drones

MDPI

Article

Efficient Uncertainty Propagation in Model-Based Reinforcement Learning Unmanned Surface Vehicle Using Unscented Kalman Filter

Jincheng Wang [1,2], Lei Xia [1,3], Lei Peng [1], Huiyun Li [1] and Yunduan Cui [1,*]

[1] Shenzhen Institute of Advanced Technology (SIAT), Chinese Academy of Sciences, Shenzhen 518055, China
[2] College of Engineering, Southern University of Science and Technology, Shenzhen 518055, China
[3] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: cuiyunduan@gmail.com

Abstract: This article tackles the computational burden of propagating uncertainties in the model predictive controller-based policy of the probabilistic model-based reinforcement learning (MBRL) system for an unmanned surface vehicles system (USV). We proposed filtered probabilistic model predictive control using the unscented Kalman filter (FPMPC-UKF) that introduces the unscented Kalman filter (UKF) for a more efficient uncertainty propagation in MBRL. A USV control system based on FPMPC-UKF is developed and evaluated by position-keeping and target-reaching tasks in a real USV data-driven simulation. The experimental results demonstrate a significant superiority of the proposed method in balancing the control performance and computational burdens under different levels of disturbances compared with the related works of USV, and therefore indicate its potential in more challenging USV scenarios with limited computational resources.

Keywords: unmanned surface vehicle; model-based reinforcement learning; Gaussian process

Citation: Wang, J.; Xia, L.; Peng, L.; Li, H.; Cui, Y. Efficient Uncertainty Propagation in Model-Based Reinforcement Learning Unmanned Surface Vehicle Using Unscented Kalman Filter. *Drones* **2023**, *7*, 228. https://doi.org/10.3390/drones7040228

Academic Editors: Bo Li and Sanjay Sharma

Received: 20 February 2023
Revised: 17 March 2023
Accepted: 23 March 2023
Published: 24 March 2023

1. Introduction

With the rapid development of machine learning technology in recent years, unmanned surface vehicles (USVs) are becoming more and more intelligent and have been widely deployed in various scenarios [1]. They not only improve the efficiency of the shipping industry but also alleviate the shortage of human resources. On the other hand, although USV has achieved impressive achievements in specific tasks, including position control [2–5], trajectory tracking [6–12], and obstacle avoidance [13,14], it is still a long-term goal in both industry and academia to develop a fully autonomous system that does not rely on human intervention and numerical models based on human prior knowledge to properly handle complex ocean disturbances [15].

As an appealing approach to learning a fully autonomous system without human prior knowledge, reinforcement learning (RL) iteratively learns optimal and sub-optimal control policies that maximize the long-term reward function through trial-and-error interactions with unknown environments [16–18]. The traditional model-free RL approaches aim to directly learn the target task in the Markov decision process (MDP) without modeling the system. These methods have been extensively studied in USV simulations [19–24]. However, their implementation to the real-world USV remains limited due to the deteriorated control capability under real ocean environments, where the observation and prediction of the frequently changing disturbances are extremely difficult and expensive.

To address this issue, model-based reinforcement learning (MBRL) attempted to introduce the probabilistic model to properly process environmental disturbances as uncertainty. One of the most popular probabilistic MBRL, probabilistic inference for learning control (PILCO) [25], employs Gaussian processes (GP) [26] to model the system uncertainties as a collection of Gaussian distributions and learns its policy with a long-term propagation

of uncertainties via analytic moment matching [27]. It successfully improved the control performance results under system uncertainties with great sample efficiency in robot control [28,29]. On the other hand, PILCO assumes that all disturbances are controllable and predictable in the full horizon of the rollout, which turns to a strenuous implementation of USV due to the heavy computational burden of the frequent decision making against the changing disturbances. To tackle the limitation of handling the real-time disturbances, GP-MPC integrates PILCO and model predictive control (MPC) in [30]. This work optimizes its policy with an expanded dimensionality for a deterministic dynamical system with Lagrange parameters and state constraints under Pontryagin's maximum principle (PMP), which resulted in heavy computational cost and therefore limited its application on simulated cart-pole and double pendulum tasks. Based on this work, probabilistic model predictive control (SPMPC) [31] is developed for better efficiency without expanded dynamics and PMP constraint. It is successfully implemented to USV with detailed evaluation in the real ocean environment. Its control capability is further improved in filtered probabilistic model predictive control (FPMPC) [32], where the state space is implicitly extended from MDP to partially observed MDP (POMDP). Although these works have demonstrated a great potential of probabilistic MBRL as a novel direction toward the fully autonomous USV, they are still limited by the bottleneck of computational complexity and are difficult to be widely applied in practice. The non-parametric characteristic of GP turned to a rapidly increasing computational burden with the expanding data size. It results in a difficult trade-off between the model expressiveness of GP and the control frequency of MPC based on long-term uncertainty propagation, especially with the limited computational resource on USV. Employing sparse GP [33] to balance the control performance and the computational complexity, the real-world USV in [31] was only controlled at 0.33 Hz, which is insufficient for more challenging scenarios.

To tackle the computational efficiency issue of the uncertainty propagation of GP, unscented Kalman filter (UKF) [34,35] estimates the target distribution by the weighted σ sampling rather than integrating the GP model over a Gaussian distribution input and therefore becomes an alternative solution besides analytic moment matching [27]. It achieved a superior computational efficiency from an engineering perspective in various applications of unmanned ground vehicles (UGVs) and unmanned aerial vehicles (UAVs) [36–39]. In the field of USV, UKF has been employed for the neural network controller [40] and model parameters estimation [41]. Despite successful applications of UKF in engineering, less work focused on using its efficient uncertainty propagation under RL's trial-and-error framework. This prevents the engineering implementation of UKF when the human's prior knowledge is unavailable.

Table 1. Relationship between the proposed approach and other related works, ◯ and × denote involve and uninvolved.

Approach	Uncertainty Propagation	MBRL	MPC	USV
PILCO [25]	analytic moment-matching	◯	×	×
GP-MPC [30]	analytic moment-matching	◯	◯	×
SPMPC [31], FPMPC [32]	analytic moment-matching	◯	◯	◯
RC-LB-NMPC [36]	UKF	×	◯	×
RBFNN-UKF [40]	UKF	×	×	×
FPMPC-UKF (proposed)	UKF	◯	◯	◯

In this work, we address the computational efficiency of uncertainty propagation in GP-based MBRL USV. Extending UKF to the MBRL system specific for USV, filtered probabilistic model predictive control with unscented Kalman filter (FPMPC-UKF) is proposed to alleviate the computational burden of GP uncertainty propagation, especially under larger sparse scales, i.e., the GP model is approximated by less sparse pseudo input.

With the evaluation in position-keeping and target-reaching tasks on the simulation based on the real boat-driven data [31] under different ocean conditions, the proposed FPMPC-UKF demonstrated a significant superiority of the proposed method in the balance between the control capability and the computational cost compared with the related MBRL USV approach [32]. With a proper setting of the GP sparse pseudo input, it improved 20% to 28% computational efficiency while achieving 15% to 29% better control performance. Based on the relationship between the proposed method and other related approaches summarized in Table 1, the contributions of this work are summarized as follows:

1. Algorithmically, the proposed FPMPC-UKF first attempts to extend the potential of UKF uncertainty propagation to the GP-based MBRL. It not only contributes to a more effective solution to trade off the control performance and computational burden compared with the existing approaches with MPC-based policy [30–32] but also demonstrates the broad prospects of traditional optimal filtering methods in enhancing related MBRL approaches based on different prediction horizons [25] in more challenging control tasks.
2. On the side of traditional optimal filtering technologies, this work can be seen as extending the current state-of-the-art optimal filtering implementations in unmanned systems [36–39] to the trial-and-error RL framework. It expanded the usage of optimal filters in unmanned systems by adaptively obtaining an efficient filter to propagate system uncertainties without human prior knowledge of the target model.
3. On the side of the application of MBRL, a USV control system based on FPMPC-UKF was developed. We investigated the effect of sparse GP scales on both control capability and model prediction error in position-keeping and target-reaching tasks under different levels of disturbances. The proposed method significantly outperformed existing MBRL systems specific to USV [31,32] with over 15% less offset while reducing more than 20% computational burden. It enabled the higher control frequency in the existing work without damaging control performance and therefore expanded the practicability of probabilistic MBRL in the USV domain.

The remainder of this paper is organized as follows. The USV control problem and MBRL settings are introduced in Section 2. Section 3 explains the proposed FPMPC-UFK and the corresponding USV control system. The experimental results are demonstrated in Section 4. Section 5 presents the conclusions.

2. Preliminaries

2.1. Markov Decision Process of USV

The target USV dynamics is described in the bottom of Figure 1. The USV observation states include its position in GPS $(X_{\text{usv}}, Y_{\text{usv}})$, orientation Ψ_{usv}, velocity V_{usv}, the engine throttle τ_{usv} and rudder angle δ_{usv}. The wind is defined as the observable disturbances in this study with direction Ψ_{wind} and velocity V_{wind}, while other disturbances, such as current, are considered unknown noises. The USV control problem is described as the Markov decision process (MDP) in the RL domain [16] with state space $\mathcal{S}$, action space $\mathcal{A}$ and reward function $\mathcal{R}$. The state vector is defined as $\boldsymbol{x} = [X_{\text{usv}}, Y_{\text{usv}}, \Psi_{\text{usv}}, V_{\text{usv}}, \tau_{\text{usv}}, \delta_{\text{usv}}, V_{\text{wind}} \cdot \cos(\Psi_{\text{wind}}), V_{\text{wind}} \cdot \sin(\Psi_{\text{wind}})]$ in $\mathcal{S}$. The control signals to engine throttle and rudder angle are defined as a vector $\boldsymbol{u} = [\tau_{\text{action}}, \delta_{\text{action}}]$ in $\mathcal{A}$. Please note that the throttle and rudder signals $\tau_{\text{usv}}, \delta_{\text{usv}}$ are the current states read from sensors, which are different from the current command sending to the throttle and rudder $[\tau_{\text{action}}, \delta_{\text{action}}]$. A reward function $\mathcal{R}(\boldsymbol{x}, \boldsymbol{u})$ is designed to evaluate the performance of USV in the target task.

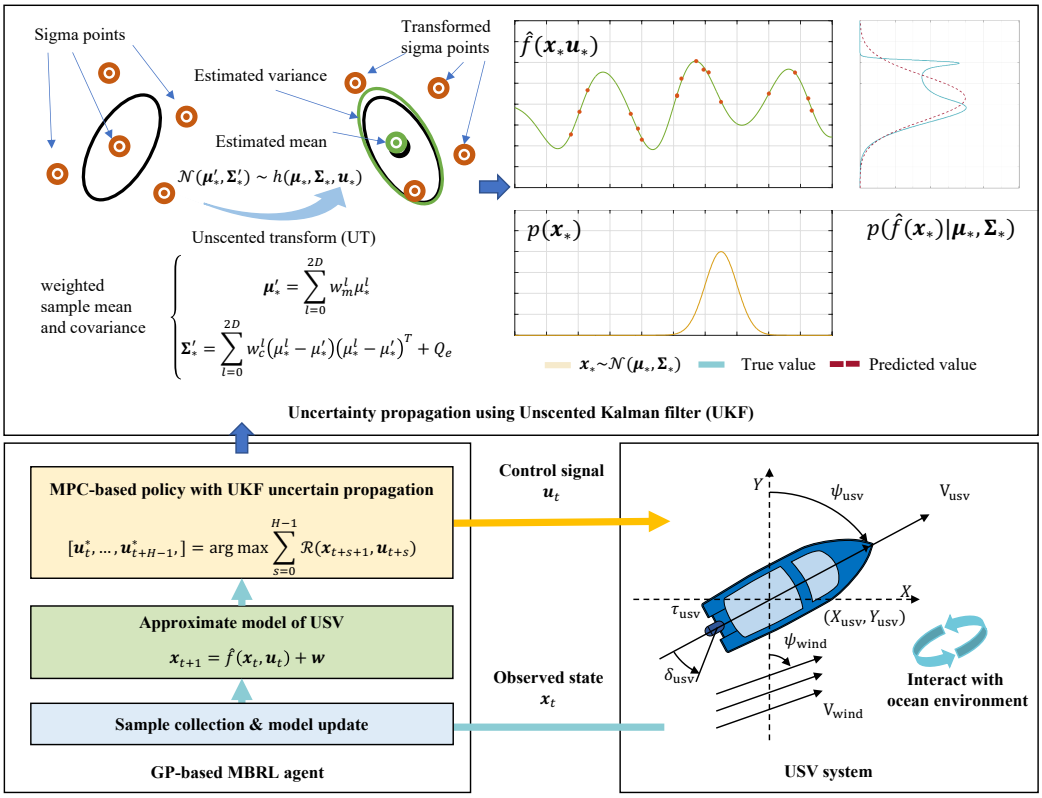

Figure 1. Principle of the uncertainty propagation in the GP-based MPC policy using UKF (**top**); overview of MBRL USV control system in this work (**bottom**).

2.2. Model-Based Reinforcement Learning

The goal of MBRL is to iteratively learn a policy $\pi : x_t \rightarrow u_t$ that maximizes the reward function in the long term by not only interacting with the environment but also approximating the following system transition model at time step t:

$$x_{t+1} = \hat{f}(x_t, u_t) + w \tag{1}$$

where w represents the model error. Define a data set $\mathcal{D}$ to store the exploration sample for updating the policy and system model, and the interaction process of MBRL can be generalized as a loop: (1) execute control signals via the current policy $u_t^* \sim \pi(x_t)$; (2) get the state in the next step x_{t+1} and expand data set; and (3) move to the next step $t = t + 1$, then return to 1.

As one appealing solution for alleviating the impact of complex and frequently changing disturbances, the MPC controller is widely utilized in several engineering MBRL systems [30–32]. At time step t, it plans a trajectory of control signals from the current state x_t to maximize the expected long-term reward over a pre-defined horizon H:

$$\begin{aligned} [u_t^*, \ldots, u_{t+H-1}^*] = & \underset{u_t, \ldots, u_{t+H-1}}{\arg\max} \sum_{s=0}^{H-1} \mathcal{R}(x_{t+s+1}, u_{t+s}). \\ \text{s.t.} \quad & x_{t+s+1} = \hat{f}(x_{t+s}, u_{t+s}) + w \\ & x_{t+s+1} \in \mathcal{S}, u_{t+s} \in \mathcal{A}. \end{aligned} \tag{2}$$

The nonlinear optimization approaches are applied to search for the optimal action sequence based on the approximated system model $\hat{f}(\cdot)$.

3. Approach

3.1. Probabilistic Model of USV

In this work, GP [26] is utilized to model the USV dynamics under a noisy environment in a probabilistic perspective. For each dimension of the state vector $a = 1, ..., D$, the approximated GP model of the target USV is as follows:

$$\boldsymbol{y}_{a,t+1} = \hat{f}_a(\tilde{\boldsymbol{x}}_t) + w_a \tag{3}$$

where $\tilde{\boldsymbol{x}}_t := (\boldsymbol{x}_t, \boldsymbol{u}_t)$ combines state and action vectors, and the unobservable noises in this dimension are simplified as one Gaussian noise $w_a \sim \mathcal{N}(0, \sigma_{w_a}^2)$. Since there is no assumption that the USV state must be fully observable or predicted, the output vector of model $\boldsymbol{y}_{t+1}$ is defined by elements $X_{\text{usv}}, Y_{\text{usv}}, \Psi_{\text{usv}}, V_{\text{usv}}, \tau_{\text{usv}}, \delta_{\text{usv}}$ without unpredictable wind information. As a general model of USV, it is also free to expand the observable states with pitch, yaw, roll and wave information using the specific sensors or remove any state that cannot be easily accessed due to the limited hardware. In this work, the observable states of USV are defined following the existing work [31], which has been implemented in real-ocean boat with relatively limited observation states.

Let $\tilde{\boldsymbol{\Lambda}}_a$ and $\alpha_{\hat{f}_a}^2$ denote a diagonal matrix of squared characteristic length-scales and the noise variances of the target system, then the measurement of the distance of any two inputs $k_a(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{x}}_j)$ is achieved by the squared exponential (SE) kernel function

$$k_a(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{x}}_j) = \alpha_{f_a}^2 e^{\left(-\frac{(\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j)^T \tilde{\boldsymbol{\Lambda}}_a^{-1} (\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j)}{2}\right)}. \tag{4}$$

Although the SE kernel is selected based on related works [30–32], it is also valuable and possible to investigate the power of other kernel functions [26] in modeling USV for challenging control tasks. We leave this issue as our future work.

Define the kernel function of one input $\tilde{\boldsymbol{x}}_*$ as $k_{a**} = k_a(\tilde{\boldsymbol{x}}_*, \tilde{\boldsymbol{x}}_*)$, the distance measurement of $\tilde{\boldsymbol{x}}_*$ over the whole sample set as $\boldsymbol{k}_{a*} = k_a(\tilde{\boldsymbol{X}}, \tilde{\boldsymbol{x}}_*)$, and the kernel matrix $\boldsymbol{K}_a$ with $K_{a,i,j} = k_a(\tilde{\boldsymbol{x}}_i)$ as the distance measurement of each sample in the whole set. Setting $\boldsymbol{\beta}_a = (\boldsymbol{K}_a + \sigma_{w_a}^2 \boldsymbol{I})^{-1} \boldsymbol{Y}_a$ as the irrelevant term to the input, its prediction is determined by the posterior mean $m_{\hat{f}_a}(\tilde{\boldsymbol{x}}_*)$ and the corresponding variance $\sigma_{\hat{f}_a}^2(\tilde{\boldsymbol{x}}_*)$:

$$m_{\hat{f}_a}(\tilde{\boldsymbol{x}}_*) := \mathbb{E}[\hat{f}_a(\tilde{\boldsymbol{x}}_*) | \tilde{\boldsymbol{X}}, \boldsymbol{Y}_a] = \boldsymbol{k}_{a*}^T (\boldsymbol{K}_a + \sigma_{w_a}^2 \boldsymbol{I})^{-1} \boldsymbol{Y}_a = \boldsymbol{k}_{a*}^T \boldsymbol{\beta}_a, \tag{5}$$

$$\sigma_{\hat{f}_a}^2(\tilde{\boldsymbol{x}}_*) := \text{var}[\hat{f}_a(\tilde{\boldsymbol{x}}_*) | \tilde{\boldsymbol{X}}, \boldsymbol{Y}_a] = k_{a**} - \boldsymbol{k}_{a*}^T (\boldsymbol{K}_a + \sigma_{w_a}^2 \boldsymbol{I})^{-1} \boldsymbol{k}_{a*}. \tag{6}$$

Given a training data set $\tilde{\boldsymbol{X}}, \boldsymbol{Y}$, the GP model's hyper-parameters $\alpha_{f_a}^2, \boldsymbol{\Lambda}_a, \sigma_{w_a}^2$ for each output dimension are determined by maximizing the log marginal likelihood via evidence maximization [26], which is called the training of the GP model:

$$\begin{aligned} [\tilde{\boldsymbol{\Lambda}}_a^*, \alpha_{\hat{f}_a}^*, \sigma_{w_a}^*] &= \arg\max \log p(\boldsymbol{Y}_a | \tilde{\boldsymbol{X}}, \tilde{\boldsymbol{\Lambda}}_a, \alpha_{\hat{f}_a}, \sigma_{w_a}) \\ p(\boldsymbol{Y}_a | \tilde{\boldsymbol{X}}, \tilde{\boldsymbol{\Lambda}}_a, \alpha_{\hat{f}_a}, \sigma_{w_a}) &= \log \int p(\boldsymbol{Y}_a | \hat{f}_a, \tilde{\boldsymbol{X}}, \tilde{\boldsymbol{\Lambda}}_a, \alpha_{\hat{f}_a}, \sigma_{w_a}) p(\hat{f}_a | \tilde{\boldsymbol{X}}, \tilde{\boldsymbol{\Lambda}}_a, \alpha_{\hat{f}_a}, \sigma_{w_a}) \mathrm{d}\hat{f}_a \\ &= -\frac{1}{2} \boldsymbol{Y}_a^T (\boldsymbol{K}_a + \sigma_{w_a}^2 \boldsymbol{I})^{-1} \boldsymbol{Y}_a - \frac{1}{2} \log |\boldsymbol{K}_a + \sigma_{w_a}^2 \boldsymbol{I}| - \frac{D}{2} \log(2\pi). \end{aligned} \tag{7}$$

3.2. Uncertainty Propagation Using Unscented Kalman Filter

One core characteristic of existing probabilistic MBRL approaches, such as PILCO [25] and GP-MPC [30], is the characteristic of propagating the uncertainty predicted by GP in their policy. They employed analytic moment matching [27], which propagates the

uncertainty by integrating the GP model over the Gaussian distribution input $p(\tilde{\boldsymbol{x}}_*) \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_*, \tilde{\boldsymbol{\Sigma}}_*)$ rather than the deterministic state $\boldsymbol{x}_*$:

$$p(\hat{f}(\tilde{\boldsymbol{x}}_*)|\tilde{\boldsymbol{\mu}}_*, \tilde{\boldsymbol{\Sigma}}_*) = \int p(\hat{f}(\tilde{\boldsymbol{x}}_*)|\tilde{\boldsymbol{x}}_*)p(\tilde{\boldsymbol{x}}_*)\mathrm{d}\tilde{\boldsymbol{x}}_*, \quad p(\tilde{\boldsymbol{x}}_*) \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_*, \tilde{\boldsymbol{\Sigma}}_*). \tag{8}$$

Since the resulting distribution is non-Gaussian without an analytical solution [25], it is estimated by a Gaussian distribution in analytic moment matching [27]:

$$\begin{aligned} [\boldsymbol{\mu}'_*, \boldsymbol{\Sigma}'_*] &= h(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*, \boldsymbol{u}_*) \\ \mathcal{N}(\boldsymbol{\mu}'_*, \boldsymbol{\Sigma}'_*) &\approx \int p\big(\hat{f}(\boldsymbol{x}_*, \boldsymbol{u}_*)|\boldsymbol{x}_*, \boldsymbol{u}_*\big)p(\boldsymbol{x}_*)\mathrm{d}\boldsymbol{x}_*. \end{aligned} \tag{9}$$

Although Equation (9) successfully captured uncertainties in various MBRL applications from the toy pendulum simulator to UGV and USV, it resulted in a heavy computational burden due to the frequent inversions of the kernel matrix over the full data set [31,42].

In this paper, we propose FPMPC-UKF to utilize UKF [34,35] to alleviate the computational burden of the uncertainty propagation of GP from an engineering perspective. Assuming the GP model's input follows D-dimensional Gaussian distribution $p(\boldsymbol{x}_*) \sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*)$, where the action $\boldsymbol{u}_*$ is deterministic following the setting of [31], SSUFK-MPC employs unscented transform (UT) to approximate the mean and variance of the target distribution by $2D+1$ sigma points, where D is the dimension of the GP input as demonstrated on the top of Figure 1. The first point is initialized as the current input mean $\boldsymbol{\chi}^0_* = \boldsymbol{\mu}_*$. Other $2D$ points are selected by

$$\boldsymbol{\chi}^l_* = \begin{cases} \boldsymbol{\mu}_* + \left(\sqrt{(D+\lambda)\boldsymbol{\Sigma}_*}\right)_l & , \quad l = 1, 2, \ldots, D, \\ \boldsymbol{\mu}_* - \left(\sqrt{(D+\lambda)\boldsymbol{\Sigma}_*}\right)_l & , \quad l = D+1, D+2, \ldots, 2D \end{cases} \tag{10}$$

where $\lambda = \alpha^2(D+\kappa) - N$ is the scale factor for reducing the prediction error. We set $\alpha = 1$ for a common distribution of sigma points. κ is set to 1 for the positive semi-definite of UKF. Calculating the GP prediction over these points,

$$\boldsymbol{\mu}^l_* = m_{f_a}(\boldsymbol{\chi}^l_*, \boldsymbol{u}_*), \quad l = 0, 1, \ldots, 2D. \tag{11}$$

The integration in Equation (9) can be approximated by $2D+1$ sigma points as follows:

$$\begin{aligned} \boldsymbol{\mu}'_* &= \sum_{l=0}^{2D} w^l_m \boldsymbol{\mu}^l_*, \\ \boldsymbol{\Sigma}'_* &= \sum_{l=0}^{2D} w^l_c \left(\boldsymbol{\mu}^l_* - \boldsymbol{\mu}'_*\right)\left(\boldsymbol{\mu}^l_* - \boldsymbol{\mu}'_*\right)^T + Q_e \end{aligned} \tag{12}$$

where $Q_e = 10^{-4}$ is an additional noise of variance for better robustness, and the weights of both predicting mean w^l_m and variance w^l_c for all $2D+1$ sigma points are calculated as

$$w^l_m = \begin{cases} \frac{\lambda}{D+\lambda}, & l = 0, \\ \frac{1}{2(D+\lambda)}, & l = 1, \ldots, 2D. \end{cases} \tag{13}$$

$$w^l_c = \begin{cases} \frac{\lambda}{D+\lambda} + (1 - \alpha^2 + \beta), & l = 0, \\ \frac{1}{2(D+\lambda)}, & l = 1, \ldots, 2D \end{cases} \tag{14}$$

where $\beta = 2$ is the weight coefficient.

According to [42], the computational complexity of analytic moment matching is $\mathcal{O}(D^3M^2)$, where D is the dimension number of the GP model, and M is the number of samples or sparse pseudo inputs. As a comparison, GP-based UKF achieved $\mathcal{O}(N^2M^2)$ in uncertainty propagation, which outperformed analytic moment matching, especially with large M. Despite the prediction error caused by the limited sampling of UKF, its superior computational efficiency resulted in wide engineering implementations [36–39]. With the equations above, we could efficiently calculate Equation (9) and employ it in the H steps prediction of Equation (2) to properly consider the propagated uncertainties in the MPC-based policy.

3.3. Filtered Probabilistic Model Predictive Control Using UKF

We detailed the workflow of the proposed method FPMPC-UKF for the USV control task in this subsection. Following the existing GP-based MBRL USV control system [31,32], the sample set $\mathcal{D} = (\tilde{X}, Y)$, where $\tilde{X}$ collects the state and action $\boldsymbol{x}, \boldsymbol{u}$ at the current time step t, $\tilde{Y}$ contains the controllable state $\boldsymbol{x} = [X_{\text{usv}}, Y_{\text{usv}}, \Psi_{\text{usv}}, V_{\text{usv}}, \tau_{\text{usv}}, \delta_{\text{usv}}]$ in the next step $t+1$. The corresponding learning loop is shown in Algorithm 1. At the beginning, the GP model of USV was initially trained by a pre-prepared sample set $\mathcal{D} = (\tilde{X}, Y)$ (usually generalized by random policy or human driver). The MBRL is updated in the N_{trial} rollout during the learning. At the start of each rollout, the USV is reset to its initial state with $X_{\text{usv}} = 0, Y_{\text{usv}} = 0, \Psi_{\text{usv}} = 0, V_{\text{usv}} = 0$. The initial observation at this moment was recorded as $\boldsymbol{\mu}_{0|0}$ with an initial variance $\boldsymbol{\Sigma}_{0|0}$, an empty control action that would be executed in the first step is set as $\boldsymbol{u}_0^* = [0, 0]$.

Algorithm 1: Learning loop of FPMPC-UKF USV control system.

Input initial prior variance $\boldsymbol{\Sigma}_{0|0}$, constant observation variance $\boldsymbol{\Sigma}_y$, executing time Δt, horizon of MPC-based policy H, sample set $\mathcal{D} = (\tilde{X}, Y)$.
Initialize GP model
h = `Train_GP(`$\tilde{X}, Y$`)`
for $i = 1, 2, ..., N_{trial}$ **do**
  $[\boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0}]$ =`Reset_USV_State()`, $\boldsymbol{u}_0^* = [0, 0]$
  for $t = 1, 2, ..., L_{rollout}$ **do**
    # CPU core 1
    # Execute USV signal
    `Operate_USV_Actions(`$\boldsymbol{u}_{t-1}^*$`)`
    # Observe USV state after Δt
    $\boldsymbol{x}_t^o$ = `Observe_USV_State()`
    # CPU core 2
    if $t > 1$ **then**
      # Bayesian filter process following Equations (15) and (16)
      $[\boldsymbol{\mu}_{t-1|t-2}, \boldsymbol{\Sigma}_{t-1|t-2}] = h(\boldsymbol{\mu}_{t-2|t-2}, \boldsymbol{\Sigma}_{t-2|t-2}, \boldsymbol{u}_{t-2}^*)$
      $\boldsymbol{\mu}_{t-1|t-1} = \boldsymbol{W}_f \boldsymbol{\mu}_{t-1|t-2} + \boldsymbol{W}_o \boldsymbol{y}_{t-1}$
      $\boldsymbol{\Sigma}_{t-1|t-1} = \boldsymbol{W}_f \boldsymbol{\Sigma}_{t-1|t-2}$
    # Bias compensation following Equation (17)
    $[\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\Sigma}}_t] = h(\boldsymbol{\mu}_{t-1|t-1}, \boldsymbol{\Sigma}_{t-1|t-1}, \boldsymbol{u}_{t-1}^*)$
    # Search MPC policy following Equation (2) using UKF
    $\boldsymbol{u}_t^*$ = `MPC_Policy(`$\boldsymbol{\mu}_{t-1|t-1}$, $\boldsymbol{\Sigma}_{t-1|t-1}$, h, H`)`
    # Expand sample set
    $\tilde{\boldsymbol{X}} = \{\tilde{\boldsymbol{X}}, (\boldsymbol{x}_{t-1}^o, \boldsymbol{u}_{t-1}^*)\}, \boldsymbol{Y} = \{\boldsymbol{Y}, \boldsymbol{y}_t\}$
  # Iteratively update GP model
  h = `Train_GP(`$\tilde{\boldsymbol{X}}, \boldsymbol{Y}$`)`
Return h

Following GPMPC [31] and FPMPC [32], the proposed method employs a double-prediction process by two separate CPU cores for swift control against the frequently

changing environmental disturbances. At time step $t = 1, ..., L_{\text{rollout}}$, two CPU parallelly worked. The first CPU core operated the pre-prepared control signal $\boldsymbol{u}^*_{t-1}$ and stored the currently observed state $\boldsymbol{x}^o_t$ to buffer. During the same period, the second CPU core first applied a Bayesian filter when $t > 1$ for superior robustness under complex environmental disturbances [32]. Given the previous state and control signal $[\boldsymbol{\mu}_{t-2|t-2}, \boldsymbol{\Sigma}_{t-2|t-2}, \boldsymbol{u}^*_{t-2}]$ at step $t-2$, the belief of the next step prediction was calculated by one-step prediction of Equations (11) and (12):

$$[\boldsymbol{\mu}_{t-1|t-2}, \boldsymbol{\Sigma}_{t-1|t-2}] = h(\boldsymbol{\mu}_{t-2|t-2}, \boldsymbol{\Sigma}_{t-2|t-2}, \boldsymbol{u}^*_{t-2}). \tag{15}$$

Defining $\boldsymbol{y}_{t-1}$ as the real observation $\boldsymbol{x}^o_{t-1}$ without unpredictable information, such as wind, and $\boldsymbol{\Sigma}_y$ as a constant observation variance, a Kalman filter based on the predicted belief was applied to estimate the posterior belief of the prediction with weights $\boldsymbol{W}_f$ and $\boldsymbol{W}_o$:

$$\begin{aligned} \boldsymbol{\mu}_{t-1|t-1} &= \boldsymbol{W}_f \boldsymbol{\mu}_{t-1|t-2} + \boldsymbol{W}_o \boldsymbol{y}_{t-1}, \\ \boldsymbol{\Sigma}_{t-1|t-1} &= \boldsymbol{W}_f \boldsymbol{\Sigma}_{t-1|t-2}, \\ \boldsymbol{W}_f &= \boldsymbol{\Sigma}_y (\boldsymbol{\Sigma}_{t-1|t-2} + \boldsymbol{\Sigma}_y)^{-1}, \\ \boldsymbol{W}_o &= \boldsymbol{\Sigma}_{t-1|t-2} (\boldsymbol{\Sigma}_{t-1|t-2} + \boldsymbol{\Sigma}_y)^{-1}. \end{aligned} \tag{16}$$

Based on the posterior belief, which was reported as a more robust and accurate representation of the state in noisy environments [32] and previous control signal $\boldsymbol{u}^*_{t-1}$, the USV state after executing the $\boldsymbol{u}^*_{t-1}$ was estimated to compensate for the potential bias caused by the current action:

$$[\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\Sigma}}_t] = h(\boldsymbol{\mu}_{t-1|t-1}, \boldsymbol{\Sigma}_{t-1|t-1}, \boldsymbol{u}^*_{t-1}) \tag{17}$$

which was achieved by one-step uncertainty propagation using UFK. The calculated state $[\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\Sigma}}_t]$ was set as the input state of searching the optimal actions of horizon H $[\boldsymbol{u}^*_t, ..., \boldsymbol{u}^*_{t+H-1}]$ following Equation (2). The first action $\boldsymbol{u}^*_t$ is executed by the first CPU core in the next step. After each step, the sample set was expanded by $\tilde{\boldsymbol{X}} = \{\tilde{\boldsymbol{X}}, (\boldsymbol{x}^o_{t-1}, \boldsymbol{u}^*_{t-1})\}, \boldsymbol{Y} = \{\boldsymbol{Y}, \boldsymbol{y}_t\}$, which is used for iteratively updating the GP model of USV at the end of each rollout.

4. Experimental Results

4.1. Simulation Settings

The proposed FPMPC-UKF was evaluated by the USV simulation developed based on the real boat-driven data under different ocean condition in [31], whose appendix detailed the parameters of dynamics. Three levels of disturbances, including observable wind (angle ψ_{wind} and velocity v_{wind}) and unobservable current (angle ψ_{current} and velocity v_{current}), were set to simulate increasing challenging ocean environments following the simulation settings in [31]:

* Level 1, $(\psi_{\text{wind}}) = 37^\circ + U(-30, 30)^\circ$, $\psi_{\text{current}} = 100^\circ + U(-30, 30)^\circ$, $v_{\text{wind}} = 2.0 + U(0, 0.1)$ m/s, $v_{\text{current}} = 0.25 + U(0, 0.1)$ m/s,
* Level 2, $\psi_{\text{wind}} = 37^\circ + U(-30, 30)^\circ$, $\psi_{\text{current}} = 100^\circ + U(-30, 30)^\circ$, $v_{\text{wind}} = 4.0 + U(0, 0.1)$ m/s, $v_{\text{current}} = 0.5 + U(0, 0.1)$ m/s,
* Level 3, $\psi_{\text{wind}} = 37^\circ + U(-30, 30)^\circ$, $\psi_{\text{current}} = 100^\circ + U(-30, 30)^\circ$, $v_{\text{wind}} = 6.0 + U(0, 0.1)$ m/s, $v_{\text{current}} = 0.5 + U(0, 0.1)$ m/s.

The wind and current changedper step following uniform distributions U. The simulated boat had a single outboard engine with rudder range $\delta_{\text{usv}} \in [-30^\circ, +30^\circ]$, and the engine speed was mapped to $\tau_{\text{usv}} \in [0\%, 100\%]$ up to 800 rpm.

In this paper, we focused on two USV tasks: position keeping and target reaching. Define the reward function in Equation (2) as

$$\mathcal{R}(\boldsymbol{p}_{t+1}) = -||\boldsymbol{p}_{t+1} - \boldsymbol{p}_{target}||^2 \tag{18}$$

where $\boldsymbol{p} = [\boldsymbol{X}_{\text{usv}}, \boldsymbol{Y}_{\text{usv}}]$ represents the location of USV at the corresponding time step. For the position-keeping task, the target position is set as $\boldsymbol{p}_{target} = [0, 0]$. For the target-reaching task, it is set to $\boldsymbol{p}_{target} = [20, 25]$. For the setting of the MPC-based policy, we select the prediction horizon as $H = 3$, and the initial variance $\boldsymbol{\Sigma}_{0|0}$ and observation variance $\boldsymbol{\Sigma}_y$ are set as 10^{-4}. The GP model of USV is initially trained with 500 samples with a random control policy. The agent is iteratively trained in $N_{trial} = 20$ rollouts with length $L_{\text{rollout}} = 100$. Exploration noise was added to actions $\boldsymbol{w}_e$, set to $\boldsymbol{\Sigma}_e = 0.1$. After the training process, it is evaluated by another 30 rollouts. For the baseline approach, we select FPMPC [32] based on its superior performance among related works. It can be treated as the proposed method without employing UKF uncertainty propagation. All approaches were developed GPflow [43], while the bound optimization by quadratic approximation [44] in NLopt (http://github.com/stevengj/nlopt accessed on 20 May 2021) was used to optimize the MPC-based policy. All experimental results were conducted on a computational server with an Intel Xeon(R) W-2275 CPU and 64 GB memory by five independent trials with different random seeds for statistical evidence.

4.2. Evaluation of Learning Capability

We first evaluated the learning capability of the proposed method in position keeping and target reaching with different sparse GP pseudo input sizes. According to the learning curves in Figure 2, both FPMPC-UKF and the baseline approach achieved convergence behaviors within 20 rollouts under various numbers of pseudo input in sparse GP. Compared with FPMPC using analytic moment matching, the proposed method enjoyed more stable learning curves with less standard deviation in the average position offset and usually converged faster with less sparse GP pseudo input ($\leq$10). These results indicate the superiority of FPMPC-UKF in convergence with less pseudo input in sparse GP. With the increasing level of disturbances and pseudo input, both methods achieved a larger standard deviation in the position-keeping task, while the learning curves of the target-reaching task are relatively flat. The main reason is the higher USV velocity in the target-reaching task, which resulted in better control capability against disturbances.

After training, the performance results of the proposed method in the testing procedures were studied, shown in Tables 2 and 3, respectively. Several terms are compared in these tables, including the average offsets, the median offsets and the average optimization time. The success rates (final) indicate the rates of successfully holding the USV near the target within 7 m at the end of each testing rollout. The success rates (overall), which are only for the position-keeping task, indicate the success rate of holding the USV during the whole rollout.

The result of the position-keeping task is shown in Table 2. The proposed FPMPC-UKF outperformed the baseline approach FPMPC under all three levels of disturbances with sparse GP pseudo input ≤ 10. With a 5 sparse size, despite its convergence in the training procedure, FPMPC had insufficient generalization capability to finish the test. With the increasing level of disturbances, it achieved very large average offsets (from 7 m to 28 m) with low success rates (from 18% to 63%). As a comparison, FPMPC-UKF significantly outperformed the baseline; it achieved 65% less average offset and 66% higher success rate in the level 3 disturbances. Setting the sparse size to 10, FPMPC-UKF enjoyed better performance with significant superiority to FPMPC: it achieved 29% to 78% less average offsets and 17% to 70% higher success rates. With a sparse size larger than 10, the baseline approach usually outperformed the proposed method as the power of analytical moment matching in uncertainty propagation was fully released. However, it turned to heavy computational burdens, which are reported in the next section.

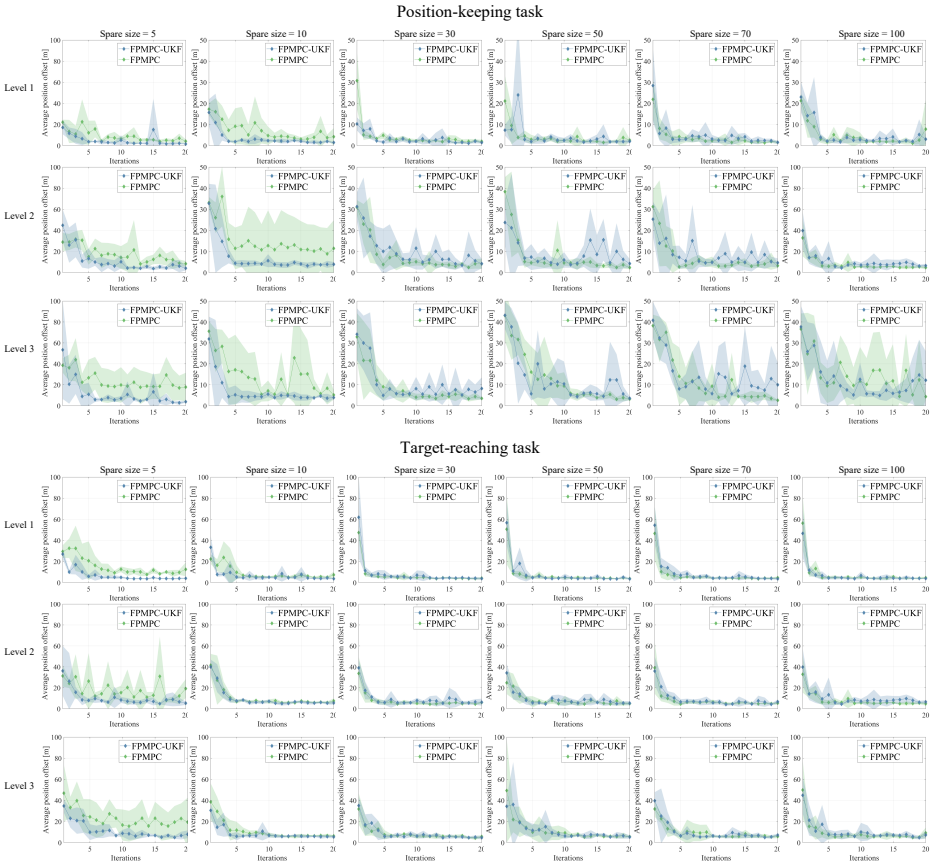

Figure 2. Learning curves under different disturbances levels in position keeping and target reaching with different sparse GP pseudo input size, the lines and opaque regions representing position offsets' mean and standard deviation.

The result in the target-reaching task is shown in Table 3, which is close to the one in the position-keeping task. FPMPC-UKF outperformed the baseline approach at all three levels of disturbances with sparse GP pseudo input ≤ 10 in both average offset and success rate. With a 5 sparse size, FPMPC achieved large average offsets (from 10 m to 31 m) with low success rates (from 24% to 47%) in all three levels of disturbances. As a comparison, FPMPC-UKF reduced more than 50% average offset with 34% to 44% higher success rate. With 10 sparse size, FPMPC-UKF still outperformed FPMPC: it achieved 11% to 44% less average offsets and 4% to 21% higher success rates. On the other hand, it obtained less benefit from more sparse GP pseudo input compared with FPMPC using analytic moment matching.

4.3. Evaluation of Computational Efficiency and Model Quality

We further evaluate the computational efficiency (including one-step prediction time and the overall prediction time in the MPC policy) and quickly model the orientation and velocity, i.e., the model error of position, in one-step MPC prediction in this section. According to the results of the proposed and the baseline approaches (represented by green and red colors) in both position-keeping and target-reaching tasks, demonstrated in Figures 3 and 4, the proposed FPMPC-UKF demonstrated a significant advantage in the average optimization time with sparse size > 5, especially under larger environmental disturbances. With 5 sparse GP pseudo inputs, although the average optimization time of the proposed method was increased (up to 104% in the target-research task with level 1 disturbances), it was limited within 0.5 s, which is superior to the existing MBRL USV control system [31]. Meanwhile, FPMPC-UKF achieved great improvements in the model prediction error, which contributed to over 50% decrease in the average offset and a significant increase in success rate, reported in Tables 2 and 3.

Table 2. Test result of the proposed method and baseline approach in the position-keeping.

Disturbances	Method	Sparse Sizes	Average Offset [m]	Median Offset [m]	Success Rate (Final)	Success Rate (Overall)
Level 1	FPMPC	5	7.82 ± 14.35	4.25	62.67%	11.33%
		10	5.10 ± 5.66	3.20	80.00%	20.00%
		30	2.22 ± 2.90	1.54	96.00%	72.00%
		50	2.30 ± 4.31	1.26	92.67%	77.33%
		70	2.21 ± 2.20	1.56	95.33%	69.33%
		100	5.06 ± 11.06	1.42	85.33%	67.33%
	FPMPC-UKF	5	2.41 ± 4.79 (−69.18%)	1.47 (−65.41%)	97.33% (+34.66%)	78.00% (+66.67%)
		10	1.65 ± 1.21 (−67.65%)	1.35 (−57.81%)	99.33% (+19.33%)	90.67% (+70.67%)
		30	2.49 ± 5.11 (+12.16%)	1.47 (−4.55%)	94.67% (−1.33%)	77.33% (+5.33%)
		50	1.93 ± 2.34 (−16.09%)	1.34 (+6.35%)	95.33% (+2.66%)	80.67% (+3.34%)
		70	2.55 ± 3.13 (+15.38%)	1.67 (+7.05%)	92.00% (−3.33%)	52.67% (−16.66%)
		100	2.31 ± 1.77 (−54.35%)	1.86 (+30.99%)	97.33% (+12.00%)	62.67% (−4.66%)
Level 2	FPMPC	5	14.38 ± 19.20	9.01	26.00%	1.33%
		10	18.80 ± 33.99	4.88	64.67%	4.00%
		30	4.90 ± 8.99	3.05	88.00%	18.00%
		50	4.71 ± 7.21	3.03	88.67%	24.67%
		70	3.27 ± 3.47	2.34	94.67%	37.33%
		100	5.84 ± 10.48	3.26	85.33%	10.00%
	FPMPC-UKF	5	4.94 ± 6.43 (−65.65%)	3.73 (−58.60%)	77.33% (+51.33%)	13.33% (+12.00%)
		10	4.01 ± 4.20 (−78.67%)	3.35 (−31.35%)	95.33% (+30.66%)	24.67% (+20.67%)
		30	6.40 ± 10.56 (+30.61%)	3.36 (+10.16%)	78.67% (−9.33%)	15.33% (−2.67%)
		50	4.95 ± 5.94 (+5.10%)	3.05 (+0.66%)	82.00% (−6.67%)	12.67% (−12.00%)
		70	6.28 ± 8.44 (+92.05%)	3.44 (+47.01%)	75.33% (−19.34%)	8.00% (−29.33%)
		100	7.16 ± 13.32 (+22.60%)	3.60 (+10.43%)	74.00% (−11.33%)	5.33% (−4.67%)
Level 3	FPMPC	5	28.51 ± 35.60	14.59	18.67%	0.00%
		10	6.05 ± 5.37	4.98	70.67%	1.33%
		30	3.52 ± 2.64	2.95	94.00%	20.97%
		50	4.17 ± 6.98	2.86	90.67%	19.33%
		70	5.04 ± 7.84	3.44	86.00%	13.33%
		100	7.16 ± 11.99	4.04	76.00%	1.33%
	FPMPC-UKF	5	9.87 ± 34.20 (−65.38%)	4.27 (−70.73%)	85.33% (+66.66%)	14.67% (+14.67%)
		10	4.29 ± 2.69 (−29.09%)	3.88 (−22.09%)	82.00% (+1.33%)	18.67% (+17.34%)
		30	6.67 ± 10.09 (+89.49%)	4.02 (+36.27%)	70.00% (−24.00%)	12.00% (−8.67%)
		50	10.09 ± 16.18 (+141.97%)	4.07 (+42.31%)	72.00% (−18.67%)	9.33% (−10.00%)
		70	5.85 ± 5.91 (+16.07%)	4.13 (+20.06%)	78.67% (−7.33%)	2.67% (−10.66%)
		100	9.01 ± 12.12 (+25.84%)	4.89 (+21.04%)	67.33% (−8.67%)	0.67% (−0.66%)

Table 3. Test result of the proposed method and baseline approach in the target-reaching.

Disturbances	Method	Sparse Sizes	Average Offset [m]	Median Offset [m]	Success Rate (Final)
Level 1	FPMPC	5	10.91 ± 12.45	7.01	47.33%
		10	5.99 ± 6.55	4.19	78.00%
		30	3.53 ± 5.87	1.72	96.00%
		50	3.19 ± 5.58	1.28	95.33%
		70	3.57 ± 5.92	1.67	92.00%
		100	4.27 ± 6.79	1.94	88.00%
	FPMPC-UKF	5	4.89 ± 10.16 (−55.18%)	1.76 (−74.89%)	92.00% (+44.67%)
		10	3.33 ± 5.06 (−44.41%)	1.89 (−54.89%)	99.33% (+21.33%)
		30	2.87 ± 5.15 (−18.70%)	1.35 (−21.51%)	98.00% (+2.00%)
		50	3.40 ± 5.33 (+6.58%)	1.76 (+37.50%)	89.33% (−6.00%)
		70	3.45 ± 5.23 (−3.36%)	1.84 (+10.18%)	94.67% (+2.67%)
		100	3.22 ± 5.15 (−24.59%)	1.75 (−9.79%)	98.67% (+10.67%)
Level 2	FPMPC	5	15.43 ± 19.86	12.13	24.00%
		10	5.85 ± 5.71	4.33	75.33%
		30	4.48 ± 5.69	2.89	92.67%
		50	5.42 ± 7.28	3.37	83.33%
		70	4.49 ± 5.77	2.72	90.00%
		100	5.90 ± 7.80	3.67	80.00%
	FPMPC-UKF	5	7.64 ± 12.61 (−50.49%)	4.38 (−63.89%)	66.00% (+42.00%)
		10	5.15 ± 5.18 (−11.97%)	3.85 (−11.09%)	83.33% (+8.00%)
		30	6.53 ± 8.52 (+45.76%)	3.43 (+18.69%)	70.67% (−22.00%)
		50	4.62 ± 5.51 (−14.76%)	2.81 (−16.62%)	82.67% (−0.66%)
		70	6.52 ± 8.98 (+45.21%)	4.04 (+48.53%)	68.67% (−21.33%)
		100	6.09 ± 6.49 (3.22%)	4.17 (+13.62%)	72.67% (−7.33%)
Level 3	FPMPC	5	31.64 ± 39.31	16.15	27.33%
		10	6.38 ± 10.04	3.93	77.33%
		30	7.89 ± 19.91	2.86	84.67%
		50	5.11 ± 5.52	3.47	78.67%
		70	5.49 ± 6.68	3.39	87.33%
		100	5.17 ± 6.07	3.66	80.67%
	FPMPC-UKF	5	8.37 ± 14.42 (−73.55%)	5.46 (−66.19%)	62.00% (+34.67%)
		10	5.43 ± 6.41 (−14.89%)	3.94 (+0.25%)	81.33% (+4.00%)
		30	7.30 ± 11.61 (−7.48%)	3.76 (+31.47%)	78.00% (−6.67%)
		50	5.71 ± 6.22 (+11.74%)	4.15 (+19.60%)	72.00% (−6.67%)
		70	5.81 ± 5.71 (+5.83%)	4.22 (+24.48%)	72.67% (−14.66%)
		100	6.27 ± 6.59 (+21.28%)	4.22 (+15.30%)	68.00% (−12.67%)

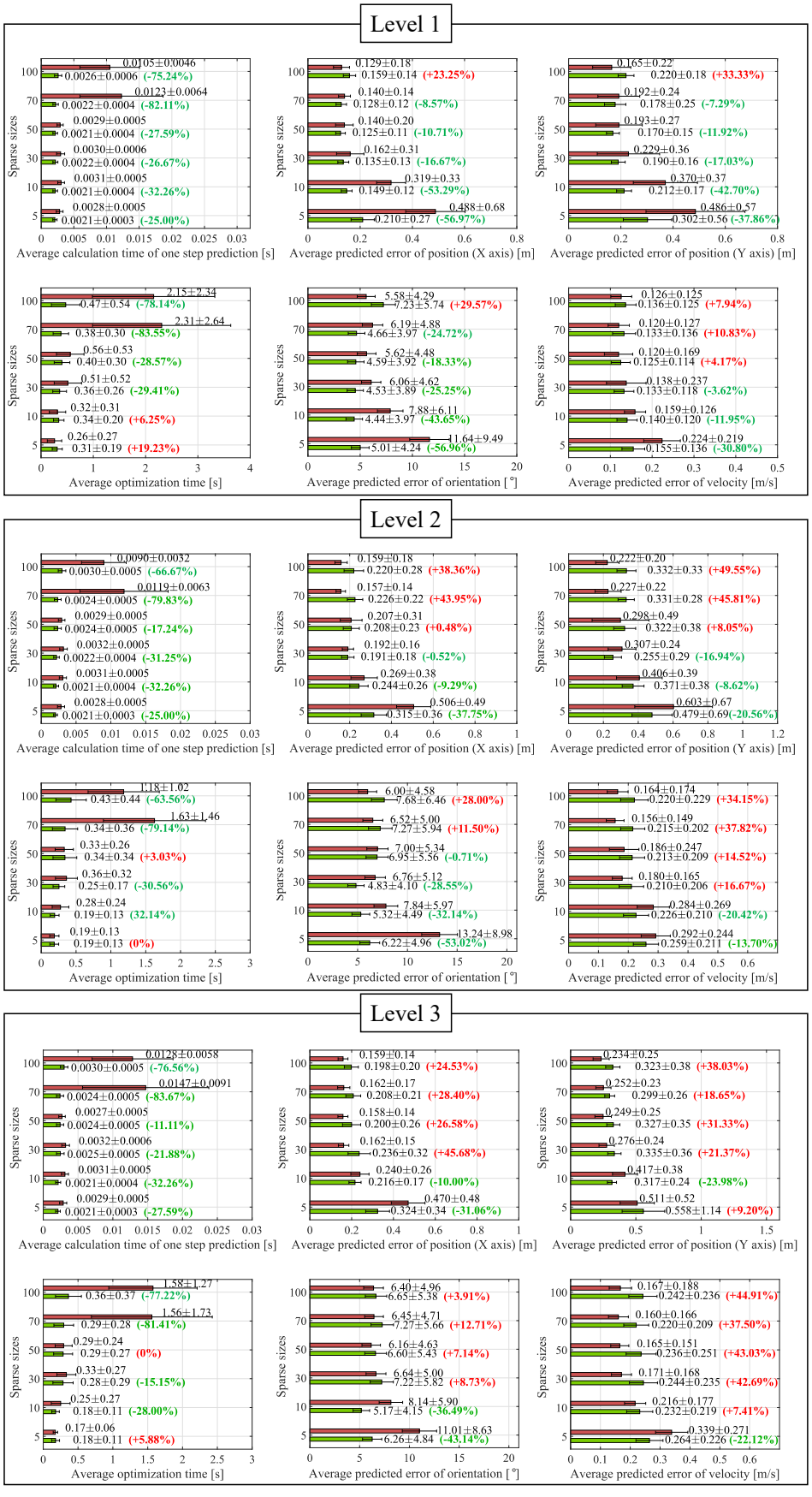

Figure 3. Average predicted time and errors of MPC-based policy in position keeping. The green and red colors indicate the proposed method and baseline.

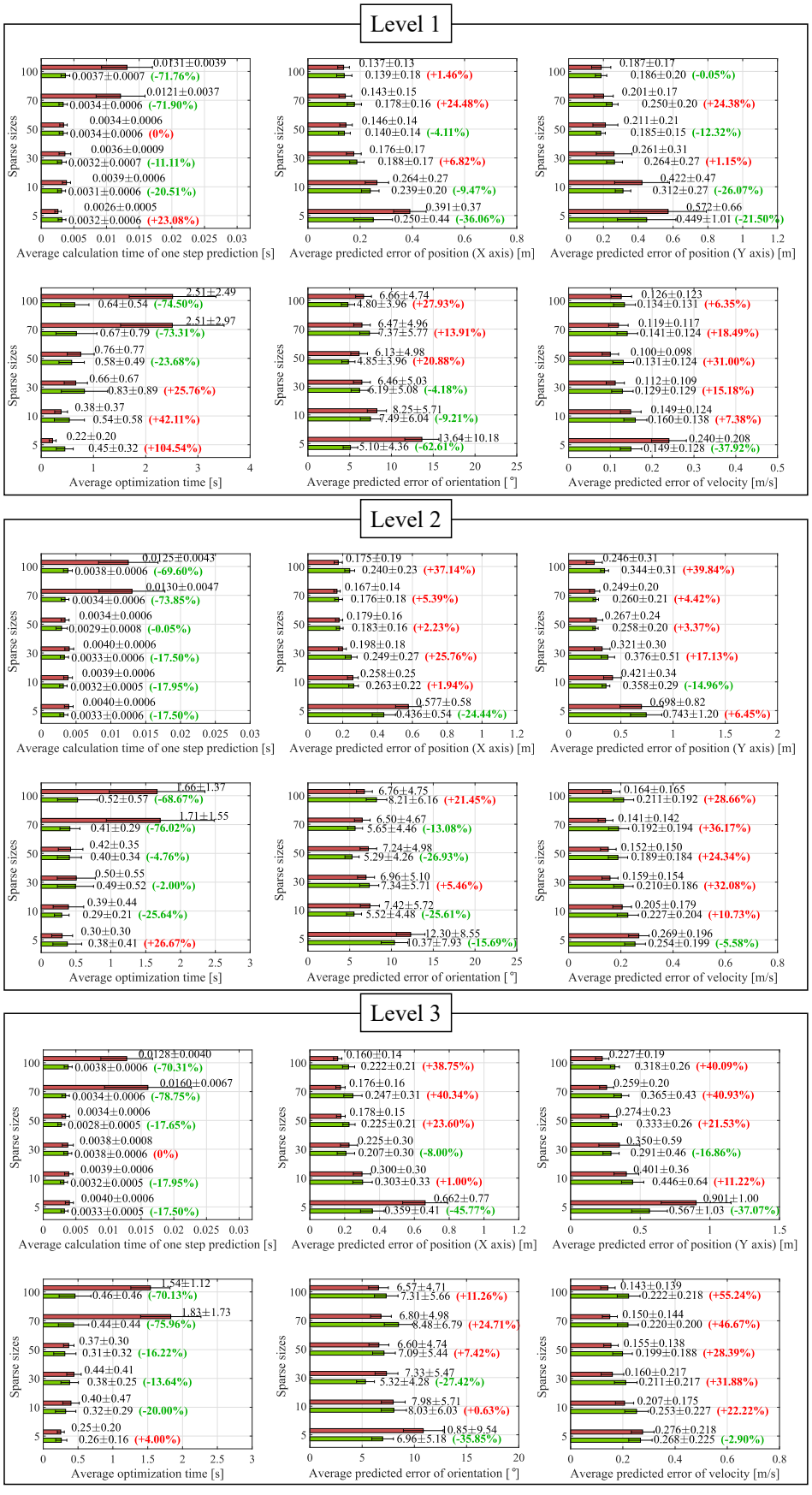

Figure 4. Average predicted time and errors of MPC-based policy in target reaching. The green and red colors indicate the proposed method and baseline.

Increasing the sparse GP pseudo inputs from 10 to 100, the baseline approach improved its model quality with heavy computational burdens (from 0.5 s to more than 2 s). As a comparison, the proposed method still achieved close model prediction errors under these sparse sizes while enjoying a far faster optimization. Compared with FPMPC, which required over 2 s to search the optimal action with 100 sparse GP pseudo inputs, the proposed method only needs less than 0.5 s (more than 70% faster), which is more suitable in a real USV control system with a limited computational resource. Overall, the proposed method considerably outperformed the baseline approach in control performance and model prediction error, while enjoying a faster optimization time (about 0.3 s) with less sparse GP pseudo input. With the increase in sparse GP pseudo input, the computational burden of the proposed method increased significantly less than the baseline approach, while maintaining close control performance and model prediction errors. These results demonstrated the great potential of FPMPC-UKF in balancing its learning capability and computational complexity and therefore contributed to a higher control frequency in MBRL USV.

Based on our experimental results, one proper balance between the control performance and computational efficiency was met when the sparse GP pseudo input was set to 10. In the position-keeping task, the proposed method achieved 67%, 78%, and 29% less average offsets, and 70%, 20%, and 17% higher success rates in three levels of disturbances, while achieving 28% to 32% superior computational efficiency, except for one tiny degradation in optimization time in level 1 disturbances compared with the baseline approach. In the target-reaching task, the proposed method achieved 15%, and 12% less average offsets, and 21%, 8%, and 4% higher success rates in three levels of disturbances while achieving 20% to 25% superior computational efficiency, except for one degradation in optimization (42.11% slower but still be close to 0.5 s) time in level 1 disturbances compared with FPMPC.

Figure 5 illustrates the trajectories of USV in position-keeping and target-reaching tasks under level 3 disturbances during one test rollout of FPMPC and FPMPC-UKF with 10 sparse pseudo inputs as one case study. The environmental disturbances for all approaches were generated by the same random seed. It can be observed that in both tasks, the proposed FPMPC-UKF enjoyed better control performance. The fewer prediction errors in the orientation and velocity resulted in a more sophisticated control policy that smoothly drove the USV to hold its position/reach its target without over-large control signals in both the rudder and throttle. After reaching the target, the proposed method fully utilized uncertainty propagation to keep its position against disturbances. On the other hand, the analytic moment matching in the baseline approach did not work well in such a large GP sparse scale and therefore resulted in unstable driving trajectories and struggled with position keeping.

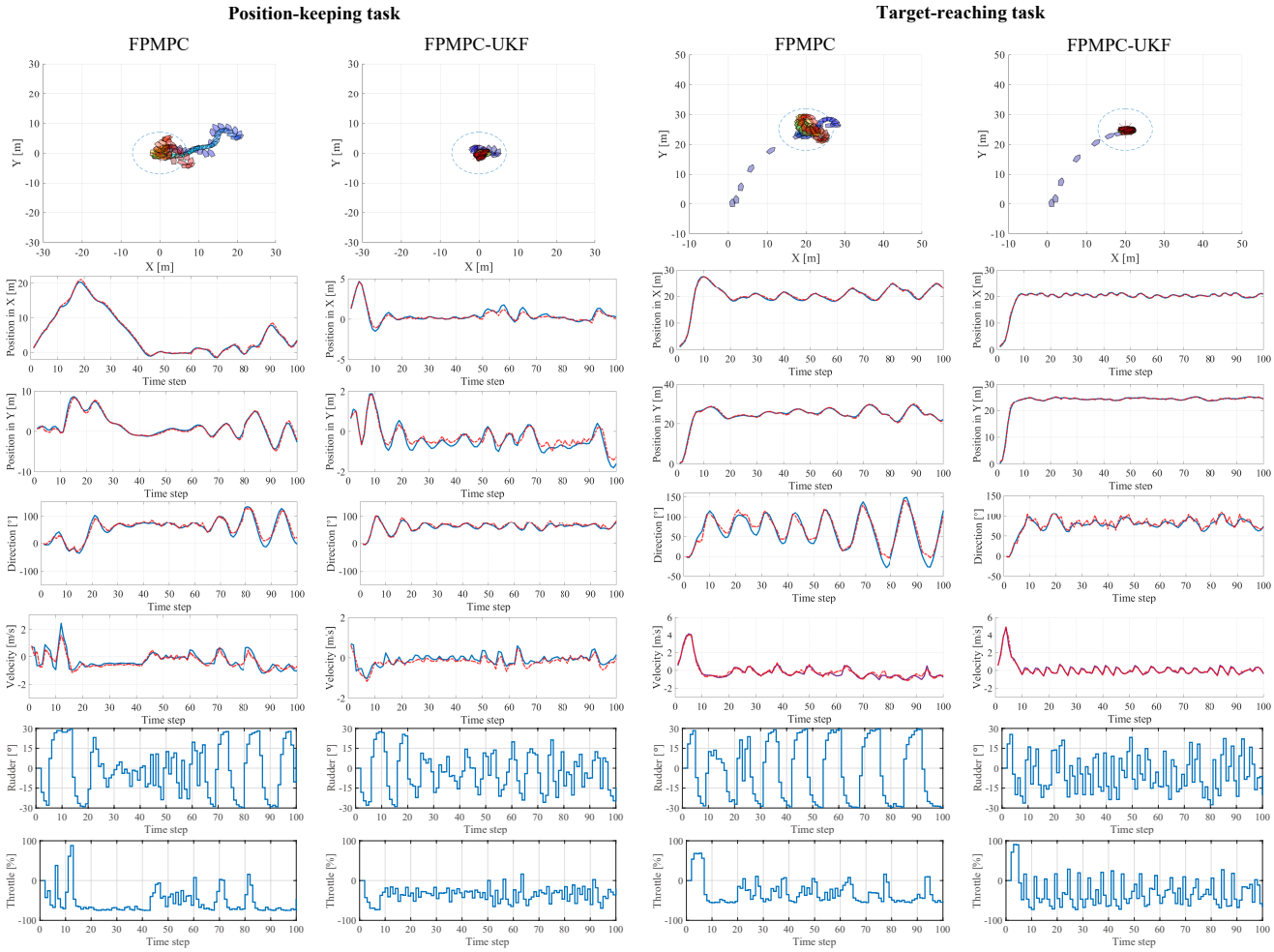

Figure 5. Trajectories of USV including states and control signals in one test rollout of position-keeping and target-reaching tasks under level 3 disturbances with 10 sparse pseudo inputs. The real value is presented as blue lines, and the predicted mean of model is shown as red lines.

5. Conclusions

In this work, we proposed FPMPC-UKF, a novel GP-based MBRL approach specific for USV by introducing an efficient uncertainty propagation using UKF to the MPC-based policy for a superior control performance of USV against the real-time disturbances, especially with less sparse GP pseudo input. As a bridge connecting probabilistic MBRL and optimal filtering technologies toward fully autonomous USV, FPMPC-UKF naturally released the efficiency of the optimal filter in uncertainty propagation under the trial-and-error framework of RL with a system specific to USV. The proposed method was validated in both position-keeping and target-reaching tasks under different levels of environmental disturbances. The comprehensive comparisons with the related baseline approach in the learning capability, computation efficiency and model quality show the superiority of the proposed method in balancing the learning capability, control performance and computational burdens, which expands the potential of probabilistic MBRL in more challenging USV scenarios with limited computational resources.

Author Contributions: Conceptualization, J.W. and Y.C.; methodology, J.W. and Y.C.; software, J.W. and L.X.; validation, H.L. and Y.C.; formal analysis, J.W.; investigation, J.W.; writing—original draft preparation, J.W. and Y.C.; writing—review and editing, Y.C.; visualization, J.W.; supervision, L.P., H.L. and Y.C.; project administration, Y.C.; funding acquisition, L.P., H.L. and Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported in part by the National Natural Science Foundation of China under Grants 62103403; in part by the National Key Research and Development Program of China under Grant 2020YFB2104300; in part by Guangdong Basic and Applied Basic Research Foundation under Grant No. 2020B515130004; in part by the Science and Technology Development Fund, Macao S.A.R. (FDCT) under Grant 0015/2019/AKP.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

USV	Unmanned Surface Vehicle
RL	Reinforcement Learning
MDP	Markov Decision Process
POMDP	Partially observed Markov Decision Process
MBRL	Model-based Reinforcement Learning
PILCO	Probabilistic Inference for Learning Control
GP	Gaussian Processes
MPC	Model Predictive Control
SPMPC	Sample-efficient Probabilistic Model Predictive Control
FPMPC	Filtered Probabilistic Model Predictive Control
UKF	Unscented Kalman Filter
UGV	Unmanned Ground Vehicle
UAV	Unmanned Aerial Vehicle
FPMPC-UKF	Filtered Probabilistic Model Predictive Control with Unscented Kalman Filter
UT	Unscented Transform

References

1. Sarda, E.I.; Qu, H.; Bertaska, I.R.; von Ellenrieder, K.D. Station-keeping control of an unmanned surface vehicle exposed to current and wind disturbances. *Ocean. Eng.* **2016**, *127*, 305–324. [CrossRef]
2. Guo, G.; Zhang, P. Asymptotic Stabilization of USVs With Actuator Dead-Zones and Yaw Constraints Based on Fixed-Time Disturbance Observer. *IEEE Trans. Veh. Technol.* **2020**, *69*, 302–316. [CrossRef]
3. Zhou, W.; Wang, Y.; Ahn, C.K.; Cheng, J.; Chen, C. Adaptive Fuzzy Backstepping-Based Formation Control of Unmanned Surface Vehicles With Unknown Model Nonlinearity and Actuator Saturation. *IEEE Trans. Veh. Technol.* **2020**, *69*, 14749–14764. [CrossRef]
4. Yang, Y.; Wu, J.; Zheng, W. Station-keeping control for a stratospheric airship platform via fuzzy adaptive backstepping approach. *Adv. Space Res.* **2013**, *51*, 1157–1167. [CrossRef]
5. Vu, M.T.; Le Thanh, H.N.N.; Huynh, T.T.; Thang, Q.; Duc, T.; Hoang, Q.D.; Le, T.H. Station-keeping control of a hovering over-actuated autonomous underwater vehicle under ocean current effects and model uncertainties in horizontal plane. *IEEE Access* **2021**, *9*, 6855–6867. [CrossRef]
6. Wang, N.; Karimi, H.R. Successive Waypoints Tracking of an Underactuated Surface Vehicle. *IEEE Trans. Ind. Informatics* **2020**, *16*, 898–908. [CrossRef]
7. Zhao, Z.; Zhu, B.; Zhou, Y.; Yao, P.; Yu, J. Cooperative Path Planning of Multiple Unmanned Surface Vehicles for Search and Coverage Task. *Drones* **2023**, *7*, 21. [CrossRef]
8. Qin, H.; Wu, Z.; Sun, Y.; Chen, H. Disturbance-Observer-Based Prescribed Performance Fault-Tolerant Trajectory Tracking Control for Ocean Bottom Flying Node. *IEEE Access* **2019**, *7*, 49004–49013. [CrossRef]
9. Wu, Y.; Low, K.H.; Lv, C. Cooperative Path Planning for Heterogeneous Unmanned Vehicles in a Search-and-Track Mission Aiming at an Underwater Target. *IEEE Trans. Veh. Technol.* **2020**, *69*, 6782–6787. [CrossRef]
10. Wang, N.; He, H. Extreme Learning-Based Monocular Visual Servo of an Unmanned Surface Vessel. *IEEE Trans. Ind. Informatics* **2021**, *17*, 5152–5163. [CrossRef]
11. Divelbiss, A.W.; Wen, J.T. Trajectory tracking control of a car-trailer system. *IEEE Trans. Control. Syst. Technol.* **1997**, *5*, 269–278. [CrossRef]
12. Yu, R.; Shi, Z.; Huang, C.; Li, T.; Ma, Q. Deep reinforcement learning based optimal trajectory tracking control of autonomous underwater vehicle. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017; pp. 4958–4965.

13. Eriksen, B.O.H.; Breivik, M.; Wilthil, E.F.; Flåten, A.L.; Brekke, E.F. The branching-course model predictive control algorithm for maritime collision avoidance. *J. Field Robot.* **2019**, *36*, 1222–1249. [CrossRef]
14. Wang, N.; Su, S.F.; Pan, X.; Yu, X.; Xie, G. Yaw-guided trajectory tracking control of an asymmetric underactuated surface vehicle. *IEEE Trans. Ind. Informatics* **2018**, *15*, 3502–3513. [CrossRef]
15. United Nations Conference on Trade and Development. Review of Maritime Transport 2018; United Nations: Geneva, Switzerland, 2018; p. 115. [CrossRef]
16. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press Cambridge, UK, 1998.
17. Kober, J.; Bagnell, J.A.; Peters, J. Reinforcement learning in robotics: A survey. *Int. J. Robot. Res.* **2013**, *32*, 1238–1274. [CrossRef]
18. Arulkumaran, K.; Deisenroth, M.P.; Brundage, M.; Bharath, A.A. Deep reinforcement learning: A brief survey. *IEEE Signal Process. Mag.* **2017**, *34*, 26–38. [CrossRef]
19. Wang, N.; Gao, Y.; Zhang, X. Data-Driven Performance-Prescribed Reinforcement Learning Control of an Unmanned Surface Vehicle. *IEEE Trans. Neural Networks Learn. Syst.* **2021**, *32*, 1–12. [CrossRef]
20. Zhao, Y.; Qi, X.; Ma, Y.; Li, Z.; Malekian, R.; Sotelo, M.A. Path Following Optimization for an Underactuated USV Using Smoothly-Convergent Deep Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 1–13. [CrossRef]
21. Zhao, L.; Roh, M.I. COLREGs-compliant multiship collision avoidance based on deep reinforcement learning. *Ocean Eng.* **2019**, *191*, 106436. [CrossRef]
22. Woo, J.; Yu, C.; Kim, N. Deep reinforcement learning-based controller for path following of an unmanned surface vehicle. *Ocean Eng.* **2019**, *183*, 155–166. [CrossRef]
23. Wang, N.; Zhang, Y.; Ahn, C.K.; Xu, Q. Autonomous Pilot of Unmanned Surface Vehicles: Bridging Path Planning and Tracking. *IEEE Trans. Veh. Technol.* **2022**, *71*, 2358–2374. [CrossRef]
24. Woo, J.; Kim, N. Collision avoidance for an unmanned surface vehicle using deep reinforcement learning. *Ocean Eng.* **2020**, *199*, 107001. [CrossRef]
25. Deisenroth, M.P.; Fox, D.; Rasmussen, C.E. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 408–423. [CrossRef] [PubMed]
26. Rasmussen, C.E.; Williams, C.K. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, UK, 2006.
27. Girard, A.; Rasmussen, C.E.; Candela, J.Q.; Murray-Smith, R. Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, UK, 2003, pp. 545–552.
28. Bischoff, B.; Nguyen-Tuong, D.; van Hoof, H.; McHutchon, A.; Rasmussen, C.E.; Knoll, A.; Peters, J.; Deisenroth, M.P. Policy search for learning robot control using sparse data. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 3882–3887.
29. Cutler, M.; How, J.P. Efficient reinforcement learning for robots using informative simulated priors. In Proceedings of the 2015 IEEE international conference on robotics and automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2605–2612.
30. Kamthe, S.; Deisenroth, M. Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Laguna Hills, CA, USA, 25–26 March 2018; pp. 1701–1710.
31. Cui, Y.; Osaki, S.; Matsubara, T. Autonomous boat driving system using sample-efficient model predictive control-based reinforcement learning approach. *J. Field Robot.* **2021**, *38*, 331–354. [CrossRef]
32. Cui, Y.; Peng, L.; Li, H. Filtered Probabilistic Model Predictive Control-Based Reinforcement Learning for Unmanned Surface Vehicles. *IEEE Trans. Ind. Informatics* **2022**, *18*, 6950–6961. [CrossRef]
33. Snelson, E.; Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, UK, 2005; Volume 18.
34. Wan, E.A.; Van Der Merwe, R. The unscented Kalman filter. In *Kalman Filtering and Neural Networks*; Oregon Graduate Institute of Science & Technology: Beaverton, OR, USA, 2001; pp. 221–280.
35. Ko, J.; Klein, D.J.; Fox, D.; Haehnel, D. GP-UKF: Unscented Kalman filters with Gaussian process prediction and observation models. In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, 29–30 June 2007; pp. 1901–1907.
36. Ostafew, C.J.; Schoellig, A.P.; Barfoot, T.D. Robust constrained learning-based NMPC enabling reliable mobile robot path tracking. *Int. J. Robot. Res.* **2016**, *35*, 1547–1563. [CrossRef]
37. Liu, Z.; Li, Y.; Wu, Y.; He, S. Formation control of nonholonomic unmanned ground vehicles via unscented Kalman filter-based sensor fusion approach. *ISA Trans.* **2022**, *125*, 60–71. [CrossRef]
38. Zhai, C.; Wang, M.; Yang, Y.; Shen, K. Robust vision-aided inertial navigation system for protection against ego-motion uncertainty of unmanned ground vehicle. *IEEE Trans. Ind. Electron.* **2020**, *68*, 12462–12471. [CrossRef]
39. Song, W.; Wang, J.; Zhao, S.; Shan, J. Event-triggered cooperative unscented Kalman filtering and its application in multi-UAV systems. *Automatica* **2019**, *105*, 264–273. [CrossRef]
40. Wang, Y.; Chai, S.; Nguyen, H.D. Unscented Kalman filter trained neural network control design for ship autopilot with experimental and numerical approaches. *Appl. Ocean. Res.* **2019**, *85*, 162–172. [CrossRef]
41. Shen, H.; Wen, G.; Lv, Y.; Zhou, J.; Wang, L. USV Parameter Estimation: Adaptive Unscented Kalman Filter-Based Approach. *IEEE Trans. Ind. Informatics* **2022**, 1–10. [CrossRef]

42. Deisenroth, M.P. Efficient Reinforcement Learning using Gaussian Processes. Ph.D. Thesis, Fakultat fur Informatik , Karlsruhe, Germany, 2010. [CrossRef]
43. Matthews, A.G.d.G.; Van Der Wilk, M.; Nickson, T.; Fujii, K.; Boukouvalas, A.; León-Villagrá, P.; Ghahramani, Z.; Hensman, J. GPflow: A Gaussian Process Library using TensorFlow. *J. Mach. Learn. Res.* **2017**, *18*, 1–6.
44. Powell, M.J. *The BOBYQA Algorithm for Bound Constrained Optimization without Derivatives;* Cambridge NA Report NA2009/06; University of Cambridge: Cambridge, UK, 2009; Volume 26, pp. 1–39.