Applied Ocean Research 137 (2023) 103591

Contents lists available at ScienceDirect

Applied Ocean Research

journal homepage: www.elsevier.com/locate/apor

Integrated path planning and control through proximal policy optimization for a marine current turbine ☆

Arezoo Hasankhani [a], Yufei Tang [a,*], James VanZwieten [b]

[a] Department of Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431, USA
[b] Department of Ocean and Mechanical Engineering, Florida Atlantic University, Boca Raton, FL 33431, USA

ARTICLE INFO

Keywords:
Path planning
Path tracking
Proximal policy optimization (PPO)
Reinforcement learning (RL)
Autonomous underwater vehicle (AUV)
Marine current turbine (MCT)

ABSTRACT

This paper presents an integrated path planning and tracking control framework for a marine current turbine (MCT), where the MCT is treated as an energy-harvesting autonomous underwater vehicle (AUV). Considering the ocean (space of action) is continuous, the proposed framework employs two modules to address path planning and path tracking enabled by the proximal policy optimization (PPO) algorithm, which is a policy gradient deep reinforcement learning (RL) method. To enable fully autonomous operation in a stochastic oceanic environment, the proposed path planning seeks a primary objective of maximizing the harvested energy; then, the path tracking module is designed to minimize the tracking error and avoid collisions with static and dynamic obstacles. Using field-collected acoustic Doppler current profiler (ADCP) data, the performance of the proposed framework is evaluated. Comparative studies with baseline algorithms in three different scenarios of path planning, path tracking without an obstacle, and path tracking with collision avoidance verify the effectiveness of our proposed approach.

1. Introduction

Autonomous underwater vehicles (AUVs) have gained ever-increasing attention recently. To deal with a general framework for autonomous and unmanned driving, two major tasks, including path planning and path tracking, should be taken into account. Path planning and tracking have been widely addressed as two separate duties in the literature to realize smooth path and autonomous driving operations. However, embracing planning and tracking in an integrated framework is necessary to reach a completely autonomous AUV. This paper deals with an energy-harvesting AUV, which requires an integrated framework in a real-time application to realize an autonomous and intelligent system capable of harnessing maximum power through path planning and following the optimal path without collision with static or dynamic obstacles, such as marine animals or underwater infrastructures. In the light of classical control, an efficient path control has entailed optimized ultimate goals such as collision-free optimal path (Ali et al., 2005; Wiig et al., 2019; Yao et al., 2023), minimum path length (Steinhauser and Swevers, 2018; Bortoff, 2000), minimum consumed time (Zeng et al., 2014), minimum energy consumption (Di Franco and Buttazzo, 2015), maximum harnessed energy (Cobb et al., 2021), etc., followed smoothly with minimized tracking error.

To address an independent task of path planning, research has been done in the classical control literature, followed by a multitude of algorithms grouped into Debnath et al. (2019) (i) combinatorial algorithms (i.e., c-space and graph search-based algorithms); (ii) sampling-based algorithms; and (iii) biologically inspired and evolutionary-based algorithms (Xu and Mohseni, 2013). Popular methods for path planning include Dijkstra (Dijkstra et al., 1959), A^* (Hart et al., 1968), probabilistic road map (PRM) (Geraerts and Overmars, 2004), rapidly-exploring random trees (RRT) (LaValle et al., 1998), artificial potential field (APF) (Lee and Park, 2003), heuristic-based algorithm such as genetic algorithm (GA) (Tuncer and Yildirim, 2012), and particle swarm optimization (PSO) (Roberge et al., 2012; Krell et al., 2022). The classical graph search methods (Dijkstra and A^*) have been enabled to cope with a minimized cost path through a weighted graph, suffering from a so-called *curse of dimensionality* due to discrete precision increase (Ferguson et al., 2005). Suppose that the environment graph is updated through sensors, several methods (i.e., Focused Dynamic A^* (D^*) (Stentz et al., 1995), D^* Lite (Koenig and Likhachev, 2002), Anytime Repairing A^* (ARA^*) (Likhachev et al., 2003), etc.) have been proposed instead of replanning from the scratch; still facing with optimality issues and occasionally high computational complexity than

☆ This work was supported in part by the National Science Foundation under Grant No. CMMI-2145571 and the U.S. Department of Energy under Grant No. DE-EE0008955.

* Corresponding author.
E-mail addresses: ahasankhani2019@fau.edu (A. Hasankhani), tangy@fau.edu (Y. Tang), jvanzwi@fau.edu (J. VanZwieten).

https://doi.org/10.1016/j.apor.2023.103591
Received 1 March 2023; Received in revised form 16 April 2023; Accepted 3 May 2023
Available online 6 June 2023

planning from scratch (Ferguson et al., 2005). These algorithms face challenges when dealing with an uncertain and dynamic environment, such as in the open ocean. Moreover, we need such an intelligent planning algorithm to seek a feasible path while avoiding collisions. To meet these requirements of path planning, other new methods, e.g., reinforcement learning (RL) (Xi et al., 2022; Hasankhani et al., 2023, 2021a; Hadi et al., 2022), model predictive control (Hasankhani et al., 2021a; Bin-Karim et al., 2017), and extremum seeking (Bafande and Vermillion, 2016), have been used. From a path planning perspective, RL is a powerful and intelligent algorithm due to its capability to extract robust features from an uncertain and noisy environment. However, developing planning algorithms is still an active research topic to cope with a partially observable dynamic environment.

For path tracking, the major task is to follow the path with minimized tracking error through the smooth path with continuous velocity and acceleration functions. Line-of-sight (LOS) guidance law has been employed to connect waypoints using straight lines (Fossen et al., 2003). This approach is limited to following the lines, even though enhanced as advanced types of LOS (Wu et al., 2021; Fossen et al., 2014; Fossen and Lekkas, 2017; Weng et al., 2022), or combined with complicated methods like MPC (Zhang et al., 2020; Yan et al., 2020) and fuzzy controller (Mu et al., 2018). To follow a curve-based smooth path, in addition to the classical approach of PID controller (Fossen, 2011), more intricate methods have been investigated, including MPC (Ji et al., 2016; Cheng et al., 2020; Li and Yan, 2016), sliding mode control (Truong et al., 2021), adaptive control (Antonelli et al., 2001; Yu et al., 2021; Guerrero et al., 2019), fuzzy control (Zhang et al., 2021), back-stepping control (Cho et al., 2020), recently artificial intelligence-based approaches like RL (He et al., 2021; Sun et al., 2019; Wang et al., 2023), or any combination thereof. To endow the path following, a set of control commands (such as velocity, acceleration, and actuator instructions) should be generated due to the autonomous vehicle's model and environment model. Meanwhile, the utmost concern is defined in relation to the underactuated autonomous vehicles (i.e., plants with fewer actuators than their degree-of-freedom) (Aguiar and Hespanha, 2007) and underactuated AUV (Jin et al., 2015) intensified subject to application in a dynamic and unpredictable environment.

An integrated framework for path control, contributing to a single complex task of path planning and tracking, represents a possible solution to the fully autonomous systems operating in an inherent stochastic environment. Such a framework deals with high-level path planning and low-level path tracking in real-time, subject to nonholonomic constraints. As an instance, a multiconstrained model predictive control (MMPC) has been developed to construct a collision-free path for autonomous driving and successfully follow the path (Ji et al., 2016). An integrated path planning and tracking for an AUV has been proposed in Shen et al. (2017) following a spline path defined due to AUV's dynamic. *Deep reinforcement learning (DRL)*, as an approach in machine learning widely applied for autonomous control applications, has demonstrated impressive results in the field of path control. For example, DRL has been applied to address collision avoidance as a path planning task and path following for the AUV in the presence of stationary obstacles (Meyer et al., 2020b; Havenstrøm et al., 2021) and a complex layout of dynamic obstacles (Meyer et al., 2020a). For a similar application of AUV, path planning and tracking have been developed in a bi-level framework taking advantage of DRL in both levels (He et al., 2021), thereby enabling more complicated planning objectives but increasing the complexity and simulation time. The DRL algorithm as a model-free algorithm learning policy from its interactions with the environment seems a better choice compared to the MPC (Hasankhani et al., 2021a), which is identified as a model-based algorithm and would be sensitive to the model precision. For our specific application of AUV and MCT, using the DRL would be very helpful, where the real-recorded data from the ocean environment can be directly used to train the integrated path planning and tracking. The DRL will then capture the uncertainties in the ocean environment and learn how to react when facing an obstacle. Hence, the DRL is by its nature a wise candidate for the integrated path planning and tracking of the AUV systems in the uncertain oceanic environment.

In this paper, we focus on an *oceanic environment*, and a *marine current turbine (MCT)* interpreted as the AUV (Hasankhani et al., 2021b) with the primary task of energy generation. Several prerequisites should be entailed to formulate the challenging path control problem for the MCT. The path planning task should be defined by a major objective of power maximization, introducing a different problem from the AUV's common path planning. Further, to enable path tracking in the ocean, waypoints selected by the path planner can be connected through either a straight-line path (Martinsen and Lekkas, 2018) or a curved path (Marrtinsen and Lekkas, 2018); however, the main difficulties arise in the necessity of defining a feasible curved path for the MCT system. For path tracking, the tracking error should be minimized, and the collision should be avoided due to the stochastic oceanic environment. Hence, to address path control, this paper will advance the integrated framework of path planning and tracking for a highly nonlinear dynamic MCT while modeling a stochastic oceanic environment. The underactuated MCT controlled by three actuators is targeted to achieve two major objectives of power maximization and collision avoidance, which distinguishes us from previous works in AUV literature (Meyer et al., 2020b; Havenstrøm et al., 2021; Meyer et al., 2020a; He et al., 2021), by defining a smooth curved reference path and following that path while considering MCT dynamics and avoiding any aggressive motion through an integrated framework to deal with the real-time autonomous path control.

Building on our previous work in MCT dynamic modeling (Hasankhani et al., Accepted), this paper will develop a powerful path planning and tracking framework enabled by proximal policy optimization (PPO) algorithm, with the following major contributions:

- We present an integrated path planning and tracking control framework to enable path control for a fully autonomous underactuated energy-harvesting AUV in a real-time application. The proposed algorithm takes advantage of PPO capable of learning dynamic and robust features from the field-recorded ocean current profile from an uncertain environment.
- We formulate a novel path planning algorithm to maximize the harnessed power from the energy-harvesting AUV. To ensure a feasible path for AUV and avoid any abrupt movement, PPO-based path planning leverages primary AUV constraints on position, velocity, and acceleration. Then, by employing a path smoother, the planned path is smoothed for the path tracking controller, who is responsible for following the optimal path without collision. To the best of the authors' knowledge, it is the first time to propose such an integrated path planning and tracking control for the energy-harvesting AUV.
- We verify the efficiency of the proposed approach to tame an underactuated MCT with 7 Degree-of-freedom (DOF) and 3 actuators to cope with power maximization and collision avoidance with static and dynamic obstacles when the agent operating in the stochastic oceanic environment.

The rest of the paper is organized as follows. Section 2 formulates the framework for integrated path planning and tracking, consisting of the system model, path model, environment model, and integration strategy. Section 3 presents a DRL-based integrated control architecture. Section 4 describes a specific application of our integrated model for the MCT system. Section 5 presents the simulation results, and Section 6 draws conclusions and future works.

2. Problem statement for underactuated AUV

In this section, the AUV model, underwater environment model, and proposed integrated path planning and tracking framework for the AUV are discussed.

2.1. System modeling

AUV Model: Consider a dynamic model for an underactuated AUV:

$$\begin{aligned} \dot{X}(t) &= f(X(t), U(t)) \\ Y(t) &= g(X(t)) \end{aligned} \tag{1}$$

with the state vector $X \in \mathbb{R}^n$, the control inputs $U \in \mathbb{R}^m$, and $Y \in \mathbb{R}^o$ such that the control inputs are fewer than the DOF of the system. Should the nonlinear model be linearized around an equilibrium operating point, and using the linearized model reduces the computational complexity of the problem, the nonlinear model may be replaced with a linear model.

The AUV system is generally described with twelve states, including $\eta = [x\ y\ z\ \phi\ \theta\ \psi]^T$, with x, y, z representing the surge, sway, and heave, as well as ϕ, θ, and ψ denoting the roll, pitch, and yaw, respectively. The six remaining states represent the linear velocity and angular velocity of the AUV system denoted by $\mathcal{V} = [u\ v\ w\ p^b\ q\ r]^T$. Note that in a specific AUV system, the other states, such as rotor angular velocity, can be added to these twelve states.

Uncertain Underwater Environment Model: To perform successful path planning and tracking, an underwater environment should be carefully observed and modeled with respect to the spatial and temporal uncertainties arising from turbulence, wave, and lower frequency flow structures. Meanwhile, the learning-based integrated path planning and tracking framework entails the requirement to rely on previously recorded underwater data from historical observations. The underwater environment data, including current speed, northward current velocity, and eastward current velocity, can be recorded by an acoustic Doppler current profiler (ADCP). Note that the ADCP can measure the water velocity directly above the instrument at depth increments of about 5 m to 8 m (depth resolution depends on the configuration); hence, to extend the recorded velocity data spatially, multiple ADCPs should be deployed at the same time. Also, to justify the collision avoidance objective, the environment should be observed to detect the obstacles; hence, a sonar sensor is needed to mount on the top of AUV (Havenstrøm et al., 2021).

Path Model: Let $p = \{p_{i=1:n_w} \in \mathbb{R}^3 | p_i(x_i, y_i, z_i)\}$ denote a set of n_w 3D waypoints expressed in an inertial frame; a well-defined path associated with these waypoints should satisfy a smooth and at least a G^2 continuous path (Chang and Huh, 2015). The G^2 continuity evokes the continuous velocity and acceleration functions and thus a continuous curvature to satisfy a real-world application. Hence, the proposed framework includes a path smoother to yield a smooth G^2 continuous path.

2.2. Proposed integration framework

The proposed framework targets the integrated path planning and tracking control by generating an energy-optimized reference path, as well as following this path with a minimized tracking error and avoiding the collision. The overall proposed framework is represented in Fig. 1, consisting of five primary modules of (i) underwater environment, (ii) PPO-based path planner, (iii) PPO-based path tracking, (iv) AUV model, and (v) path smoother. Let us suppose that the sampling time for "path planner" is T^{spp} and the sampling time for "path tracking" is T^{spt}, where $T^{\text{spt}} \leq T^{\text{spp}}$.

The underwater environment inputs the ADCP data and sonar sensor data to the PPO-based path tracking module to enable the observations from the environment. The PPO-based path planner is responsible for generating the energy-optimized reference path due to the current speed data (ADCP data) received from the underwater environment and the AUV's current position, which output the reference path with the position $[x^*\ y^*\ z^*]^{\mathrm{T}}$ for a 3D path planning; this module is discussed in detail in Section 3.1. The path smoother takes care of generating a smooth path from the reference points received from the path planner subject to the AUV constraints, continuous velocity and acceleration, and the sampling time of path tracking module T^{spt}, so resulting in a reference vector of position and velocity $[x^*\ y^*\ z^*\ u^*\ v^*\ w^*]^{\mathrm{T}}$. The main goal for the path smoother is to generate several points between every two points coming from the path planner and switch from the coarse path planning sampling time T^{spp} (e.g., 1 h) to the higher resolution path tracking sampling time T^{spt} (e.g., 2 s). The generated path from the path smoother will ensure a smooth and feasible path for the path tracking controller. The AUV model interprets the movement of the AUV system and outputs the states and dominant operation constraints to the PPO-based path planner and path tracker, respectively, which also receives the actuators (control inputs) from the PPO-based path tracking module as inputs. Eventually, the PPO-based path tracking takes care of the safe collision-free path, as well as complying with a minimized error tracking of the reference path; this module is also discussed in detail in Section 3.2.

Note that generally the path tracking is executed in a higher frequency than the path planning (Falcone et al., 2008) (i.e., $T^{\text{spt}} \leq T^{\text{spp}}$). Both path planning and tracking are fed with the system model and environmental model; however, these modules are constructed based on different levels of model fidelity constrained by computational complexity and accuracy. Developing a path planner subject to the detailed dynamic model of the system reduces the computation burden for the path tracker still increases the complexity of the planning module. In the proposed framework, the initial path planner will leverage the primary AUV constraints (i.e., the constraints on the position, velocity, and acceleration in a 3D movement).

3. PPO-based integrated control design

The PPO, a policy gradient deep reinforcement learning method (Schulman et al., 2017) is nominated to address the integrated path planning and path tracking problem. The PPO algorithm is a favorable candidate to solve the problem at hand over a continuous space of action. Here, the preliminaries on the PPO algorithm are discussed; then, the details on the PPO algorithm application for path planning and path tracking are explained.

The AUV as an agent observes states s, accomplishes an action a, and receives a reward R accordingly. Let define the advantage function $A(s,a) = Q(s,a) - V(s)$, where $V(s)$ denotes the state-value function and $Q(s,a)$ denotes the action-value function. An estimator from the advantage function over T timesteps is built as a generalized advantage estimate (GAE), namely:

$$\widehat{A}_t = \delta_t + (\gamma\lambda)\delta_{t+1} + \cdots + (\gamma\lambda)^{T-t+1}\delta_{T-1} \tag{2}$$

where

$$\delta_t = R_t + \gamma\widehat{V}(s_{t+1}) - \widehat{V}(s_t) \tag{3}$$

with γ denoting the discount factor and λ denoting a parameter for scaling the state-value function. A surrogate objective function is defined as follows:

$$L^{\text{CLIP}}(\theta) = \widehat{\mathbb{E}}_t[\min(\frac{\pi_\theta(a|s)}{\pi_{\theta_o}(a|s)}\widehat{A}_t, \text{clip}(\frac{\pi_\theta(a|s)}{\pi_{\theta_o}(a|s)}, 1-\epsilon, 1+\epsilon)\widehat{A}_t)] \tag{4}$$

where $\pi_\theta(a|s)$ denotes the policy with θ being a learnable parameter in the PPO network, and ϵ denotes the clipping parameter.

3.1. PPO-based path planning

To endow the PPO algorithm for the AUV path planning, the first step is to define s^{pp}, a^{pp}, and R^{pp}. The state space as an input to the PPO network is built upon the underwater environment velocity (ADCP data) v_{e} with $(.)_{\text{e}}$ denoting the environment and the position of the AUV system, i.e., $s^{\text{pp}} \triangleq [v_{\text{e}}\ x\ y\ z]$. For path planning, the action, which is represented as an output of the PPO network, interprets the optimal position of the AUV, so the action space should contain the

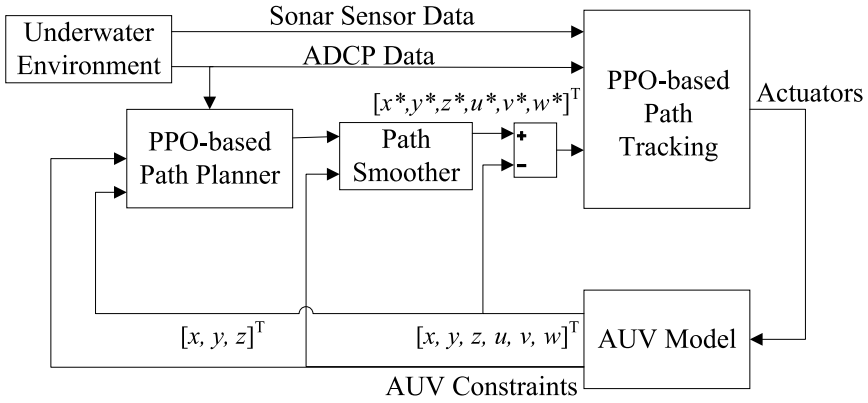

Fig. 1. Proposed integrated path planning and tracking control for an AUV, consisting of five major modules.

Algorithm 1 PPO for path planning (training phase)

1: **Input:** underwater environment velocity, AUV position, feasible underwater environment positions for AUV, and PPO parameters;
2: **for** each iteration **do**
3: Initialize state $s^{\mathrm{pp}} \leftarrow [\upsilon_{\mathrm{e}}\ x\ y\ z]$;
4: Proceed policy $\pi_{\theta_{\mathrm{o}}^{\mathrm{pp}}}(a^{\mathrm{pp}}|s^{\mathrm{pp}})$ over N^{pp} timesteps and accomplish action a^{pp};
5: Compute the path planning reward function R^{pp}:

$$R^{\mathrm{pp}} = \vartheta_1 R_{\mathrm{P}}^{\mathrm{pp}} + \vartheta_2 R_{\mathrm{v}}^{\mathrm{pp}}$$

6: Compute the generalized advantage function estimate:

$$\widehat{A}_t^{\mathrm{p}} = \delta_t + (\gamma^{\mathrm{pp}}\lambda^{\mathrm{pp}})\delta_{t+1} + \ldots + (\gamma^{\mathrm{pp}}\lambda^{\mathrm{pp}})^{N^{\mathrm{pp}}-t+1}\delta_{T-1}$$

7: Optimize the surrogate function:

$$L^{\mathrm{CLIP}}(\theta^{\mathrm{pp}}) = \widehat{\mathbb{E}}_t[\min(\frac{\pi_{\theta^{\mathrm{pp}}}(a^{\mathrm{pp}}|s^{\mathrm{pp}})}{\pi_{\theta_{\mathrm{o}}^{\mathrm{pp}}}(a^{\mathrm{pp}}|s^{\mathrm{pp}})}\widehat{A}_t, \mathrm{clip}(\frac{\pi_{\theta^{\mathrm{pp}}}(a^{\mathrm{pp}}|s^{\mathrm{pp}})}{\pi_{\theta_{\mathrm{o}}^{\mathrm{pp}}}(a^{\mathrm{pp}}|s^{\mathrm{pp}})}, 1-\epsilon, 1+\epsilon)\widehat{A}_t)]$$

8: Update $\theta_{\mathrm{o}}^{\mathrm{pp}} \leftarrow \theta^{\mathrm{pp}}$ every b^{pp} iterations;
9: **end for**
10: **Output:** optimal PPO for path planning;

feasible positions from the underwater environment to be occupied by the AUV, i.e., $a^{\mathrm{pp}} \triangleq [x_{\mathrm{e}}\ y_{\mathrm{e}}\ z_{\mathrm{e}}]$. Finally, the reward function should be characterized based on the ultimate objective of power maximization, where the second objective of collision avoidance is included in path tracking. The reward function for path planning is formulated by two terms to reward the AUV in case of following the maximum velocity and maximum power, namely:

$$R^{\mathrm{pp}} = \vartheta_P R_{\mathrm{P}}^{\mathrm{pp}} + \vartheta_\upsilon R_{\mathrm{v}}^{\mathrm{pp}} \tag{5}$$

where

$$R_{\mathrm{P}}^{\mathrm{pp}} = \mathrm{clip}(\frac{P - P^{\mathrm{d}}}{P^{\mathrm{d}}}, -1, +1) \tag{6}$$

$$R_{\mathrm{v}}^{\mathrm{pp}} = \mathrm{clip}(\frac{\upsilon_{\mathrm{e}} - \upsilon_{\mathrm{e}}^{\mathrm{d}}}{\upsilon_{\mathrm{e}}^{\mathrm{d}}}, -1, +1) \tag{7}$$

with ϑ_{P} and ϑ_{v} represent the coefficients for two terms of the reward function; the maximum values for velocity and power are denoted by desired values of P^{d} and $\upsilon_{\mathrm{e}}^{\mathrm{d}}$. The algorithm for the PPO-based path planning is presented in Algorithm 1 with θ^{pp} showing the PPO-based path planning network coefficient.

3.2. PPO-based path tracking

For path tracking, the main objective is to follow an optimal path determined by the path planner while avoiding collision with an obstacle. To perform this task, the state space is defined by sonar sensor data $\boldsymbol{\mathcal{D}}_{ss}$, underwater environment velocity (ADCP recorded data), linear position error $\Delta\eta^{\mathrm{L}} = [\Delta x\ \Delta y\ \Delta z]$ and linear velocity error from the optimal values $\Delta\boldsymbol{\mathcal{V}}^{\mathrm{L}} = [\Delta u\ \Delta v\ \Delta w]$, so the PPO path tracking states are $s^{\mathrm{pt}} \triangleq [\upsilon_{\mathrm{e}}\ \boldsymbol{\mathcal{D}}_{ss}\ \Delta\eta^{\mathrm{L}}\ \boldsymbol{\mathcal{V}}^{\mathrm{L}}]$. The action space a^{pt} that is characterized as the output in the PPO-based path tracking is represented by the AUV actuators.

The reward function is defined by four terms: (i) reward (or penalty) for actuators to limit the changes in actuators; (ii) reward for the position following to penalize any error between the actual position and reference position; (iii) reward for velocity following to penalize any error between actual velocity and reference velocity; and (iv) reward for collision avoidance to penalize the collision. The reward function for path tracking is formulated as follows:

$$R^{\mathrm{pt}} = -\varsigma_{\mathrm{a}}(a^{\mathrm{pt}})^2 - \varsigma_{\eta^{\mathrm{L}}}(R_{\eta^{\mathrm{L}}}^{\mathrm{pt}})^2 - \varsigma_{\mathbf{V}^{\mathrm{L}}}(R_{\mathbf{V}^{\mathrm{L}}}^{\mathrm{pt}})^2 - (1-\varsigma_{\eta^{\mathrm{L}}})(R_{\mathrm{ca}}^{\mathrm{pt}})^2 \tag{8}$$

where

$$R_{\eta^{\mathrm{L}}}^{\mathrm{pt}} = \mathrm{clip}(\kappa_{\eta^{\mathrm{L}}}\frac{\eta^{\mathrm{L}} - \eta^{\mathrm{L}}}{\eta^{\mathrm{Lr}}}, -1, +1) \tag{9}$$

$$R_{\mathcal{V}^{\mathrm{L}}}^{\mathrm{pt}} = \mathrm{clip}(\kappa_{\mathcal{V}^{\mathrm{L}}}\frac{\mathcal{V}^{\mathrm{L}} - \mathcal{V}^{\mathrm{L}}}{\boldsymbol{\mathcal{V}}^{\mathbf{Lr}}}, -1, +1) \tag{10}$$

with ς_{a}, $\varsigma_{\eta^{\mathrm{L}}}$, and $\varsigma_{\mathbf{V}^{\mathrm{L}}}$ represent the coefficients for different terms of the reward function; $\kappa_{\eta^{\mathrm{L}}}$ and $\kappa_{\mathcal{V}^{\mathrm{L}}}$ show the coefficients to enlarge the impact of the position and velocity error; η^{Lr} and $\boldsymbol{\mathcal{V}}^{\mathbf{Lr}}$ denote the constant reference values for position and velocity to normalize the errors. The collision avoidance term of the reward function is defined as follows:

$$R_{\mathrm{ca}}^{\mathrm{pt}} = (\delta_{\mathrm{ca}}(\max(1-c, \epsilon_{\mathrm{ca}}))^2)^{-1} \tag{11}$$

where $c = \mathrm{clip}(1 - \frac{d}{d_{\max}}, 0, 1)$ is the obstacle closeness with d showing the distance measurement and $d_{\max}$ denoting the maximum sensor range; δ_{ca} represents the coefficient for the collision avoidance term; and ϵ_{ca} denotes the constant value. The algorithm for the PPO-based path tracking is presented in Algorithm 2 with θ^{pt} representing the PPO-based path tracking network coefficient.

3.3. Integrated path planning and tracking for real-time application

After constructing the optimal PPO networks for path planning and path tracking, these two trained networks are combined as an integrated framework and then applied in a real-time manner to seek the optimal path through PPO-based path planning. Afterward, the optimal path is smoothed to find the reference path and velocity according to the path tracking timestep T^{spt}. Finally, the reference path is given to the PPO-based path tracking module to determine the optimal actuators for an AUV. The algorithm for real-time application of the integrated path planning and tracking is presented in Algorithm 3.

Algorithm 2 PPO for path tracking (training phase)

Input: underwater environment velocity, sonar sensor data, position error, velocity error, and PPO parameters;
for each iteration **do**
  Initialize state $s^{\mathrm{pt}} \leftarrow [v_{\mathrm{e}}\ \mathcal{D}_{ss}\ \Delta\eta^{\mathrm{L}}\ \mathcal{V}^{\mathrm{L}}]$;
  Proceed policy $\pi_{\theta_{\mathrm{o}}^{\mathrm{pt}}}(a^{\mathrm{pt}}|s^{\mathrm{pt}})$ over N^{pt} timesteps and accomplish action a^{pt};
  Compute the path planning reward function R^{pt}:

$$R^{\mathrm{pt}} = -\varsigma_{\mathrm{a}}(a^{\mathrm{pt}})^2 - \varsigma_{\eta^{\mathrm{L}}}(R_{\eta^{\mathrm{L}}}^{\mathrm{pt}})^2 - \varsigma_{\mathbf{V}^{\mathrm{L}}}(R_{\mathbf{V}^{\mathrm{L}}}^{\mathrm{pt}})^2 - (1-\varsigma_{\eta^{\mathrm{L}}})(R_{\mathrm{ca}}^{\mathrm{pt}})^2$$

  Compute the generalized advantage function estimate:

$$\widehat{A}_t^{\mathrm{t}} = \delta_t + (\gamma^{\mathrm{pt}}\lambda^{\mathrm{pt}})\delta_{t+1} + \ldots + (\gamma^{\mathrm{pt}}\lambda^{\mathrm{pt}})^{N^{\mathrm{pt}}-t+1}\delta_{T-1}$$

  Optimize the surrogate function:

$$L^{\mathrm{CLIP}}(\theta^{\mathrm{pt}}) = \widehat{\mathbb{E}}_t[\min(\frac{\pi_{\theta^{\mathrm{pt}}}(a^{\mathrm{pt}}|s^{\mathrm{pt}})}{\pi_{\theta_{\mathrm{o}}^{\mathrm{pt}}}(a^{\mathrm{pt}}|s^{\mathrm{pt}})}\widehat{A}_t, \mathrm{clip}(\frac{\pi_{\theta^{\mathrm{pt}}}(a^{\mathrm{pt}}|s^{\mathrm{pt}})}{\pi_{\theta_{\mathrm{o}}^{\mathrm{pt}}}(a^{\mathrm{pt}}|s^{\mathrm{pt}})}, 1-\epsilon, 1+\epsilon)\widehat{A}_t)]$$

  Update $\theta_{\mathrm{o}}^{\mathrm{pt}} \leftarrow \theta^{\mathrm{pt}}$ every b^{pt} iterations;
end for
Output: optimal PPO for path tracking;

Algorithm 3 PPO for integrated path planning and tracking (real-time application phase)

Input: real-time measured current velocity, real-time sonar sensor data, optimal PPO for path planning, optimal PPO for path tracking;
for each timestep T^{spp} **do**
  Initialize state $s^{\mathrm{pp}} \leftarrow [v_{\mathrm{e}}\ x\ y\ z]$;
  Select action a^{pp} (optimal path) according to the optimal PPO for path planning;
  Output: optimal path;
  Smooth the optimal path and generate the optimal reference vector of position and velocity $[x^*\ y^*\ z^*\ u^*\ v^*\ w^*]^{\mathrm{T}}$;
  for each timestep T^{spt} **do**
    Initialize state $s^{\mathrm{pt}} \leftarrow [v_{\mathrm{e}}\ \mathcal{D}_{ss}\ \Delta\eta^{\mathrm{L}}\ \mathcal{V}^{\mathrm{L}}]$;
    Select action a^{pt} (optimal actuators) according to the optimal PPO for path tracking;
    Output: optimal actuators;
  end for
end for

4. Case study on an energy harvesting AUV: Marine current turbine

4.1. Marine current turbine model

The MCT considered in this study consists of a turbine tethered to an anchor through a mooring cable as shown in Fig. 2. This system is designed to operate in the Gulf Stream off Florida's East Coast to deliver a rated power of 700 kW under nominal operation following the prototype MCTs from IHI Corp. Ueno et al. (2018) and the University of Naples (Coiro et al., 2017). The represented MCT consists of four major elements: body, variable pitch rotor, variable buoyancy tank including two variable buoyancy sections, and mooring cable. The MCT system is primarily controlled to move in a vertical direction.

The investigated MCT system is modeled with 14 states X, consisting of the position $\eta = [x\ y\ z\ \phi\ \theta\ \psi]^{\mathrm{T}}$ and the velocity of the MCT system $\mathcal{V} = [u\ v\ w\ p^{\mathrm{b}}\ q\ r]^{\mathrm{T}}$; two remaining states of the MCT system are the angular velocity of the rotor p^{r}, and rotation angle of the rotor blade ϕ^r, thereby symbolizing the state vector by $X = [\eta\ \mathcal{V}\ p^{\mathrm{r}}\ \phi^r]^{\mathrm{T}}$.

Kinematics and Coordinate Frame: To derive the equations of motion for the MCT system, five coordinate frames are used, including (i) an inertial coordinate frame ($\mathcal{O}_{\mathrm{I}}$), (ii) a body-fixed coordinate frame ($\mathcal{O}_{\mathrm{B}}$), (iii) a momentum mesh coordinate frame ($\mathcal{O}_{\mathrm{M}}$), (iv) a shaft coordinate frame ($\mathcal{O}_{\mathrm{S}}$), and (v) a rotor blade coordinate frame ($\mathcal{O}_{\mathrm{R}}$) (see Fig. 2) (VanZwieten et al., 2012). The transformation matrix from the inertial coordinate frame ($\mathcal{O}_{\mathrm{I}}$) to the body-fixed coordinate frame ($\mathcal{O}_{\mathrm{B}}$), $L_{\mathcal{O}_{\mathrm{I}}}^{\mathcal{O}_{\mathrm{B}}}$, is defined as follows (Fossen, 1999):

$$L_{\mathcal{O}_I}^{\mathcal{O}_B} = \begin{bmatrix} c_\psi c_\theta & s_\psi c_\theta & -s_\theta \\ c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi c_\phi + s_\psi s_\theta s_\phi & c_\theta s_\phi \\ c_\psi s_\theta c_\phi - s_\psi s_\phi & -c_\psi s_\phi + s_\psi s_\theta c_\phi & c_\theta c_\phi \end{bmatrix} \tag{12}$$

$$\dot{\mathcal{V}} = \begin{bmatrix} m & 0 & 0 & 0 & m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}} & 0 \\ 0 & m & 0 & -m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}} & 0 & mx_{\mathrm{G}} \\ 0 & 0 & m & 0 & -mx_{\mathrm{G}} & 0 \\ 0 & -m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}} & 0 & I_x^{\mathrm{b}} & 0 & -I_{xz}^{\mathrm{b}} \\ m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}} & 0 & -mx_{\mathrm{G}} & 0 & I_y & 0 \\ 0 & mx_{\mathrm{G}} & 0 & -I_{xz}^{\mathrm{b}} & 0 & I_z \end{bmatrix}^{-1} \begin{bmatrix} F_x + m(vr - wq) + mx_{\mathrm{G}}\left(q^2 + r^2\right) - m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}}p^{\mathrm{b}}r \\ F_y - mur + w\left(m^{\mathrm{b}}p^{\mathrm{b}} + m^{\mathrm{r}}p^{\mathrm{r}}\right) - m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}}qr \\ -m^{\mathrm{b}}x_{\mathrm{G}}^{\mathrm{b}}qp^{\mathrm{b}} - m^{\mathrm{r}}x_{\mathrm{G}}^{\mathrm{r}}qp^{\mathrm{r}} \\ F_z + muq - v\left(m^{\mathrm{b}}p^{\mathrm{b}} + m^{\mathrm{r}}p^{\mathrm{r}}\right) + m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}}\left(p^{\mathrm{b}2} + q^2\right) \\ -m^{\mathrm{b}}x_{\mathrm{G}}^{\mathrm{b}}rp^{\mathrm{b}} - m^{\mathrm{r}}x_{\mathrm{G}}^{\mathrm{r}}rp^{\mathrm{r}} \\ M_x + M_x^{\mathrm{s}} - qr\left(I_z^{\mathrm{b}} - I_y^{\mathrm{b}}\right) + I_{xz}^{\mathrm{B}}p^{\mathrm{b}}q \\ -m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}}\left(wp^{\mathrm{b}} - ur\right) \\ M_y - rp^{\mathrm{b}}\left(I_x^{\mathrm{b}} - I_z^{\mathrm{b}}\right) - rp^{\mathrm{r}}\left(I_x^{\mathrm{r}} - I_z^{\mathrm{r}}\right) - I_{xz}^{\mathrm{b}}\left(p^{\mathrm{b}2} - r^2\right) \\ +m^{\mathrm{b}}z_{\mathrm{G}}^{\mathrm{b}}(vr - wq) - mx_{\mathrm{G}}uq + m^{\mathrm{b}}x_{\mathrm{G}}^{\mathrm{b}}vp^{\mathrm{b}} + m^{\mathrm{r}}x_{\mathrm{G}}^{\mathrm{r}}vp^{\mathrm{r}} \\ M_z - qp^{\mathrm{b}}\left(I_y^{\mathrm{b}} - I_x^{\mathrm{b}}\right) - qp^{\mathrm{r}}\left(I_y^{\mathrm{r}} - I_x^{\mathrm{r}}\right) - I_{xz}^{\mathrm{b}}rq \\ -mx_{\mathrm{G}}\ \mathrm{u}\ r + m^{\mathrm{b}}x_{\mathrm{G}}^{\mathrm{b}}wp^{\mathrm{b}} + m^{\mathrm{r}v}x_{\mathrm{G}}^{\mathrm{r}}wp^{\mathrm{r}} \end{bmatrix} \tag{13}$$

$$\dot{p}_{\mathrm{r}} = \frac{M_{x_{\mathrm{r}}} - M_x^{\mathrm{s}} - qr(I_{z_{\mathrm{r}}} - I_{y_{\mathrm{r}}})}{I_{x_{\mathrm{r}}}} \tag{14}$$

where $s_{(.)} = sin(.)$ and $c_{(.)} = cos(.)$.

Equations of Motion: Given that the motion of the MCT system is defined in $\mathcal{O}_{\mathrm{B}}$, instead of the rotation about the x-axis, the location of the center of gravity and center of buoyancy of the MCT system are represented by $r_{\mathrm{G}} = [x_{\mathrm{G}}\ y_{\mathrm{G}}\ z_{\mathrm{G}}]^{\mathrm{T}}$ and $r_{\mathrm{B}} = [x_{\mathrm{B}}\ y_{\mathrm{B}}\ z_{\mathrm{B}}]^{\mathrm{T}}$, respectively. A set of twelve equations of motion are reduced to the seven equations representing an MCT system with 7-DOF (VanZwieten et al., 2012), with 6-DOF describing the main body's rotation, and the last DOF representing the rotor's rotation about the x-axis. These equations of motion are summarized in a matrix form as presented in .

In this matrix representation 13, $F_{(.)}$ denotes the force; $M_{(.)}$ is the moment; $(.)_x$, $(.)_y$, and $(.)_z$ are the portion $(.)$ about $x-$, $y-$, and $z-$ axes; $(.)^{\mathrm{r}}$ and $(.)^{\mathrm{b}}$ denote the rotor and body portions; M_x^s denotes the electromechanical torque. The mass $m^{(.)}$, the moment of inertia $I^{(.)}$, and the center of gravity $(.)_{\mathrm{G}}$ are defined with respect to both the actual inertial properties and added inertial properties of the MCT (denoted as *virtual* in VanZwieten et al. (2012)).

The total external forces acting on the MCT, F, consists of forces due to gravitational and buoyancy forces, rotor force, body force, variable buoyancy force, and tether force, namely:

$$F = F^{\mathrm{GB}} + F^{\mathrm{r}} + F^{\mathrm{b}} + F^{\mathrm{vb}} + F^{\mathrm{t}} \tag{15}$$

Similarly, the total moment acting about the center of mass of the MCT is equal to the sum of moments due to buoyancy moment, rotor moment, body moment, variable buoyancy moment, and tether moment, as follows:

$$M = M^{\mathrm{B}} + M^{\mathrm{r}} + M^{\mathrm{b}} + M^{\mathrm{vb}} + M^{\mathrm{t}} \tag{16}$$

The forces and moments acting on the MCT system defined through the hydrostatics, hydrodynamic, and tether forces are detailed in VanZwieten et al. (2012).

Linear Model: The linear model of the MCT system is constructed by averaging the equations of motion around the equilibrium point and

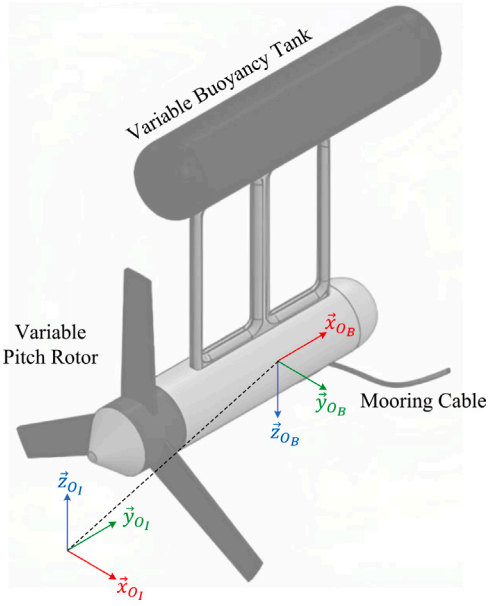

Fig. 2. Schematic of the marine current turbine system representing the inertial frame ($\mathcal{O}_{\mathrm{I}}$) and the body-fixed frame ($\mathcal{O}_{\mathrm{B}}$).

Table 1
Dimensions of a simulated buoyancy controlled marine current turbine.

Symbol	Description	Unit	Value
I_x	Moment of Inertia about x	kg m^3	1.35×10^7
I_y	Moment of Inertia about y	kg m^3	4.74×10^7
I_z	Moment of Inertia about z	kg m^3	3.45×10^7
I_x^r	Moment of Inertia of rotor about x-axis	kg m^3	4.78×10^5
m	Total Mass not including buoyancy water	kg	4.98×10^5
B	Buoyancy	kN	3.2×10^4
d^r	Rotor Diameter	m	20
d^b	Body Average Diameter	m	3
l^b	Body Length	m	10.653
l^c	Cable Length	m	607
d^c	Cable Diameter	m	0.16
v^{vb}	Volume of each buoyancy tank	m^3	31.215
$z^{\min}$	Minimum bound of vertical position	m	50
$z^{\max}$	Maximum bound of vertical position	m	150
w	Maximum linear velocity about z	m/s	0.21
$\dot{w}$	Maximum linear acceleration about z	m/s^2	0.0015

the homogeneous current speed of 1.6 m/s. For the MCT linear model, the rotation angle of the rotor blade ϕ^{r} is removed, and the number of states is decreased from 14 states in the nonlinear model to 13 states by averaging the MCT response over one rotor blade rotation. Also, the control input vector is established by the MCT actuators, including forward and aft buoyancy tank fill fractions b_f and b_a, as well as the electromechanical torque M_x^s, i.e., $U = [b_f \ b_a \ M_x^s]^{\mathrm{T}}$.

4.2. Gulf stream environment model

To model the current flow speed of the Gulf Stream off Florida's East Coast, the historical observations are collected by a 75 kHz ADCP, recorded at a latitude of 26.09° N and longitude of −79.80° E with a resolution of 6 m within 400 m depth (see Fig. 3). Given that the bad data (primarily happen above a depth of 50 m) is removed through filtering the measured data (Maria Carolina et al., 2016).

4.3. MCT-specific integrated control

The MCT is primarily controlled in the vertical direction, so the movement is limited to the 1D direction. Hence, the PPO-based path planner input is characterized by $[z]$ and the PPO-based path tracker input is defined by $[z^* \ w^*]^{\mathrm{T}}$, where the remaining inputs are similar to the general framework introduced in Section 2.2. The actuators are then updated subject to the MCT actuators, i.e., $[b_f \ b_a \ M_x^s]^{\mathrm{T}}$.

Table 2
Parameters of PPO-based path planning and PPO-based path tracking.

Symbol	Description	Unit	Value
Path planning			
γ^{pp}	Discount factor	–	0.5
λ^{pp}	Scaling parameter	–	0.9
ϑ_P	Coefficient of power term in (5)	–	0.8
ϑ_v	Coefficient of velocity term in (5)	–	0.2
P^{d}	Desired power in (6)	kW	700
$v_{\mathrm{e}}^{\mathrm{d}}$	Desired velocity in (7)	m/s	2
Path tracking			
γ^{pt}	Discount factor	–	0.6
λ^{pt}	Scaling parameter	–	0.95
ς_{a}	Coefficient of action term in (8)	–	0.1
$\varsigma_{\eta^{\mathrm{L}}}$	Coefficient of linear position term in (8)	–	0.5
$\varsigma_{\mathcal{V}^{\mathrm{L}}}$	Coefficient of linear velocity term in (8)	–	0.5
η^{Lr}	Constant position in (9)	m	20
$\kappa_{\eta^{\mathrm{L}}}$	Coefficient of position error in (9)	–	200
$\mathcal{V}^{\mathrm{Lr}}$	Constant velocity in (10)	m/s	0.0056
$\kappa_{\mathcal{V}^{\mathrm{L}}}$	Coefficient of velocity error in (10)	–	50
δ_{ca}	Coefficient of collision avoidance in (11)	–	2
ϵ_{ca}	Constant value in (11)	–	0.05
$d_{\max}$	Sonar sensor range in (11)	m	20

5. Simulation results

5.1. Simulation setup

The simulations are implemented in Python 3.7 and Tensorflow 1.14 on a PC with a 2.6 GHz CPU and 16 GB of RAM. The parameters of the MCT system and PPO networks are presented below.

MCT system: The primary dimensions of the simulated MCT system are presented in Table 1. The MCT constraints, i.e., maximum linear velocity and linear acceleration about z, as well as the minimum and maximum movement bound of the MCT, are shown in this table. The MCT system operates in the Gulf Stream off Florida's East Coast, where the real ocean current speed data are recorded by an ADCP (as discussed in Section 4.2).

PPO-based benchmarks: Two PPO networks are applied for path planning and path tracking. The sampling time for path tracking is $T^{\mathrm{spt}} = 2$ s, and the parameters of the PPO-based path tracking are presented in Table 2. Furthermore, the sampling time for path planning is $T^{\mathrm{spp}} = 60$ min, where the path planning parameters are listed in Table 2. Note that the path smoother module takes care of smoothing the path to shift from the path planning time step to the path tracking time step.

5.2. Quantitative results

PPO-based Path Planning: To evaluate the performance of the path planning for the MCT system, we compare with multiple methods, including A* algorithm, model predictive control (MPC) algorithm, and two other candidate RL algorithms, i.e., Q-learning, and deep Q-network (DQN). Those methods have been previously applied to MCT either path planning or path tracking, detailed in Hasankhani et al. (2023, 2021a). These baseline methods are briefly introduced here:

- A* Algorithm: This algorithm finds the optimal path in the ocean environment modeled with a discrete grid of depth, which utilizes a greedy strategy to find the maximum power and, accordingly the optimal path.

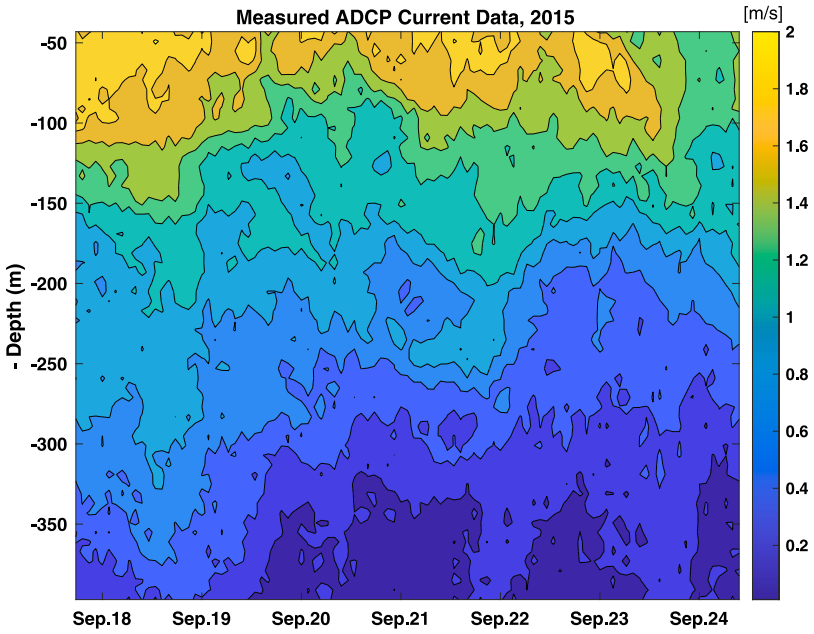

Fig. 3. Histories of the flow recorded by a 75 kHz ADCP at a latitude of 26.09° N and longitude of −79.80° E (Maria Carolina et al., 2016).

- MPC Algorithm: In the MPC algorithm, power maximization is defined as an objective function to seek the optimal path over a prediction horizon subject to the operational constraints of the MCT system.
- Q-learning Algorithm: This algorithm solves the path planning optimization problem through a constructed Q-value table, where the Q-value is calculated for each cell of the Q-value table representing a discrete depth at a specific time.
- DQN Algorithm: In the DQN algorithm, the Q-value is approximated through the neural networks to avoid computational complexity arising from the Q-value table in a large environment. Note that both DQN and Q-learning algorithms are defined for the discrete environment.

We first show the cumulative reward obtained in the offline training of the PPO-based path planning and DQN-based path planning in Fig. 4. As shown in this figure, the PPO needs almost three times more episodes to be fully trained than the DQN algorithm, which is predictable due to the continuous states and actions defined for the PPO algorithm compared to the finite number of actions (depths) and states (current depth and ocean current velocity in the discrete depths) in the DQN algorithm. From the figure, we can observe the convergence of both rewards, confirming successful training.

Moreover, the reward values are different for the PPO algorithm (left axis in Fig. 4) from the DQN reward values (right axis in Fig. 4) since the reward function defined for the DQN algorithm is different from the PPO reward function (5) to improve the performance of the DQN algorithm according to its discrete nature (see Hasankhani et al. (2023) for more details). To verify the efficiency of two reward functions for DQN and PPO algorithms, similar trends in increasing the cumulative energy of MCT during training are illustrated. The DQN reward function includes two terms of power and velocity, which are defined as follows:

$$R^{\mathrm{DQN}} = R_{\mathrm{P}}^{\mathrm{DQN}} + 0.5 R_{\mathrm{v}}^{\mathrm{DQN}} \tag{17}$$

$$R_{\mathrm{P}}^{\mathrm{DQN}} = \begin{cases} \zeta_1, & \Delta P_{\mathrm{net}} > \delta_1 \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

$$R_{\mathrm{v}}^{\mathrm{DQN}} = \begin{cases} \zeta_2, & \Delta v_{\mathrm{e}} > \delta_2 \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

with $\zeta_1 = \zeta_2 = 1$, $\delta_1 = 1$ kW, $\delta_2 = 0.001$ m/s, and ΔP_{net} and Δv_{e} showing changes in the net power and ocean current velocity due to changes in the depth.

The comparative results on the optimal depths, optimal ocean current velocity, and optimal power for different algorithms are represented in Fig. 5. The optimal path chosen by each algorithm verifies different policies of multiple approaches, where the A* algorithm tends to pick sharp changes in the vertical position for the MCT. However, the MPC algorithm limits the MCT movement, still experiencing high values of harnessed power than the A* algorithm. The Q-learning and DQN algorithms show almost the same performance with minor differences, justifying that the DQN algorithm successfully estimates the Q-value table, but the precision of these algorithms are limited to the discrete depths. Finally, the PPO algorithm outperforms other methods, finding the optimal path with the maximum power. The cumulative harnessed energy after 100-hour operation is 29.799 MWh (A*), 31.089 MWh (MPC), 32.250 (QL), 32.168 (DQN), and 34.930 (PPO).

It should be noted that we consider two modes of application for the PPO-based path planning: (i) offline path planning: the optimal path is planned offline, and (ii) online path planning: in case the offline planned path is not tracked, the optimal path will be re-planned online (The importance of the second case will be highlighted in the collision avoidance scenario).

PPO-based Path Tracking: We evaluate the PPO-based path tracking to follow the optimal path commanded by the PPO-based path planning. It should be noted that we first evaluate the capability of the path tracking algorithm to follow a reference path successfully; then, the collision avoidance is simulated in the next section. A PI controller is introduced as a baseline algorithm to assess the performance of our proposed algorithm. The main objective of the path tracking module is to minimize the tracking error while the fill fractions (MCT's actuators) remain within the allowable limit. It is noteworthy to mention that the electromechanical torque is set to stay constant since the fill fractions are the main actuators affecting the MCT.

The path tracking results for a sample reference path over 24 h through the PPO-based path tracking and PI algorithm are shown in Fig. 6 (Basic Scenario). The simulated results confirm a successful path tracking for both the PI controller and PPO-based path tracking module, where the main error for the PI controller happens at the beginning of the tracking procedure. Also, the actuators in the PI controller change

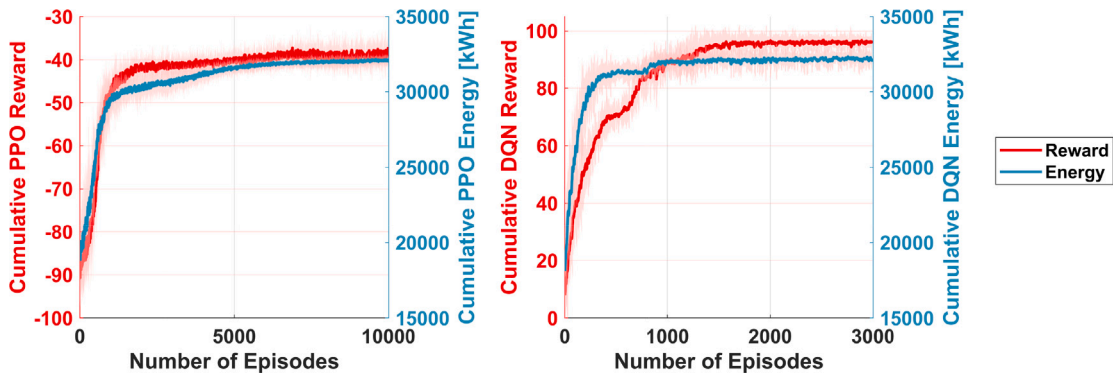

Fig. 4. Cumulative reward and energy for (i) PPO-based path planning; and (ii) DQN-based path planning.

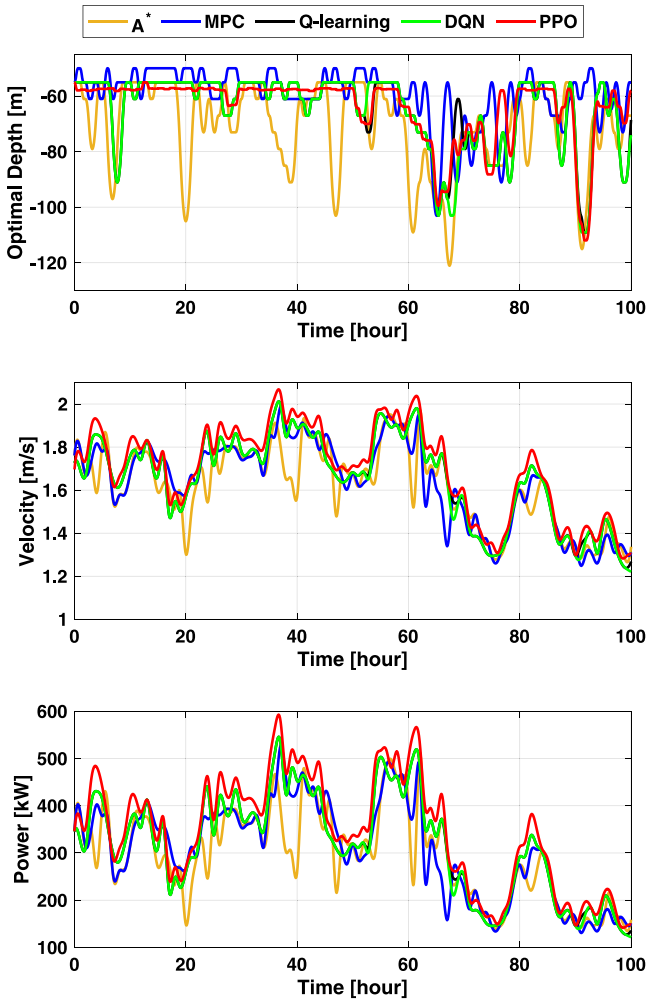

Fig. 5. Comparing optimal results obtained by PPO-based path planning over 100 h with A*, MPC, Q-learning, and DQN algorithms. (a) Optimal vertical path, (b) Optimal velocity, and (c) Optimal power.

in a larger interval than the PPO algorithm, still keeping within the allowable range. Meanwhile, for the PPO-based path tracking method, the tracking error is mostly visible within $t = 5$ to $t = 10$ while making an effort to minimize the changes in the fill fractions. The main reason for this small error is that we use a single trained PPO network for path tracking to meet both minimizing tracking error and collision avoidance; the PPO network is trained under different scenarios of path tracking and obstacles, and the trained PPO network is the best one to operate for both successful path tracking and collision avoidance. Hence, some small errors in path tracking are anticipated.

PPO-based Path Tracking with Collision Avoidance: The second primary task for the PPO-base path tracking module is to avoid collision with an obstacle, which is the main motivation to apply a learning-based and intelligent path tracking algorithm. An intelligent RL-based path tracking is capable of identifying and avoiding an obstacle, unlike the conventional PI controller. In this case, we simulate two scenarios of a stationary obstacle at a constant depth and a dynamic obstacle changing its depth, where the MCT operating depth is demonstrated regarding the reference path and the obstacle. It should be noted that in this scenario, the path tracking module completed an offline training phase considering both stationary and dynamic obstacles, where the reward term for collision avoidance in (8) is enabled.

The path tracking results and MCT actuators for the static obstacle are summarized in Fig. 6 (Static Obstacle Scenario), where the obstacle remains at a depth of 66 m. Two application modes of PPO path planner are active in this scenario. For an offline path planner, the reference path hits the obstacle at four points, where the path tracking module updates the MCT's actual path to avoid collision. Hence, the MCT system keeps its operating depth near the obstacle at an acceptable distance to ensure safe movement. Also, the MCT continues following the optimal path when its sensor does not detect the obstacle at $t = 10$, justifying the intelligence of the PPO-based path tracking to distinguish the collision avoidance and path tracking scenarios. For the online path planner, the reference path is re-planned after detecting the obstacle to avoid the collision. Both modes of application are successful in collision avoidance, while the online path planner is able to re-planning the path for harnessing the maximum power.

The final scenario interprets a dynamic moving obstacle (like a large fish) in a vertical direction, where the simulation results are presented in Fig. 6 (Dynamic Obstacle Scenario). The dynamic obstacle moves between four different depths, so the task of the path tracking module is complicated to see the obstacle, identify the case of collision occurrence, and avoid this case, as well as follow the optimal path during a safe movement. As the results show in the offline planner, the path tracking module successfully detects the collision scenarios in four cases, such as an interval between $t = 4$ to $t = 10$, and defines a new reference path for the MCT to stay near the obstacle. Also, the path tracking module follows the optimal path after $t = 10$ when the PPO ensures a safe path without collision, thereby fulfilling its primary task of path tracking with minimized error. In the online planner, the planned path is actively updated according to the obstacle. During the path tracking scenario with collision avoidance, the fill fractions are also kept within the allowable limits.

5.3. Discussion

The PPO-based path planning and tracking framework, as a candidate among RL algorithms, entails offline training with a dataset

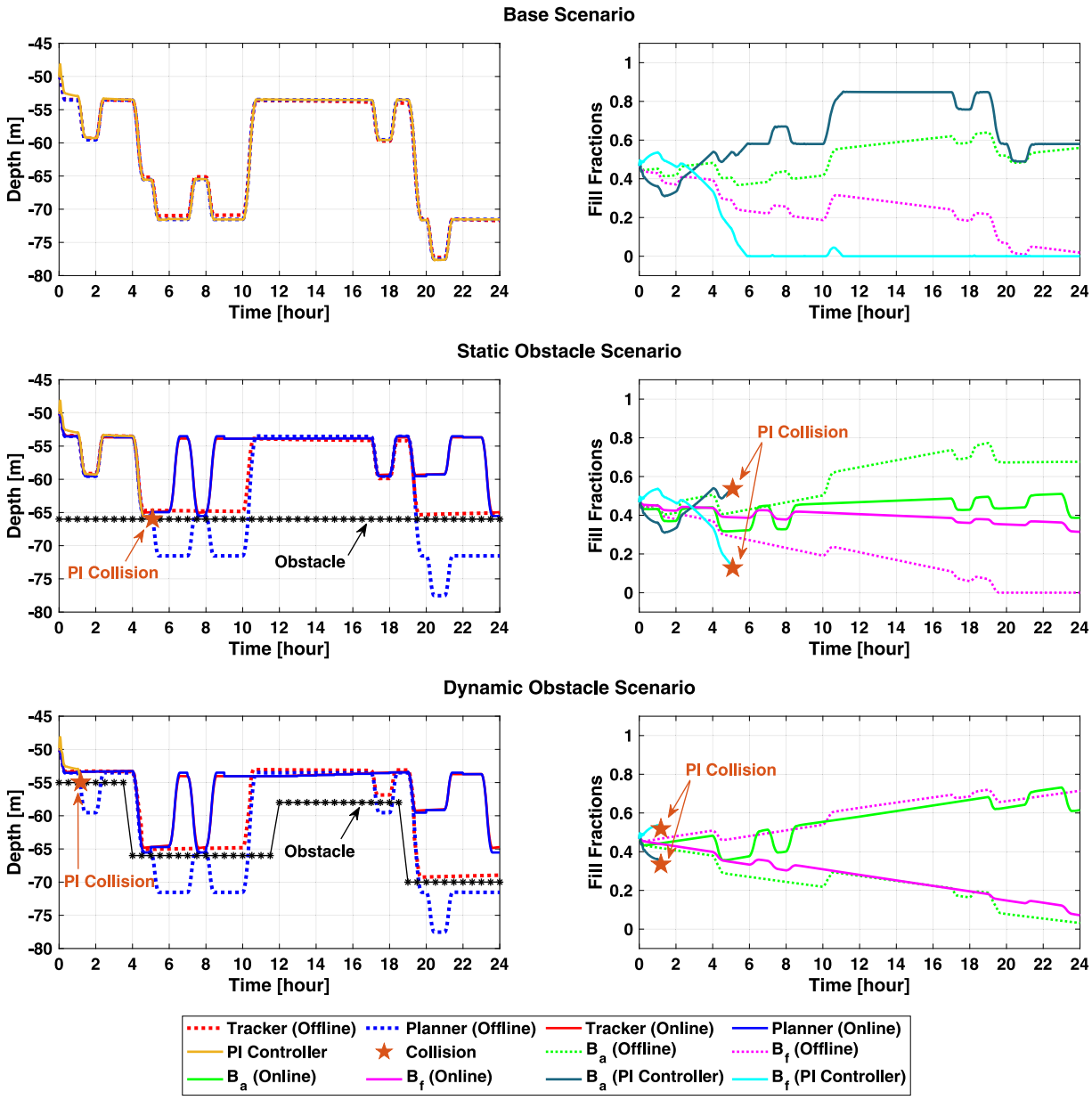

Fig. 6. Simulation results for integrated path planning and path tracking control: (i) Base Scenario: The left plot shows the path followed by the PPO-based path tracking and PI controller along with the reference optimal path obtained by the PPO-based path planning. The right plot shows the actuators (fill fractions) for the PPO-based path tracking and PI controller. (ii) Static Obstacle Scenario: The left plot shows the path followed by the PPO-based path tracking along with the reference optimal path obtained by the offline and online PPO-based path planning and obstacle. The right plot shows the actuators (fill fractions) for the PPO-based path tracking and PI controller. The collision that occurred in the case of the PI controller is shown. (iii) Dynamic Obstacle Scenario: The left plot shows the path followed by the PPO-based path tracking along with the reference optimal path obtained by the offline and online PPO-based path planning and obstacle. The right plot shows the actuators (fill fractions) for the PPO-based path tracking and PI controller. Again, the collision that occurred in the case of the PI controller is illustrated.

of the recorded ocean current velocity, which is then applied in an online operation of the MCT system. The main feature and superiority of the RL algorithm to other approaches in both path planning and path tracking tasks are its capability of learning from experiencing different scenarios and then making the best decision (choosing the best action) in any similar scenario. In the path planning task, the PPO-based path planning module shows better performance than the MPC and A* algorithms by learning from the real recorded ocean current velocity data. Also, the PPO-based path planning considers a continuous set of actions (depths), which improves its ability to solve the path planning problem and find the optimal power than the discrete algorithms with limited choices in the depth changes, such as Q-learning and DQN algorithms.

The path tracking module enabled with the PPO algorithm outperforms a classical PI controller considering the complicated task of collision avoidance. In this case, the path tracking module should be qualified with intelligence to detect the obstacle and distinguish the collision avoidance and path tracking scenarios. More specifically, the path tracking module should follow the commanded path with a minimized error; still, it would avoid the collision facing an obstacle. The PPO-based path tracking for the MCT can detect both stationary and dynamic obstacles and avoid collision while keep following the optimal path in the absence of the obstacle.

Although the proposed approach is a successful attempt to address path planning and tracking control for the MCT system, we need further details to meet all the complexities in the real-world application. The first limitation is that the current approach uses the MCT's linear model,

which is precise enough, still requires further analysis and a comparison with the model enabled with the dynamic model. The proposed approach relies on the assumption that the ocean environment is fully observable, whereas the real ocean environment deals with partial observability, which should be considered in future studies. Also, the proposed approach should be tested for other collision scenarios, which can verify the performance or detect the probable limitations. This testing procedure would be necessary to justify the generalization of our proposed approach and ensure a safe operation for various conditions.

6. Conclusions

In this paper, we presented an integrated path planning and tracking control framework for turbines operating in a dynamic marine environment, where the system was treated as an energy-harvesting autonomous underwater vehicle. The whole framework was designed based on the proximal policy optimization, where the main objective of the path planning module was to find the optimal path with the maximum energy harvesting, and the path tracking module was trained to follow the optimal path with minimum error and avoid a collision. The simulation results verified the successful operation of the proposed framework in comparison with several baseline approaches in different scenarios of path planning, simple path tracking, and path tracking with collision avoidance. Future work is anticipated to extend the integrated path planning and tracking framework to apply to other energy-harvesting AUVs. Also, in the proposed framework, we currently use the linear model of the MCT, which can be replaced with a dynamic and nonlinear model of the MCT. The presented framework can become more mature by proposing a solution for partial observability in an underwater environment.

CRediT authorship contribution statement

Arezoo Hasankhani: Conceptualization, Methodology, Original draft preparation. **Yufei Tang:** Supervision, Review & editing. **James VanZwieten:** Data creation, Review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yufei Tang reports financial support was provided by National Science Foundation. Yufei Tang reports financial support was provided by US Department of Energy. Yufei Tang reports a relationship with National Science Foundation that includes: funding grants.

Data availability

Data will be made available on request.

References

Aguiar, A. Pedro, Hespanha, Joao P., 2007. Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. IEEE Trans. Automat. Control 52 (8), 1362–1379.

Ali, M.S. Ajmal Deen, Babu, N. Ramesh, Varghese, Koshy, 2005. Collision free path planning of cooperative crane manipulators using genetic algorithm. J. Comput. Civ. Eng. 19 (2), 182–193.

Antonelli, Gianluca, Chiaverini, Stefano, Sarkar, Nilanjan, West, Michael, 2001. Adaptive control of an autonomous underwater vehicle: experimental results on ODIN. IEEE Trans. Control Syst. Technol. 9 (5), 756–765.

Bafande, Alireza, Vermillion, Chris, 2016. Altitude optimization of airborne wind energy systems via switched extremum seeking—design, analysis, and economic assessment. IEEE Trans. Control Syst. Technol. 25 (6), 2022–2033.

Bin-Karim, Shamir, Bafandeh, Alireza, Baheri, Ali, Vermillion, Christopher, 2017. Spatiotemporal optimization through gaussian process-based model predictive control: A case study in airborne wind energy. IEEE Trans. Control Syst. Technol. 27 (2), 798–805.

Bortoff, Scott A., 2000. Path planning for UAVs. In: Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No. 00CH36334), Vol. 1, No. 6. IEEE, pp. 364–368.

Chang, Seong-Ryong, Huh, Uk-Youl, 2015. G 2 continuity smooth path planning using cubic polynomial interpolation with membership function. J. Electr. Eng. Technol. 10 (2), 676–687.

Cheng, Shuo, Li, Liang, Chen, Xiang, Wu, Jian, et al., 2020. Model-predictive-control-based path tracking controller of autonomous vehicle considering parametric uncertainties and velocity-varying. IEEE Trans. Ind. Electron. 68 (9), 8698–8707.

Cho, Gun Rae, Li, Ji-Hong, Park, Daegil, Jung, Je Hyung, 2020. Robust trajectory tracking of autonomous underwater vehicles using back-stepping control and time delay estimation. Ocean Eng. 201, 107131.

Cobb, Mitchell, Reed, James, Daniels, Joshua, Siddiqui, Ayaz, Wu, Max, Fathy, Hosam, Barton, Kira, Vermillion, Chris, 2021. Iterative learning-based path optimization with application to marine hydrokinetic energy systems. IEEE Trans. Control Syst. Technol.

Coiro, DP, Troise, G, Scherillo, F, De Marco, A, Calise, G, Bizzarrini, N, 2017. Development, deployment and experimental test on the novel tethered system GEM for tidal current energy exploitation. Renew. Energy 114, 323–336.

Debnath, Sanjoy Kumar, Omar, Rosli, Latip, Nor Badariyah Abdul, 2019. A review on energy efficient path planning algorithms for unmanned air vehicles. In: Computational Science and Technology. Springer, pp. 523–532.

Di Franco, Carmelo, Buttazzo, Giorgio, 2015. Energy-aware coverage path planning of UAVs. In: 2015 IEEE International Conference on Autonomous Robot Systems and Competitions. IEEE, pp. 111–117.

Dijkstra, Edsger W., et al., 1959. A note on two problems in connexion with graphs. Numer. Math. 1 (1), 269–271.

Falcone, Paolo, Borrelli, Francesco, Tseng, H Eric, Asgari, Jahan, Hrovat, Davor, 2008. A hierarchical model predictive control framework for autonomous ground vehicles. In: 2008 American Control Conference. IEEE, pp. 3719–3724.

Ferguson, Dave, Likhachev, Maxim, Stentz, Anthony, 2005. A guide to heuristic-based path planning. In: Proceedings of the International Workshop on Planning under Uncertainty for Autonomous Systems, International Conference on Automated Planning and Scheduling. ICAPS, pp. 9–18.

Fossen, Thor I., 1999. Guidance and Control of Ocean Vehicles (Doctors thesis). University of Trondheim, Norway, ISBN: 0 471 94113 1, Printed By John Wiley & Sons, Chichester, England.

Fossen, Thor I., 2011. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons.

Fossen, Thor I., Breivik, Morten, Skjetne, Roger, 2003. Line-of-sight path following of underactuated marine craft. IFAC Proc. Vol. 36 (21), 211–216.

Fossen, Thor I., Lekkas, Anastasios M., 2017. Direct and indirect adaptive integral line-of-sight path-following controllers for marine craft exposed to ocean currents. Internat. J. Adapt. Control Signal Process. 31 (4), 445–463.

Fossen, Thor I., Pettersen, Kristin Y., Galeazzi, Roberto, 2014. Line-of-sight path following for dubins paths with adaptive sideslip compensation of drift forces. IEEE Trans. Control Syst. Technol. 23 (2), 820–827.

Geraerts, Roland, Overmars, Mark H., 2004. A comparative study of probabilistic roadmap planners. In: Algorithmic Foundations of Robotics V. Springer, pp. 43–57.

Guerrero, Jesus, Torres, Jorge, Creuze, Vincent, Chemori, Ahmed, 2019. Trajectory tracking for autonomous underwater vehicle: An adaptive approach. Ocean Eng. 172, 511–522.

Hadi, Behnaz, Khosravi, Alireza, Sarhadi, Pouria, 2022. Deep reinforcement learning for adaptive path planning and control of an autonomous underwater vehicle. Appl. Ocean Res. 129, 103326.

Hart, Peter E., Nilsson, Nils J., Raphael, Bertram, 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Sci. Cybern. 4 (2), 100–107.

Hasankhani, Arezoo, Ondes, Ertugrul Baris, Tang, Yufei, Sultan, Cornel, VanZwieten, James, Accepted. Integrated path planning and tracking control of marine current turbine in uncertain ocean environments. In: 2022 Annual American Control Conference. ACC.

Hasankhani, Arezoo, Tang, Yufei, VanZwieten, James, Sultan, Cornel, 2021a. Comparison of deep reinforcement learning and model predictive control for real-time depth optimization of a lifting surface controlled ocean current turbine. In: 2021 IEEE Conference on Control Technology and Applications. CCTA, IEEE, pp. 301–308.

Hasankhani, Arezoo, Tang, Yufei, VanZwieten, James, Sultan, Cornel, 2023. Spatiotemporal optimization for vertical path planning of an ocean current turbine. IEEE Transactions on Control Systems Technology 31 (2), 587–601. http://dx.doi.org/10.1109/TCST.2022.3193637.

Hasankhani, Arezoo, VanZwieten, James, Tang, Yufei, Dunlap, Broc, De Luera, Alexandra, Sultan, Cornel, Xiros, Nikolaos, 2021b. Modeling and numerical simulation of a buoyancy controlled ocean current turbine. Int. Mar. Energy J. 4 (2), 47–58.

Havenstrøm, Simen Theie, Rasheed, Adil, San, Omer, 2021. Deep reinforcement learning controller for 3D path following and collision avoidance by autonomous underwater vehicles. Front. Robot. AI 7, 211.

He, Zichen, Dong, Lu, Sun, Changyin, Wang, Jiawei, 2021. Asynchronous multithreading reinforcement-learning-based path planning and tracking for unmanned underwater vehicle. IEEE Trans. Syst. Man Cybern.

Ji, Jie, Khajepour, Amir, Melek, Wael William, Huang, Yanjun, 2016. Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints. IEEE Trans. Veh. Technol. 66 (2), 952–964.

Jin, Sangrok, Kim, Jihoon, Kim, Jongwon, Seo, TaeWon, 2015. Six-degree-of-freedom hovering control of an underwater robotic platform with four tilting thrusters via selective switching control. IEEE/ASME Trans. Mechatronics 20 (5), 2370–2378. http://dx.doi.org/10.1109/TMECH.2014.2378286.

Koenig, Sven, Likhachev, Maxim, 2002. Dˆ* lite. Aaai/Iaai 15.

Krell, Evan, King, Scott A., Carrillo, Luis Rodolfo Garcia, 2022. Autonomous surface vehicle energy-efficient and reward-based path planning using particle swarm optimization and visibility graphs. Appl. Ocean Res. 122, 103125.

LaValle, Steven M., et al., 1998. Rapidly-Exploring Random Trees: a New Tool for Path Planning. Ames, IA, USA.

Lee, Min Cheol, Park, Min Gyu, 2003. Artificial potential field based path planning for mobile robots using a virtual obstacle concept. In: Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Vol. 2. AIM 2003, IEEE, pp. 735–740.

Li, Huiping, Yan, Weisheng, 2016. Model predictive stabilization of constrained underactuated autonomous underwater vehicles with guaranteed feasibility and stability. IEEE/ASME Trans. Mechatronics 22 (3), 1185–1194.

Likhachev, Maxim, Gordon, Geoffrey J., Thrun, Sebastian, 2003. Ara*: Anytime a* with provable bounds on sub-optimality. Adv. Neural Inf. Process. Syst. 16, 767–774.

Maria Carolina, P.M. Machado, VanZwieten, James H., Pinos, Isabella, 2016. A measurement based analyses of the hydrokinetic energy in the gulf stream. J. Ocean Wind Energy 3 (1), 25–30.

Marrtinsen, Andreas B., Lekkas, Anastasios M., 2018. Curved path following with deep reinforcement learning: Results from three vessel models. In: OCEANS 2018 MTS/IEEE Charleston. IEEE, pp. 1–8.

Martinsen, Andreas B., Lekkas, Anastasios M., 2018. Straight-path following for underactuated marine vessels using deep reinforcement learning. IFAC-PapersOnLine 51 (29), 329–334.

Meyer, Eivind, Heiberg, Amalie, Rasheed, Adil, San, Omer, 2020a. COLREG-compliant collision avoidance for unmanned surface vehicle using deep reinforcement learning. IEEE Access 8, 165344–165364.

Meyer, Eivind, Robinson, Haakon, Rasheed, Adil, San, Omer, 2020b. Taming an autonomous surface vehicle for path following and collision avoidance using deep reinforcement learning. IEEE Access 8, 41466–41481.

Mu, Dongdong, Wang, Guofeng, Fan, Yunsheng, Bai, Yiming, Zhao, Yongsheng, 2018. Fuzzy-based optimal adaptive line-of-sight path following for underactuated unmanned surface vehicle with uncertainties and time-varying disturbances. Math. Probl. Eng. 2018.

Roberge, Vincent, Tarbouchi, Mohammed, Labonté, Gilles, 2012. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. IEEE Trans. Ind. Inform. 9 (1), 132–141.

Schulman, John, Wolski, Filip, Dhariwal, Prafulla, Radford, Alec, Klimov, Oleg, 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

Shen, Chao, Shi, Yang, Buckham, Brad, 2017. Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control. IEEE Trans. Ind. Electron. 65 (7), 5796–5805.

Steinhauser, Armin, Swevers, Jan, 2018. An efficient iterative learning approach to time-optimal path tracking for industrial robots. IEEE Trans. Ind. Inform. 14 (11), 5200–5207.

Stentz, Anthony, et al., 1995. The focussed dˆ* algorithm for real-time replanning. In: IJCAI, Vol. 95. pp. 1652–1659.

Sun, Yushan, Zhang, Chenming, Zhang, Guocheng, Xu, Hao, Ran, Xiangrui, 2019. Three-dimensional path tracking control of autonomous underwater vehicle based on deep reinforcement learning. J. Mar. Sci. Eng. 7 (12), 443.

Truong, Thanh Nguyen, Vo, Anh Tuan, Kang, Hee-Jun, 2021. A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators. IEEE Access 9, 31921–31931.

Tuncer, Adem, Yildirim, Mehmet, 2012. Dynamic path planning of mobile robots with improved genetic algorithm. Comput. Electr. Eng. 38 (6), 1564–1572.

Ueno, Tomohiro, Nagaya, Shigeki, Shimizu, Masayuki, Saito, Hiroyuki, Murata, Show, Handa, Norihisa, 2018. Development and demonstration test for floating type ocean current turbine system conducted in kuroshio current. In: 2018 OCEANS-MTS/IEEE Kobe Techno-Oceans. OTO, IEEE, pp. 1–6.

VanZwieten, James H., Vanrietvelde, Nicolas, Hacker, Basil L., 2012. Numerical simulation of an experimental ocean current turbine. IEEE J. Ocean. Eng. 38 (1), 131–143.

Wang, Zhanyuan, Li, Yulong, Ma, Caipeng, Yan, Xun, Jiang, Dapeng, 2023. Path-following optimal control of autonomous underwater vehicle based on deep reinforcement learning. Ocean Eng. 268, 113407.

Weng, Yang, Matsuda, Takumi, Sekimori, Yuki, Pajarinen, Joni, Peters, Jan, Maki, Toshihiro, 2022. Establishment of line-of-sight optical links between autonomous underwater vehicles: Field experiment and performance validation. Appl. Ocean Res. 129, 103385.

Wiig, Martin Syre, Pettersen, Kristin Ytterstad, Krogstad, Thomas Røbekk, 2019. Collision avoidance for underactuated marine vehicles using the constant avoidance angle algorithm. IEEE Trans. Control Syst. Technol. 28 (3), 951–966.

Wu, Wentao, Peng, Zhouhua, Wang, Dan, Liu, Lu, Han, Qing-Long, 2021. Network-based line-of-sight path tracking of underactuated unmanned surface vehicles with experiment results. IEEE Trans. Cybern.

Xi, Meng, Yang, Jiachen, Wen, Jiabao, Liu, Hankai, Li, Yang, Song, Houbing Herbert, 2022. Comprehensive ocean information enabled AUV path planning via reinforcement learning. IEEE Internet Things J.

Xu, Yiming, Mohseni, Kamran, 2013. Bioinspired hydrodynamic force feedforward for autonomous underwater vehicle control. IEEE/ASME Trans. Mechatronics 19 (4), 1127–1137.

Yan, Zheping, Gong, Peng, Zhang, Wei, Wu, Wenhua, 2020. Model predictive control of autonomous underwater vehicles for trajectory tracking with external disturbances. Ocean Eng. 217, 107884.

Yao, Peng, Sui, Xinyi, Liu, Yuhui, Zhao, Zhiyao, 2023. Vision-based environment perception and autonomous obstacle avoidance for unmanned underwater vehicle. Appl. Ocean Res. 134, 103510.

Yu, Caoyang, Zhong, Yiming, Lian, Lian, Xiang, Xianbo, 2021. An experimental study of adaptive bounded depth control for underwater vehicles subject to thruster's dead-zone and saturation. Appl. Ocean Res. 117, 102947.

Zeng, Zheng, Lammas, Andrew, Sammut, Karl, He, Fangpo, Tang, Youhong, 2014. Shell space decomposition based path planning for AUVs operating in a variable environment. Ocean Eng. 91, 181–195.

Zhang, Jialei, Xiang, Xianbo, Lapierre, Lionel, Zhang, Qin, Li, Weijia, 2021. Approach-angle-based three-dimensional indirect adaptive fuzzy path following of under-actuated AUV with input saturation. Appl. Ocean Res. 107, 102486.

Zhang, Hanwen, Zeng, Zheng, Yu, Caoyang, Jiang, Zhining, Han, Bo, Lian, Lian, 2020. Predictive and sliding mode cascade control for cross-domain locomotion of a coaxial aerial underwater vehicle with disturbances. Appl. Ocean Res. 100, 102183.