2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Real-Time Control Method for Resource-Limited HAUV Based on Dual-Modal Dynamic Triggering

Gehan Zhu[2],Qi Guo[2], Zhaoyang Wang [1],Bo Xu[1(✉)]

[1] College of Intelligent Systems Science and Engineering, Harbin Engineering University,
Harbin 150006, China
xubocarter@sina.com

[2] College of Future Technology, Harbin Engineering University, Harbin 150006, China

Abstract. Hybrid Aquatic-Aerial Underwater Vehicles (HAUVs) face significant control challenges in scenarios involving abrupt medium transitions, actuator constraints, and limited computational resources, including sudden dynamic changes, multimodal coupling, and insufficient real-time responsiveness. To address the actuator saturation issue inherent in traditional Proportional-Integral-Derivative control (PID) and the high computational load of model predictive control (MPC), this paper proposes an event-triggered disturbance observer-based tiny model predictive control (ET-TMPC) method. First, a HAUV rigid-body dynamic model is established, where lumped disturbances during cross-medium transitions are estimated using a nonlinear disturbance observer. Second, the MPC optimization process is restructured by integrating the alternating direction method of multipliers (ADMM) with precomputation techniques, significantly reducing online computational complexity. Furthermore, a dual-modal FAL dynamic triggering strategy is introduced, which dynamically adjusts triggering thresholds for disturbance errors and state errors through FAL, thereby achieving co-optimization of control performance and resource efficiency in cross-domain trajectory tracking. Simulation results demonstrate that, compared to conventional PID and standard MPC, ET-TMPC substantially enhances trajectory tracking stability and anti-disturbance capability during water-to-air transition phases while effectively suppressing attitude fluctuations and reducing computational load.

Keywords: Event-Triggered, Model Predictive Control, Disturbance Observer, Alternating Direction Method of Multipliers

1 Introduction

Cross-domain air-water navigation has become a research hotspot in recent years. However, achieving unified control and maintaining stability for hybrid air-underwater vehicles (HAUVs) in both water and air remains a significant challenge. Unified control refers to ensuring consistent performance in attitude stability, trajectory tracking, and propulsion transitions under a single control framework across different media. The core difficulty arises from dynamic discontinuities during medium transitions, multimodal coupling, and the nonlinear control complexities involved in switching between air and water environments. Sliding mode control (SMC), as a robust method based on switching strategies, offers high-gain responses based on the current system state. Nonetheless, this often results in control signals that exceed actuator limits and lacks the capability to incorporate system constraints such as thrust saturation or small-angle restrictions, especially during cross-medium operations.

Model predictive control (MPC) offers a more comprehensive framework by naturally incorporating system and path constraints through its receding horizon optimization mechanism. For instance, during transitions through the air-water interface, sudden variations in hydrodynamic forces can cause required thrust to surpass physical limits. MPC directly enforces constraints on control inputs to prevent such violations. MPC methods based on the alternating direction method of multipliers (ADMM) have shown strong capabilities in managing highly dynamic systems with complex constraints while reducing computational load, making them suitable for small robotic platforms. However, the robustness of MPC in dynamic environments still presents challenges. To enhance real-time performance, event-triggered mechanisms based on fractional-order adaptive laws (FAL) have been proposed, reducing computational demands and memory usage, and making real-world applications feasible. Yet, implementing event-triggered MPC in HAUVs is complicated by disturbances and state estimation errors during cross-domain navigation.

This paper proposes an event-triggered disturbance observer-based tiny model predictive control (ET-TMPC) framework to address the above challenges. The main contributions are as follows: 1) A nonlinear event-triggering mechanism based on FAL is introduced, which dynamically adjusts the control update frequency based on a joint decision rule involving disturbances and state errors. This balances real-time performance with robustness and ensures trajectory tracking accuracy with minimal computational effort. 2) A rigid-body dynamics model combined with a lumped disturbance observer is established to replace complex hydrodynamic modeling, enhancing modeling efficiency and control reliability during air-water transitions. 3) The proposed method overcomes the actuator saturation and constraint enforcement limitations of traditional SMC while achieving coordinated control of dynamic discontinuities and multimodal coupling. Simulation experiments validate the proposed approach in scenarios with strong disturbances and complex medium transitions, confirming its effectiveness and generalizability.

2 HAUV Dynamic Model

To effectively capture the motion characteristics of a hybrid air-underwater vehicle (HAUV) operating across multiple domains, this section presents a simplified dynamic model grounded in rigid-body theory. The model is developed to retain the essential physical attributes of the vehicle, such as mass distribution, symmetry, and dominant motion dynamics, while omitting higher-order or coupled hydrodynamic effects that contribute minimally to control-relevant behavior. This approach strikes a balance between model fidelity and computational efficiency, making it well-suited for subsequent control strategy design and real-time application.

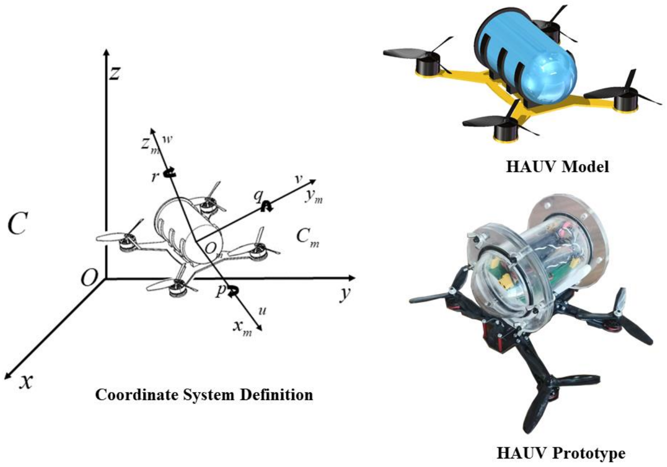


Fig. 1. Conceptual Diagram of HAUV

Main Assumptions as below:

Assumption 1: The primary mass and volume of the HAUV are uniformly distributed along a cylindrical body, with the mass and volume of arms and propellers neglected.

Assumption 2: The HAUV is considered a uniformly symmetric rigid body.

Assumption 3: Coupled hydrodynamic coefficients and higher-order hydrodynamic coefficients are neglected.

Assumption 4: The rotation and revolution of Earth are neglected to ensure that the reference sea surface of HAUV is considered an inertial frame.

Define two coordinate systems to describe the motion of the HAUV. One coordinate system is the navigation coordinate system $\{\boldsymbol{R_N}\}(O_e, x_e, y_e, z_e)$,the other is the body-fixed coordinate system $\{\boldsymbol{R_B}\}(O_b, x_b, y_b, z_b)$ .Position $\boldsymbol{\eta_1} = [x, y, z]^T$, velocity $\boldsymbol{v_1} = [u, v, w]^T$, and attitude $\boldsymbol{\eta_1} = [\varphi, \theta, \psi]^T$ are physical quantities described in the navigation coordinate system. Angular velocity $\boldsymbol{v_2} = [p, q, r]^T$ is a physical quantity described in the body-fixed coordinate system. The rotation matrix of the body-fixed coordinate system $\boldsymbol{R_B}$ relative to the navigation coordinate system $\boldsymbol{R_N}$ and the attitude transformation matrix that relates angular velocity to the rate of change of attitude angles is denoted as $\boldsymbol{C_b^n}$ and $\boldsymbol{W}$ respectively.

$$\boldsymbol{C}_b^n = \begin{bmatrix} c\theta c\psi & -c\varphi s\psi + s\varphi c\psi s\theta & s\varphi s\psi + c\varphi s\psi s\theta \\ c\theta s\psi & c\varphi s\psi + s\varphi \mathrm{s}\phi s\theta & s\varphi c\psi + c\varphi s\psi s\theta \\ -s\theta & s\varphi c\theta & c\theta c\varphi \end{bmatrix}$$

$$\boldsymbol{W} = \begin{bmatrix} 1 & s\varphi t\theta & c\varphi t\theta \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi / c\theta & c\varphi / c\theta \end{bmatrix}$$

where s(·), c(·), and t(·) individually represent sin(·), cos(·), and tan(·), respectively. According to the Newton-Euler equations, the HAUV is mathematically described as:

$$\begin{cases} \dot{\boldsymbol{\eta}}_{\boldsymbol{1}} = \boldsymbol{v}_{\boldsymbol{1}} \\ \boldsymbol{f} = m\dot{\boldsymbol{v}}_{\boldsymbol{1}} + m\boldsymbol{v}_{\boldsymbol{2}} \times \boldsymbol{v}_{\boldsymbol{1}} \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{2}} = \boldsymbol{W}\boldsymbol{v}_{\boldsymbol{2}} \\ \boldsymbol{\tau} = \boldsymbol{J}\dot{\boldsymbol{v}}_{\boldsymbol{2}} + \boldsymbol{v}_{\boldsymbol{2}} \times \boldsymbol{J}\boldsymbol{v}_{\boldsymbol{2}} \end{cases} \tag{1}$$

where $\boldsymbol{f}$ and $\boldsymbol{\tau}$ represent the total force and total moment acting on the body, respectively, and m and $\boldsymbol{J}$ denote the mass and inertia matrix of the vehicle, respectively.

$$\boldsymbol{J} = \begin{bmatrix} J_{xx} & -J_{xy} & -J_{xz} \\ -J_{xy} & J_{yy} & -J_{yz} \\ -J_{xz} & -J_{yz} & J_{zz} \end{bmatrix} \tag{2}$$

$$\begin{cases} \boldsymbol{f} = \boldsymbol{f}_{\boldsymbol{C}} + \boldsymbol{f}_{\boldsymbol{G}} + \boldsymbol{f}_{\boldsymbol{B}} + \boldsymbol{f}_{\boldsymbol{D}} + \boldsymbol{f}_{\boldsymbol{W}} \\ \boldsymbol{\tau} = \boldsymbol{\tau}_{\boldsymbol{C}} + \boldsymbol{\tau}_{\boldsymbol{G}} + \boldsymbol{\tau}_{\boldsymbol{B}} + \boldsymbol{\tau}_{\boldsymbol{D}} + \boldsymbol{\tau}_{\boldsymbol{W}} + \boldsymbol{\tau}_{\boldsymbol{gyro}} \end{cases} \tag{3}$$

where $\boldsymbol{f}_{\boldsymbol{C}}$ and $\boldsymbol{\tau}_{\boldsymbol{C}}$ represent the control input forces and moments, $\boldsymbol{f}_{\boldsymbol{G}}$ and $\boldsymbol{\tau}_{\boldsymbol{G}}$ denote the gravitational forces and their corresponding moments, and $\boldsymbol{f}_{\boldsymbol{B}}$ and $\boldsymbol{\tau}_{\boldsymbol{B}}$ represent the buoyant forces and the moments they generate， and $\boldsymbol{f}_{\boldsymbol{D}}$ and $\boldsymbol{\tau}_{\boldsymbol{D}}$ represent the hydrodynamic damping forces and moments, $\boldsymbol{f}_{\boldsymbol{W}}$ and $\boldsymbol{\tau}_{\boldsymbol{W}}$ denote the forces and moments exerted on the body by wind and waves, and $\boldsymbol{\tau}_{\boldsymbol{gyro}}$ represents the gyroscopic moment.

Since the calculation of wave forces, hydrodynamic damping forces, buoyancy, and gyroscopic moments requires establishment of complex hydrodynamic models, which is not the focus of this study, a disturbance observer is designed in this paper to estimate the aggregated disturbance force $\hat{\boldsymbol{d}}_f$ and the aggregated disturbance moment $\hat{\boldsymbol{d}}_\tau$.

$$\hat{\boldsymbol{d}}_f = \boldsymbol{f}_{\boldsymbol{B}} + \boldsymbol{f}_{\boldsymbol{D}} + \boldsymbol{f}_{\boldsymbol{W}} \tag{4}$$

$$\hat{\boldsymbol{d}}_\tau = \boldsymbol{\tau}_{\boldsymbol{B}} + \boldsymbol{\tau}_{\boldsymbol{D}} + \boldsymbol{\tau}_{\boldsymbol{W}} + \boldsymbol{\tau}_{\boldsymbol{gyro}} \tag{5}$$

The system dynamic model can now be transformed into:

$$\begin{cases} \dot{\boldsymbol{\eta}}_{\boldsymbol{1}} = \boldsymbol{v}_{\boldsymbol{1}} \\ \boldsymbol{f}_{\boldsymbol{C}} + \boldsymbol{f}_{\boldsymbol{G}} + \hat{\boldsymbol{d}}_f = m\dot{\boldsymbol{v}}_{\boldsymbol{1}} + m\boldsymbol{v}_{\boldsymbol{2}} \times \boldsymbol{v}_{\boldsymbol{1}} \\ \dot{\boldsymbol{\eta}}_{\boldsymbol{2}} = \boldsymbol{W}\boldsymbol{v}_{\boldsymbol{2}} \\ \boldsymbol{\tau}_{\boldsymbol{G}} + \hat{\boldsymbol{d}}_\tau = \boldsymbol{J}\dot{\boldsymbol{v}}_{\boldsymbol{2}} + \boldsymbol{v}_{\boldsymbol{2}} \times \boldsymbol{J}\boldsymbol{v}_{\boldsymbol{2}} \end{cases} \tag{6}$$

3 Event-Triggered Disturbance Observer-based Tiny Model Predictive Control

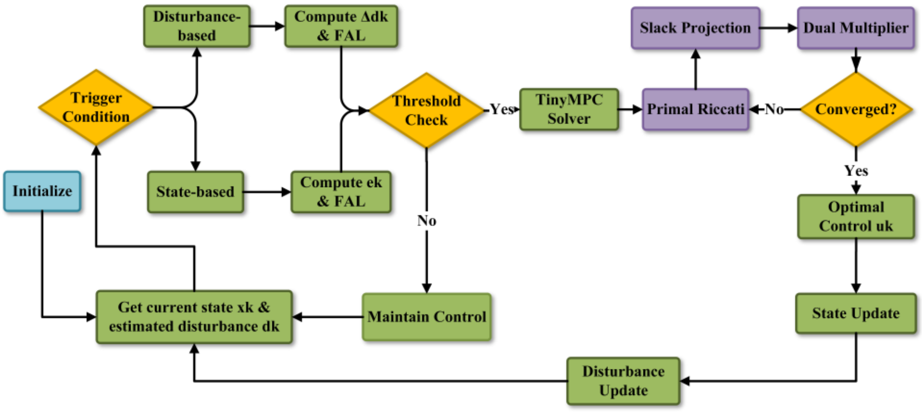


Fig. 2. Event-triggered disturbance observer-based tiny model predictive control

3.1 Introduction of Relaxation Variables and Alternating Direction Method of Multipliers

The optimization is performed by minimizing a cost function subject to state and control input constraints.

$$\min J = \sum_{k=1}^{N-1} \frac{1}{2} \boldsymbol{x}_k^T \boldsymbol{Q} \boldsymbol{x}_k + \boldsymbol{q}_k^T \boldsymbol{x}_k + \frac{1}{2} \boldsymbol{u}_k^T \boldsymbol{R} \boldsymbol{u}_k + \boldsymbol{r}_k^T \boldsymbol{u}_k \tag{7}$$

where, $\boldsymbol{x}_k \in \boldsymbol{R}^n$ is the 12-dimensional state vector of the HUAV at a given time, $\boldsymbol{u}_k \in \boldsymbol{R}^m$ is the 4-dimensional control input vector for the HUAV motors, $\boldsymbol{Q},\boldsymbol{R}$ is a positive definite weight matrix, and $\boldsymbol{q}_k, \boldsymbol{r}_k$ is the linear cost vector.

TinyMPC employs the Alternating Direction Method of Multipliers (ADMM) to handle the aforementioned optimization problem. By introducing relaxation variables, the original problem is transformed into an equivalent form, enabling stepwise solution using ADMM.

The relaxation variables $\boldsymbol{z}_k$ and $\boldsymbol{w}_k$ (representing the relaxation variables for the state and control input, respectively) are introduced, and the original problem is transformed into the following form:

$$\begin{aligned}\min J = &\sum_{k=1}^{N-1} \left(\frac{1}{2} \boldsymbol{x}_k^T \boldsymbol{Q} \boldsymbol{x}_k + \boldsymbol{q}_k^T \boldsymbol{x}_k + \frac{1}{2} \boldsymbol{u}_k^T \boldsymbol{R} \boldsymbol{u}_k + \boldsymbol{r}_k^T \boldsymbol{u}_k \right) + \boldsymbol{\mu}_k^T (\boldsymbol{u}_k - \boldsymbol{w}_k) + \\ &\rho \sum_{k=1}^{N-1} (\|\boldsymbol{x}_k - \boldsymbol{z}_k\|^2 + \|\boldsymbol{u}_k - \boldsymbol{w}_k\|^2) + \sum_{k=1}^{N-1} \boldsymbol{\lambda}_k^T (\boldsymbol{x}_k - \boldsymbol{z}_k)\end{aligned} \tag{8}$$

where λ_k and $\boldsymbol{\mu}_k$ are the Lagrange multipliers, and ρ is the penalty coefficient.

The core of Alternating Direction Method of Multipliers involves alternating between three steps: Primal Update, Slack Update, and Dual Update.

The objective of the primal problem is to minimize the cost associated with the state and control inputs. In each ADMM iteration, we update the state and control inputs. Specifically, the state update formula is:

$$\boldsymbol{x}_k^+ = \arg\min_x \left(\frac{1}{2} \boldsymbol{x}_k^T \boldsymbol{Q} \boldsymbol{x}_k + \boldsymbol{q}_k^T \boldsymbol{x}_k \right) \tag{1}$$

Similarly, the update formula for the control input is:

$$\boldsymbol{u}_k^+ = \arg\min_u \left(\frac{1}{2} \boldsymbol{u}_k^T \boldsymbol{R} \boldsymbol{u}_k + \boldsymbol{r}_k^T \boldsymbol{u}_k \right) \tag{2}$$

In the Slack Update step, the relaxation variables undergo a linear projection to ensure compliance with the constraint conditions. The update formulas for the relaxation variables are as follows:

$$\begin{cases} \boldsymbol{z}_k^+ = \mathrm{proj}_{\boldsymbol{X}} (\boldsymbol{x}_k^+ + \dfrac{\boldsymbol{\lambda}_k}{\rho}) \\ \boldsymbol{w}_k^+ = \mathrm{proj}_{\boldsymbol{U}} (\boldsymbol{u}_k^+ + \dfrac{\boldsymbol{\mu}_k}{\rho}) \end{cases} \tag{3}$$

where $proj_X$ and $proj_U$ represent the projection operations on the state and control input, respectively, ensuring they remain within the constraint sets X and $\boldsymbol{U}$.

The dual variables are updated using gradient ascent to reinforce constraint enforcement:

$$\begin{cases} \boldsymbol{\lambda}_k^+ = \boldsymbol{\lambda}_k + \rho(\boldsymbol{x}_k^+ - \boldsymbol{z}_k^+) \\ \boldsymbol{\mu}_k^+ = \boldsymbol{\mu}_k + \rho(\boldsymbol{u}_k^+ - \boldsymbol{w}_k^+) \end{cases} \tag{4}$$

These dual updates ensure that the constraints are effectively enforced in each iteration.

In the implementation of TinyMPC , the most computationally intensive step is solving linear equations, particularly the Riccati equation. To accelerate the computation, TinyMPC employs precomputation techniques. By precomputing the Riccati gain matrix $\boldsymbol{K}_\infty$ and the Hessian matrix $\boldsymbol{P}_\infty$, the need for expensive matrix decomposition in each iteration is eliminated. The precomputed formulas are as follows:

$$\begin{cases} \boldsymbol{C}_1 = (\boldsymbol{R} + \boldsymbol{B}^T \boldsymbol{P}_\infty \boldsymbol{B})^{-1} \\ \boldsymbol{C}_2 = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}_\infty)^T \end{cases} \tag{5}$$

Through these precomputations, TinyMPC simplifies the computation to matrix-vector multiplications, significantly improving computational efficiency and reducing memory usage.

3.2 Nonlinear Disturbance Observer.

$\boldsymbol{d}_f, \boldsymbol{d}_\tau$ need to be estimated using a disturbance observer, and a nonlinear disturbance observer is chosen. The observation equation for $\hat{\boldsymbol{d}}_f$ is given by

$$\begin{cases} \dot{\boldsymbol{z}}_{\boldsymbol{f}} = -\dfrac{1}{m}\boldsymbol{l}_{\boldsymbol{f}}(m(\boldsymbol{v}_{\boldsymbol{1}} \times \boldsymbol{v}_{\boldsymbol{2}}) + \boldsymbol{f}_{\boldsymbol{G}} + \boldsymbol{f}_{\boldsymbol{C}} + \hat{\boldsymbol{d}}_{\boldsymbol{f}}) \\ \hat{\boldsymbol{d}}_{\boldsymbol{f}} = \boldsymbol{z}_{\boldsymbol{f}} + \boldsymbol{p}_{\boldsymbol{f}} \end{cases} \tag{6}$$

where the auxiliary parameters $\boldsymbol{p}_{\boldsymbol{f}} = m[\zeta_1\boldsymbol{u}, \zeta_2\boldsymbol{v}, \zeta_3\boldsymbol{w}]^T$, $\boldsymbol{l}_{\boldsymbol{f}} = \partial\boldsymbol{p}_{\boldsymbol{f}} / \partial\boldsymbol{v}_{\boldsymbol{1}} = m \cdot diag\{\zeta_1, \zeta_2, \zeta_3\}$ and ζ_i in the observer are dimensionless scaling factors used for directional weighting of the state variables. The observation of $\hat{\boldsymbol{d}}_{\boldsymbol{\tau}}$ is given by

$$\begin{cases} \dot{\boldsymbol{z}}_{\boldsymbol{\tau}} = -l_\tau I_\tau^{-1}((I_0\boldsymbol{v}_{\boldsymbol{2}}) \times \boldsymbol{v}_{\boldsymbol{2}} + \boldsymbol{\tau}_{\boldsymbol{C}} + \hat{\boldsymbol{d}}_{\boldsymbol{\tau}}) \\ \hat{\boldsymbol{d}}_{\boldsymbol{\tau}} = \boldsymbol{z}_{\boldsymbol{\tau}} + \boldsymbol{p}_{\boldsymbol{\tau}} \end{cases} \tag{7}$$

where the auxiliary parameters $\boldsymbol{p}_{\boldsymbol{\tau}} = diag[\mu_1, \mu_2, \mu_3] \cdot I_0 \cdot \boldsymbol{v}_{\boldsymbol{2}}$, $\boldsymbol{l}_{\boldsymbol{\tau}} = m \cdot diag\{\mu_1, \mu_2, \mu_3\} \cdot I_0$ and μ_i in the observer serve as gain adjustment coefficients for rotational directions, enhancing or suppressing the observer's sensitivity in specific directions.

3.3 Bimodal FAL-Based Dynamic Triggering Strategy.

To refine the sensitivity of the triggering mechanism, a nonlinear function known as the Fractional Power Function (FAL) is introduced to modulate the threshold dynamically. This function is specifically designed to provide nonlinear feedback based on the magnitude of the tracking error, enabling the system to respond more gently to large deviations while remaining responsive to smaller discrepancies. The mathematical expression of the FAL function is given as follows:

$$\mathrm{FAL}(e, \alpha, \delta) \square \begin{cases} |e|^\alpha \ \mathrm{sign}(e), & |e| > \delta \\ \dfrac{e}{\delta^{1-\alpha}}, & |e| \le \delta \end{cases} \tag{8}$$

where $0 < \alpha < 1$ is used to control the degree of nonlinearity. and $\delta > 0$ defines a linear interval for small errors, preventing excessive triggering sensitivity when the error is minimal， $|e| > \delta$ ensures that when a certain condition is met, the trigger value grows according to a fractional power law, accelerating the triggering process, and $|e| \le \delta$ ensures that when another condition is met, the trigger value is adjusted linearly to avoid excessive sensitivity.

By designing a FAL-based triggering threshold, the system imposes stricter triggering conditions under small disturbances, while allowing more frequent control updates when disturbances become significant.

The disturbance triggering condition is given by:

$$\left\| \hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1} \right\| > \varepsilon_d \tag{9}$$

The triggering threshold ϵ_d is adaptively adjusted using the FAL function:

$$\grave{o}_d = \alpha_d \cdot \mathrm{FAL}(\left\| \hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1} \right\|, \gamma_d, \delta_d) + \beta_d \tag{10}$$

In other words:

$$FAL(\|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\|, \gamma_d, \delta_d) = \begin{cases} \|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\|^{\gamma_{\mathrm{d}}}, \|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\| > \delta_d \\ \dfrac{\|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\|}{\delta_d^{1-\gamma_d}}, \|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\| \le \delta_d \end{cases}, \quad \delta_d > 0, 0 < \gamma_{\mathrm{d}} < 1 \tag{11}$$

where: γ_d controls the degree of nonlinearity. A smaller γ_d makes the triggering curve approach linear, while a larger γ_d enhances nonlinearity. δ_d is the threshold switching point. When the disturbance variation is smaller than δ_d, the triggering condition follows a linear growth; otherwise, it follows a nonlinear growth.

α_d is a scaling factor that adjusts the overall magnitude of the triggering threshold. β is an offset to ensure the threshold does not become too low, thereby reducing unnecessary computational overhead.

Similarly, we employ the Fast Adaptive Law function for the state error-based triggering condition:

$$\|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\| > \varepsilon_x \tag{12}$$

$$\grave{o}_x = \alpha_x \cdot \mathrm{FAL}(\|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\|, \gamma_x, \delta_x) + \beta_x \tag{13}$$

In other words:

$$FAL(\|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\|, \gamma_x, \delta_x) = \begin{cases} \|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\|^{\gamma_x}, \|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\| > \delta_x \\ \dfrac{\|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\|}{\delta_x^{1-\gamma_x}}, \|\boldsymbol{x}_k - \boldsymbol{x}_k^{ref}\| \le \delta_x \end{cases}, \quad \delta_x > 0, 0 < \gamma_x < 1 \tag{9}$$

Thus, when the state error is small (i.e., the system is close to stability), the triggering threshold increases, reducing the frequency of triggering. Conversely, when the state error becomes larger (i.e., the system deviates from the desired trajectory), the triggering threshold decreases, allowing the control mechanism to activate earlier, thereby improving the system's response speed.

Finally, the ET-TMPC bimodal dynamic triggering condition can be formulated as:

$$\begin{cases} \|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\| > \alpha_d \cdot \mathrm{FAL}(\|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\|, \gamma_d, \delta_d) + \beta_d \\ \|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\| > \alpha_x \cdot \mathrm{FAL}(\|\hat{\boldsymbol{d}}_k - \hat{\boldsymbol{d}}_{k-1}\|, \gamma_x, \delta_x) + \beta_x \end{cases} \tag{14}$$

This condition integrates both linear and nonlinear triggering mechanisms, ensuring adaptive control updates based on disturbance variations.

The selection of parameters γ_d, δ_i and α_d directly influences the trade-off between control precision and computational efficiency. For instance, In practical applications, γ_d is typically chosen within the range $[0.3, 0.7]$ to balance nonlinearity and stability. A value too close to 1 may lead to oversensitivity, while a value near 0 renders the system quasi-linear. And the parameter δ_i should be dynamically adjusted based on the system's operating conditions. For high-frequency disturbance environments, a smaller δ_i ensures rapid response, whereas a larger δ_i suits steady-state operations.

Table 1. ET-TMPC with State Observer for HAUV

Algorithm ET-TMPC with State Observer for HAUV

Input: Initial state estimate $\hat{x}_0$, disturbance estimate $\hat{d}_0$, Previous control input u_{-1}, Event thresholds ε_e, ε_d, Nonlinear evaluation function FAL$(\cdot)$, Control horizon T, max iterations $N_{\max}$

Output: Control sequence $\{u_k\}_{k=0}^{K}$, State estimates $\{\hat{x}_k\}_{k=1}^{K}$, Trigger indicators $\{\text{flag}_k\}_{k=0}^{K}$

```
$k \leftarrow 0$, Trigger $\leftarrow$ True
Obtain $\hat{x}_0$, $\hat{d}_0$ from observer

for $k = 0,1,2,\ldots$ do
    $\hat{x}_k, \hat{d}_k \leftarrow$ Observer_Update()
    $F_e \leftarrow \mathrm{FAL}(\|\hat{x}_k - \hat{x}_{k-1}\|)$
    $F_d \leftarrow \mathrm{FAL}(\|\hat{d}_k - \hat{d}_{k-1}\|)$
    Trigger $\leftarrow (F_e > \varepsilon_e) \vee (F_d > \varepsilon_d)$

    if Trigger then
        $\{u_k\} \leftarrow$ TMPC_Solver($\hat{x}_k, \hat{d}_k, T, N_{\max}$)
            Solve via:
              1. Primal Riccati recursion
              2. Slack variable projection
              3. Dual multiplier update
    else
        $u_k \leftarrow u_{k-1}$
    end if

    $\hat{x}_{k+1} \leftarrow$ State_Observer($\hat{x}_k, u_k$)
    $\hat{d}_{k+1} \leftarrow$ Disturbance_Observer($\hat{d}_k, \hat{x}_k$)
    $k \leftarrow k + 1$
end for
```

4 Simulation Experiments

Based on the previous discussion and analysis, the theoretically developed method is expected to be effective. In this section, numerical simulations are conducted to evaluate the position and attitude errors in trajectory tracking. The simulations cover both aerial and underwater scenarios, with particular focus on the transition phase between mediums, where sudden dynamic changes may occur. Additionally, computational resource consumption is assessed by recording algorithm iteration counts and memory usage under different trajectory complexities, offering insights into the method's practical applicability. To further examine the robustness of the proposed approach, various external disturbances with different amplitudes and durations are introduced to simulate real-world uncertainty. The system's response under such disturbances is analyzed to validate the control algorithm’s stability and effectiveness across diverse and challenging operating conditions.

4.1 Simulation Conditions

Table 2. Simulation Conditions for Convergence Analysis Parameters

ETDOTMPC	PID
$Q = diag(100,100,100,... 4,4,400,4,4,4,2,2,4)$	$PIDController\ pid\_x(8,\ 0.0,\ 1)$
	$PIDController\ pid\_y(8,\ 0.0,\ 1)$
	$PIDController\ pid\_z(8,\ 0.0,\ 1)$
$R = diag(4,4,4,4)$	$PIDController\ pid\_roll(4,\ 0.0,\ 0.5)$
	$PIDController\ pid\_pitch(4,\ 0.0,\ 0.5)$
	$PIDController\ pid\_yaw(4,\ 0.0,\ 0.5);$

Table 2 outlines the controller-specific tuning parameters used in the convergence simulations. The weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are defined to penalize deviations in state and control inputs, respectively.

Table 3. Simulation parameters that measure computing resource usage

Classes	TRAJECTORY [i][0]	TRAJECTORY [i][1]	TRAJECTORY [i][2]
(a)	$0.5*\sin(0.5*t)$	$0.4*\sin(0.6*t)$	$0.6*\sin(0.4*t)$
(b)	$0.8*\cos(0.3*t)$	$0.6*\cos(0.2*t)$	$0.4*\cos(0.5*t)$
(c)	$\sin(0.4*t)$	$\sin(0.5*t)$	$\sin(0.3*t)$

Table 3 denotes different trajectory classes used to evaluate the computational demands and tracking robustness of the controllers. Classes (a), (b), and (c) represent various levels of oscillatory motion in 3D space, simulating realistic and dynamically rich reference paths for the hybrid vehicle.

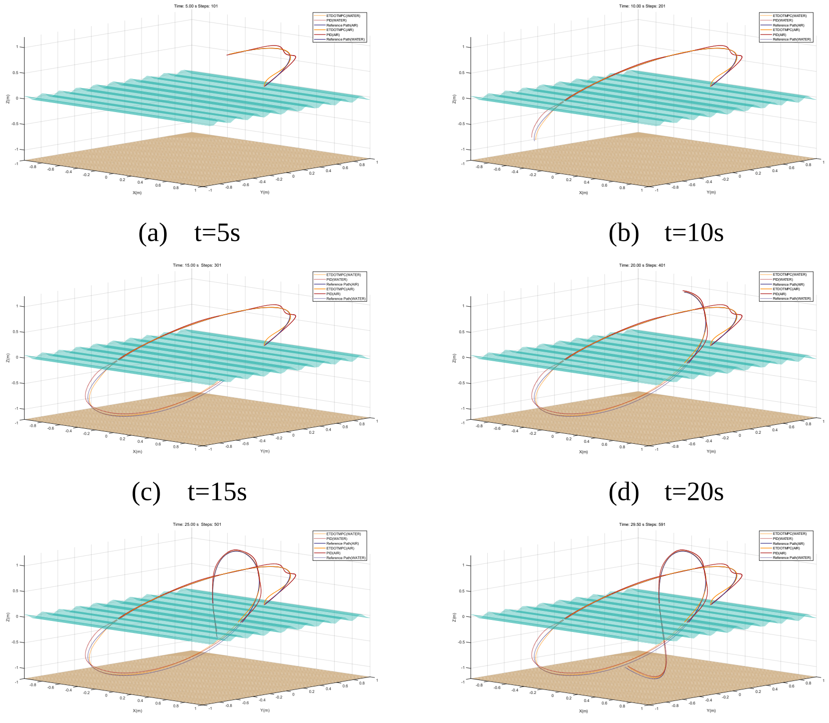
Table 4. Parameters of Applied Strong Disturbances.

Time	START_STEP	DURATION	FORCE_X	FORCE_Y	FORCE_Z
t=5s	100	0.5s	2.50N	0.00	0.00
t=10s	200	0.5s	0.00	2.50N	0.00
t=20s	400	0.5s	0.00	0.50	2.50N

Table 4 details the external disturbance forces applied at specific time intervals to assess the disturbance rejection capability of each control algorithm.

4.2 Convergence Analysis

In this section, we compare the trajectory tracking performance of the ETDOTMPC and PID algorithms in the cross-domain operation of a hybrid aerial-underwater vehicle (HAUV) to demonstrate the stability and broad applicability of the ETDOTMPC algorithm. This comparison not only highlights the advantages and limitations of each algorithm but also illustrates the feasibility of their application in different medium environments. The simulation results present the trajectory tracking performance of both algorithms at six different time instants and analyze the convergence of their position and attitude errors.



(a) t=5s

(b) t=10s

(c) t=15s

(d) t=20s

(e) t=25s

(f) t=29.5s

Fig. 3. Different Control Algorithms in Cross-Domain Trajectory Tracking

Fig 3 compares the trajectory tracking performance of the ETDOTMPC and PID control algorithms at six different time instants. In the air, the PID algorithm exhibits significantly larger errors compared to ETDOTMPC. Underwater, both algorithms achieve similar tracking accuracy, though ETDOTMPC demonstrates overall superior performance. In cross-domain transitions between air and water, the PID algorithm shows a tendency to diverge, indicating lower stability com-pared to ETDOTMPC. Specifically, at t = 5 s, the HAUV performs a sharp turn, during which the PID controller exhibits significant oscillations. Additionally, during water entry (t = 10 s), the deviation under PID control is approximately twice that of ETDOTMPC.

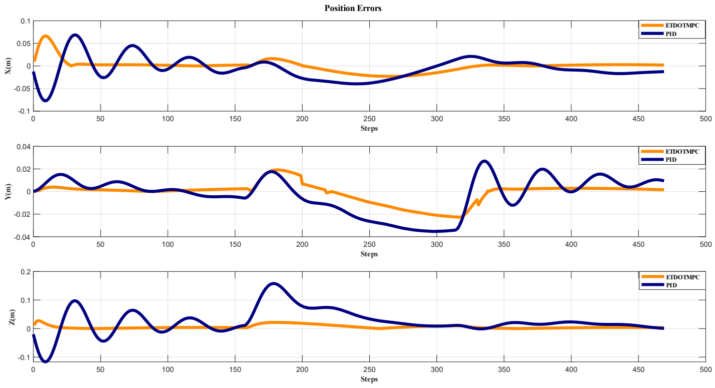


Fig. 4. Comparison of HAUV Position Errors Under Different Control Algorithms

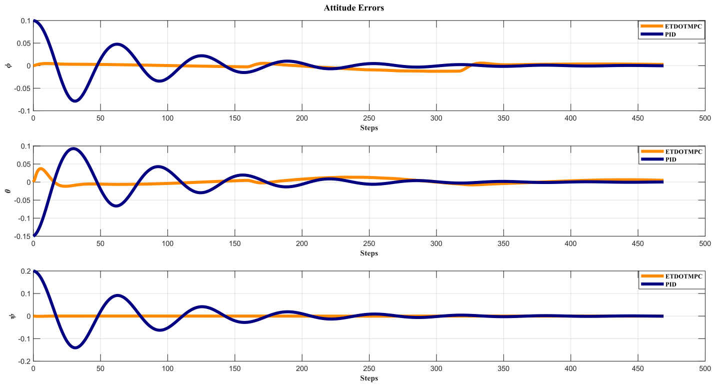


Fig. 5. Comparison of HAUV Attitude Errors Under Different Control Algorithms

Fig 4 and Fig 5 illustrate the convergence of position and attitude errors for the two control algorithms. Fig 4 shows that the ET-TMPC algorithm achieves faster and more stable convergence of position errors compared to PID. However, both algorithms exhibit significant disturbances in the Y-direction. This is because the applied wave disturbance during the vehicle's water-exit phase has a considerable component in the Y-direction, leading to larger perturbations. Fig 5 clearly demonstrates that ET-TMPC achieves superior convergence in attitude errors.

4.3 Analysis of Computational Resource Consumption.

To evaluate the computational efficiency of the proposed control method, we conducted a comparative analysis of the number of execution steps and memory usage under different trajectory constraints. This assessment helps quantify suitability of the control algorithms for real-time applications on resource-constrained platforms.

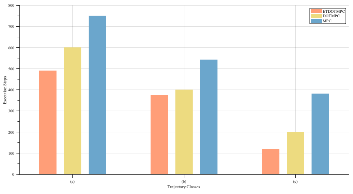


Fig. 6. Comparing Execution Steps Under Different Trajectory Constraints

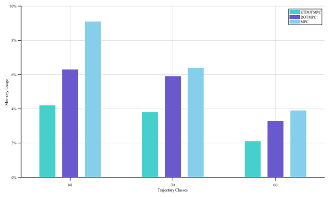


Fig. 7. Comparison of Memory Usage Under Different Trajectory Constraints

We measured the number of execution steps and memory usage of the two control algorithms under trajectories with varying complexities to reflect computational resource consumption. From Fig 6 and Fig 7, it is evident that ET-TMPC consistently requires the least computational resources across all three trajectories, while MPC consumes the most. Additionally, in low-complexity trajectories, ET-TMPC and DOMPC perform similarly, whereas in high-complexity trajectories, ET-TMPC's performance is significantly superior. This indicates that ET-TMPC is particularly suitable for high-dynamic environments.

4.4 Disturbance Rejection Analysis

In this section, strong disturbances are applied to the system at 5s, 10s, and 20s, and the disturbance rejection performance of the ETDOTMPC and PID algorithms is observed to validate the disturbance rejection capability of ETDOTMPC.

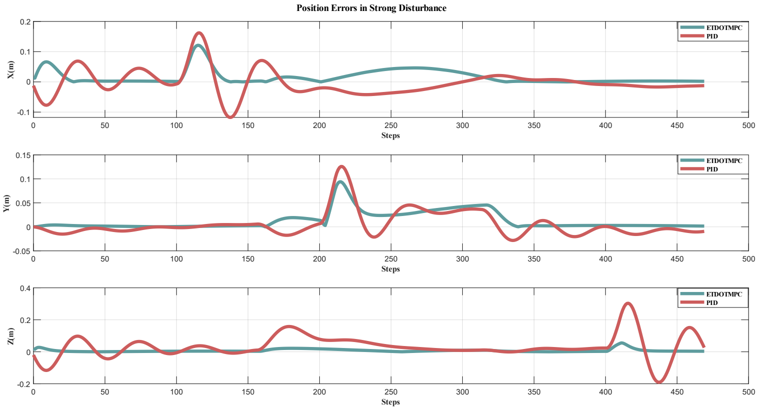


Fig. 8. Comparing Position Errors Under Strong Disturbances for Different Control

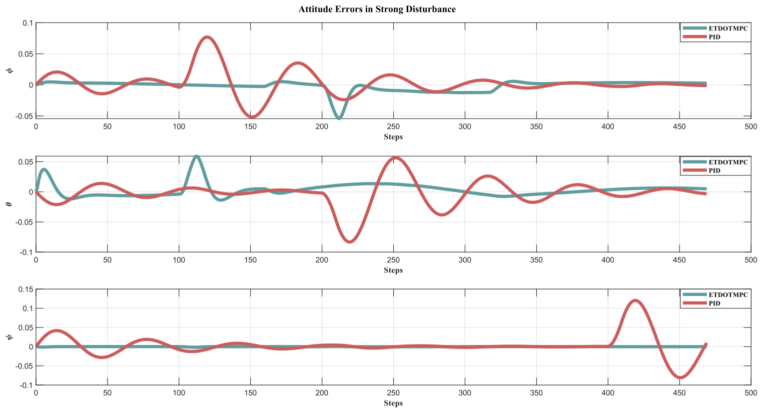


Fig. 9. Comparing Attitude Errors Under Strong Disturbances for Different Control

In Fig 8 and Fig 9, among the three strong disturbances applied, ETDOTMPC exhibits significantly lower position errors in the Z-direction and heading angle errors compared to PID, with greater stability. In other directions or attitudes, ETDOTMPC shows a noticeable surge in errors at the disturbance moments, which may be caused by the delay in the joint event-triggered mechanism. However, the subsequent convergence speed is noticeably faster than PID, and the overall error performance is superior to PID.

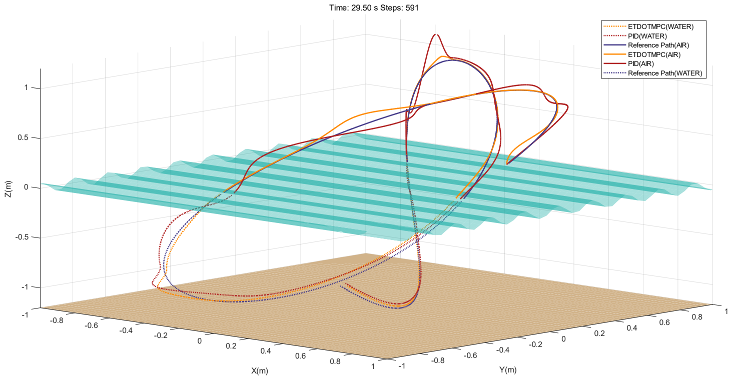


Fig. 10. Trajectory Tracking Under Strong Disturbances for Different Algorithms

From Fig 10, it can be observed that under disturbance, the fluctuations and amplitudes generated by PID are significantly larger, indicating that ETDOTMPC has stronger disturbance rejection capabilities. This also highlights the contribution of the disturbance observer in enhancing the algorithm's disturbance rejection performance.

5 Conclusion

This paper proposes an event-triggered disturbance observer-based miniature MPC for hybrid aerial-underwater vehicles (HAUVs) operating under medium transitions and computational constraints. By integrating a nonlinear disturbance observer with ADMM optimization, ETDOTMPC mitigates actuator saturation from traditional sliding mode control and reduces MPC's computational load. A FAL-based event-triggered mechanism further optimizes control updates while ensuring stable cross-domain tracking. Simulations demonstrate superior robustness and disturbance rejection during air-water transitions, balancing performance and efficiency for cross-medium autonomy.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (62473106), Hainan Province key research and development project (ZDYF2024GXJS009）.

References

1. Y. Bi, D. Lu, Z. Zeng, and L. Lian, "Dynamics and control of hybrid aerial underwater vehicle subject to disturbances," Ocean Eng., vol. 250, p. 110933, 2022.
2. Z. Zeng, C. Lyu, Y. Bi, Y. Jin, D. Lu, and L. Lian, "Review of hybrid aerial underwater vehicle: Cross-domain mobility and transitions control," *Ocean Eng.*, vol. 248, p. 110840, 2022.
3. M. Lu, F. Liao, B. Xing, Z. Fan, Y. Su, and W. Wu, "Adaptive finite-time trajectory tracking control for coaxial HAUVs facing uncertainties and unknown environmental disturbances," *Appl. Sci.*, vol. 13, no. 24, p. 8026, 2023.
4. T. Vu et al., "Station-keeping control of a hovering over-actuated autonomous underwater vehicle under ocean current effects and model uncertainties in horizontal plane," *IEEE Access*, vol. 9, pp. 6855–6867, 2021.
5. N. Hui, Y. Guo, X. Han, and B. Wu, "Robust H-infinity dual cascade MPC-based attitude control study of a quadcopter UAV," *Actuators*, vol. 13, no. 10, p. 392, 2024.
6. Z. Cheng, J. Ma, X. Zhang, and T. H. Lee, "Semi-proximal ADMM for model predictive control problem with application to a UAV system," in *Proc. 20th Int. Conf. Control, Autom. Syst. (ICCAS)*, Busan, South Korea, pp. 82–87, 2020.
7. K. Nguyen, S. Schoedel, A. Alavilli, B. Plancher, and Z. Manchester, "TinyMPC: Model-predictive control on resource-constrained microcontrollers," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Yokohama, Japan, pp. 1–7, 2024.
8. Y. Y. Nazaruddin, A. Widyotriatmo, T. A. Tamba, M. S. Arifin, and R. A. Santosa, "Communication-efficient optimal-based control of a quadrotor UAV by event-triggered mechanism," in *Proc. 5th Asian Conf. Defense Technol. (ACDT)*, Hanoi, Vietnam, pp. 96–101, 2018.
9. J. Liu, W. Gai, J. Zhang et al., "Nonlinear adaptive backstepping with ESO for the quadrotor trajectory tracking control in the multiple disturbances," *Int. J. Control Autom. Syst.*, vol. 17, no. 6, pp. 2754–2768, 2019.

10. R. Jiao, W. Chou, and Y. Rong, "Disturbance observer-based backstepping control for quadrotor UAV manipulator attitude system," in *Proc. Chinese Autom. Congr. (CAC)*, Shanghai, China, pp. 2523–2526, 2020.