error of $0.0829°$. For the feedback-linearized controllers we measure an error of $1.3459°$ for the linear drag model, and $1.6897°$ for the quadratic drag model. The reason INDI has a smaller steady-state error is that it has an incremental control-law, which is eliminating the nonlinear effects successively. Table III shows the RMS tracking errors for the linear and angular velocities. According to the data, INDI keeps the roll axis more stable than the model-based controllers, which indicates that INDI captures the coupled dynamics better. A surprising result is that the tracking performance of the quadratic drag model is worse than that of the linear drag model. One possible explanation could be that the velocities of our system are considerably low, which makes the quadratic drag terms hard to identify correctly.

2) Station Keeping: To evaluate the station keeping performance, we measure the positional drift while maintaining an upright pose for 300 seconds. The drift is plotted in Figure 6 for INDI and FBL. It can clearly be noticed that the feedback-linearized controllers have much higher drift. One

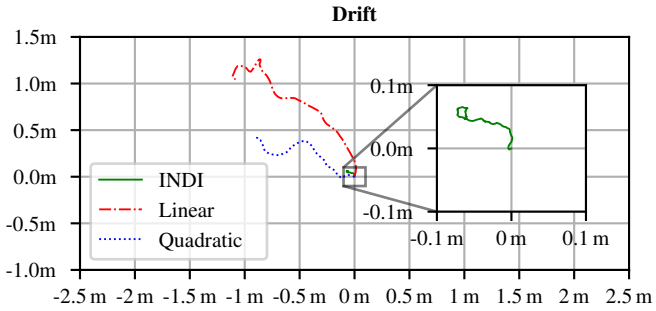

Fig. 6: Drift after 300 s in x and y world coordinates

TABLE III: Tracking Error (RMSE)

	INDI	Linear FBL	Quadratic FBL
Surge	0.39 mm/s	0.55 mm/s	1.04 mm/s
Sway	0.12 mm/s	0.34 mm/s	0.64 mm/s
Heave	0.73 mm/s	0.69 mm/s	1.11 mm/s
Roll	0.022 °/s	0.043 °/s	0.063 °/s
Pitch	0.328 °/s	0.321 °/s	0.317 °/s
Yaw	0.053 °/s	0.051 °/s	0.108 °/s

explanation is that feedback linearization tries to compensate the damping terms. However, if the modeled drag is higher than the actual drag, the linearization loop applies too much force, resulting in instabilities. A detailed overview of the error velocities is given in Table IV. During station keeping, we could measure a continuous power consumption of 2231 Watts using INDI, 2246 Watts using linear-drag FBL, and 2274 Watts using quadratic-drag FBL.

IV. CONCLUSION

In this paper, we conducted a first study on using incremental nonlinear dynamic inversion for attitude control of a 6-DOF underwater vehicle. Our results show that INDI clearly outperforms the model-based controller in stabilizing the upright pose with minimum drift. Keeping the drift of the AUV minimal is crucial for our intended use in intervention tasks. INDI results in a much lower steady-state error than the model-based controller, while keeping the roll axis much more stable during transition phase. Our findings are as expected, since the installed PHINS Compact C3 is a state-of-the-art military grade INS. It is possible that INDI might not have outperformed the model-based controller when less accurate sensor hardware or more advanced models had been used. However, modeling the dynamic behavior of marine vehicles is difficult, and not all hydrodynamic effects can be modeled exactly. For our sensor-based approach, only a 6×6 mass-inertia matrix together with an actuation model is needed, reducing the modeling effort significantly. This also comes with the advantage that altered dynamics have less impact on controller performance.

The results of this work can be seen as a basis for further studying INDI on marine vehicles. According to [14], INDI is also capable of adapting to actuator failure to a certain extent, which has not been studied in this work. For this reason, the authors plan to conduct upcoming research on Fault-Tolerant Control using INDI.

TABLE IV: Station Keeping Error (RMSE)

	INDI	Linear FBL	Quadratic FBL
Surge	0.128 mm/s	0.449 mm/s	0.279 mm/s
Sway	0.016 mm/s	0.171 mm/s	0.088 mm/s
Heave	0.017 mm/s	0.191 mm/s	0.117 mm/s
Roll	0.003 °/s	0.015 °/s	0.015 °/s
Pitch	0.007 °/s	0.066 °/s	0.049 °/s
Yaw	0.007 °/s	0.027 °/s	0.029 °/s

REFERENCES

[1] J. Lee and F. Zhao, "Global Wind Report 2021," Global Wind Energy Council, Tech. Rep., 2021.

[2] A. Woods, "A report on fatalities in commercial diving," Institute of Energy and Environmental Flows, Universtiy of Cambridge, Tech. Rep., 2022.

[3] P. Ridao, M. Carreras, D. Ribas, P. J. Sanz, and G. Oliver, "Intervention AUVs: The next challenge," *19th IFAC World Congress*, vol. 47, no. 3, pp. 12 146–12 159, 2014, doi: 10.3182/20140824-6-ZA-1003.02819.

[4] L. Christensen, J. Hilljegerdes, M. Zipper, A. Kolesnikov, B. Hülsen, C. E. S. Koch, M. Hildebrandt, and L. C. Danter, "The hydrobatic dual-arm intervention AUV Cuttlefish," in *OCEANS 2022, Hampton Roads*, 2022, pp. 1–8, doi: 10.1109/OCEANS47191.2022.9977150.

[5] S. Bhat and I. Stenius, "Hydrobatics: A review of trends, challenges and opportunities for efficient and agile underactuated AUVs," in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, 2018, pp. 1–8, doi: 10.1109/AUV.2018.8729805.

[6] T. Rossol, C. E. S. Koch, R. Bachmayer, and F. Kirchner, "Necessity of hydrostatic stability in autonomous underwater vehicles on intervention missions," in *OCEANS 2022, Hampton Roads*, 2022, pp. 1–10, doi: 10.1109/OCEANS47191.2022.9977030.

[7] P. Acquatella, W. Falkena, E.-J. Van Kampen, and Q. Chu, "Robust Nonlinear Spacecraft Attitude Control using Incremental Nonlinear Dynamic Inversion," in *AIAA Guidance, Navigation, and Control Conference 2012*, 01 2012, doi: 10.2514/6.2012-4623.

[8] N. Chaturvedi, S. Amit, and N. Mcclamroch, "Rigid-body Attitude Control," *Control Systems, IEEE*, vol. 31, pp. 30 – 51, 07 2011, doi: 10.1109/MCS.2011.940459.

[9] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011, doi: 10.1002/9781119994138.

[10] G. Antonelli, *Underwater Robots*. Springer International Publishing, 2018, doi: doi.org/10.1007/978-3-319-77899-0.

[11] G. Antonelli, S. Chiaverini, N. Sarkar, and M. West, "Adaptive control of an autonomous underwater vehicle: experimental results on ODIN," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 5, pp. 756–765, 2001, doi: 10.1109/87.944470.

[12] G. Marani, S. Choi, and J. Yuh, "Real-time center of buoyancy identification for optimal hovering in autonomous underwater intervention," *Intelligent Service Robotics*, vol. 3, pp. 175–182, 07 2010, doi: 10.1007/s11370-010-0068-1.

[13] P. R. Smith, "A simplified approach to nonlinear dynamic inversion based flight control," in *23rd Atmospheric Flight Mechanics Conference*, 1998, doi: 10.2514/6.1998-4461.

[14] S. Sun, X. Wang, Q. Chu, and C. d. Visser, "Incremental Nonlinear Fault-Tolerant Control of a Quadrotor With Complete Loss of Two Opposing Rotors," *IEEE Transactions on Robotics*, vol. 37, no. 1, pp. 116–130, 2021, doi: 10.1109/TRO.2020.3010626.

[15] G. Chen, A. Liu, J. Hu, J. Feng, and Z. Ma, "Attitude and Altitude Control of Unmanned Aerial-Underwater Vehicle Based on Incremental Nonlinear Dynamic Inversion," *IEEE Access*, vol. 8, pp. 156 129–156 138, 2020, doi: 10.1109/ACCESS.2020.3015857.

[16] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks, Fuzzy Models, and Gaussian Processes*. Springer International Publishing, 2020, doi: doi.org/10.1007/978-3-662-04323-3.

[17] A. Isidori, M. Thoma, E. D. Sontag, B. W. Dickinson, A. Fettweis, J. L. Massey, and J. W. Modestino, *Nonlinear Control Systems*, 3rd ed. Berlin, Heidelberg: Springer-Verlag, 1995, doi: 10.1007/978-1-84628-615-5.