

Energy-optimal linear quadratic tracking control for unmanned underwater vehicles in offshore aquaculture fish net-pen visual inspection

The International Journal of
Robotics Research
2026, Vol. 0(0) 1–30
© The Author(s) 2026



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/02783649261453998
journals.sagepub.com/home/ijr



Thein Than Tun^{1,2} , Loulin Huang¹  and Mark Anthony Preece³

Abstract

Unmanned underwater vehicles (UUVs) have been deployed for fish net-pen visual inspection (FNVI) in offshore aquaculture. Limited energy capacity of onboard power supplies constrains the UUV's working range and operating time. To minimize the energy consumption by the UUV during the FNVI of the Blue Endeavour Project (an offshore salmon farm of the New Zealand King Salmon Company), an energy-optimal linear quadratic tracking (EO-LQT) control scheme is proposed in this paper. For EO-LQTs implementation, a new Linear-Parameter-Varying (LPV) system that approximates the nonlinear UUV dynamics model with an accuracy of approximately 99% regardless of the operating points in real-time, with the modified versions of Bhāskara I's sine approximation and Shirali's cosine approximation, is developed. The use of the Lagrangian under the Principle of Least Action with the UUV's kinetic energy and the non-quadratic thruster power function in the EO-LQT performance index (PI) is demonstrated. The steps to solve the Hamilton-Jacobi-Bellman (HJB) equation with the non-quadratic Hamiltonian $\mathcal{H}$ are detailed to derive the new analytical EO-LQT optimal control form. Five EO-LQT controllers with different PIs are tested against the conventional LQT (CO-LQT) controller in both high-fidelity simulations under simulated disturbance speed up to 0.9 m/s and pool experiments, reducing energy consumption up to 37.1%. As key comparison metrics for the pose tracking and energy consumption, the mean-absolute-error (MAE) and T200 thruster power function are used to validate the effectiveness of the proposed EO-LQT controllers, compared to the CO-LQT controller.

Keywords

Unmanned Underwater Vehicle (UUV), Fish Net-pen Visual Inspection (FNVI), Linear-Parameter-Varying (LPV) system, Linear Quadratic Tracking (LQT) control, Robot Operating System (ROS), Gazebo Physics Engine

1. Introduction

In fulfilling the market demand for production, traditional inshore or nearshore aquaculture is facing the shortage of a suitable aquaculture environment for species such as King Salmon which requires cooler water temperatures (e.g., 12°C – 16°C) and adequate water flow, catering a natural environment for ideal fish growth (Preece, 2021). Other challenges of the existing aquaculture inshore or nearshore are environmental impact and conflicting coastal usages such as tourism, recreation, and conservation (Chu et al., 2020). As the inshore and nearshore areas reach the social carrying capacity, the New Zealand Government Aquaculture Strategy specifically mentions that by expanding into the open ocean, aquaculture presents transformational opportunities for its \$3 billion industry by 2035, together with sustainable land-based farming (The New Zealand

Government (Accessed 29 February 2024)). However, offshore aquaculture requires technical advancements for the farm and the workforce to withstand the harsh working environment. To minimize the safety concerns of the workforce and high operating costs in the long term, unmanned underwater vehicle (UUV) deployments are being

¹Department of Mechanical Engineering, School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland, New Zealand

²Blue Economy Cooperative Research Centre, Launceston, TAS, Australia

³The New Zealand King Salmon Company, Nelson, New Zealand

Corresponding author:

Loulin Huang, Department of Mechanical Engineering, School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland 1010, New Zealand.
Email: loulin.huang@aut.ac.nz

reported mostly for remote operations in the industry and for autonomous operations in research and development (Amundsen et al., 2021). One of the most common use cases of UUVs is to perform a fish net-pen visual inspection (FNVI) (Akram et al., 2022; Liao et al., 2022). For instance, the New Zealand King Salmon (NZKS) company deploys a remotely operated underwater vehicle (ROV), a type of UUV¹, to perform fish net-pen cleaning and visual inspection (1) to prevent net-occlusion and (2) to identify fish net-pen holes, respectively.

The NZKS has a dedicated deployment team that routinely conducts fish net-pen cleaning and visual inspection tasks in its fish farm. It takes approximately 2.5 hours to complete the task for a square net-pen: $30\text{ m} \times 30\text{ m} \times 15\text{ m}$ as shown in Figure 1. So, for offshore aquaculture with larger fish net-pen in harsh working environments, the visual inspection task will become a more hazardous, time-consuming, and costly process. As such, an option is to deploy autonomous UUVs to conduct fish net-pen visual inspection as a first step to solve the above problems facing the manual operations.

Unlike other torpedo-shaped UUVs in defense or oceanographic research, UUVs used for aquaculture are rectangular in shape, offering high maneuverability (with 6 or 8 thrusters) in a constrained environment. However, this type of UUV encounters more hydrodynamic effects such as added mass, damping, and other coupled nonlinear dynamic parameters (Kim et al., 2021). As a result, these increased hydrodynamic effects lead to higher energy demands for UUVs used in aquaculture applications. To solve this problem, the optimal design of a UUV with less hydrodynamic effects is proposed (Ao et al., 2024). However, due to the geometrical shape of the UUV and its effects on maneuverability as mentioned above, optimizing UUV shape alone will not resolve the task-specific requirements (e.g., the optimal slender body-shaped UUV cannot maneuver in the constrained operational environment of a fish farm.).



Figure 1. Fish farm with square fish net-pens of the New Zealand King Salmon company in Nelson, New Zealand.

Alternatively, energy-optimality can be achieved at the UUV's trajectory planning stage and real-time control stage with optimal control approaches. An online time-optimal trajectory planner for a slender body-shaped AUV called "nupiri muka" in a dynamic environment is reported in Lim et al. (2022). Energy-aware route optimization problem (EA-OP) is proposed and tested on an IVER3 AUV and a Nessie VII AUV in sea trials in De Carolis et al. (2018). Online Energy-Efficient Stochastic Trajectory Optimization (EESTO) using historical ocean currents data from Regional Ocean Modeling System (ROMS) is presented in Jones and Hollinger (2017). Another energy-optimal path planning with active flow perception of ocean current is proposed in Yang et al. (2021). Such planning approaches considering environmental disturbances are useful in oceanographic research covering a large range where the waypoints are not heavily constrained. In other words, there is a freedom to select the waypoints aligned with underwater current flow to minimize energy consumption. However, for FNVI operation, the waypoints (or areas of interest around the fish net-pen) are fixed. For the floating flexible cage system, the fish net-pen deforms due to the time-varying underwater current, but the visual inspection waypoints remain relatively the same around the fish net-pen. Therefore, the visual inspection trajectory usually covers areas where the underwater current might be against or aligned with. As a result, energy-optimal control, regardless of the trajectory planning, is more crucial for FNVI operation.

Energy-optimal depth tracking of an AUV using model predictive control (MPC) is proposed in Yao et al. (2019). Still, the proposed cost function or PI in the form of quadratic inputs (thrust) does not express power or energy explicitly and accurately according to either's definition. Energy-optimal depth control using an infinite time linear quadratic regulator (LQR) on a linear time-invariant (LTI) model of an autonomous underwater glider (AUG), a special type of UUV, is proposed in Claus and Bachmayer (2016). Energy sub-optimal sliding mode control (SOSMC) is presented in Sarkar et al. (2016) in which the output from SMC is fed into finite-time LQR as the control input to be minimized. However, still, no specific power or energy function is used in the cost function. Using the thruster's power function of DROP-Sphere AUV with 4 degrees of freedom (DoF), the development of a real-time energy-optimal MPC called RTEO-MPC, which can switch between solving static and dynamic surge motion optimizations, is proposed in Yang et al. (2018). The experiments of tracking two horizontal waypoints without obstacles or ocean currents are conducted using MATLAB/Simulink at the control loop rate of 10 Hz and a prediction horizon of 15 steps, and the results of RTEO-MPC are found close to that of the open-loop global optimal solution obtained from direct collocation (DC). Similarly, on the same platform, an energy-optimal economic MPC called EO-EMPC is

developed considering the terminal cost (energy-to-go) in which static (surge and heave) and dynamic (surge, heave, and yaw) costs are considered (Yang et al., 2019). The results of EO-EMPC are compared to those of DC and line-of-sight MPC (LOS-MPC), achieving close performance to DC with substantially less computation time. In another work, the performances of conventional LOS-MPC (CLOS-MPC), nominal energy-optimal LOS-MPC (ELOS-MPC), ideal ELOS-MPC, and robust ELOS-MPC are compared in Yang et al. (2020), which reports that robust ELOS-MPC demonstrates higher energy efficiency, with reduced UUV travel time and lower root-mean-square error (RMSE), compared to other ELOS-MPC approaches. However, it is less efficient than CLOS-MPC in UUV traveling time and RMSE. In the controllers presented in the aforementioned literature, surge and yaw are controlled by MPC, and pitch and heave are controlled by proportional-integral-derivative (PID) control to reduce computational load. In Yang et al. (2024) for 3D path following under ocean currents, vertical setpoint (heave and pitch) tracking MPC and horizontal setpoint (yaw and surge) tracking MPC under 3D EO-MPC are proposed and compared with integral LOS-based control and 2D EO-MPC from Yang et al. (2020) and details of implementation of such decentralized MPC (DMPC) can be found in Shen et al. (2016); Shen and Shi (2020). This decoupled control approach is only suitable for the lawnmower-type operation with weak coupling between the vertical and horizontal planes of the system's dynamics. It reports that, generally, at the expense of a substantial increase in UUV traveling time and a small variation in path-following error, 3D EO-MPC is more energy-efficient than others. An energy-optimal motion planning (trajectory generation and control) formulated as nonlinear robust MPC (NRMPC) and solved by an A*-like algorithm is proposed in Huynh et al. (2015). Its energy function takes into consideration hotel load (used by computing and sensor systems without propulsion), energy to overcome inertia and drag forces, and estimated remaining energy for waypoints clearance. Energy-optimal motion planning for the Norwegian Experimental Remotely Operated Vehicle (NEROV) using direct shooting is proposed to solve the non-quadratic dissipated energy function of the thruster in Spangelo and Egeland (1992).

It is noted that in the above-mentioned literature relevant to UUV energy-optimal control, most of the UUVs are in slender-body/torpedo shapes, which suffer less from hydrodynamic effect than square/rectangular/box-shaped UUVs that encounter large hydrodynamic effect but offer high maneuverability. They also have large inertia to withstand high disturbances and have low DoF or are assumed to operate in a simplified scenario such as pure horizontal plane motion or depth control. They are not suitable for operation in constrained operational environments like fish farms. It is also noted that in the energy-optimal control schemes reported, most of them have no

explicit power/energy functions used to define the cost functions. Additionally, they are not conducted in high-fidelity simulation environments with well-integrated hydrodynamic effects, which can be readily extended to field study. Alternatively, the literature on high-fidelity simulation does not necessarily cover the aspects of energy-optimality for use in offshore aquaculture. Moreover, numerical optimization-based control strategies (e.g., MPC) can handle the highly nonlinear and coupled UUV dynamics well but require high computing capacity. On the other hand, with a lower computation load, an analytical approach using a linear control strategy (e.g., conventional LQR or linear quadratic tracking) can only operate around the operating point at which it is linearized.

To address the issues identified above, the main contributions of this paper include:

- Formulation of the fish net-pen visual inspection task as an energy-optimal finite-horizon constrained optimization problem, namely energy-optimal linear quadratic tracking (EO-LQT) control

- New development on Bhāskara I's sine approximation and Shirali's cosine approximation to extend their respective original domain to the full range ($-\pi$ to π)

- New development of a Linear-Parameter-Varying (LPV) system to approximate the highly nonlinear and coupled UUV dynamics model with an accuracy of approximately 99% regardless of the operating points in real-time by adopting the proposed modified versions of Bhāskara I's sine approximation and Shirali's cosine approximation

- Use of the Lagrangian under the Principle of Least Action (PLA) with linear and rotational kinetic energy, transformed into error states, to be adopted in the EO-LQT controller's performance index (PI)

- Use of the non-quadratic power function of UUV's thruster, partially reflecting the unmodeled energy consumption normally ignored in the energy-optimal control, in the EO-LQT controller's PI

- Derivation of the new analytical optimal control form by re-solving the HJB equation with non-quadratic Hamiltonian $\mathcal{H}$ due to the adoption of the non-quadratic thruster power function

- High-fidelity simulations of the proposed LQT controllers (CO-LQT and EO-LQTs), in the existing Robot Operating System (ROS)-based high-fidelity simulation platform, integrated with Gazebo Physics Engine called UUV Simulator (Manhães et al., 2016 (Accessed 20 November 2023))

- Performance validation of the proposed LQT controllers under three different underwater current speeds (e.g., 0.0 m/s to 0.9 m/s) in high-fidelity simulations

Experiments of the proposed LQT controllers in the pool, along with their respective high-fidelity simulations in the pool environment

Use of T200 thruster power function, approximated by the 6th-order polynomial regression, to compare real-time energy consumption among controllers in both high-fidelity simulations and pool experiments. Use of the publicly available specifications and the constraints of the Blue Endeavour Project (the first of its kind in New Zealand) of the New Zealand King Salmon Company and the lightweight and highly maneuverable BlueROV2 Heavy Configuration in these simulations ([The New Zealand King Salmon Company](#) (Accessed 25 November 2023); [BlueRobotics](#) (Accessed 02 August 2023))

The remaining part of this paper is structured as follows. Section 2 presents UUV's 6-DoF nonlinear model, its state-space representation (SSR), and the subsequent Linear-Parameter-Varying (LPV) model using the modified versions of Bhāskara I's sine approximation and Shirali's cosine approximation. Section 3 discusses the CO-LQT control problem. Section 4 covers the formulation of energy terms which are used in Section 5 that details the EO-LQT control problem. Section 6 describes the Hamilton-Jacobi-Bellman equation, the necessary and sufficient condition for optimality, and the detailed steps of finding the conventional optimal control and non-quadratic optimal control. Section 7 explains high-fidelity simulations and experiments, covering the experimental setup for vision-based state-estimation that is implemented using a Kalman Filter and is crucial to conducting experiments in the pool. Section 8 discusses the results of FNVI simulations, pool experiments, and simulations, in terms of trajectory tracking, pose tracking, and energy consumption. Section 9 suggests future work to address the factors that potentially cause the performance differences of the controller between pool experiments and simulations, and highlights future research direction, targeted to field trials. Section 10 concludes the summarized findings of this research work.

2. Dynamic model of 6-DoF UUV

2.1. Nonlinear model

The 6-DoF nonlinear dynamic model consists of rigid body dynamics and the additional hydrodynamics due to the added mass, linear and quadratic damping ([Antonelli, 2018](#); [Fossen, 2011](#)). The control wrench depends on the thrusters' static configurations in the UUV frame $\{b\}$. The Society of Naval Architects and Marine Engineers (SNAME) nomenclature as shown in [Figure 2](#) is used for the UUV's dynamic modeling and the relevant reference frames are illustrated in [Figure 3](#).

	Surge			Sway			Heave		
Position	x	y	z	ϕ	θ	ψ	Euler Angles		
Linear Velocity	u	v	w	p	q	r	Angular Velocity		
Force	X	Y	Z	K	M	N	Moment		
				Roll	Pitch	Yaw			

Figure 2. SNAME nomenclature for the UUV's dynamic modeling.

Assigning Frame $\{b\}$ at the center of mass/gravity, the simplified mass matrix of the rigid body can be represented by

$$M_R = \begin{bmatrix} m_r I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_G \end{bmatrix}$$

where m_r is the mass of the UUV, $I_G \in \mathbb{R}^{3 \times 3}$ is the UUV's moment of inertia expressed in Frame $\{b\}$, $I_{3 \times 3} \in \mathbb{R}^{3 \times 3}$ is the identity matrix, and $0_{3 \times 3} \in \mathbb{R}^{3 \times 3}$ is the zero matrix. Likewise, the simplified Coriolis-Centripetal acceleration matrix of the rigid body can be expressed as

$$C_R(v) = \begin{bmatrix} m_r[v_2 \times] & 0_{3 \times 3} \\ 0_{3 \times 3} & -[(I_G v_2) \times] \end{bmatrix}$$

where $[(\cdot) \times]$ is the skew-symmetric operator on the vector $(\cdot)$.

Assumption 2.1. *The UUV has three symmetric planes and is completely submerged in the water and operated at low speed during the operation.*

With Assumption 2.1, only the diagonal components of the mass matrix, contributed by the added mass, are taken into account,

$$M_A = \text{diag}([X_u \ Y_v \ Z_w \ K_{\dot{p}} \ M_{\dot{q}} \ N_{\dot{r}}]^T)$$

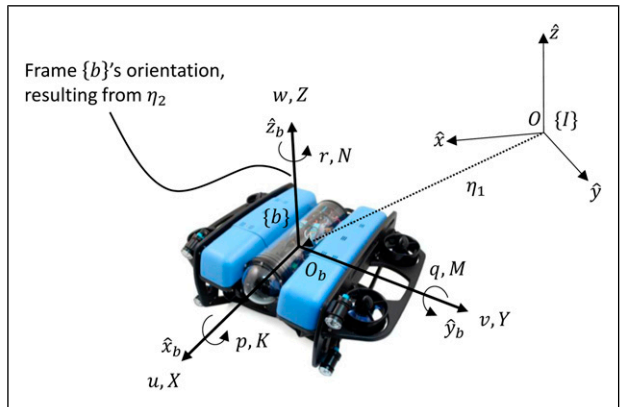


Figure 3. BlueROV2 Heavy Configuration with SNAME nomenclature in East-North-Up (ENU) reference frame.

where X_u , Y_v , Z_w , K_p , M_q , and N_r are added mass coefficients.

Using the added mass matrix, the Coriolis-Centripetal acceleration matrix, contributed by the added mass, can be described by

$$C_A(v) = \begin{bmatrix} 0_{3 \times 3} & -[(A_{11}v_1 + A_{12}v_2) \times] \\ -[(A_{11}v_1 + A_{12}v_2) \times] & -[(A_{21}v_1 + A_{22}v_2) \times] \end{bmatrix}$$

where

$$M_A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

The approximated linear and quadratic damping matrices can be represented by

$$\begin{aligned} D_l &= \text{diag}([X_u \ Y_v \ Z_w \ K_p \ M_q \ N_r]^T) \\ D_q(v) &= \text{diag}([X_{u|u}|u| \ Y_{v|v}|v| \ Z_{w|w}|w| \ K_{p|p}|p| \\ &\quad M_{q|q}|q| \ N_{r|r}|r|]^T) \end{aligned}$$

where X_u , Y_v , Z_w , K_p , M_q , and N_r are linear damping coefficients. $X_{u|u}|$, $Y_{v|v}|$, $Z_{w|w}|$, $K_{p|p}|$, $M_{q|q}|$, and $N_{r|r}|$ are quadratic damping coefficients. The restoring forces and moments due to the gravitational force and buoyancy can be described by

$$g(\eta) = \begin{bmatrix} R_b^l(\eta)(f_G^l + f_B^l) \\ r_G^b \times R_b^l(\eta)f_G^l + r_B^b \times R_b^l(\eta)f_B^l \end{bmatrix} \quad (1)$$

where $f_G^l = [0 \ 0 \ -w_G]^T$, $f_B^l = [0 \ 0 \ w_B]^T$, w_G is the UUV's weight and w_B is its buoyancy force. $R_b^l(\eta)$ is the rotation matrix from Frame $\{I\}$ to Frame $\{b\}$. r_G^b and r_B^b are the position vectors of the center of gravity (CoG) and buoyancy (CoB) in Frame $\{b\}$, respectively.

Suppose the j th thruster's direction vector expressed in Frame $\{T_j\}$, which is defined with respect to Frame $\{b\}$, is denoted by

$$d_{T_j} = [x_{T_j} \ y_{T_j} \ z_{T_j}]^T$$

Then, with the thrust magnitude of $f_{T,j}$, the j th thrust vector is $d_{T_j}f_{T,j}$ and its resulting wrench can be described by

$$\tau_{f_{T,j}} = \begin{bmatrix} R_{T_j}^b d_{T_j} f_{T,j} \\ [r_{T_j}^b \times] R_{T_j}^b d_{T_j} f_{T,j} \end{bmatrix} = B_{T_j} f_{T,j}$$

Therefore, the total wrench propagated by m number of thrusters can be described by

$$\tau = [B_{T_1} \ B_{T_2} \ \dots \ B_{T_m}] f_T = B_T f_T$$

Using the terms mentioned above, the compact nonlinear dynamic model can be described in the form of

$$M\dot{v} + (C(v) + D(v))v + g(\eta) = \tau (\triangleq B_T f_T) \quad (2)$$

where $M = (M_R + M_A)$, $C(v) = C_R(v) + C_A(v)$, and $D(v) = D_l + D_q(v)$. $\eta_1 = [x \ y \ z]^T$ is the position vector, $\eta_2 = [\phi \ \theta \ \psi]^T$ is the orientation (Euler angles) vector, and $\eta = [\eta_1 \ \eta_2]^T \in \mathbb{R}^6$ is

the UUV's pose vector with respect to (w.r.t) and expressed in Frame $\{I\}$. $v_1 = [u \ v \ w]^T$ is the linear velocity vector, $v_2 = [p \ q \ r]^T$ is the angular velocity vector, and $v = [v_1 \ v_2]^T \in \mathbb{R}^6$ is the UUV's twist vector w.r.t Frame $\{I\}$ and expressed in Frame $\{b\}$. $\tau_1 = [X \ Y \ Z]^T$ is the force vector, $\tau_2 = [K \ M \ N]^T$ is the moment vector, and $\tau = [\tau_1 \ \tau_2]^T \in \mathbb{R}^6$ is the UUV's wrench vector w.r.t and expressed in Frame $\{b\}$. $B_T \in \mathbb{R}^{6 \times m}$ is the thruster control/allocation matrix (TCM/TAM). $f_T \in \mathbb{R}^m$ is the input (Thrust) vector.

The UUV's twist can be converted to the rates of change of position and Euler angles using the Jacobian J_b^l as follows to adopt in the full 12-state dynamic model.

$$\dot{\eta} = J_b^l(\eta)v \quad (3)$$

where

$$J_b^l(\eta) = \begin{bmatrix} R_b^l(\eta) & 0_{3 \times 3} \\ 0_{3 \times 3} & T_b^l(\eta) \end{bmatrix}, \quad J_b^b(\eta) = \begin{bmatrix} R_b^l(\eta)^{-1} & 0_{3 \times 3} \\ 0_{3 \times 3} & T_b^l(\eta)^{-1} \end{bmatrix},$$

$$T_b^l(\eta) = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix}, \theta \neq \pm \frac{\pi}{2},$$

$$T_b^b(\eta) = T_b^l(\eta)^{-1} = \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & s(\phi)c(\theta) \\ 0 & -s(\phi) & c(\phi)c(\theta) \end{bmatrix},$$

$$R_b^l(\eta) = R_z(\psi)R_y(\theta)R_x(\phi),$$

$$s(\cdot) = \sin(\cdot), c(\cdot) = \cos(\cdot), t(\cdot) = \tan(\cdot)$$

Let

$$\zeta_1 = \eta$$

$$\zeta_2 = v$$

$$\dot{\zeta}_1 = \dot{\eta} = J_b^l(\eta)v = J_b^l(\zeta_1)\zeta_2$$

$$\dot{\zeta}_2 = \dot{v} = M^{-1}[B_T f_T - (C(\zeta_2) + D(\zeta_2))\zeta_2 - g(\zeta_1)]$$

By rearranging the equations above, 12-state nonlinear state-space representation (SSR) model can be obtained as

$$\dot{\zeta} = A_\zeta(\zeta)\zeta + g_{\zeta_1}(\zeta_1) + B f_T \quad (4)$$

$$\ddot{\zeta} = C \zeta \quad (5)$$

where $\zeta = [\eta \ v]^T$, $\dot{\zeta} = [\dot{\eta} \ \dot{v}]^T$,

$$A_\zeta(\zeta) = \begin{bmatrix} 0_{6 \times 6} & J_b^l(\zeta_1) \\ 0_{6 \times 6} & -M^{-1}(C(\zeta_2) + D(\zeta_2)) \end{bmatrix},$$

$$g_{\zeta_1}(\zeta_1) = \begin{bmatrix} 0_{6 \times 1} \\ -M^{-1}g(\zeta_1) \end{bmatrix}, \quad B = \begin{bmatrix} 0_{6 \times 8} \\ M^{-1}B_T \end{bmatrix}, \quad C = I_{12 \times 12}$$

To express all the states w.r.t Frame $\{b\}$, ζ_1 and $\dot{\zeta}_1$ can be rewritten as $\zeta_1 = {}^b\eta$ and $\dot{\zeta}_1 = {}^b\dot{\eta} = v$. Therefore, equation (4)

now consists of the modified representations of states, $\zeta = [{}^b\eta \ v]^T$, $\dot{\zeta} = [{}^b\dot{\eta} \ \dot{v}]^T$ and $A_\zeta(\zeta)$ as follows:

$$A_\zeta(\zeta) = \begin{bmatrix} \mathbf{0}_{6 \times 6} & I_{6 \times 6} \\ \mathbf{0}_{6 \times 6} & -M^{-1}(C(\zeta_2) + D(\zeta_2)) \end{bmatrix} \quad (6)$$

In this work, the UUV's 12-state nonlinear SSR model, represented by equations (4) and (6), will be used in the following subsection to develop the Linear-Parameter-Varying (LPV) system.

2.2. Linearized model

Conventionally, a nonlinear system is linearized to a Linear-Parameter-Varying (LPV) system via Jacobian at each operating point (MathWorks (Accessed 12 August 2023)). This method is based on the Taylor series expansion of the nonlinear system truncated at the first partial derivative around an operating point $(\zeta_o, f_{T,o})$. For instance, equation (4) can be written as $\dot{\zeta} = h(\zeta, f_T)$ and its linearized model can be described as

$$\dot{\zeta} \approx h(\zeta_o, f_{T,o}) + \frac{\partial h(\zeta_o, f_{T,o})}{\partial \zeta} (\zeta - \zeta_o) + B(f_T - f_{T,o}) \quad (7)$$

However, equation (7) is not suitable to produce a standard linear system (e.g., $\dot{\zeta} = A(\zeta)\zeta + Bf_T$) of a highly nonlinear dynamic system such as a UUV. If the control objective is the trajectory tracking, one possible approach is to reformulate equation (7) as the error dynamics as follows.

Suppose $\dot{e} = \dot{\zeta} - \dot{\zeta}_o (\triangleq \dot{\zeta}_d)$,

$$\dot{e} \approx \frac{\partial h(\zeta_o, f_{T,o})}{\partial \zeta} e + B\bar{f}_T = Ae + B\bar{f}_T \quad (8)$$

where ζ_d is the desired trajectory and $\bar{f}_T = f_T - f_{T,o}$. Although equation (8) is now in the form of a standard linear system, there are two main issues arising from calculating the matrix A . Firstly, the function $h(\cdot)$ consists of multiple squared parameters (e.g., $u|u|$, $v|v|$ in state equations). Due to the absolute term (e.g., $|u|$), $\partial u|u|/\partial u$ produces the piecewise solution and its linearized function derived via Jacobian returns twice the actual value of the original function. Secondly, the function $h(\cdot)$ consists of sine and cosine functions which are hard to approximate as linear functions over a large operating range, introducing large approximation inaccuracies. In addition to those two issues, due to the nature of Taylor series expansion, if the desired trajectory is far from the UUV's states (e.g., $e \gg 0$), equation (8) will not reflect the UUV's dynamics anymore due to the lack of truncated higher-order terms.

Therefore, instead of using the Jacobian method, the method of factorizing and extracting states ζ from equation (4) is proposed. In the term $u|u|$, u can be easily extracted, but this is not the case for sine or cosine functions. One way to deal with it is to represent sine (e.g., $s(\gamma)$) and cosine (e.g.,

$c(\beta)$) as rational functions such as Joseph (2009, p. 57) and Shirali (2011, p. 99),

$$s(\gamma) \approx \frac{16\gamma(\pi - \gamma)}{5\pi^2 - 4\gamma(\pi - \gamma)}, \quad 0 \leq \gamma \leq \pi \quad (9)$$

$$c\left(\frac{\pi\zeta}{2}\right) \approx \frac{4(1 - \zeta^2)}{4 + \zeta^2}, \quad -1 \leq \zeta \leq 1 \quad (10)$$

Letting

$$\beta = \frac{\pi\zeta}{2} \Rightarrow \zeta = \frac{2\beta}{\pi}$$

Equation (10) can be rewritten as

$$c(\beta) \approx \frac{\pi^2 - 4\beta^2}{\pi^2 + \beta^2}, \quad -\frac{\pi}{2} \leq \beta \leq \frac{\pi}{2} \quad (11)$$

where $c_\beta(\beta) = \frac{\pi^2 - 4\beta^2}{\pi^2 + \beta^2}$. $\text{sgn}(\beta)$ is the sign function of β which returns 1 for non-negative values or -1 otherwise. $\max(|\beta|, \epsilon)$ returns either $|\beta|$ or ϵ whichever has a larger value.

Note: in extracting β from $\pi^2 - 4\beta^2$, the denominator of $\frac{\pi^2}{\beta}$ is replaced with $\text{sgn}(\beta)\max(|\beta|, \epsilon)$ to prevent it from becoming undefined when $\beta = 0$.

Equations (9) and (11) are modified further so that they can be factorized in a larger domain.

$$s(\gamma) \approx s_\gamma(\gamma)\gamma, \quad -\pi \leq \gamma \leq \pi \quad (12)$$

where $s_\gamma(\gamma) = \frac{16(\pi - |\gamma|)}{5\pi^2 - 4|\gamma|(\pi - |\gamma|)}$.

Using the properties of $c_\beta(\beta)^2\beta^2 + s_\beta(\beta)^2\beta^2 \approx 1$,

$$c(\beta) \approx c_\beta(\beta)\beta, \quad -\pi \leq \beta \leq \pi \quad (13)$$

where

$$c_\beta(\beta) \approx \begin{cases} -\sqrt{\frac{1}{\max(\beta^2, \epsilon)} - s_\beta(\beta)^2}, & \frac{\pi}{2} < \beta < \pi \\ \frac{\pi^2}{\frac{\text{sgn}(\beta)\max(|\beta|, \epsilon)}{\pi^2 + \beta^2}} - 4\beta, & -\frac{\pi}{2} \leq \beta \leq \frac{\pi}{2} \\ \sqrt{\frac{1}{\max(\beta^2, \epsilon)} - s_\beta(\beta)^2}, & -\pi < \beta < -\frac{\pi}{2} \end{cases}$$

where $\frac{1}{\max(\beta^2, \epsilon)} \geq s_\beta(\beta)^2$. The original formulations are reported to achieve an accuracy of approximately 99% within their defined domain. To investigate our modified versions over the full domain as shown in equations (12) and (13), Figures 4 and 5 are produced using 1×10^6 data points and compared with the values of sine and cosine function of Numpy Python library as follow. The maximum relative approximation errors are

$$e_s = \frac{|\text{np.sin}(\gamma) - s_\gamma(\gamma)\gamma|}{|\text{np.sin}(\gamma)|}, \max(e_s) = 1.85914\% \quad (14)$$

$$e_c = \frac{|\text{np.cos}(\beta) - c_\beta(\beta)\beta|}{|\text{np.cos}(\beta)|}, \max(e_c) = 1.85915\% \quad (15)$$

Therefore, our proposed method produces the approximated sine and cosine functions over the full domain with an accuracy of approximately 98 %.

Using the techniques described above, equation (4) needs to be modified in the standard form for a Linear-Parameter-Varying (LPV) system. Firstly, due to the frame assignment of Frame $\{b\}$ at the center of gravity, and the center of buoyancy in the positive z -axis (upward) of Frame $\{b\}$, the gravity vector and the buoyancy vector from equation (1) becomes $r_G^b = [x_G \ y_G \ z_G]^T = [0 \ 0 \ 0]^T$ and $r_B^b = [x_B \ y_B \ z_B]^T = [0 \ 0 \ z_B]^T$, resulting in a simplified version of $g_{\zeta_1}(\zeta_1)$ in equation (4) as follows.

The simplified equation (1) becomes

$$g(\zeta_1) = \begin{bmatrix} -k_1 s(\theta) \\ k_1 c(\theta) s(\phi) \\ k_1 c(\theta) c(\phi) \\ -k_2 c(\theta) s(\phi) \\ -k_2 s(\theta) \\ 0 \end{bmatrix} \quad (16)$$

where $k_1 = w_B - w_G$, $k_2 = z_B w_B$.

Subsequently, the proposed technique above for sine and cosine approximation will be applied on equation (16) such that it results in

$$g_{\zeta_1}(\zeta_1) \approx \begin{bmatrix} 0_{6 \times 6} & 0_{6 \times 6} \\ -M^{-1}G_r(\zeta_1) & 0_{6 \times 6} \end{bmatrix} \zeta \approx A_{g_{\zeta_1}}(\zeta_1) \zeta \quad (17)$$

where

$$G_r(\zeta_1) = \begin{bmatrix} 0 & -k_1 s_\theta(\theta) & 0 \\ k_1 c(\theta) s_\phi(\phi) & 0 & 0 \\ k_1 c(\theta) c_\phi(\phi) & 0 & 0 \\ -k_2 c(\theta) s_\phi(\phi) & 0 & 0 \\ 0 & -k_2 s_\theta(\theta) & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Finally, by substituting equation (17) into equation (4), the UUV's LPV system can be described by

$$\dot{\zeta} = A(\zeta)\zeta + Bf_T, \quad (18)$$

where

$$A(\zeta) = \overbrace{A_\zeta(\zeta)}^{\text{Eqn. (6)}} + A_{g_{\zeta_1}}(\zeta_1)$$

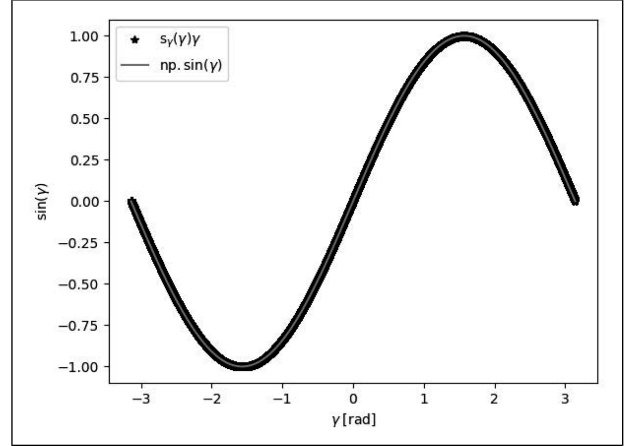


Figure 4. Sine approximation, resulting from the modified version (equation (12)) of Bhāskara I's rational expression.

where $A(\zeta) \in \mathbb{R}^{n \times n}$ is the state-varying matrix. As both M^{-1} and B_T are constant matrices in equation (4), $B \in \mathbb{R}^{n \times m}$ is a constant matrix. To estimate the maximum relative approximation error between the nonlinear SSR model equation (4) and the proposed LPV system equation (18), the elementwise operation on the vector is performed as shown below.

$$e_{\dot{\zeta}} = \frac{|\dot{\zeta}_{(4)} - \dot{\zeta}_{(18)}|}{|\dot{\zeta}_{(4)}|} \quad (19)$$

$$= \frac{\underbrace{|\underbrace{g_{\zeta_1}(\zeta_1)}_{\text{Hydrostatics}} - \underbrace{A_{g_{\zeta_1}}(\zeta_1)\zeta}_{\text{Approximated Hydrostatics}}|}_{\underbrace{\quad}_{\text{Full System Dynamics}}}}{|\dot{\zeta}_{(4)}|}$$

In a similar fashion to equations (14) and (15) for the elementwise comparison on equation (19),

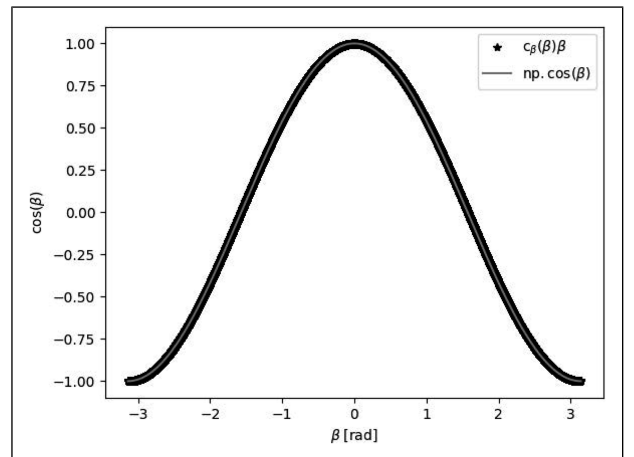


Figure 5. Cosine approximation, resulting from the modified version (equation (13)) of Shirali's rational expression.

$|\dot{\zeta}_{(4)}| \gg |g_{\zeta_1}(\zeta_1) - A_{g_{\zeta_1}}(\zeta_1)\zeta| \Rightarrow e_{\zeta} \ll 1.85915\%$. Therefore, as an LPV system, equation (18) approximates the nonlinear UUV dynamic system described by equation (4) with an accuracy of approximately 99% regardless of the operating points in real-time, whereas conventional methods, such as the Jacobian linearization or truncated Taylor series expansion, fail to achieve such a level of accuracy due to their reliance on operating points and the extent of the system's nonlinearity.

Remark 2.1. The state-varying matrix A is also denoted as $A(x, t)$, $A(t)$, or $A(x, k)$ (Morato et al., 2020).

3. Linear quadratic tracking control problem

A general trajectory tracking problem in the sense of LQT is illustrated in Figure 6. In a generic LQT control for the fish net-pen visual inspection task, the performance index (PI) will be formulated as the finite-horizon Bolza type which includes both terminal cost and integral tracking cost, as shown in equation (20).

$$J(\zeta, f_T, t) = J_{f_T} + \int_{t_0}^{t_f} \underbrace{[J_T + J_E]}_{\text{Running Cost: } \ell(\cdot)} dt \quad (20)$$

Therefore, the minimum cost function $J^*(\zeta, t)$ is

$$\begin{aligned} \min_{f_T(\cdot) \in \mathcal{F}_{\text{admissible}}[t_0, t_f]} & \left\{ J_{f_T} + \int_{t_0}^{t_f} [J_T + J_E] dt \right\} \\ \text{subject to} & \quad \dot{\zeta} = A(\zeta)\zeta + Bf_T \\ & \quad \ddot{\zeta} = C\zeta \end{aligned} \quad (21)$$

where the terminal tracking cost: $J_{f_T} = e_{f_T}^T Q_f e_{f_T}$, the terminal tracking error: $e_{f_T} = \zeta_{d,f} - \zeta_f$, the terminal tracking penalty weight: $Q_f = Q_f^T \in \mathbb{R}_{0+}^{n \times n}$, the tracking cost: $J_T = e^T Q e$, the tracking error: $e = \zeta_d - \zeta$, the tracking penalty weight: $Q = Q^T \in \mathbb{R}_{0+}^{n \times n}$, the control-effort cost: $J_E = f_T^T R f_T$ (known as the energy penalty term in the conventional sense of LQT), and the control-effort penalty weight: $R = R^T \in \mathbb{R}_{0+}^{m \times m}$. Considering the specifications of the UUV and its working conditions, constraints are imposed on the controller outputs via a thrust saturation function as shown in equation (22).

$$\text{sat}(f_{T,j}) = \begin{cases} f_{T,j}, & |f_{T,j}| < f_{T,j,\text{rated}}, \\ \text{sgn}(f_{T,j})f_{T,j,\text{rated}}, & |f_{T,j}| \geq f_{T,j,\text{rated}} \end{cases} \quad (22)$$

where $j = \{1, 2, \dots, m\}$, $f_{T,j,\text{rated}}$ is the rated thrust limit to ensure that the thruster does not operate at the maximum limit continuously.

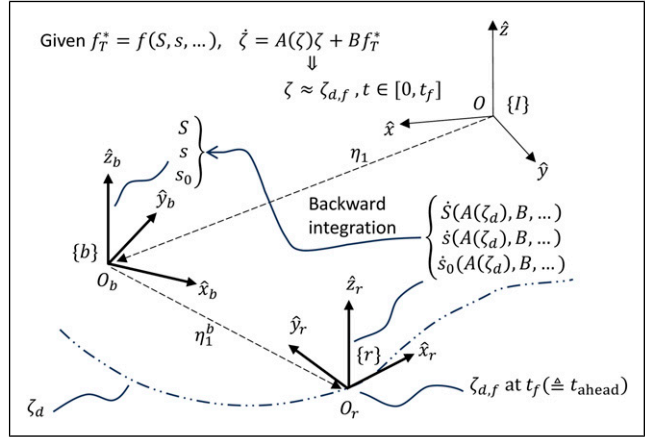


Figure 6. Illustration of trajectory tracking problem in the sense of LQT.

Note: The system from equations (18) and (5) is controllable and observable. Given the controllability matrix:

$$C = [B \ AB \ A^2B \ \dots \ A^{n-1}B] \text{ and the observability matrix: } \mathcal{O} = [C \ CA \ CA^2 \ \dots \ CA^{n-1}]^T, \text{ rank}\{C\} = \text{rank}\{\mathcal{O}\} = 12.$$

4. Formulation of energy terms in the PI

The energy terms in the PI need to be established first before an EO-LQT controller can be designed. As shown in the UUV system's dynamic model in equation (2), there are UUV's states and inputs which are directly linked to the total energy consumption. According to the law of conservation of energy,

$$E_m + E_h = E_{f_T} \quad (23)$$

where E_m is the energy associated with the UUV's motion as a rigid body, E_h is the energy due to the hydrodynamics and unmodeled dynamics which is hard to model, E_{f_T} is the energy provided by the UUV's thrusters. E_m and E_{f_T} can be obtained via the energy of motion and thruster's model, respectively. In the following subsections, those terms will be examined and transformed into state or input so that they can be embedded into PI as shown in equation (21).

4.1. Energy associated with the UUV's motion: E_m

Generally, the energy associated with the UUV's motion E_m is related to the kinetic energy E_k and the potential energy E_p . Therefore, if these energy terms are embedded into the PI as a part of J_E as shown in equation (21), the integration of the change in energy will result in action according to the Principle of Least Action (PLA), which states that the true trajectory is the one that results in the minimum action value of equation (24). Its use in optimal trajectory planning is demonstrated in Huang et al. (2023).

$$S = \int_{t_0}^{t_f} \mathcal{L} dt \quad (24)$$

where the Lagrangian: $\mathcal{L} = E_k - E_p$.

However, as mentioned in Remark 4.1, the potential energy: E_p will not be considered. Therefore, equation (24) can be reflected in equation (21) as part of the energy-related term as shown below.

$$\min [S = \int_{t_0}^{t_f} \mathcal{L} (\triangleq E_k - \cancel{E_p}) dt = \int_{t_0}^{t_f} J_E dt] \quad (25)$$

Remark 4.1. As the UUV can be configured mechanically to become neutrally buoyant, the potential energy resulting from the gravitational acceleration can be neglected. Although there are other types of potential energy (e.g., fluid memory effects), resulting from the environments and disturbances, they are not considered, as, unlike an unmanned surface vehicle (USV), the UUV will be fully submerged and will not operate at the free surface most of the time.

As $\mathcal{L}$ now consists of only E_k (Battista et al., 2016; Jung et al., 2021) and involves the system's states, those energy terms will be written in tracking error states via state transformation to have a compatible form (either in error state or thrust input) in LQT control. Firstly, the relationship between the state v and its respective error e_v needs to be established. As the tracking twist error is $e_v = v_d - v$ where v_d is the desired twist, v can be written as a function of v_d and e_v as follows:

$$\begin{aligned} v &= v_d - e_v \\ &= \left(\frac{v_d}{e_v} - 1 \right) e_v \quad : (\text{Elementwise operation}) \end{aligned} \quad (26)$$

Subsequently, equation (26) can be rewritten, as a function of the full-state tracking error $e = \zeta_d - \zeta \in \mathbb{R}^{12}$, as follows:

$$v = C_v e \quad (27)$$

where

$$C_v = \begin{bmatrix} 0_{6 \times 6} & \begin{bmatrix} k_u & 0 & 0 & \cdots & 0 \\ 0 & k_v & 0 & \cdots & 0 \\ 0 & 0 & k_w & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & k_r \end{bmatrix} \end{bmatrix},$$

$$k_i = \left(\frac{i_d}{\text{sgn}(e_i) \max(|e_i|, \epsilon)} - 1 \right), i = \{u, v, w, p, q, r\}$$

Using equation (27), the kinetic energy can be expressed as follows:

$$E_k = \frac{1}{2} v^T M_v v = \frac{1}{2} e^T Q_{E_k} e \quad (28)$$

where $Q_{E_k} = C_v^T M_v C_v$, $M_v = \begin{bmatrix} M_{v_1} & 0 \\ 0 & M_{v_2} \end{bmatrix}$,

$$M_{v_1} = \begin{bmatrix} m_r & 0 & 0 \\ 0 & m_r & 0 \\ 0 & 0 & m_r \end{bmatrix}, M_{v_2} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}.$$

As E_k consists only of v as shown in equation (28), in the compatible form of LQT's PI (either in state or control-effort), it now becomes a part of J_T in equation (21) instead of J_E . However, $\mathcal{L} \triangleq E_k$ still holds and thus, the form of PLA as shown in equation (25) does not change.

4.2. Energy from the UUV thrusters E_{f_T}

To determine the energy consumed by thrusters: E_{f_T} , the load torque and thrust force of the j th thruster can be written as follows (Spangelo and Egeland, 1992; Whitcomb and Yoerger, 1999; Antonelli, 2018):

$$\tau_{T,j} = c_\tau \omega_j |\omega_j| \quad (29)$$

$$f_{T,j} = c_f \omega_j |\omega_j| \quad (30)$$

where $j = \{1, \dots, m\}$, $c_\tau = \rho D^5 K_\tau$, $c_f = \rho D^4 K_f$, ρ is the water density, D is the thruster diameter, K_τ , K_f are the torque and thrust coefficients, respectively, and ω is the thruster's angular velocity.

Therefore, the mechanical power of the j th thruster can be represented by

$$p_{T,j} = \tau_{T,j} \omega_j \quad (31)$$

To write the power as a function of $f_{T,j}$, the following relationship between $\tau_{T,j}$ and $f_{T,j}$ can be established.

$$\frac{\tau_{T,j}}{f_{T,j}} = \frac{c_\tau \omega_j |\omega_j|}{c_f \omega_j |\omega_j|} \Rightarrow \tau_{T,j} = c_p f_{T,j} \quad (32)$$

where $c_p = \frac{c_\tau}{c_f}$ is the thruster power constant. By substituting equation (32) into equation (31), thruster power can be presented as follows:

$$p_{T,j}(\omega_j, f_{T,j}) = c_p \omega_j f_{T,j} \quad (33)$$

If ω_j is not directly available from the thruster, it can be approximated as shown in equation (34) based on equation (30) as $f_{T,j}$ is available from the LQT controller output in real-time.

$$\omega_j \approx \text{sgn}(f_{T,j}) \sqrt{\frac{|f_{T,j}|}{c_f}} \quad (34)$$

After establishing the individual thrusters' power term as a function of ω_j and $f_{T,j}$, the total power term and quadratic power term of all thrusters as a function of f_T can be written as follows:

$$p_T = k_p^T f_T \quad (35)$$

$$p_T^T p_T = f_T^T R_p f_T \quad (36)$$

where $k_p = [c_p \omega_1 \dots c_p \omega_8]^T$ and

$$R_p = \begin{bmatrix} c_p^2 \omega_1^2 & & \\ & \ddots & \\ & & c_p^2 \omega_8^2 \end{bmatrix}.$$

The approximated linear relationship described by equation (33) can be observed in Figure 7. The least square linear regression estimates $c_p = 0.022$ with R^2 score (coefficient of determination) of 0.96 and thus, the regression model fits the data well.

To improve the accuracy of the estimation of the energy consumption across various controllers during the analysis of experimental results, a higher-order polynomial function can be used to predict the power consumption $p_{T,j}$ of the j th thruster.

Therefore, using the publicly available dataset of T200 thruster, operating at 16 V, the polynomial regression on the relationship between thrust (N) and Power (W) will be generated, mapping $f_{T,j}$ to $p_{T,j}$ (T200 Thruster (Accessed 3 November 2023)). As shown in Figure 8, the resulting 6th-order polynomial regression yields, with R^2 score of 0.99, in the form of

$$p_{T,j} = c_0 + c_1 f_{T,j} + c_2 f_{T,j}^2 + c_3 f_{T,j}^3 + c_4 f_{T,j}^4 + c_5 f_{T,j}^5 + c_6 f_{T,j}^6 \quad (37)$$

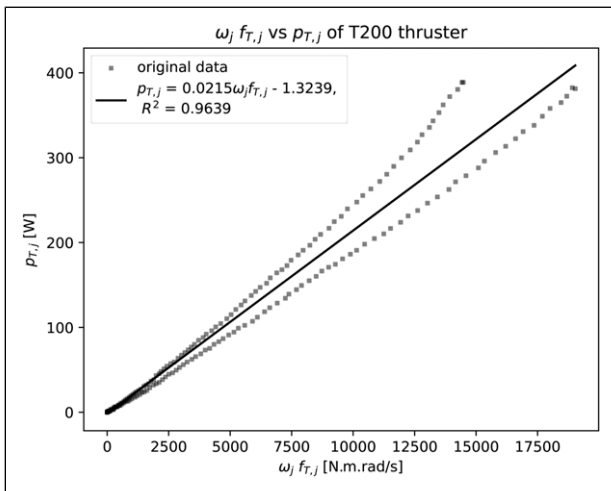


Figure 7. Mapping of $\omega_j f_{T,j}$ to $p_{T,j}$ of the j th T200 thruster.

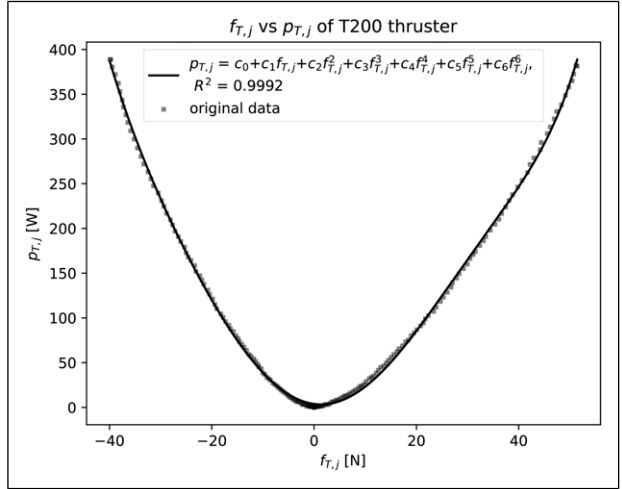


Figure 8. Mapping of $f_{T,j}$ to $p_{T,j}$ of the j th T200 thruster.

Note: Figure 8 shows that T200 thruster generates a slightly higher thrust in the forward direction, compared to the reverse direction.

where $c_0 = 3.8266e + 00$, $c_1 = -7.6590e-01$, $c_2 = 2.8014e-01$, $c_3 = -1.1633e-04$, $c_4 = -9.2529e-05$, $c_5 = -3.1842e-07$, and $c_6 = 2.4910e-08$.

The total energy consumption of the thrusters at each control loop which is running with the control loop duration Δt is calculated using equation (37) as follows:

$$E_{\Delta t} = \sum_{j=1}^m p_{T,j} \Delta t \quad (38)$$

In the following section, E_m (Specifically E_k) in terms of tracking error states e and E_{f_T} in terms of thrust inputs f_T will be embedded into the standard quadratic PI form of a CO-LQT controller and a new non-quadratic form to formulate EO-LQT controllers.

Remark 4.2. Although E_h is not dealt with directly in equation (21), it is part of equation (23). As such, the inclusion of E_m and E_{f_T} in equation (21) takes it into account in the design of the energy-optimal control.

5. Energy-optimal linear quadratic tracking control

Conventionally, J_T and J_E of CO-LQT control as shown in equation (21) are in quadratic form. Therefore, with the energy terms formulated above, the PI for EO-LQT control in quadratic form using E_k as shown in equation (28) and quadratic power term of all thrusters as shown in equation (36) can be defined.

$$J_T = e^T Q e + \frac{1}{2} \overbrace{e^T Q_{E_k} e}^{E_k} = e^T Q_E e \quad (39)$$

$$J_{E,q} = f_T^T R f_T + \underbrace{f_T^T R_p f_T}_{p_T^T p_T} = f_T^T R_E f_T \quad (40)$$

where

$$Q_E = \begin{bmatrix} \begin{bmatrix} q_1 & & \\ & \ddots & \\ & & q_6 \end{bmatrix} & 0_{6 \times 6} \\ 0_{6 \times 6} & \begin{bmatrix} q_7 + \frac{k_u^2 m_r}{2} & & \\ & \ddots & \\ & & q_{12} + \frac{k_r^2 I_{zz}}{2} \end{bmatrix} \end{bmatrix} \quad (41)$$

and

$$R_E = \begin{bmatrix} r_1 + c_p^2 \omega_1^2 & & \\ & \ddots & \\ & & r_m + c_p^2 \omega_m^2 \end{bmatrix}. \quad (42)$$

When using the total power term of all thrusters as shown in equation (35), there will be non-quadratic term involved as follows:

$$J_{E,nq} = f_T^T R f_T + k_p^T f_T \quad (43)$$

In $J_{E,nq}$, the summation of $f_T^T R f_T$ to the total power term ensures the existence of f_T in equation (60) of the non-quadratic Hamiltonian which will be discussed later.

Note: $Q_E = Q_E^T \in \mathbb{R}_{0+}^{n \times n}$, $R_E = R_E^T \in \mathbb{R}_+^{m \times m}$. They are the functions of the system's parameters, states and inputs, whereas the equivalent terms in the CO-LQT controller are constants. Q_E will auto-adjust the tracking penalty weights according to v_d and e_v . R_E will also auto-adjust the energy penalty weights based on the thruster power constant and the respective angular velocity of each thruster. The faster the thruster spins, the higher the energy penalty weight will be. Therefore, the effect of energy penalty weight to achieve energy-optimality can be observed in R_E with a clear physical interpretation.

After formulating the relevant tracking cost J_T and energy costs $J_{E,q}$, $J_{E,nq}$ for the EO-LQT control, the following six controllers with different PIs will be proposed to test the efficacy of those costs individually and in different combinations. The first controller is the conventional LQT controller, named CO-LQT1, which is exactly the same as equation (21) and will be used as a baseline controller for the comparison. **Note:** J_{T_f} and system constraints are the same across all controllers, and

thus only the running cost $\ell(\cdot)$ will be highlighted in the following.

The second controller, named EO-LQT1, is designed from the tracking cost as shown in equation (39) which utilizes the Lagrangian $\mathcal{L}$ in tracking error cost and the standard constant R . This is to verify the effects of the Lagrangian $\mathcal{L}$ in addition to the constant Q .

$$\ell(\cdot)_{\text{EO-LQT1}} = \int_{t_0}^{t_f} [e^T Q_E e + f_T^T R f_T] dt \quad (44)$$

The third controller, named EO-LQT2, is formulated with the standard constant tracking penalty weight Q and the energy penalty weight with quadratic power term as shown in equation (40). Due to the auto-adjusted R_E , this controller is expected to save energy consumption substantially.

$$\ell(\cdot)_{\text{EO-LQT2}} = \int_{t_0}^{t_f} [e^T Q e + f_T^T R_E f_T] dt \quad (45)$$

In the fourth controller, named EO-LQT3, the standard constant weight Q and R and the total power term in non-quadratic form as shown in equation (43) are used. As this controller involves non-quadratic term, the standard LQT equations cannot be used to find the energy-optimal control and thus, the detailed derivation will be presented in the next section.

$$\ell(\cdot)_{\text{EO-LQT3}} = \int_{t_0}^{t_f} [e^T Q e + f_T^T R f_T + k_p^T f_T] dt \quad (46)$$

The fifth controller, named EO-LQT4, is the combination of EO-LQT2 and EO-LQT3. This controller is designed with the standard constant tracking penalty weight Q and the quadratic and non-quadratic power terms. It is also expected to minimize energy consumption with a similar performance as EO-LQT3.

$$\ell(\cdot)_{\text{EO-LQT4}} = \int_{t_0}^{t_f} [e^T Q e + f_T^T R_E f_T + k_p^T f_T] dt \quad (47)$$

While energy-optimality is an important consideration, achieving a proper fish net-pen visual inspection requires a balanced trade-off between energy-optimality and trajectory tracking. The quadratic power term in EO-LQT2 and EO-LQT4 is expected to minimize energy consumption substantially and, at the same time, to have relatively larger tracking errors. However, another combination of the Lagrangian $\mathcal{L}$ and the non-quadratic power term is expected to achieve relatively good performance in both trajectory tracking and energy-optimality. Therefore, the sixth controller, named EO-LQT5, is proposed as follows:

$$\ell(\cdot)_{\text{EO-LQT5}} = \int_{t_0}^{t_f} [e^T Q_E e + f_T^T R f_T + k_p^T f_T] dt \quad (48)$$

As mentioned above, the six controllers are proposed with the minimally possible combinations of the formulated energy/power terms to investigate the trajectory tracking and energy-optimality performance.

So far, the fish net-pen visual inspection task is formulated into energy-optimal finite-horizon optimization problem. Although the constraints on the input cannot be directly imposed on the optimization problem, it will be dealt at the controller output to represent energy-optimal finite-horizon constrained optimization problem. Table 1 summarizes the trajectory tracking cost and energy cost of the six LQT controllers. Now, after the discussion of LQT and EO-LQT control problems, the detailed discussion on the necessary and sufficient condition for optimality will be discussed in the following section.

6. Hamilton-Jacobi-Bellman equation: Necessary and sufficient condition for optimality

To obtain the optimal control, there exists a necessary and sufficient condition for optimality known as the Hamilton-Jacobi-Bellman (HJB) equation. As mentioned earlier, some of the proposed controllers have non-quadratic terms in PI and thus, there will be two separate derivations, one for purely quadratic PI and another for the PI with quadratic and non-quadratic terms. **Note:** The Q and R in the following derivation can be assigned according to J_T and J_E as shown in Table 1. For instance, in EO-LQT1: $Q = Q_E$ and $R = R$. In EO-LQT2: $Q = Q$ and $R = R_E$.

6.1. Quadratic HJB and solution of the optimal control f_T^*

Firstly, quadratic Hamiltonian $\mathcal{H}$ can be defined as follows:

$$\mathcal{H} \triangleq e^T Q e + f_T^T R f_T + \frac{\partial J^*}{\partial \zeta} [A(\zeta)\zeta + B f_T]$$

Therefore, the HJB equation based on $\mathcal{H}$ becomes

$$0 = \min_{f_T(\cdot)} \left[\mathcal{H} + \frac{\partial J^*}{\partial t} \right] \quad (49)$$

Unlike the infinite linear quadratic tracking problem, the optimal cost-to-go function $J^*(\zeta, t)$ consists of the state and time components (Tedrake, 2023). One possible definition for $J^*(\zeta, t)$ can be

$$J^*(\zeta, t) := \zeta^T S \zeta + 2\zeta^T s + s_0, \quad S = S^T > 0 \quad (50)$$

where $S \in \mathbb{R}^{n \times n}$, $s \in \mathbb{R}^n$, and $s_0 \in \mathbb{R}$ are the variables to be determined from the resulting HJB equation in the followings.

Take the partial derivative of $J^*(\zeta, t)$ with respect to state ζ and time t , respectively.

$$\begin{aligned} \frac{\partial J^*}{\partial \zeta} &= 2\zeta^T S + 2s^T, \\ \frac{\partial J^*}{\partial t} &= \zeta^T \dot{S} \zeta + 2\zeta^T \dot{s} + \dot{s}_0 + G \end{aligned}$$

where

$$G = 2\zeta^T S A(\zeta)\zeta + 2\zeta^T S B f_T + 2\zeta^T A(\zeta)^T s + 2f_T^T B^T s$$

(Note: The term G is not considered in Tedrake (2023), and thus it is important to note that the resulting matrix differential equations are not the same. Especially, for the derivation based on the non-quadratic term, the matrix differential equations have additional terms.)

Setting the differentiation of equation (49) with respect to f_T to be zero, we have

$$2f_T^T R + 4\zeta^T S B + 4s^T B = 0$$

Therefore, the ideal optimal control without thrust saturation is

$$f_{T, \text{unsat}}^* = -2R^{-1}B^T[S\zeta + s] = -2K_1[S\zeta + s] \quad (51)$$

where $K_1 = R^{-1}B^T$.

Table 1. LQT controller types.

Controller	J_T	J_E	Description
CO-LQT1	$e^T Q e$	$f_T^T R f_T$	: Conventional LQT controller
EO-LQT1	$e^T Q_E e$	$f_T^T R f_T$	: LQT controller with the Lagrangian $\mathcal{L}$ embedded in Q
EO-LQT2	$e^T Q e$	$f_T^T R_E f_T$	: LQT controller with quadratic power term embedded in R
EO-LQT3	$e^T Q e$	$f_T^T R f_T + k_p^T f_T$	: Conventional LQT controller with the non-quadratic thruster power term
EO-LQT4	$e^T Q e$	$f_T^T R_E f_T + k_p^T f_T$	: LQT controller with quadratic power term embedded in R and the non-quadratic thruster power term
EO-LQT5	$e^T Q_E e$	$f_T^T R f_T + k_p^T f_T$	: LQT controller with the Lagrangian $\mathcal{L}$ embedded in Q and the non-quadratic thruster power term

Note: For the actual implementation, using the thrust saturation function represented by equation (22), it becomes

$$f_T^* = \text{sat}(f_{T,\text{unsat}}^*) \quad (52)$$

With equation (51), the HJB equation as shown in equation (49) can be solved.

$$\begin{aligned} \mathcal{H}|_{f_T^*} &= \zeta^T [Q + 2SA(\zeta)]\zeta + \zeta_d^T Q\zeta_d - 2\zeta^T Q\zeta_d \\ &\quad + 2s^T A(\zeta)\zeta \\ \left. \frac{\partial J^*}{\partial t} \right|_{f_T^*} &= \zeta^T [\dot{S} + 2SA(\zeta) - 4SK_2S]\zeta + \dot{s}_0 - 4s^T K_2s + \\ &\quad 2\zeta^T [\dot{s} + A(\zeta)^T s - 4SK_2s] \end{aligned}$$

where $K_2 = BK_1$.

Hence, equation (49) becomes

$$\begin{aligned} &\zeta^T [\dot{S} + 4SA(\zeta) + Q - 4SK_2S]\zeta + \\ &2\zeta^T [\dot{s} + A(\zeta)^T s - 4SK_2s - Q\zeta_d] + \\ &\dot{s}_0 + \zeta_d^T Q\zeta_d - 4s^T K_2s + 2s^T A(\zeta)\zeta = 0 \end{aligned} \quad (53)$$

The resulting equation (53) implies that

$$\dot{S} + 4SA(\zeta) + Q - 4SK_2S = 0 \quad (54)$$

$$\dot{s} + [A(\zeta)^T - 4SK_2]s - Q\zeta_d = 0 \quad (55)$$

$$\dot{s}_0 + \zeta_d^T Q\zeta_d - 4s^T K_2s + 2s^T A(\zeta)\zeta = 0 \quad (56)$$

Using the final conditions,

$$S_f = Q_f \quad (57)$$

$$s_f = -Q_f \zeta_{d,f} \quad (58)$$

$$s_{0,f} = \zeta_{d,f}^T Q_f \zeta_{d,f} \quad (59)$$

the backward integration of equations (54)–(56) will produce S , s , and s_0 which must yield equation (50) as a uniformly positive function. Subsequently, the resulting S and s are used in equation (51), followed by equation (52), to compute f_T^* .

6.2. Non-quadratic HJB and solution of the optimal control $f_{T,\text{sq}}^*$

For the case of $\mathcal{H}_{\text{sq}}$, it involves quadratic and non-quadratic terms,

$$\mathcal{H}_{\text{sq}} = e^T Qe + f_T^T R f_T + k_p^T f_T + \frac{\partial J^*}{\partial \zeta} [A(\zeta)\zeta + Bf_T]$$

Therefore, the HJB equation based on $\mathcal{H}_{\text{sq}}$ becomes

$$0 = \min_{f_{T,\text{sq}}(\cdot)} \left[\mathcal{H}_{\text{sq}} + \frac{\partial J^*}{\partial t} \right] \quad (60)$$

Following the same process for $\mathcal{H}_{\text{sq}}$, the ideal optimal control without thrust saturation can also be derived.

$$f_{T,\text{sq},\text{unsat}}^* = -2K_1[S\zeta + s] - \frac{1}{2}R^{-1}k_p \quad (61)$$

Note: For the actual implementation, the feasible optimal control with thrust saturation is

$$f_{T,\text{sq}}^* = \text{sat}(f_{T,\text{sq},\text{unsat}}^*) \quad (62)$$

Substituting the optimal control expressed in equation (61) into the non-quadratic HJB as shown in equation (60), the resulting differential matrix and vector equations become

$$\dot{S} + 4SA(\zeta) + Q - 4SK_2S = 0 \quad (63)$$

$$\dot{s} + [A(\zeta)^T - 4SK_2]s - Q\zeta_d - \frac{1}{2}SBR^{-1}k_p = 0 \quad (64)$$

$$\dot{s}_0 + \zeta_d^T Q\zeta_d - 4s^T K_2s + k_\zeta + k_R = 0 \quad (65)$$

where $k_\zeta = 2s^T A(\zeta)\zeta - k_p^T K_1 S\zeta$, $k_R = k_p^T [(1/4)R^{-1} R^{-1} k_p - 2K_1s - (1/2)R^{-1}k_p]$. Using the same final conditions associated only with the desired trajectory, the backward integration of equations (63)–(65) can be performed. Subsequently, the resulting S and s are used in equation (61), followed by equation (62), to compute $f_{T,\text{sq}}^*$.

6.3. Uniform positiveness condition for $J^*(\zeta, t)$

To ensure that $J^*(\zeta, t)$ is uniformly positive, the following conditions are needed (Tedrake, 2023).

Firstly, as shown in equation (50),

$$S = S^T \in \mathbb{R}_+^{n \times n}$$

Secondly,

$$J^*(\zeta_{\min}, t) > 0 \Rightarrow s^T S^{-1} s < s_0$$

Although the resulting S and s ensure that f_T^* from equation (52) and $f_{T,\text{sq}}^*$ from equation (62) are indeed optimal controller outputs, the backward integration over time for a long horizon (e.g., 2 hours) in each control loop is not practical for the trajectory tracking task which is conducted in real-time. Therefore, the backward integration over a short horizon for each control loop is used. In other

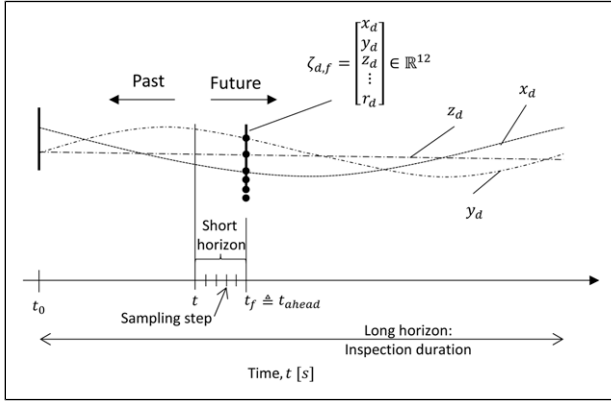


Figure 9. Extracting the desired trajectory $\zeta_{d,f}$ at $t_f \hat{=} t_{ahead}$. x_d , y_d , and z_d are illustrated and so are other desired trajectory states with circular dots.

words, t_f will become the trajectory time ahead of the current trajectory time t . Subsequently, Q_f can be set in advance and the final conditions (equations (57), (58) and (59)) can be, in real-time, updated with the extracted desired trajectory $\zeta_{d,f}$ at a specific time ahead $t_{ahead}(\hat{=} t_f)$ as shown in Figures 6 and 9.

7. High-fidelity simulations and experiments

Prior to the field experiment, conducting high-fidelity simulation is crucial to ensure the feasibility of the experiment in the estimated operational environment and to validate the controller's performance with the minimum confounding factor. Moreover, in the simulation engine, a specific disturbance, such as underwater current disturbance, can be consistently created for each controller's performance testing. Robot Operating System (ROS) with Gazebo Physics Engine is used to conduct the high-fidelity simulation UUV Simulator (Manhães et al., 2016 (Accessed 20 November 2023)). However, for the pool experiments, the underwater current disturbance is not considered due to

Table 2. Number of simulations and experiments conducted.

	FNVI Sim.	Pool Sim.	Pool Exp.
FNVI	ROS + Gazebo	ROS + Gazebo	ROS
No. of Controllers	6	6	6
No. of Disturbance ^a	3	1	1
No. of Sim. and Expt.	18	6	6
Total No. of Sim. and Expt.	30		

^aThere are three types of disturbances in total. Number 3 refers to 0.0 m/s, 0.5 m/s, and 0.9 m/s. Number 1 refers to only 0.0 m/s.

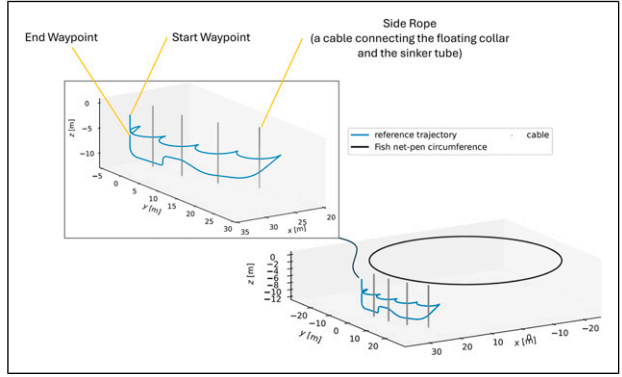


Figure 10. Illustration of the FNVI trajectory around the floating flexible fish net-pen from the operational perspective.

the lack of a suitable and reliable disturbance generator. The number of simulations and experiments conducted is reported in Table 2. In this work for the FNVI simulation, a complicated trajectory, which consists of a sudden change in twist, a steep and gradual change in heave, and varying and constant surge, is utilized as shown in Figure 10. Such a trajectory reflects a more realistic real-world operation where the areas of interest (waypoints) are not around a simple helical trajectory. These waypoints are determined based on the condition of the fish net-pens by the operation team.

However, for the field trials in the open ocean, there are a few barriers (mainly budget constraint, expensive localization sensors, industrial-grade UUV, logistics and safety concerns) to deploying BlueROV2 Heavy Configuration in the actual fish farm at the current stage. Therefore, pool experiments are conducted, along with their respective simulations in ROS and Gazebo Physics Engine. Even in the case of pool experiments, the major challenge is the UUV localization, which will be detailed and addressed in the next subsection. As a result of this localization issue, a planar trajectory is utilized for the pool experiment; however, to consider realistic maneuvering capabilities required for the actual field trials in the future, the pool-experiment trajectory is designed with a sudden change in twist, curvature, and straight lines.

Hardware & Software specifications, controller tuning parameters, the numerical solvers for matrix differential equations, and ROS launch processes used in this work are summarized in Appendix A. The main difference between simulation and experiment is the availability of comprehensive state-estimation. In simulation, via Gazebo, the full state-estimation is readily accessible, whereas in experimental settings, the full state-estimation is one of the tedious tasks, and it can directly impact the performance of the controller and introduce confounding factors that compromise the experimental results. Therefore, the experimental setup for state-estimation will be detailed in the following subsection.

7.1. Experimental setup

For the experiment, the main challenge is the affordability of a reliable localization system for state-estimation with the sensor suite (e.g., Ultra-Short Baseline (USBL), Doppler Velocity Logger (DVL), Inertial Navigation System (INS)), which usually costs much more than the BlueROV2 Heavy Configuration. Therefore, the AprilTag detection system, which many academic research institutes use to estimate the UUV's pose (position and orientation), is integrated for the UUV localization (Bauschmann et al., 2023; Jung et al., 2025; Tang et al., 2025). To minimize the computation load on RPi in BlueROV2 Heavy Configuration and to avoid a meticulous installation process of many AprilTags in the pool, the proposed setup consists of only two AprilTags and a ZED 2i camera. This approach eliminates the requirement of accurate measurements among many AprilTags and does not require a particular orientation of the camera as long as both AprilTags can be seen from the ZED 2i camera frame as shown in Figure 11. The experimental pool setup for the UUV localization using AprilTags and the installation of AprilTag on the additional fixture, attached to the UUV, are illustrated in Figure 12.

As H_a^c and H_I^c with their respective rotation matrices (e.g., R_a^c, R_I^c) and position vectors (e.g., r_a^c, r_I^c) with respect to Frame $\{c\}$ are generated by the AprilTag detection system, the homogeneous transformation from Frame $\{a\}$ to Frame $\{I\}$ can be computed in real-time as follows:

$$H_a^I = H_a^c H_I^c \quad (66)$$

where

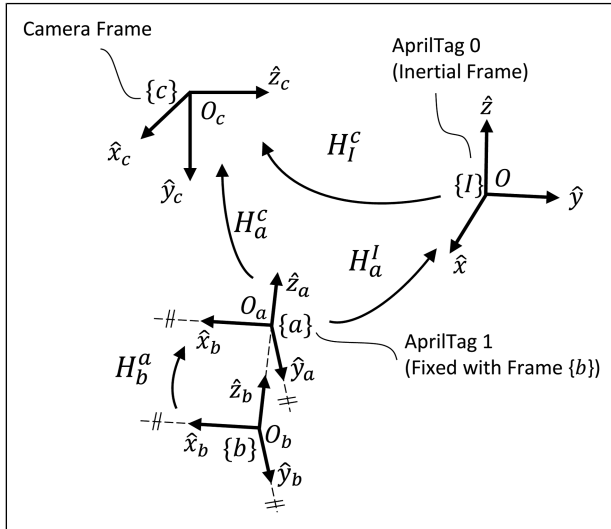


Figure 11. Illustration of the UUV Localization using AprilTags.

$$H_c^I \quad (\neq H_I^{cT}) = \begin{bmatrix} R_I^{cT} & -R_I^{cT} r_I^c \\ 0_{1 \times 3} & 1 \end{bmatrix},$$

$$H_I^c = \begin{bmatrix} R_I^c & r_I^c \\ 0_{1 \times 3} & 1 \end{bmatrix}, \quad H_a^c = \begin{bmatrix} R_a^c & r_a^c \\ 0_{1 \times 3} & 1 \end{bmatrix}.$$

Subsequently, as Frame $\{b\}$ is fixed in the negative $\hat{z}_a$ with the same orientation, H_b^a can be defined accordingly. Therefore, the homogeneous transformation from Frame $\{b\}$ to Frame $\{I\}$ can be computed in real-time as follows:

$$H_b^I = H_a^I H_b^a \quad (67)$$

where

$$H_b^a = \begin{bmatrix} R_b^a & r_b^a \\ 0_{1 \times 3} & 1 \end{bmatrix}$$

Now, as r_b^I and R_b^I are available from equation (67), the UUV's measured pose $\eta_m = [\eta_{1,m} \ \eta_{2,m}]^T$ can be calculated as follows:

$$r_b^I \Rightarrow \eta_{1,m} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad R_b^I \Rightarrow \eta_{2,m} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}. \quad (68)$$

Suppose at time instant i after the time elapse Δt , using the current measured pose $\eta_{m,i} = [\eta_{1,m,i} \ \eta_{2,m,i}]^T$ and the previous measured pose $\eta_{m,i-1} = [\eta_{1,m,i-1} \ \eta_{2,m,i-1}]^T$, the rate of change of position and Euler angles can be computed as follows:

$$\dot{\eta}_m = \frac{\eta_{m,i} - \eta_{m,i-1}}{\Delta t} \quad (69)$$

So far, using the AprilTag detection system, the full-state measurements $\zeta_{I,m} = [\eta_m \ \dot{\eta}_m]^T$ with respect to and expressed in Frame $\{I\}$ are acquired. The next step is the implementation of a Kalman filter (KF) on those measurements to filter out the noise and produce the optimal estimated states.

As the notation conventions, the full states, along with their estimates and the corresponding measured states and their estimates where all of which are represented with respect to and expressed in Frame $\{I\}$ are denoted as follows:

$$\begin{aligned} \zeta_I(\cdot) &= [\eta(\cdot) \ \dot{\eta}(\cdot)]^T \in \mathbb{R}^{12}, \\ \tilde{\zeta}_I(\cdot) &= [\tilde{\eta}(\cdot) \ \tilde{\dot{\eta}}(\cdot)]^T \in \mathbb{R}^{12}, \\ \zeta_{I,m}(\cdot) &= [\eta_m(\cdot) \ \dot{\eta}_m(\cdot)]^T \in \mathbb{R}^{12}, \\ \tilde{\zeta}_{I,m}(\cdot) &= [\tilde{\eta}_m(\cdot) \ \tilde{\dot{\eta}}_m(\cdot)]^T \in \mathbb{R}^{12} \end{aligned}$$

$\tilde{\zeta}_I(i|i)$ is the estimate of $\zeta_I(i)$ using the measurements of $\zeta_{I,m}(i)$, $\zeta_{I,m}(i-1)$,

$\tilde{\zeta}_I(i+1|i)$ is the estimate of $\zeta_I(i+1)$ using the measurements of $\zeta_{I,m}(i)$, $\zeta_{I,m}(i-1)$,

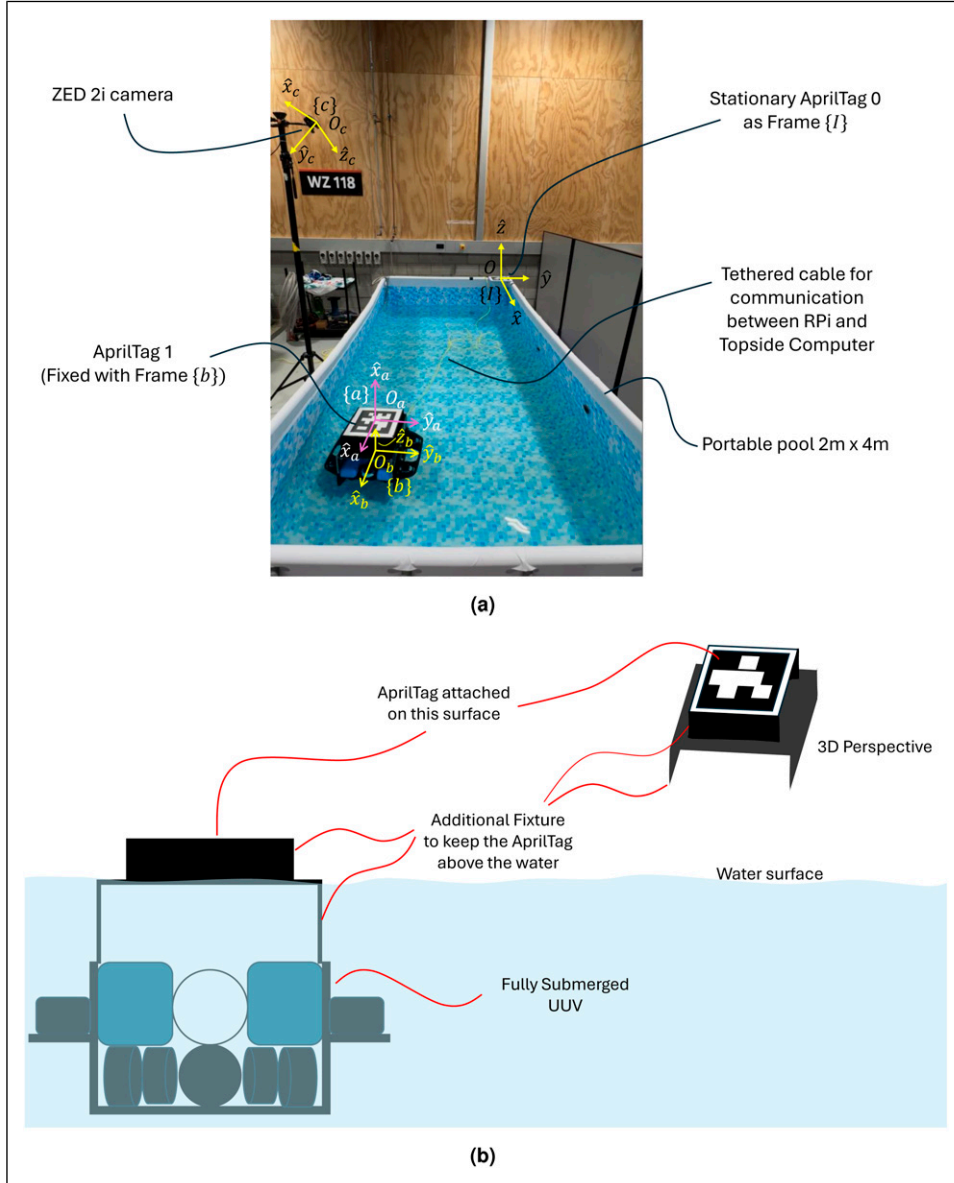


Figure 12. (a) Experimental Pool Setup for the UUV Localization using AprilTags. (b) Installation of AprilTag on the additional fixture, attached to the UUV.

$\tilde{\zeta}_I(i+1|i+1)$ is the estimate of $\zeta_I(i+1)$ using the measurements of $\zeta_{I,m}(i+1)$, $\zeta_{I,m}(i)$, $\zeta_{I,m}(i-1)$, Other parameters will be described in the same fashion.

At the time instant i after a time elapse Δt , the state prediction equation can be described as follows:

$$\tilde{\zeta}_I(i+1|i) = A(i)\tilde{\zeta}_I(i|i) \quad (70)$$

where, $A(i) = \begin{bmatrix} I_{6 \times 6} & I_{6 \times 6} \Delta t \\ 0_{6 \times 6} & I_{6 \times 6} \end{bmatrix}$, $I_{6 \times 6} \in \mathbb{R}^{6 \times 6}$ is the identity matrix.

As all the full states are acquired via the AprilTag detection algorithm and subsequent computation as shown in

equation (69), the measurement prediction equation can be written using $H = I_{12 \times 12}$ as follows:

$$\tilde{\zeta}_{I,m}(i+1|i) = H\tilde{\zeta}_I(i+1|i) \quad (71)$$

The measurement residual can be obtained by the difference between the actual measurement, $\zeta_{I,m}(i+1)$ and measurement prediction as follows:

$$e(i+1) = \zeta_{I,m}(i+1) - \tilde{\zeta}_{I,m}(i+1|i) \quad (72)$$

Given $Q = Q^T \in \mathbb{R}_+^{12 \times 12}$ and the initial $0 < P(\cdot) = P(\cdot)^T < \epsilon I \in \mathbb{R}_+^{12 \times 12}$, $0 < \epsilon \ll 1$, the state prediction covariance can be described by

$$P(i+1|i) = A(i)P(i|i)A(i)^T + Q \quad (73)$$

Given $R = R^T \in \mathbb{R}_+^{12 \times 12}$, the measurement prediction covariance can be defined by

$$S(i+1) = HP(i+1|i)H^T + R \quad (74)$$

The KF gain can be obtained by

$$K(i+1) = P(i+1|i)H^T S(i+1)^{-1} \quad (75)$$

The KF estimated state, that utilizes the latest measurement, can be finally achieved via

$$\tilde{\zeta}_I(i+1|i+1) = \tilde{\zeta}_I(i+1|i) + K(i+1)e(i+1) \quad (76)$$

Before the next KF loop, the state prediction covariance needs to be updated.

$$P(i+1|i+1) = P(i+1|i) - K(i+1)S(i+1)K(i+1)^T \quad (77)$$

As the full-state estimates from KF are with respect to and expressed in Frame $\{I\}$ as shown in equation (76) but the UUV's dynamic model utilizes the twist which is in Frame $\{b\}$ as shown in equation (2), the rates of change of position and Euler angles (the output of KF) need to be converted into the UUV's linear and angular velocities. From equation (76), the KF output: $\tilde{\zeta}_I(i+1|i+1) = [\tilde{\eta}(i+1|i+1) \tilde{\dot{\eta}}(i+1|i+1)]^T$ will be denoted as the standard UUV nomenclature: $\zeta_I = [\eta \dot{\eta}]^T$ for ease of readability. Recall equation (3) and using $J_I^b(\eta)$, the UUV's twist can be computed, acquiring all the full states $\zeta = [\eta \dot{\eta}]^T$ required for the proposed LPV system as shown in equation (18).

8. Results and discussions

In this section, there are two main subsections: firstly, FNVI Simulations, and secondly, Pool Experiments and Simulations. In each subsection, trajectory tracking, pose (position and orientation) tracking, and energy consumption will be discussed. Subsequently, a summary of the coherent findings between FNVI Simulations and Pool Experiments and Simulations is provided.

For the error comparison, mean-absolute-error (MAE) as shown in equation (78) and MAE ratio as shown in equation (79) will be used.

$$\text{MAE} = \frac{\sum_{i=0}^N |e_i|}{N} \quad (78)$$

where N is the number of data recorded, e_i represents i th element of a specific error type (e.g., $e_{x,i}$). For MAE of error norm, e_i represents i th element of a specific norm error type (e.g., $e_{\eta,i} = \sqrt{e_{x,i}^2 + e_{y,i}^2 + e_{z,i}^2}$).

$$\text{MAE_ratio}_j = \frac{(\text{MAE}_1 - \text{MAE}_j)}{\text{MAE}_1} \quad (79)$$

where the subscript 1 represents the baseline controller and the subscript $j = \{2, 3, 4, \dots\}$ represents other controllers to be compared with the baseline controller. In the same fashion, equation (79) is used to compare the energy consumption (equation (38)) of EO-LQTs with that of CO-LQT. Therefore, equation (79) can be used to quantitatively indicate the tracking error and energy consumption comparison between CO-LQT and EO-LQTs. The negative “−” sign of MAE ratio and $E_{\Delta t}$ ratio represents EO-LQTs' more tracking error and more energy consumption, respectively, compared to those of CO-LQT.

As shown in Table 1, J_T and J_E in EO-LQT control consist of the state-varying, input-varying matrix and vector in real-time, unlike CO-LQT control. Therefore, it is tedious to provide how each EO-LQT controller achieves the quantitative measures (MAE, Wh) via the tuning parameters, although it is expected to minimize energy consumption even at the expense of large tracking errors. For instance, CO-LQT1 uses constant Q and R and thus, it can be inferred that these specific values of Q and R result in particular MAE values. However, this inference method does not work for EO-LQT controllers and thus, only MAE ratio comparisons and energy consumptions are directly compared without the directly associated explanation to the tuning parameters in J_T and J_E . The recorded simulation videos are available at this hyperlink: https://autuni-my.sharepoint.com/:f/g/personal/jyb1376_autuni_ac_nz/IgD1BjFhNRRQIIDqYHVQZkJAWmWaM-sAPmGciyF3XaQ1sM.

8.1. FNVI simulations

For FNVI simulation, there are 18 simulations, conducted in total as shown in Table 2. Although the controllers are designed to track a trajectory with 12 reference states (including both pose and twist), only a subset of results is presented for clarity. Specifically, the 3D trajectory tracking performance, the tracking norms for position and orientation, and the energy consumption are reported. This information is sufficient enough to report the trajectory tracking performance and energy-optimality of the proposed controllers, avoiding an overabundance of plots that could overwhelm the overall presentation.

8.1.1. Trajectory tracking: FNVI simulations. The 3D trajectory tracking performance of all controllers are reported in Figure 13. The underwater current disturbance speed direction is illustrated with a black arrow. For a better illustration of 3D trajectory tracking, the 3D view and top views are presented side-by-side.

Generally, it can be observed that the tracking performance of EO-LQT2 and EO-LQT4 deteriorates

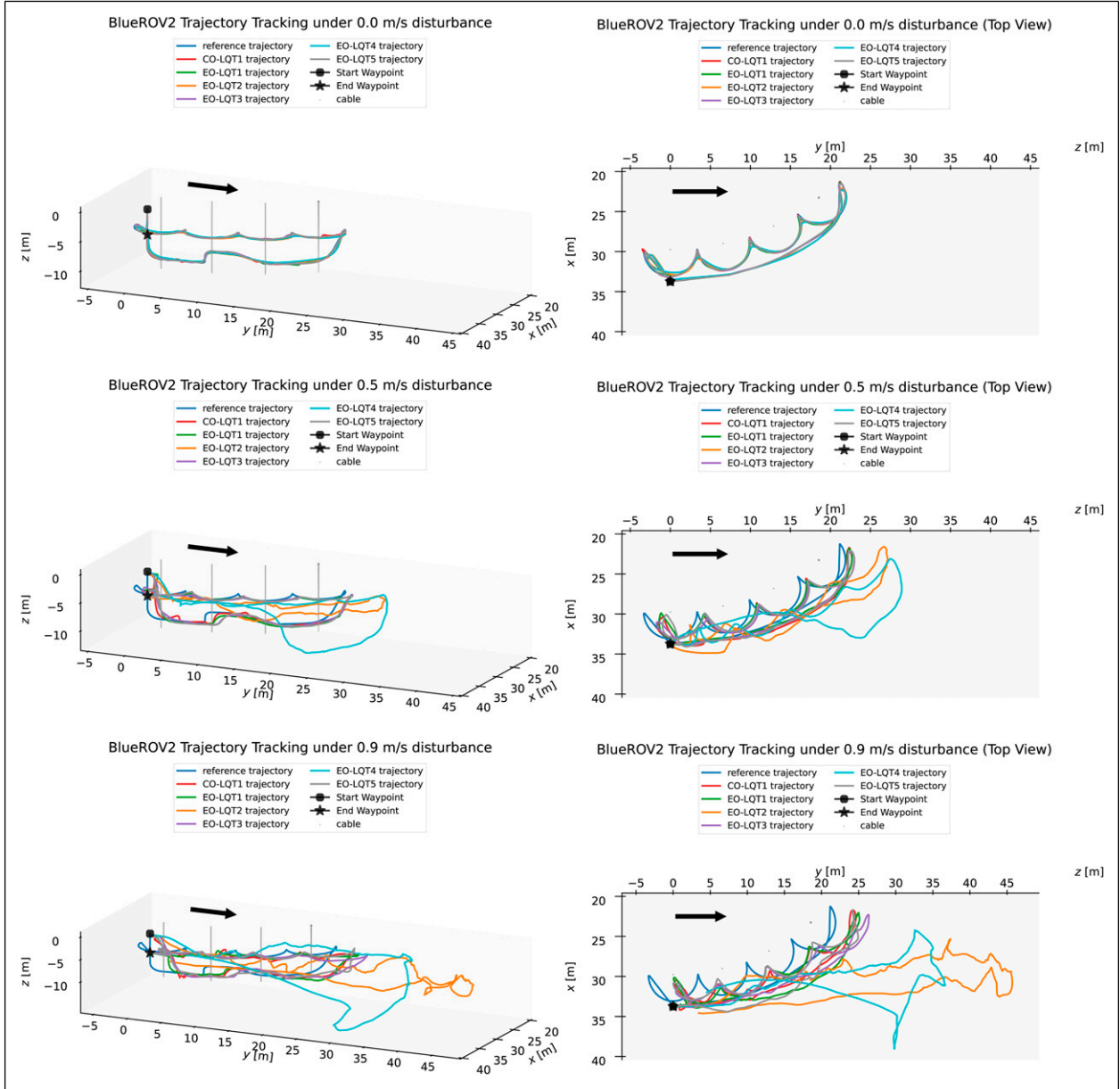


Figure 13. [FNVI Simulation] Trajectory tracking comparison under simulated underwater current speeds.

substantially with increasing underwater current speeds, but all other controllers track relatively better than EO-LQT2 and EO-LQT4. In addition to the overall 3D trajectory tracking performance, it is important to quantitatively assess the tracking norms of position and orientation. These metrics will be reported in the following sections.

8.1.2. Pose tracking: FNVI simulations. For the FNVI task, pose (position and orientation) tracking is more crucial than twist (linear and angular velocities) tracking to capture a stable visual feedback at the desired trajectory. As shown in Table 3 and Figure 14, generally all EO-LQTs have larger position and

orientation tracking error norm. However, under the increasing underwater current disturbance speeds, the pose tracking error norms of all controllers generally increase as expected. Except for EO-LQT2 and EO-LQT4, the MAE % of e_{η_1} of EO-LQT1, EO-LQT3 and EO-LQT5 reduces, when comparing the results under 0.0 m/s and 0.9 m/s disturbance speeds. In other words, although the position tracking error norms of all controllers increase under the increasing disturbance speeds, the position tracking performance difference between CO-LQT and EO-LQT1/3/5 reduces. It is important to take note that this phenomenon is not observed under 0.5 m/s disturbance speed except for EO-LQT5. On the other hand, the MAE % of e_{η_2} of

Table 3. [FNVI Simulation] Pose tracking error norm and energy consumption comparison taking CO-LQT1 as the baseline.

		e_{η_1} norm (m)			e_{η_2} norm (deg)			Energy consumption (Wh)		
		MAE (m)	MAE ratio	%	MAE (deg)	MAE ratio	%	$E_{\Delta t}$ (Wh)	$E_{\Delta t}$ Ratio	%
0.0 m/s	CO-LQT1	0.281	1.	0.0	41.116	1.	0.0	4.876	1.	0.0
	EO-LQT1	0.325	−0.159	15.9	35.524	0.136	−13.6	4.578	0.061	−6.1
	EO-LQT2	0.836	−1.978	197.8	43.848	−0.066	6.6	3.499	0.282	−28.2
	EO-LQT3	0.38	−0.351	35.1	47.661	−0.159	15.9	4.582	0.06	−6.0
	EO-LQT4	0.904	−2.217	221.7	45.342	−0.103	10.3	3.487	0.285	−28.5
0.5 m/s	EO-LQT5	0.448	−0.595	59.5	46.061	−0.12	12	4.567	0.063	−6.3
	CO-LQT1	0.997	1.	0.0	57.077	1.	0.0	6.96	1.	0.0
	EO-LQT1	1.044	−0.047	4.7	62.651	−0.098	9.8	6.931	0.004	−0.4
	EO-LQT2	5.072	−4.088	408.8	69.744	−0.222	22.2	5.135	0.262	−26.2
	EO-LQT3	1.379	−0.383	38.3	65.466	−0.147	14.7	6.372	0.085	−8.5
0.9 m/s	EO-LQT4	7.421	−6.444	644.4	82.834	−0.451	45.1	4.642	0.333	−33.3
	EO-LQT5	1.473	−0.478	47.8	70.871	−0.242	24.2	6.661	0.043	−4.3
	CO-LQT1	2.585	1.	0.0	60.95	1.	0.0	21.173	1.	0.0
	EO-LQT1	2.655	−0.027	2.7	70.975	−0.164	16.4	23.054	−0.089	8.9
	EO-LQT2	18.044	−5.981	598.1	90.541	−0.485	48.5	12.493	0.41	−41.0
	EO-LQT3	3.02	−0.169	16.9	84.359	−0.384	38.4	15.563	0.265	−26.5
	EO-LQT4	16.243	−5.284	528.4	118.105	−0.938	93.8	7.914	0.626	−62.6
	EO-LQT5	2.832	−0.096	9.6	64.989	−0.066	6.6	13.328	0.371	−37.1

Note. The negative “−” sign of MAE ratio and $E_{\Delta t}$ ratio represents EO-LQTs’ more tracking error and more energy consumption, respectively, compared to those of CO-LQT. For ease of interpretation, the sign of % is opposite to the sign of ratio.

all EO-LQTs except EO-LQT5 increases under the increasing disturbance speeds. This statement is relevant to EO-LQT5 except under 0.5 m/s disturbance speed. Therefore, generally, the pose tracking performance of EO-LQT3 and EO-LQT5 is better than that of other EO-LQTs.

Although the MAE % provides a quantitative measure for comparing the relative pose tracking performance among controllers, the selection of a controller should not rely solely on this metric. Instead, practical considerations—such as whether a given MAE value is acceptable for the intended application—should also guide the decision-making process. For instance under 0.9 m/s disturbance speed for a particular FNVI operation, if the MAE value of 2.832 m (EO-LQT5) is acceptable, compared to that of 2.585 m (CO-LQT1), the selection of EO-LQT5 is a better choice, supposed if it saves more energy. Hence, another evaluation metric for controller selection is energy consumption.

8.1.3. Energy consumption: FNVI simulations. The energy consumptions of the controllers under simulated underwater current speeds are plotted in Figure 15. Their respective $E_{\Delta t}$ Ratio and energy saving % are reported in Table 3. Generally as expected, all EO-LQTs save more energy than CO-LQT in an ideal scenario without any underwater current disturbance speed. EO-LQT1’s energy saving performance deteriorates under increasing disturbance speeds. EO-LQT2 and EO-LQT4 save more energy substantially under increasing disturbance speeds at the expense of very large pose tracking error as shown

in Figure 13. Both EO-LQT3 and EO-LQT5 save energy regardless of the increasing disturbance speeds. Based on the % metric of e_{η_1} MAE, e_{η_2} MAE, and $E_{\Delta t}$ Ratio, it can be concluded that both EO-LQT3 and EO-LQT5 are the most energy-optimal trajectory tracking controllers, suitable for operations in the presence of external disturbances. Furthermore, EO-LQT3 is more appropriate for scenarios involving mild disturbance speeds (e.g., 0.5 m/s), whereas EO-LQT5 demonstrates superior overall performance under higher disturbance conditions (e.g., 0.9 m/s).

8.2. Pool experiments and simulations

Using the vision-based state-estimation, detailed in Subsection 7.1, all proposed controller are tested on the actual hardware of BlueROV2 Heavy Configuration in the pool as shown in Figure 12. Similarly, simulations are conducted with the same trajectory as the pool experiment.

In a similar fashion to FNVI simulations, the 2D trajectory tracking performance, the tracking norms for position and orientation, and the energy consumption of pool experiments and simulations are reported in this subsection. In addition, it is important to note that there are a few factors that can cause model uncertainties in addition to the unmodeled dynamics in the actual experiments. Some important factors are listed as follows:

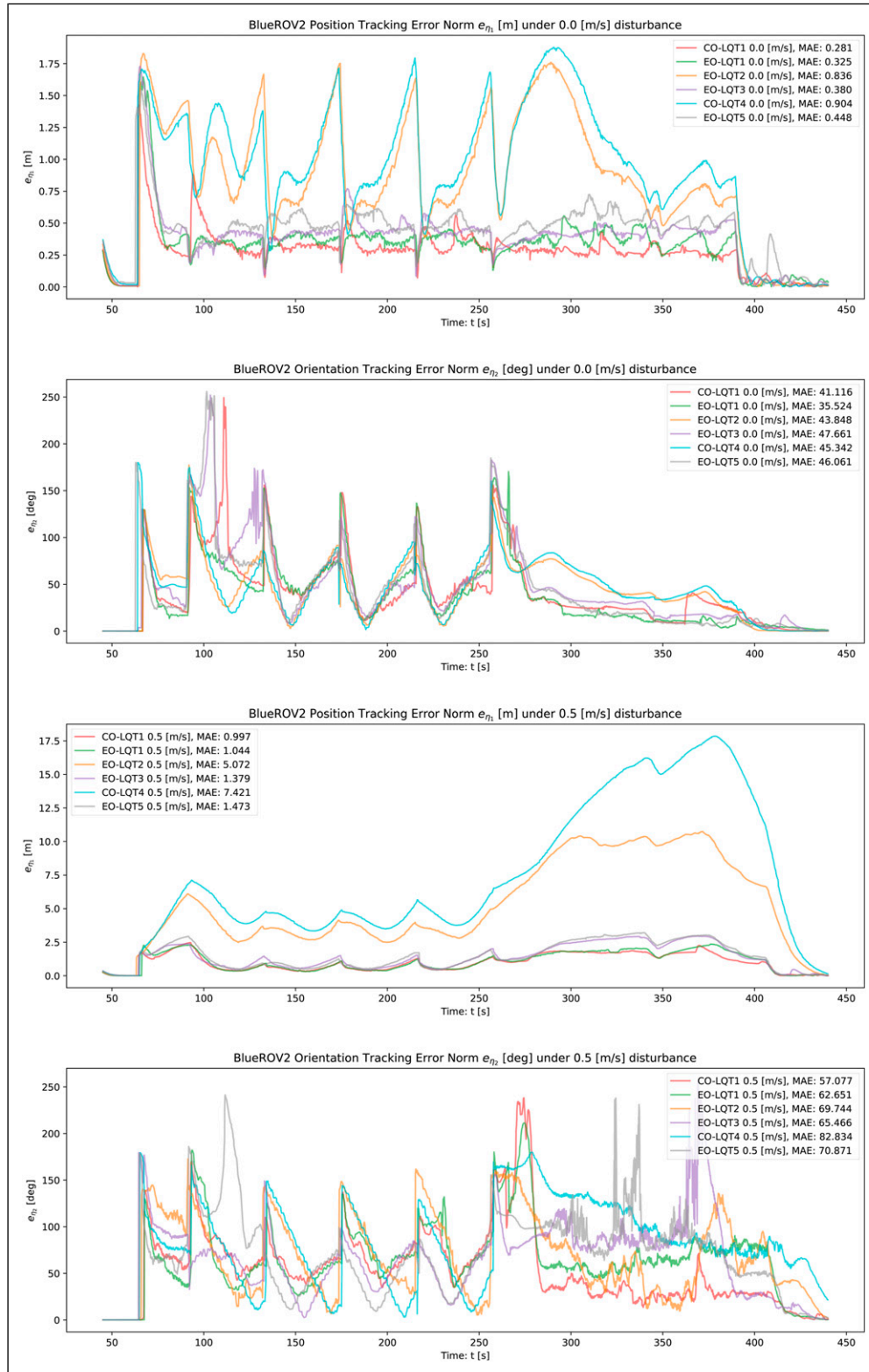


Figure 14. [FNVI Simulation] Pose tracking error norm comparison under simulated underwater current speeds.

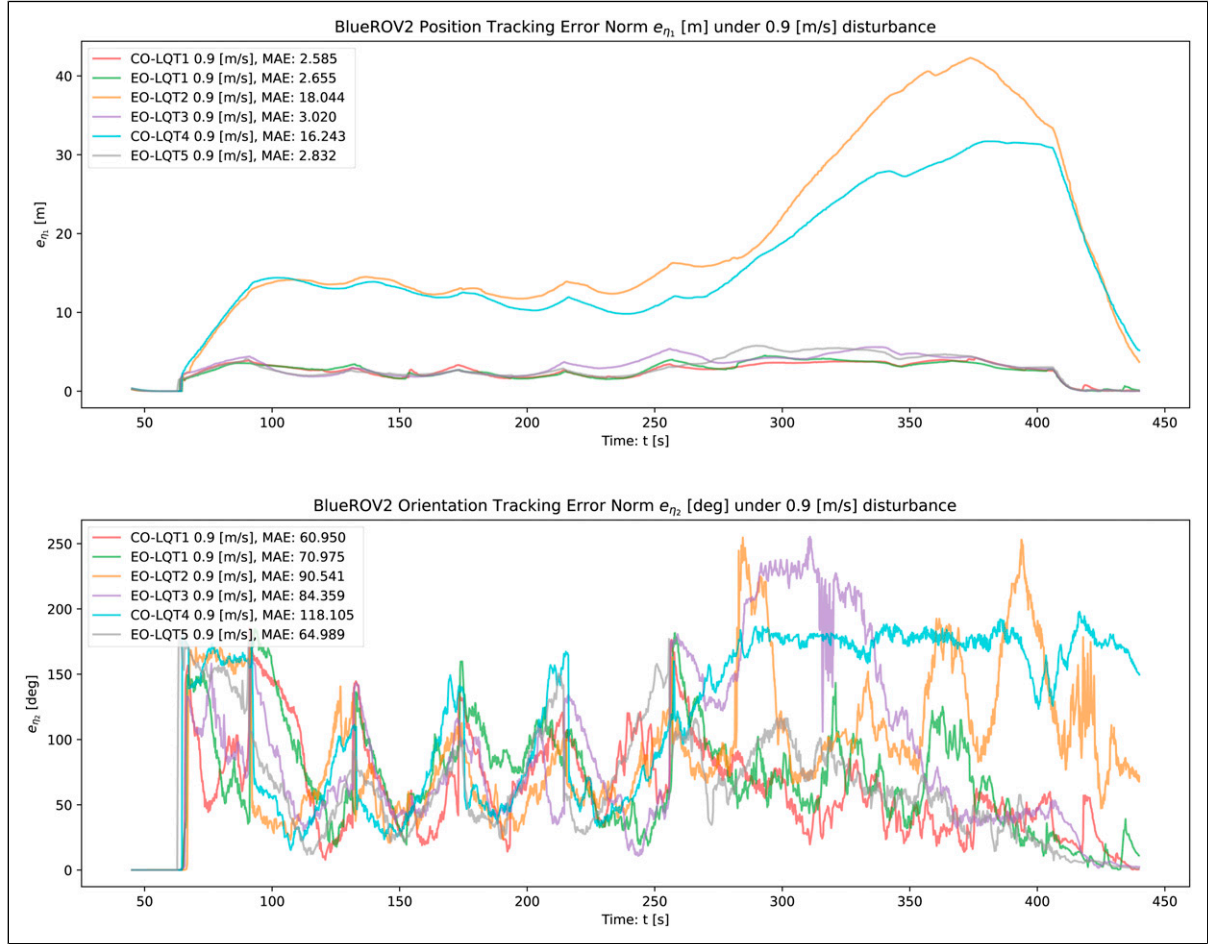


Figure 14. Continued.

The UUV is mounted with the additional corrugated plastic sheets to attach AprilTag for the vision-based state-estimation system. Although the weight can be considered negligible, the corrugated plastic sheets sealed with glue may introduce additional buoyancy, and it could potentially increase hydrodynamics parameters, especially for yaw motion.

The additional weight, tension, and drag introduced by the tethered cable could potentially affect the overall dynamics of the UUV.

Due to the requirement of the vision-based state-estimation system and the slight positive buoyancy, the UUV operates around the water surface level, although its main body is fully submerged underwater. Therefore, the near-surface operation could potentially affect the overall dynamics of the UUV.

8.2.1. Trajectory tracking: Pool experiments and simulations. The 2D trajectory tracking performance of all controllers is reported in Figure 16. In both pool experiments and simulations, inferior tracking performance is observed

for EO-LQT2 and EO-LQT4 compared to the other controllers. While the differences in trajectory tracking performance among other LQT controllers are not particularly substantial in the pool simulations, they are notably more evident in the pool experiments.

There are a few important observations between pool experiments and simulations. Firstly, unlike in simulations, it is challenging to maintain the UUV at a fixed position (Start Waypoint) in the pool prior to the activation of the controllers, resulting in multiple start waypoints as shown in Figure 16. Secondly, due to the aforementioned start waypoints drifted to the left (in the negative direction of x axis), the resulting UUV's trajectory is also shifted to the left. The quantitative performance measures will be detailed in the following.

8.2.2. Pose tracking: Pool experiments and simulations. The pose tracking error norms of pool experiments and simulations are plotted in Figure 17. The pose tracking error norms and its ratio are reported in Table 4. Due to the drift to the left (in the negative direction of x axis), there

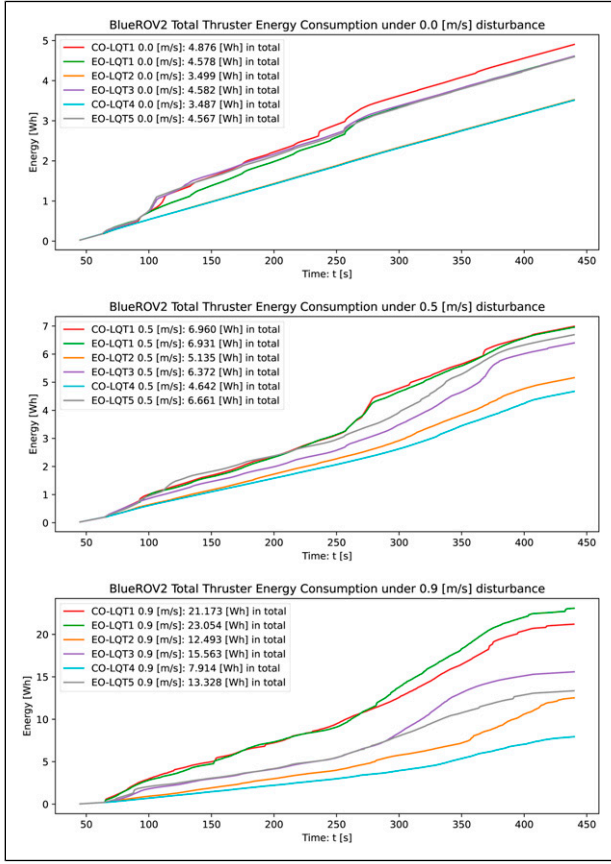


Figure 15. [FNVl Simulation] Energy consumption comparison tested under simulated underwater current speeds.

are substantial difference in position tracking error norm at the start and the overall position tracking error norm profile is shifted to the left. Most likely due to model uncertainties and unmodeled dynamics mentioned earlier and experimental setup for the start waypoint, the respective e_{η_1} norms of pool experiments and simulations are substantially different. However, the overall profiles of MAE % of e_{η_1} norm of both pool experiments and simulations are very similar.

Similar to the position tracking error norm, a substantial difference in the orientation tracking error norm is observed at the beginning. Likewise, e_{η_2} norms of pool experiments and simulations are substantially different. However, the overall profiles of MAE % of e_{η_2} norm of both pool experiments and simulations are very similar except for EO-LQT5.

Generally, the overall pose tracking performances of all controllers, especially including the baseline CO-LQT, are different in pool experiments and simulations. Based on this observation, it can be inferred that the main performance difference of the controllers during pool experiments and simulations is not due to the incoherent

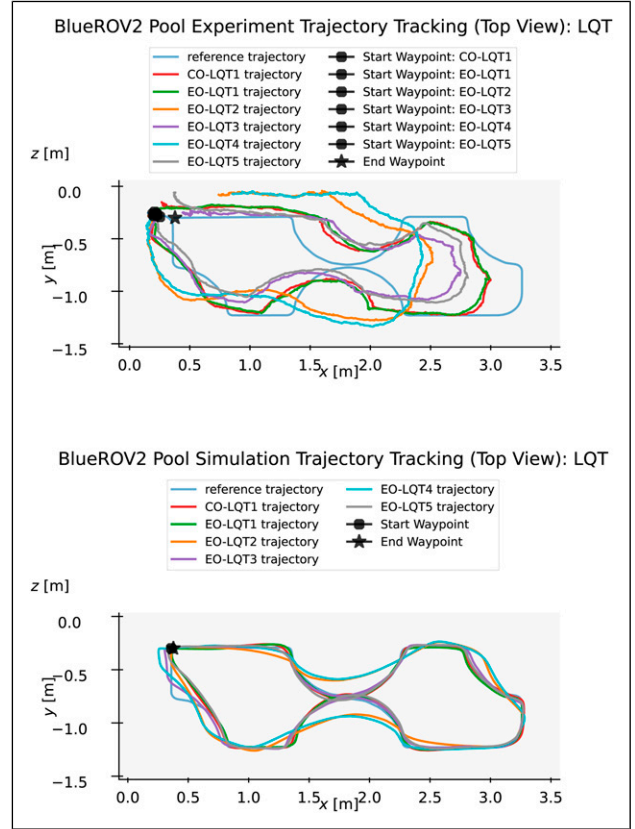


Figure 16. [Pool Experiments and Simulations] Trajectory tracking comparison (top view).

behaviors of the proposed controllers but because of the aforementioned model uncertainties, unmodeled dynamics, and experimental setup for the start waypoint, affecting all controllers.

In addition to the coherent profiles of MAE % of e_{η_1} and e_{η_2} norms of controllers between pool experiments and simulations, it is important to identify the suitable controllers for pose tracking. According to Table 4, the pose tracking performance of EO-LQT2 and EO-LQT4 is substantially inferior to EO-LQT1, EO-LQT3, and EO-LQT5. Among those superior pose tracking controllers, the energy consumption must be considered to choose the most suitable controller, and it will be addressed in the following section.

8.2.3. Energy consumption: Pool experiments and simulations. As shown in Figure 18, the overall energy consumption profiles of the controllers during pool experiments and simulations are substantially similar. As reported in Table 4 and based on pose tracking performance differences discussed earlier, the energy consumption values of the controllers in both pool experiments and simulations are considered reasonable and consistent. Although EO-LQT1 consumes a negligible amount of more energy (0.3%) in pool experiments, the overall energy consumption profiles

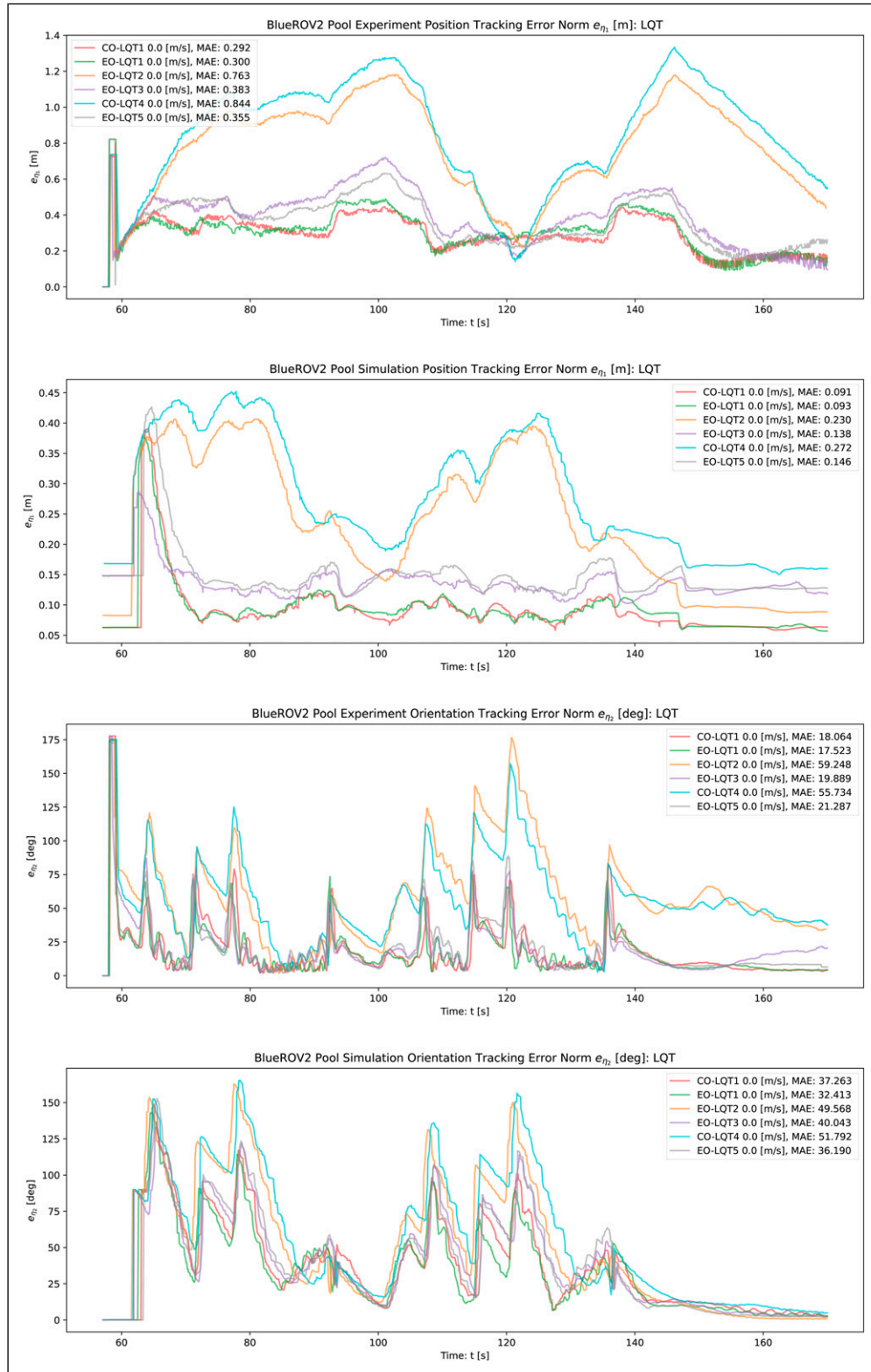


Figure 17. [Pool Experiments and Simulations] Pose tracking error norm comparison.

Table 4. [Pool Experiments and Simulations] Pose tracking norm and energy consumption comparison taking CO-LQT1 as the baseline.

		e_{η_1} norm (m)			e_{η_2} norm (deg)			Energy consumption (Wh)		
		MAE (m)	MAE ratio	%	MAE (deg)	MAE ratio	%	$E_{\Delta t}$ (Wh)	$E_{\Delta t}$ Ratio	%
Expt.	CO-LQT1	0.292	1.	0.0	18.064	1.	0.0	1.204	1.	0.0
	EO-LQT1	0.3	-0.029	2.9	17.523	0.03	-3.0	1.208	-0.003	0.3
	EO-LQT2	0.763	-1.616	161.6	59.248	-2.28	228.0	0.975	0.19	-19.0
	EO-LQT3	0.383	-0.313	31.1	19.889	-0.101	10.1	1.078	0.105	-10.5
	EO-LQT4	0.844	-1.893	189.3	55.734	-2.085	208.5	0.968	0.196	-19.6
	EO-LQT5	0.355	-0.218	21.8	21.287	-0.178	17.8	1.071	0.11	-11.0
Sim.	CO-LQT1	0.091	1.	0.0	37.263	1.	0.0	1.327	1.	0.0
	EO-LQT1	0.093	-0.021	2.1	32.413	0.13	-13.0	1.312	0.011	-1.1
	EO-LQT2	0.23	-1.517	151.7	49.568	-0.33	33.0	1.024	0.228	-22.8
	EO-LQT3	0.138	-0.507	50.7	40.043	-0.075	7.5	1.181	0.11	-11.0
	EO-LQT4	0.272	-1.979	197.9	51.792	-0.39	39.0	1.019	0.232	-23.2
	EO-LQT5	0.146	-0.601	60.1	36.19	0.029	-2.9	1.206	0.091	-9.1

The negative “-” sign of MAE ratio and $E_{\Delta t}$ ratio represents EO-LQTs’ more tracking error and more energy consumption, respectively, compared to those of CO-LQT. For ease of interpretation, the sign of % is opposite to the sign of ratio.

of all controllers are coherent across pool experiments and simulations. With the consideration of energy consumption among EO-LQT1, EO-LQT3 and EO-LQT5, which are identified to achieve superior pose tracking performance earlier, EO-LQT3 and EO-LQT5 are the most energy-optimal trajectory tracking controllers.

8.3. Inferring future FNVI field trial outcomes from pool experiments and simulations

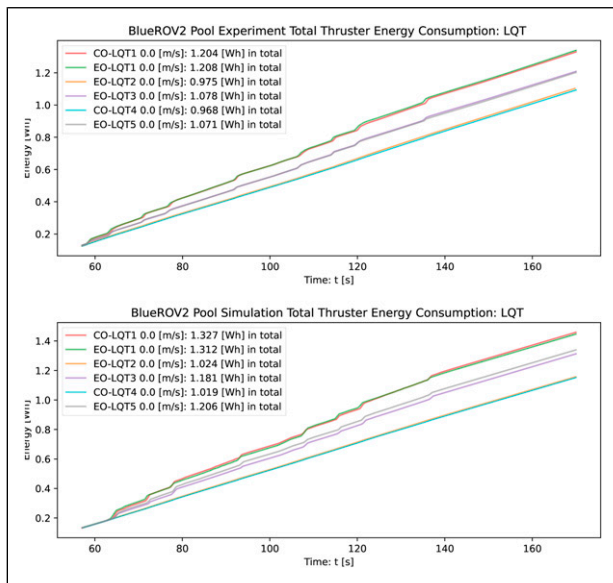
Based on the earlier discussion on pose tracking and energy consumption of pool experiments and simulations, it is

obvious that the MAE values of pose tracking and energy consumption of controllers are substantially different between pool experiments and simulations. Given that even the baseline CO-LQT exhibits the same phenomenon, it can be concluded that these substantial differences between pool experiments and simulations arise from model uncertainties, unmodeled dynamics and variations in the starting waypoint within the experimental setup. On the other hand, the respective overall profiles of MAE values of pose tracking and energy consumption of controllers are coherent between pool experiments and simulations. Given these coherent profiles between pool experiments and simulations, the next intriguing question is whether any plausible conclusion can be drawn between the pool experiments and FNVI simulations.

As shown in Tables 3 and 4, the MAE % profiles of pose tracking of controllers without disturbances in FNVI simulations are coherent with those in pool simulations. Likewise, energy consumption % profiles of controllers between FNVI simulations and pool simulations are similar. Given this coherent observation among pool experiments, pool simulations and FNVI simulations, it can be inferred that the experimental results of the proposed controllers in future FNVI field trials are likely to align with the findings from the pool experiments. Therefore, EO-LQT3 and EO-LQT5 are most likely to be the suitable energy-optimal trajectory tracking controllers, but there are substantial future works to be done before the field trials in the open ocean. The potential future works will be detailed in the following section.

9. Future work

Given the results of EO-LQT3 and EO-LQT5 under different underwater current disturbance speeds (e.g., 0.0 m/s -

**Figure 18.** [Pool Experiments and Simulations] Energy consumption comparison.

0.9 m/s), they relatively perform the trajectory tracking well, while minimizing the energy consumption. However, the performance differences between pool experiments and simulations underline the plausible factors, highlighted in Subsection 8.2, which can cause model uncertainties in addition to the unmodeled dynamics. Therefore, future work directions are related to minimizing those factors via hardware upgrades and are targeted to field trials, exploring more energy-optimal control schemes with stability and robustness analysis.

Although the current experimental setup requires only one camera and two AprilTags, the additional fixture is needed to keep the fixed AprilTag on the UUV above the water. This fixture potentially increases hydrodynamic parameters and the overall UUV dynamics (model uncertainties and unmodeled dynamics). Therefore, to remove this additional fixture, the experimental setup should be changed to multiple AprilTags inside the pool and one low-light camera (with high resolution and high frames per second) mounted on the UUV dedicated to localization.

Relating to the use of multiple AprilTags inside the pool, it requires a real-time localization strategy for state-estimation, which does not rely heavily on the accuracy of the physical installation of multiple AprilTags.

Currently, the tethered cable is used for communication to transfer data between the topside computer and the UUV. The topside computer performs AprilTag detection, vision-based full state-estimation, and control algorithm executions. To achieve a tetherless operation, the UUV needs to perform all those tasks onboard and thus, it needs to be upgraded with higher computing capacities.

In addition to achieving a tetherless UUV with high computing capacities onboard, a proper identification of the hydrodynamic and hydrostatic parameters of the UUV needs to be carried out on the new form-factor due to the hardware upgrade.

After conducting the extensive controller experiments on BlueROV2 Heavy Configuration with the aforementioned hardware upgrade in the controlled environment, a larger industrial-grade UUV is required for the open ocean field trials in the future.

10. Conclusion

In this paper, the energy-optimal control schemes based on linear quadratic tracking (LQT) principles are developed for a UUV performing the FNVI task of the Blue Endeavour Project (the first of its kind for offshore aquaculture salmon farm in New Zealand) of New Zealand King Salmon Company. The linearized model in the form of an LPV system is produced via factorizing the nonlinear UUV dynamic model and extracting

states using the proposed modified versions of Bhāskara I's sine approximation and Shirali's cosine approximation. Subsequently, the Lagrangian $\mathcal{L}$ with linear and rotational kinetic energies and energy terms using quadratic power and non-quadratic power functions of T200 thruster are formulated to modify the conventional LQT (CO-LQT) controller into an energy-optimal LQT (EO-LQT) controller. Specifically, the use of a non-quadratic term in Hamiltonian $\mathcal{H}$ requires resolving the HJB equation, and the full derivation steps are provided in detail.

Several variants of EO-LQT controllers are proposed and tested on a lightweight and highly maneuverable UUV, called BlueROV2 Heavy Configuration, in ROS-based high-fidelity simulation platform integrated with Gazebo Physics Engine called UUV Simulator under three underwater current speeds (0.0 m/s, 0.5 m/s, and 0.9 m/s). The proposed controllers are also tested on the actual hardware of BlueROV2 Heavy Configuration in the pool, along with their respective high-fidelity simulation tests in the pool environment.

From the analysis on trajectory tracking accuracy and energy-optimality, EO-LQT3, which uses a non-quadratic power function of T200 thruster, and EO-LQT5, which uses the Lagrangian $\mathcal{L}$ and non-quadratic power function of T200 thruster, are identified as the most energy-optimal trajectory tracking controllers. In FNVI simulations under 0.0 m/s–0.9 m/s underwater current disturbance speeds, EO-LQT3 and EO-LQT5 consume 6.0%–26.5% and 4.3%–37.1% less energy than CO-LQT1, respectively. In pool simulations, EO-LQT3 and EO-LQT5 consume 11.0% and 9.1% less energy than CO-LQT1, respectively. In pool experiments, EO-LQT3 and EO-LQT5 consume 10.5% and 11.0% less energy than CO-LQT1, respectively. The plausible factors that cause the controller performance difference between pool experiments and simulations in terms of trajectory tracking and energy-optimality are reported. In addition, suggestions to address those factors are provided and future works are targeted to field trials, exploring more energy-optimal control schemes with stability and robustness analysis.

Acknowledgments

The authors acknowledge the financial support of the Blue Economy Cooperative Research Centre, established and supported under the Australian Government's CRC Program, grant number CRCXX000001 (previously 20180101). The CRC Program supports industry-led collaborations between industry, researchers and the community. The authors also acknowledge the graduate research facilities and the financial support of the Auckland University of Technology (AUT) in providing the waiver of the PhD tuition fees. We greatly appreciate the technical support of Mr. Simon Hartley, the technical specialist-mechatronics at AUT, Mr. Adam Poloha, the research assistant at Mechatronics Lab, AUT and Mr. Sai Htet Moe Swe, my fellow PhD student at AUT.

ORCID iDs

Thein Than Tun  <https://orcid.org/0000-0002-2131-811X>

Loulin Huang  <https://orcid.org/0000-0002-4032-6238>

Funding

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research project is supported by the Blue Economy Cooperative Research Centre, established and supported under the Australian Government's CRC Program, grant number CRCXX000001 (previously CRC-20180101).

Declaration of conflicting interests

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

Note

1. UUV is a generic term and it covers both autonomous underwater vehicle (AUV) and remotely operated underwater vehicle (ROV). These terms will be used interchangeably in this paper.

References

- Akram W, Casavola A, Kapetanović N, et al. (2022) A visual servoing scheme for autonomous aquaculture net pens inspection using ROV. *Sensors* 22(9): 3525. <https://doi.org/10.3390/s22093525>
- Amundsen HB, Caharija W and Pettersen KY (2021) Autonomous ROV inspections of aquaculture net pens using DVL. *IEEE Journal of Oceanic Engineering* 47(1): 1–19. <https://doi.org/10.1109/JOE.2021.3105285>
- Antonelli G (2018) *Underwater Robots*. Springer International Publishing AG.
- Ao Y, Xu J, Zhang D, et al. (2024) Artificial intelligence aided design of hull form of unmanned underwater vehicles for minimization of energy consumption. *Journal of Computing and Information Science in Engineering* 24(1): 011003. <https://doi.org/10.1115/1.4062661>
- Battista T, Woolsey C, Perez T, et al. (2016) A dynamic model for underwater vehicle maneuvering near a free surface. *IFAC-PapersOnLine* 49(23): 68–73, 10th IFAC Conference on Control Applications in Marine SystemsCAMS 2016. <https://doi.org/10.1016/j.ifacol.2016.10.323>
- Bauschmann N, Duecker DA, Alff TL, et al. (2023) Evaluation of underwater AprilTag localization for highly agile micro underwater robots. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9926–9932. <https://doi.org/10.1109/IROS55552.2023.10341764>
- BlueRobotics. BlueROV2 heavy configuration retrofit kit [Online]. <https://bluerobotics.com/store/rov/bluerov2-upgrade-kits/brov2-heavy-retrofit/> (Accessed 02 August 2023).
- Chu Y, Wang CM, Park JC, et al. (2020) Review of cage and containment tank designs for offshore fish farming. *Aquaculture* 519: 734928. <https://doi.org/10.1016/j.aquaculture.2020.734928>
- Claus B and Bachmayer R (2016) Energy optimal depth control for long range underwater vehicles with applications to a hybrid underwater glider. *Autonomous Robots* 40(7): 1307–1320. <https://doi.org/10.1007/s10514-016-9555-3>
- De Carolis V, Brown KE and Lane DM (2018) Runtime energy estimation and route optimization for autonomous underwater vehicles. *IEEE Journal of Oceanic Engineering* 43(3): 608–619. <https://doi.org/10.1109/JOE.2017.2707261>
- Fossen TI (2011) *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.
- Huang H, Liu Y, Liu J, et al. (2023) General optimal trajectory planning: enabling autonomous vehicles with the principle of least action. *Engineering* 33: 63–76. <https://doi.org/10.1016/j.eng.2023.10.001>
- Huynh VT, Dunbabin M and Smith RN (2015) Predictive motion planning for AUVs subject to strong time-varying currents and forecasting uncertainties. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1144–1151. <https://doi.org/10.1109/ICRA.2015.7139335>
- Jones D and Hollinger GA (2017) Planning energy-efficient trajectories in strong disturbances. *IEEE Robotics and Automation Letters* 2(4): 2080–2087. <https://doi.org/10.1109/LRA.2017.2719760>
- Joseph GG (2009) *A Passage to Infinity: Medieval Indian Mathematics from Kerala and Its Impact*. Sage.
- Jung S, Brizzolara S and Woolsey C (2021) An approach for computing parameters for a lagrangian nonlinear maneuvering and seakeeping model of submerged vessel motion. *IEEE Journal of Oceanic Engineering* 46(3): 749–764. <https://doi.org/10.1109/JOE.2021.3052657>
- Jung J, Choi HT and Lee Y (2025) Persistent localization of autonomous underwater vehicles using visual perception of artificial landmarks. *Journal of Marine Science and Engineering* 13(5): 828. <https://doi.org/10.3390/jmse13050828>
- Kim MG, Kang H, Lee MJ, et al. (2021) UUV platform optimal design for overcoming strong current. *Journal of Ocean Engineering and Technology* 35(6): 434–445. <https://doi.org/10.26748/KSOE.2021.069>
- Liao W, Zhang S, Wu Y, et al. (2022) Research on intelligent damage detection of far-sea cage based on machine vision and deep learning. *Aquacultural Engineering* 96: 102219. <https://doi.org/10.1016/j.aquaeng.2021.102219>
- Lim HS, King P, Chin CK, et al. (2022) Real-time implementation of an online path replanner for an AUV operating in a dynamic and unexplored environment. *Applied Ocean Research* 118: 103006. <https://doi.org/10.1016/j.apor.2021.103006>
- Manhães MMM, Scherer SA, Voss M, et al. (2016) UUV simulator: a gazebo-based package for underwater intervention and multi-robot simulation. In: *OCEANS 2016 MTS/IEEE Monterey*. IEEE. <https://doi.org/10.1109/oceans.2016.7761080>
- Mate-Kole EM, Margot D and Dewji SA (2023) Mathematical solutions in internal dose assessment: a comparison of Python-based differential equation solvers in biokinetic modeling. *Journal of Radiological Protection* 43(4): 041507. <https://doi.org/10.1088/1361-6498/ad0409>

- MathWorks. ODE Jacobian matrix [Online]. <https://au.mathworks.com/help/matlab/ref/odejacobian.html#d126e1105553> (Accessed 12 August 2023).
- Morato MM, Normey-Rico JE and Sename O (2020) Model predictive control design for linear parameter varying systems: a survey. *Annual Reviews in Control* 49: 64–80. <https://doi.org/10.1016/j.arcontrol.2020.04.016>
- Newcombe E, Knight B, Smeaton M, et al. (2020) Water column assessment for a proposed salmon farm offshore of the Marlborough Sounds. In: *Prepared for the New Zealand King Salmon Co. Limited. Cawthron Report No. 3313*, 96.
- Preece MA (2021) Evidence of Mark Anthony Preece on behalf of the applicant. *Applicant's evidence - 022 - M Preece - NZKS Operational Matters*, [Online]. <https://eservices.marlborough.govt.nz/programmes/ListProgrammeEvents?id=3516198>
- Sarkar M, Nandy S, Vadali SRK, et al. (2016) Modelling and simulation of a robust energy efficient AUV controller. *Mathematics and Computers in Simulation* 121: 34–47. <https://doi.org/10.1016/j.matcom.2015.08.021>
- Scipy. Scipy.integrate.solve_ivp [Online]. https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.solve_ivp.html (Accessed 20 January 2024).
- Shen C and Shi Y (2020) Distributed implementation of nonlinear model predictive control for AUV trajectory tracking. *Automatica* 115: 108863. <https://doi.org/10.1016/j.automatica.2020.108863>
- Shen C, Shi Y and Buckham B (2016) Nonlinear model predictive control for trajectory tracking of an AUV: a distributed implementation. In: *2016 IEEE 55th Conference on Decision and Control*. CDC, pp. 5998–6003. <https://doi.org/10.1109/CDC.2016.7799190>
- Shirali SA (2011) The Bhaskara-Aryabhata approximation to the sine function. *Mathematics Magazine* 84(2): 98–107. <https://doi.org/10.4169/math.mag.84.2.098>
- Spangelo I and Egeland O (1992) Computing energy-optimal trajectories for an autonomous underwater vehicle using direct shooting. *Modeling, Identification and Control Journal* 13(3): 163–174. <https://doi.org/10.4173/mic.1992.3.4>
- T200 Thruster. T200 thruster: under technical details [Online]. <https://bluerobotics.com/store/thrusters/t100-t200-thrusters/t200-thruster-r2-rp/> (Accessed 3 November 2023).
- Tang G, Yang T, Yang Y, et al. (2025) Relative localization and dynamic tracking of underwater robots based on 3D-AprilTag. *Journal of Marine Science and Engineering* 13(5): 833. <https://doi.org/10.3390/jmse13050833>
- Tedrake R (2023) *Underactuated Robotics* [Online]. <https://underactuated.csail.mit.edu>
- The New Zealand Government. Aquaculture strategy [Online]. <https://www.mpi.govt.nz/fishing-aquaculture/aquaculture-fish-and-shellfish-farming/aquaculture-strategy-for-new-zealand/> (Accessed 29 February 2024).
- The New Zealand King Salmon Company. Blue Endeavour [Online]. <https://www.kingsalmon.co.nz/open-ocean-blue-endeavour/> (Accessed 25 November 2023).
- Tun TT, Huang L and Preece MA (2023) Development and high-fidelity simulation of trajectory tracking control schemes of a UUV for fish net-pen visual inspection in offshore aquaculture. *IEEE Access* 11: 135764–135787. <https://doi.org/10.1109/ACCESS.2023.3337872>
- UUV Simulator. uuv_simulator: unmanned underwater vehicle (UUV) simulation with Gazebo [Online]. https://github.com/uuvsimulator/uuv_simulator (Accessed 20 November 2023).
- Weckesser W. Odeintw [Online]. <https://github.com/WarrenWeckesser/odeintw> (Accessed 19 January 2024).
- Whitcomb LL and Yoerger DR (1999) Development, comparison, and preliminary experimental validation of nonlinear dynamic thruster models. *IEEE Journal of Oceanic Engineering* 24(4): 481–494. <https://doi.org/10.1109/48.809270>
- Yang N, Reza Amini M, Johnson-Roberson M, et al. (2018) Real-time model predictive control for energy management in autonomous underwater vehicle. In: *2018 IEEE Conference on Decision and Control*. CDC, pp. 4321–4326. <https://doi.org/10.1109/CDC.2018.8619844>
- Yang N, Chang D, Amini MR, et al. (2019) Energy management for autonomous underwater vehicles using economic model predictive control. In: *2019 American Control Conference (ACC)*, pp. 2639–2644. <https://doi.org/10.23919/ACC.2019.8815106>
- Yang N, Chang D, Johnson-Roberson M, et al. (2020) Robust energy-optimal path following control for autonomous underwater vehicles in ocean currents. In: *2020 American Control Conference (ACC)*, pp. 5119–5124. <https://doi.org/10.23919/ACC45564.2020.9147322>
- Yang N, Chang D, Johnson-Roberson M, et al. (2021) Energy-optimal path planning with active flow perception for autonomous underwater vehicles. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9928–9934. <https://doi.org/10.1109/ICRA48506.2021.9561865>
- Yang N, Shen C, Song Z, et al. (2024) Robust energy-optimal control for 3-D path-following of autonomous underwater vehicles under ocean currents. *IEEE Transactions on Control Systems Technology* 32(2): 680–687. <https://doi.org/10.1109/TCST.2023.3315602>
- Yao F, Yang C, Zhang M, et al. (2019) Optimization of the energy consumption of depth tracking control based on model predictive control for autonomous underwater vehicles. *Sensors* 19(1): 162. <https://doi.org/10.3390/s19010162>

Appendix A

A.1. Specification of Blue Endeavour Project

The Blue Endeavour Project is an offshore aquaculture site, a first of its kind in New Zealand, proposed by the New Zealand King Salmon Company. It is located 7 km due north of Cape Lambert, in the Cook Strait, in northern Marlborough as shown in Figure 19. Its proof-of-concept phase started in June 2025. Its specifications are gathered from publicly available information and some are directly provided by the New Zealand King Salmon Company. More information about the farm and the UUV, practical concerns and the proposed methods to mitigate those concerns were detailed in our previous work (Tun et al.,

Table 5. Hardware and software specification for ROS.

Parameters	Value	Note
Computer	HP EliteDesk	Standard hardware configuration
CPU	CORE i7 vPro	9th generation
Operating system	Ubuntu 18.04.6	Ubuntu is the recommended operating system for ROS
ROS version	Melodic Morenia	For future explorations, ROS 2 is recommended if compatible ROS-Gazebo underwater plugins are available
Gazebo version	9.0	For other ROS versions, a compatible Gazebo version has to be installed

2023)). The following is the summary of the specifications of the Blue Endeavour Project.

A.2. Simulation parameters

Underwater current disturbance: According to the technical report for the Blue Endeavour Project site, the mean underwater current speed can reach up to 0.9 m/s near the surface due to the wind and wave and it declines to near

Table 6. Specification of Blue Endeavour Project.

Parameters	Value	Descriptions
D	53.5 m	Fish net-pen internal diameter
d_m	35 mm	Fish net-pen mesh size
h	15 m	Fish net-pen depth
$v_{c,min}$	0.5 m/s	Mean underwater current
$v_{c,max}$	0.9 m/s	Surface mean underwater current



Figure 19. Blue Endeavour Project site (orange-colored rectangles) in Cook Strait (The New Zealand King Salmon Company (Accessed 25 November 2023)).

Table 7. Underwater current disturbance scenarios (All underwater current disturbances are assumed to be irrotational).

Scenario	Value	Direction
S1: No disturbance	0.0 m/s	: positive y -axis of Frame $\{I\}$
S2: Medium disturbance	0.5 m/s	: positive y -axis of Frame $\{I\}$
S3: Large disturbance	0.9 m/s	: positive y -axis of Frame $\{I\}$

0.35 m/s at 10 m depth. However, for the range of depth from 10 m to 15 m, the average underwater current speed increases to near 0.5 m/s (Newcombe et al., 2020). Therefore, three underwater current disturbance scenarios are proposed as shown in Table 7. As the operating area is relatively small, compared to the vast ocean, the underwater current disturbance is assumed to be irrotational and constant—it does not change with the position and the depth at which UUV operates. However, in reality, it varies according to the depth as mentioned above. In future work, the variation of the underwater current disturbance profile depending on the depth may be considered.

UUV: A collection of UUV parameters based on the manufacturer's website, literature and empirical study is reported as shown in Table 8. Especially, hydrodynamics parameters must be sourced from the literature, and other parameters (e.g., rotor constant and maximum rated thrust of T200 thrusters, maximum wrench by BlueROV2 Heavy Configuration) were discussed and reported in our previous work using the publicly available dataset of T200 thruster T200 Thruster (Tun et al., 2023 (Accessed 3 November 2023)).

Controller: Generally, Q and R in PI can be assigned according to J_T and J_E as shown in Table 1. For instance, EO-LQT1: $Q = Q_E$ and $R = R$, EO-LQT2: $Q = Q$ and $R = R_E$.

[FNVI Simulation] The control gains and parameters are shown below.

$$Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}, Q_f = \begin{bmatrix} Q_{1,f} & 0 \\ 0 & Q_{2,f} \end{bmatrix}$$

$$Q_1 = \text{diag}([900 \ 900 \ 900 \ 1000 \ 1000 \ 1000]^T)$$

$$Q_{1,f} = \text{diag}([500 \ 500 \ 500 \ 600 \ 600 \ 600]^T)$$

$$Q_2 = Q_{2,f} = \text{diag}([100 \ 100 \ 100 \ 100 \ 100 \ 100]^T)$$

$$R = \text{diag}([1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0]^T)$$

$$R_c = 15 \text{ Hz}, N_h = 3 \text{ steps (Sampling step)},$$

$$t_f = t_{\text{ahead}} = 0.067 \text{ s}$$

(Note: t_f is assigned with one control loop duration.)

[Pool Experiment & Simulation] The control gains and parameters are shown below.

Table 8. UUV (BlueROV2 Heavy Configuration) parameters.

Parameters	Value	Descriptions
m_r	11.5 kg	: mass
w_G	112.8 N	: weight
w_B	112.9 N	: buoyancy force
w_b, l_b, h_b	0.575 m, 0.4571 m, 0.2539 m	: width, length, height
m	8	: number of thrusters (4 in horizontal, 4 in vertical)
c_f	0.0003	: rotor constant of T200 thruster
$u_{\max}$	1.5 m/s	: maximum surge speed
$d_{\max}$	100 m	: maximum rated depth
$f_{T,\max}$	45.67 N	: maximum forward and reverse thrust of T200 thruster
$f_{T,\text{rated}}$	34.32 N	: maximum rated thrust based on thruster configuration
$\tau_{x,\max}, \tau_{y,\max}$	88.25 N	: maximum force in surge and sway (45° vectored formation)
$\tau_{z,\max}$	137.29 N	: maximum force in heave
r_G^b	[0,0,0] m	: center of gravity in the respective x, y, and z axis of Frame {b}
r_B^b	[0,0,0.02] m	: center of buoyancy in the respective x, y, and z axis of Frame {b}
$[I_{xx}, I_{yy}, I_{zz}]$	[0.21, 0.18, 0.31] kg m ²	: moment of inertia expressed in Frame {b}
$[X_u, Y_v, Z_w]$	[4.03, 6.22, 5.18] N s/m	: linear damping coefficient for surge, sway, and heave
$[K_p, M_q, N_r]$	[4.53, 4.4, 3.13] N m s/rad	: linear damping coefficient for roll, pitch, and yaw
$[X_{u u }, Y_{v v }, Z_{w w }]$	[18.18, 21.66, 36.99] N s ² /m ²	: quadratic damping coefficient for surge, sway, and heave
$[K_{p p }, M_{q q }, N_{r r }]$	[26.32, 26.9, 24.63] N m s ² /rad ²	: quadratic damping coefficient for roll, pitch, and yaw
$[X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}]$	[5.5, 12.7, 14.57] kg	: added mass coefficient for surge, sway, and heave
$[K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}]$	[3.77, 3.77, 3.77] kg m ²	: added mass coefficient for roll, pitch, and yaw

$$Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}, Q_f = \begin{bmatrix} Q_{1,f} & 0 \\ 0 & Q_{2,f} \end{bmatrix}$$

$$Q_1 = \text{diag}([1000 \ 1000 \ 1000 \ 900 \ 900 \ 900]^T)$$

$$Q_{1,f} = \text{diag}([600 \ 600 \ 600 \ 500 \ 500 \ 500]^T)$$

$$Q_2 = Q_{2,f} = \text{diag}([100 \ 100 \ 100 \ 100 \ 100 \ 100]^T)$$

$$R = \text{diag}([1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0 \ 1.0]^T)$$

$$R_c = 15 \text{ Hz}, N_h = 5 \text{ steps},$$

$$t_f = t_{\text{ahead}} = 0.067 \text{ s}$$

As shown in equations (41) and (42) and equation (35), Q_E , R_E , and k_P are varying matrices and vector in real-time. Other controller parameters are summarized in Table 9 and sampling step and t_{ahead} are illustrated in Figure 9.

Another critical factor for the high-fidelity simulation and pool experiment is the computation load. According to the Python profiler and ROS' rqt_gui, the heaviest computation loads in descending order are the controller node (due to the numerical solver for backward integration), Gazebo Physics Engine and rqt_gui (ROS visualization graphical user interface). Due to the heavy computation

load, the controller node runs at 15 Hz (Note: setting a frequency higher than 15 Hz could not actually run at a higher frequency due to the limited computation resources.). The numerical solver in the controller node is the heaviest computational load and will be a substantial matter to be tackled regardless of high-fidelity simulation or pool experiment.

Therefore, several numerical solvers such as `scipy.integrate.solve_ivp`, `scipy.integrate.odeint` and `odeintw` using different integration methods (e.g., “RK45,” “RK32,” and “LSODA”) are tested (Scipy (Accessed 21 January 2024); Weckesser (Accessed 19 January 2024)). Among them, as `odeintw` (a wrapper of `scipy.integrate.odeint` for matrix differential equations) has less overhead and uses “LSODA” integration method, which is a wrapper of Fortran ODEPACK, it is chosen as the numerical solver in this work. The detailed discussion on the numerical solver is beyond the scope of this work and further information can be referred to Mate-Kole et al. (2023). The other two heaviest computation loads (Gazebo Physics Engine and rqt_gui) are not compulsory in pool experiments but only in high-fidelity simulation. In high-fidelity simulation, Gazebo provides odometry data for localization and state-estimation, but in pool experiments, it is worth noting that it

Table 9. Controller parameters.

Parameters	Value	Descriptions
Loop rate (R_c)	15 Hz	: Controller node rate [Note. Due to the limited computing resources, such a low rate is used.]
Numerical solver	odeintw	: Integration solver which by default, uses LSODA integration method, for backward integrating equations (54)–(56), and (63)–(65)
Integration method	LSODA	: A variant of the Livermore Solver for Ordinary Differential Equations (LSODE) which can automatically switch between Adam (nonstiff) and BDF (stiff) method Mate-Kole et al. (2023)

requires relatively more computation capacity for the vision-based state-estimation.

ROS Launch Processes: ROS services such as launching the trajectory and injecting underwater current disturbances take time compared to ROS node such as running controller node. Moreover, timing

consistency in launching ROS launch files minimizes the discrepancy resulting from starting experiments at different time. Therefore, ROS Launch files are called by the automated Python script which executes the command at the specified time after starting the screen recording.