

# DOB-Net: Actively Rejecting Unknown Excessive Time-Varying Disturbances

Tianming Wang<sup>1</sup>, Wenjie Lu<sup>1,2</sup>, Zheng Yan<sup>3</sup> and Dikai Liu<sup>1</sup>

**Abstract**—This paper presents an observer-integrated Reinforcement Learning (RL) approach, called Disturbance Observer Network (DOB-Net), for robots operating in environments where disturbances are unknown and time-varying, and may frequently exceed robot control capabilities. The DOB-Net integrates a disturbance dynamics observer network and a controller network. Originated from conventional DOB mechanisms, the observer is built and enhanced via Recurrent Neural Networks (RNNs), encoding estimation of past values and prediction of future values of unknown disturbances in RNN hidden state. Such encoding allows the controller generate optimal control signals to actively reject disturbances, under the constraints of robot control capabilities. The observer and the controller are jointly learned within policy optimization by advantage actor critic. Numerical simulations on position regulation tasks have demonstrated that the proposed DOB-Net significantly outperforms conventional feedback controllers and classical RL policy.

## I. INTRODUCTION

Autonomous Underwater Vehicles (AUVs) have become vital tools in search and rescue, exploration, surveillance, monitoring, and other applications [1], [2]. For large AUVs in deep water applications, the strength and changes of underwater wave and current disturbances may be negligible to the AUVs, due to their considerable size and thrust capabilities. While small AUVs are required for some shallow water applications, like bridge pile inspection [3], where the disturbances coming from the turbulent flows may frequently exceed AUVs' thrust capabilities. These unknown disturbances inevitably bring adverse effects and may even destabilize robots [4]. Thus, this paper studies an optimal control problem of robots subject to excessive time-varying disturbances, which are in the form of forces, and presents an observer-integrated RL solution.

RL [5] is a trial-and-error method that does not require an explicit system model, and can naturally adapt to noises and uncertainties in the real system. However, the excessive disturbances are not appropriate to be regarded as noises any more, since AUV's state transition is heavily affected by the external disturbances, thus violating the assumption of Markov Decision Process (MDP). While considering the

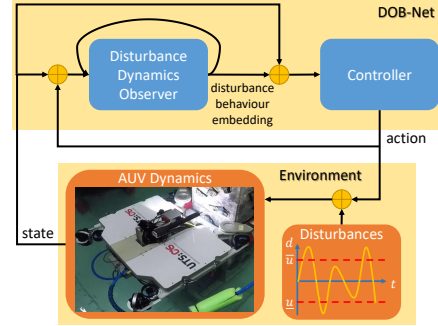


Fig. 1. Working flow of the DOB-Net ( $\bar{u}$  and  $\underline{u}$  are control limits).

time-varying characteristics of the wave and current disturbances, if future disturbances can be predicted, RL may be able to generate optimal controls.

Conventional DOB [6] and Disturbance-Observer-Based Control (DOBC) [7], [8] have been investigated in the last four decades. The main objective is to estimate the disturbances, then produce controls to compensate their influence. However, conventional DOBC does have some limitations when meeting such excessive time-varying disturbances. The first limitation is that DOB normally needs an accurate system model, which may not be available for underwater vehicles due to complex hydrodynamics. In this case, model uncertainties are lumped with external disturbances, and then estimated by DOB together. Thus, the original properties for some disturbances, such as harmonic ones, are changed. Secondly, even with an sufficiently accurate estimate of disturbances at the current time step, the optimal control solution is still unreachable, since disturbances exceeding control constraints cannot be well rejected only through feedback regulation. Thus, AUV behaviors need to be optimized over a future time horizon considering time correlation of disturbances.

This paper proposes a novel RL approach called DOB-Net, which enables integrated learning of disturbance dynamics and an optimal controller, for water wave and current disturbance rejection control of AUVs in shallow and turbulent water, as shown in Fig 1. The DOB-Net consists of a disturbance dynamics observer network and a controller network. The observer network is built and enhanced via RNNs, through imitating conventional DOB mechanisms. But this network is more flexible, since it encodes the prediction of disturbances in RNN hidden state, instead of only estimating the current value of disturbances. Also, the observer is more robust to model uncertainties and time-varying characteristics of disturbances. Based on the encoded disturbance prediction, the controller network is able to actively reject the

<sup>1</sup>Tianming Wang, Wenjie Lu and Dikai Liu are with Centre for Autonomous Systems, University of Technology Sydney, Ultimo, NSW 2007, Australia tianming.wang@student.uts.edu.au {wenjie.lu, dikai.liu}@uts.edu.au

<sup>2</sup>Wenjie Lu is with School of Mechanical Engineering and Automation, Harbin Institute of Technology (Shenzhen), Shenzhen, 518055 Guangdong, P.R. China wenjie.lu@outlook.com

<sup>3</sup>Zheng Yan is with Centre for Artificial Intelligence, University of Technology Sydney, Ultimo, NSW 2007, Australia yan.zheng@uts.edu.au

unknown disturbances. The observer and the controller are jointly learned within policy optimization by advantage actor critic. This integrated learning may achieve an optimized representation of observer outputs, compared with traditional hand-designed features. The policy is trained using simulated disturbances consisting of multiple sinusoidal waves, and evaluated using both simulated disturbances and collected disturbances, the latter is collected in a water tank with artificial waves.

In this paper, the related work is presented in Section II. Section III introduces problem formulation. Section IV provides the detailed description of the DOB-Net. Then, Section V presents validation procedures and result analysis. Some potential future improvements are discussed in the last section.

## II. RELATED WORK

### A. Feedback and Predictive Control

In the early development of disturbance rejection control, feedback control strategies are used to suppress the unknown disturbances [9]–[11]. Then, disturbance estimation and attenuation methods through adding a feedforward compensation term have been proposed and practiced, such as DOBC [6], [7]. However, these methods are built on the assumption of bounded disturbances which should be small enough, thus fail to guarantee stability under disturbances exceeding control constraints [12].

To this end, Model Predictive Control (MPC) [13] is often applied due to its constraint handling capacity through optimizing plant behaviour over a certain time horizon [12]. The MPC requires a prediction model of the system to optimize future behaviour, this model includes not only the robot dynamics, but also the predicted disturbances over next optimization horizon. Thus, a compound control scheme consisting of DOB and MPC (DOB-MPC) [14] has been developed. However, its performance heavily relies on the accuracy of given system model, and the requirement for online optimization at each time step leads to a low computational efficiency. Besides, such separated modeling and control optimization process might not be able to produce models and controls that jointly optimize robot behaviours, as evidenced in [15]. In contrast, the DOB-Net uses neural networks to construct both the observer and the controller, achieving model-free control, high computational efficiency as well as a joint optimization of the observer and the controller.

### B. Classical RL

RL has drawn a lot of attention in finding optimal controllers for systems that are difficult to model accurately. Recently, deep RL algorithms based on Q-learning [16], policy gradients [17], and actor-critic methods [18] have been shown to learn complex skills in high-dimensional state and action spaces. RL generally considers stochastic systems of the form  $x_{t+1} = f(x_t, u_t) + \varepsilon$  [19], with state variables  $x \in \mathbb{R}^D$ , control signal  $u \in \mathbb{R}^K$  and i.i.d. system noise marginalized over time  $\varepsilon \sim \mathcal{N}(0, E)$ , where  $E = \text{diag}(\sigma_1^2, \dots, \sigma_D^2)$ . While

the wave and current disturbances should be regarded as functions of time instead of random noises, due to its large amplitudes and time-varying characteristics, as discussed in Section V.

### C. History Window Approach

When using RL to deal with unknown disturbances, the problem cannot be defined as a MDP since the transition function does not only depend on the current state and action, but also heavily on the disturbances. The history window approaches [20] attempt to resolve the hidden state by making the selected action depend not only on the current state, but also on a fixed number of the most recent states and actions. Wang et al. [21] applied this approach to handle the external disturbances of an AUV through characterizing the disturbed AUV dynamics model as a multi-order Markov chain  $x_{t+1} = f_h(H_t, x_t, u_t)$ , and assuming the unobserved time-varying disturbances and their prediction over next planning horizon are encoded in state-action history of fixed length  $H_t = \{x_{t-N}, u_{t-N}, \dots, x_{t-1}, u_{t-1}\}$ . Then, the policy is trained to generate control signals based on a fixed length of state-action history along with current state. However, it is difficult to determine an optimal length of the history. Shorter history may not provide sufficient information about disturbances, longer history will make the training difficult. Wang et al. [21] considered the history length as a hyperparameter that was statistically optimized during training.

### D. Recurrent Policy

Due to the difficulty in determining optimal history length through history window approach, RNN is then utilized to automatically learn how much past experience should be explored to achieve optimal performance. Using RNN to represent policies is a popular approach to handle partial observability and solve tasks that require memory [22]–[24]. The idea being that the RNN will be able to retain information from observations and actions further back in time, and incorporate this information into predicting better actions and value functions, thus performing better on tasks that require long term planning. Compared with the recurrent RL approach, the contributions of DOB-Net lie in the exploration and application of the architectural similarities between Gated Recurrent Unit (GRU) and conventional DOB, and then the ability to encode the prediction of disturbance dynamics function.

## III. PROBLEM FORMULATION

### A. System Description

In this research, we assume that a 6 Degree Of Freedom (DOF) AUV is designed to be sufficiently stable in orientation even under strong disturbances, thanks to its large restoring forces. Thus, we only consider the disturbance rejection control of the vehicle's 3-DOF position, and assume its orientation is well controlled all the time. However, the framework can be easily extended to 6-DOF case, where a larger network and longer training time may be required. And it is also applicable for other kinds of mobile robots,

such as quadrotors [25], gliders, and surface vessels, but the effectiveness needs further investigation.

The AUV model can be considered as a floating rigid body with external disturbances, which can be represented by

$$\begin{aligned} M(q)\ddot{q} + G(q, \dot{q}) &= u + d(t), \\ G(q, \dot{q}) &= C(q, \dot{q})\dot{q} + D(q, \dot{q})\dot{q} + g(q), \end{aligned} \quad (1)$$

where  $M(q) \in \mathbb{R}^{3 \times 3}$  is the inertia matrix,  $C(q, \dot{q}) \in \mathbb{R}^{3 \times 3}$  is the matrix of Coriolis and centripetal terms,  $D(q, \dot{q}) \in \mathbb{R}^{3 \times 3}$  is the matrix of drag force,  $g(q) \in \mathbb{R}^3$  is the vector of the gravity and buoyancy forces,  $q, \dot{q}, \ddot{q} \in \mathbb{R}^3$  represent replacements, velocities and accelerations of the AUV,  $u \in \mathbb{R}^3$  represents the control forces.  $d(t) \in \mathbb{R}^3$  represents the time-varying disturbance forces, and the variation of  $d(t)$  with time from the past to the future is the disturbance dynamics, which is what the observer network tries to produce. We also assume that the magnitudes of the disturbances will exceed the AUV control constraints  $\bar{u} \in \mathbb{R}^3$  and  $\underline{u} \in \mathbb{R}^3$ , but are constrained within reasonable ranges, ensuring the controller is able to stabilize the AUV in a sufficiently small region around the target position. The whole AUV dynamic model is assumed to have fixed parameters, which are unknown for the learning algorithm or the controller.

### B. Problem Definition

In RL, the goal is to learn a policy that chooses actions  $u_t$  at each time step  $t$  in response to the current state  $x_t$ , such that the total expected sum of discounted rewards is maximized over all time. The state of the robot consists of position as well as the corresponding velocities  $x = [q^T \dot{q}^T]^T \in \mathcal{X} \in \mathbb{R}^6$ . The action includes the control forces  $u \in \mathcal{U} \in \mathbb{R}^3$ . At each time step, the system transitions from  $x_t$  to  $x_{t+1}$  in response to the chosen action  $u_t$  and the transition dynamics function  $f: \mathcal{X} \times \mathcal{U} \rightarrow \mathcal{X}$ , collecting a reward  $r_t$  according to reward function  $r(x_t, u_t) = x_t^T Q x_t + u_t^T R u_t$ , where  $Q \in \mathbb{R}^{6 \times 6}$  and  $R \in \mathbb{R}^{3 \times 3}$  represent weight matrices. The discounted sum of future rewards is then defined as  $\sum_{t'=t}^{T-1} \gamma^{t'-t} r(x_{t'}, u_{t'})$ , where  $\gamma \in [0, 1]$  is a discount factor that prioritizes near-term rewards over distant rewards [26].

## IV. METHODOLOGY

The underwater disturbances present great challenges for stabilization control due to its excessive amplitudes as well as time-varying characteristics. In this section, a conventional DOB is first compared with a GRU, the results show some similarities in the structure of processing hidden information. Thus, an enhanced observer network for excessive time-varying disturbances is designed using GRUs, encoding the disturbance dynamics into GRU hidden state. A controller

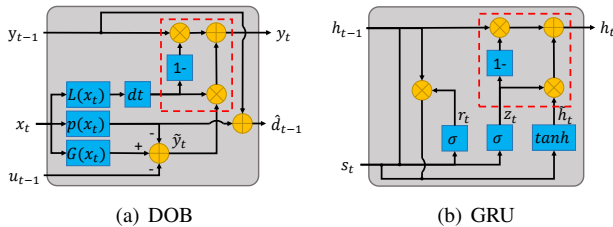


Fig. 2. Architecture of DOB and GRU.

network is then built upon this encoding in order to generate optimal controls.

### A. Conventional DOB

The basic idea of conventional DOB is to estimate disturbances based on robot state and executed controls, its formulation is proposed as

$$\begin{aligned} \dot{y} &= -L(q, \dot{q})y + L(q, \dot{q})(G(q, \dot{q}) - p(q, \dot{q}) - u), \\ \hat{d} &= y + p(q, \dot{q}), \end{aligned} \quad (2)$$

where  $\hat{d} \in \mathbb{R}^3$  is the estimated disturbances,  $y \in \mathbb{R}^3$  is the internal state of the nonlinear observer,  $p(q, \dot{q})$  is the nonlinear function to be designed, and  $L(q, \dot{q})$  is the DOB gain. It has been shown in [6] that DOB is globally asymptotically stable by choosing  $L(q, \dot{q}) = \text{diag}\{c, c\}$ , where  $c > 0$ . The convergence and the performance of the DOB have been established for slowly time-varying disturbances and disturbances with bounded rate in [27]. A discrete version of DOB is also provided (illustrated in Fig. 2(a))

$$\begin{aligned} \tilde{y}_t &= G(x_t) - p(x_t) - u_{t-1}, \\ y_t &= (1 - L(x_t)dt)y_{t-1} + L(x_t)dt\tilde{y}_t, \\ \hat{d}_{t-1} &= y_{t-1} + p(x_t). \end{aligned} \quad (3)$$

### B. Gated Recurrent Unit (GRU)

The architecture of GRU [28] is shown in Fig. 2(b). The formulations are given below:

$$\begin{aligned} z_t &= \sigma(W_z[h_{t-1}, s_t] + b_z), \\ r_t &= \sigma(W_r[h_{t-1}, s_t] + b_r), \\ \tilde{h}_t &= \tanh(W_h[r_t \circ h_{t-1}, s_t] + b_h), \\ h_t &= (1 - z_t) \circ h_{t-1} + z_t \circ \tilde{h}_t, \end{aligned} \quad (4)$$

where  $s_t$  is the input vector,  $h_t$  is the output vector,  $z_t$  is the update gate vector,  $r_t$  is the reset gate vector,  $W$  and  $b$  are the weight matrices and bias vectors,  $\sigma$  and  $\tanh$  are the activation functions (sigmoid function and hyperbolic tangent). The operator  $\circ$  denotes the Hadamard product.

### C. DOB-Net

The DOB-Net is constructed based on classical actor-critic architecture. As described in Fig. 2(a) and Fig. 2(b), the DOB and GRU have a similar architecture, especially the part in the red box.  $y_t$  of DOB acts as the hidden state, similar to the role of  $h_t$  in GRU, which preserves past processed information. In order to imitate the function of DOB, a GRU is first employed to process the state-action pair  $[x_t, u_{t-1}]$ . Then, fully connected layers are required to further extract

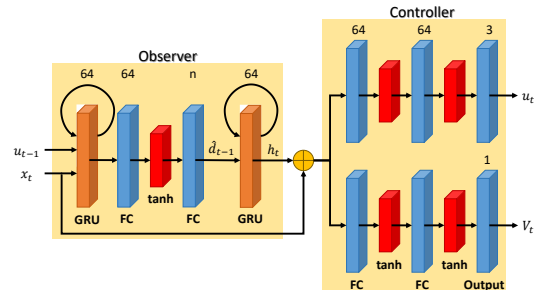


Fig. 3. Network architecture of DOB-Net.

**Algorithm 1: DOB-Net - pseudocode for each thread**


---

Assume global params  $\theta$ ,  $\theta_v$  and thread params  $\theta'$ ,  $\theta'_v$   
Initialize global counter  $T=0$  and thread counter  $t=1$   
**repeat**  
    Reset gradients:  $d\theta \leftarrow 0$  and  $d\theta_v \leftarrow 0$   
    Synchronize parameters  $\theta' = \theta$  and  $\theta'_v = \theta_v$   
     $t_{start} = t$   
    Get state  $x_t$ , last action  $u_{t-1}$ , last hidden state  $h_{t-1}$   
    **repeat**  
        Sample  $u_t$  according to  $\pi(u_t|x_t, u_{t-1}, h_{t-1}; \theta')$ ,  
        receive  $h_t$   
        Perform  $u_t$ , receive  $r_t$  and  $x_{t+1}$   
         $t \leftarrow t+1$  and  $T \leftarrow T+1$   
    **until** terminal  $x_t$  or  $t - t_{start} == t_{max}$ ;  
     $R = \begin{cases} 0 & \text{for terminal } x_t \\ V(x_t, u_{t-1}, h_{t-1}; \theta'_v) & \text{for non-terminal } x_t \end{cases}$   
    **for**  $i \in \{t-1, \dots, t_{start}\}$  **do**  
         $R \leftarrow r_i + \gamma R$   
        Accumulate gradients wrt  $\theta'$ :  $d\theta \leftarrow$   
         $d\theta + \nabla_{\theta'} \log \pi(u_i|x_i, u_{i-1}, h_{i-1}; \theta') (R -$   
         $V(x_i, u_{i-1}, h_{i-1}; \theta'_v))$   
        Accumulate gradients wrt  $\theta'_v$ :  $d\theta_v \leftarrow$   
         $d\theta_v + \partial (R - V(x_i, u_{i-1}, h_{i-1}; \theta'_v))^2 / \partial \theta'_v$   
    **end**  
    Perform update of  $\theta$ ,  $\theta_v$  using  $d\theta$ ,  $d\theta_v$   
**until**  $T > T_{max}$ ;

---

embedding of estimated disturbances  $\hat{d}_{t-1}$ . After disturbance estimation, the observer network can be further enhanced through feeding this embedding into another GRU, in order to encode a sequence of past estimated disturbances. The hidden state of the second GRU  $h_t$  is supposed to represent the disturbance dynamics. It can then be combined with the current state  $x_t$ , becoming the actual inputs of the controller network. One design parameter of the DOB-Net is the embedding dimension of  $\hat{d}_{t-1}$ . In this paper, 3-dimension (the dimension of disturbances) and 64-dimension (the dimension of RNN hidden state) are chosen and compared in simulation. Such comparison shows the flexibility of neural networks after building the observer from GRU.

**Training:** Advantage Actor Critic (A2C) [18] is a conceptually simple and lightweight framework for RL that uses synchronous gradient descent for optimization of neural network controllers. The algorithm synchronously executes multiple agents in parallel, on multiple instances of environment. This parallelism also decorrelates the agents' data into a more stationary process, since at any given time step the parallel agents will be experiencing a variety of different states. The algorithm is developed in A2C style. Pseudocode of the DOB-Net is shown in Algorithm 1. Each thread interacts with its own copy of environment. The disturbances are also different in each thread, and each of them are randomly sampled. We found this setting helps accelerate the convergence of learning and improve performance.

## V. SIMULATION EXPERIMENTS

### A. Simulation Setup

A position regulation task is simulated to test the DOB-Net algorithm. The simulated AUV has the mass  $m = 60 \text{ kg}$  with

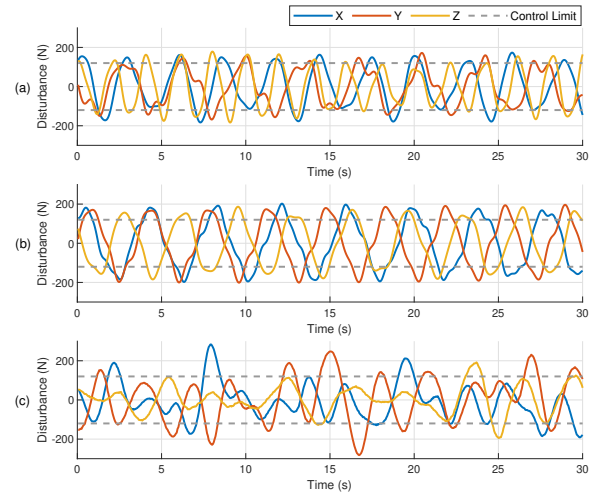


Fig. 4. Example disturbances in X, Y and Z directions, (a) small simulated disturbances; (b) large simulated disturbances; (c) collected disturbances.

the size of  $0.8 \times 0.8 \times 0.25 \text{ m}^3$ . Only positional motion and control are considered, thus the AUV has a 6-dimensional state space and a 3-dimensional action space. The control constraints  $|\bar{u}| = |\underline{u}| = [120\text{N } 120\text{N } 120\text{N}]^T$ . Each training episode contains 200 steps with 0.05s per step. In each episode, the robot starts at a random position with a random velocity, and it is controlled to reach a target position and stay within a region (refer to as converged region) thereafter.

In these simulations, the RL algorithms are trained using simulated disturbances, and tested using both simulated disturbances and collected disturbances. The simulated disturbances are constructed as a superposition of multiple sinusoidal waves (three in these simulations) with different amplitudes, frequencies and phases. Two different scenarios are considered, one has close or slightly excessive amplitudes (around 100-120% of control constraints, Fig. 4 (a)), the other one has larger amplitudes (130-150% of control constraints, Fig. 4 (b)). Our purpose is to enable the trained policy to deal with unknown time-varying disturbances, thus their amplitudes, frequencies and phases are randomly sampled from the given distributions in each training episode. In order to further validate the efficacy of the DOB-Net, we also collected wave and current disturbance data in a water tank using wave generators (refer to as collected disturbances), as shown in Fig. 4 (c). The data is collected through an onboard Inertial Measurement Unit (IMU) of an unactuated AUV, the measured linear accelerations are mapped to forces, which can be assumed as the disturbances. We notice that the amplitudes of the collected disturbances are not constrained within the ranges seen during training, leading to a more challenging scenario.

Ten different algorithms for disturbance rejection control are tested and compared:

- (1) Robust Integral of Sign Error (RISE) Control [29]
- (2) DOBC
- (3) A2C
- (4) History Window A2C with state history (HWA2C-x)
- (5) HWA2C with state-action history (HWA2C-xu)
- (6) Recurrent A2C with state history (RA2C-x)



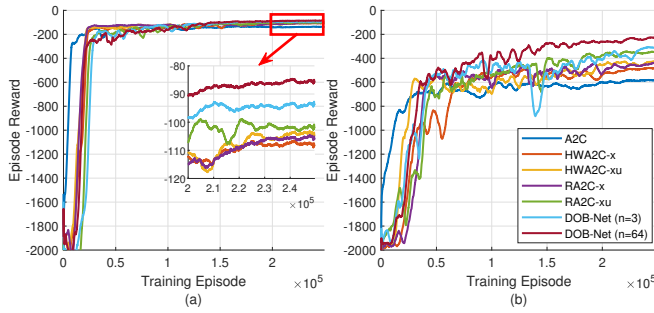


Fig. 5. Training rewards, (a) small simulated disturbances (amplitude between 100-120% of control constraints); (b) large simulated disturbances (amplitude between 130-150% of control constraints).

- (7) RA2C with state-action history (RA2C-xu)
- (8) DOB-Net ( $n = 3$ )
- (9) DOB-Net ( $n = 64$ )
- (10) Trajectory Optimization

Notice that, among these algorithms, the trajectory optimization assumes that the full knowledge of the disturbances over the whole episode is given in advance, while all other algorithms deal with unknown disturbances. The comparison is obviously not fair, the trajectory optimization is used only to provide optimal performance under ideal conditions, it can be regarded as an informal upper bound for all the algorithms. Our goal is to narrow the gap between the performance of the proposed algorithm and that of the trajectory optimization solution. RISE control is a conventional feedback controller; HWA2C applies the history window approach into the A2C framework, the used window length is 10 time steps, which is 0.5s in our simulation setup; and RA2C employs RNNs to deal with the past states and actions. The applied A2C framework employs a parallel training mode, 16 agents are used at the same time, the equivalent real-world training time for each agent is 43.4 hours. In the remaining part of this section, we first evaluate the training process of different algorithms, then test and compare the control performance among them using either the simulated disturbances or the collected disturbances.

### B. Training and Test Results on Simulated Disturbances

Fig. 5 shows the change of cumulative reward over time during training. But the training reward is not sufficient to compare the performance among different algorithms, we are also interested in state distribution and bounded response (i.e. converged region) of the AUV disturbed by underwater waves and currents. As shown in Fig. 6, the box plot is used to represent and compare the distribution of the AUV's distance from the target position during last 100 steps of each episode among different algorithms.

It is clear that both the history window policies and the recurrent policies perform better than the classical RL policy (A2C) and the conventional controllers (RISE and DOBC), which means considering history information does improve the disturbance rejection capability. When only small disturbances occur, different approaches to use the history information have nearly the same results. When the disturbances become larger, the recurrent policies achieve better results compared with the history window policies,

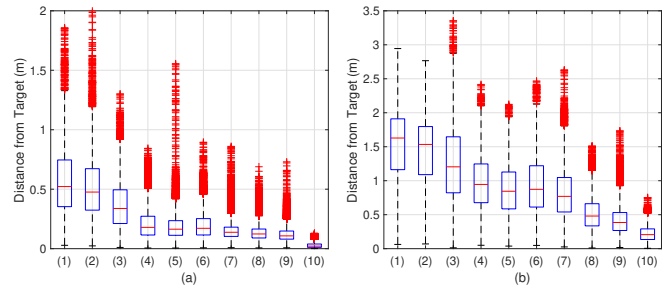


Fig. 6. Distribution of distance from target during last 100 steps, (a) small simulated disturbances (amplitude between 100-120% of control constraints); (b) large simulated disturbances (amplitude between 130-150% of control constraints).

proving that RNN is able to utilize the history information more efficiently than naively combining past states and actions into policy input space. Furthermore, including past actions as additional input besides states yields performance improvement for both HWA2C and RA2C. The DOB-Net achieves the best performance among all these algorithms, using larger embedding dimension ( $n = 64$ ) of disturbance estimate produces higher cumulative reward and smaller movement range. We believe enlarging the embedding dimension of disturbance estimate can provide better representation of disturbance dynamics. Transforming this embedding from a 64-dimensional variable to a 3-dimensional variable may cause loss of information. However, even using the best RL algorithms we mentioned so far, the control performance still has a large gap from the trajectory optimization solution. There is still room for further improvements.

In addition, it is obvious that stronger disturbances lead to worse performance. But we also found larger amplitude range of disturbances gives a more similar results between the DOB-Net and the trajectory optimization (the ratio of medians between the DOB-Net and the trajectory optimization is 502.31% and 186.65% respectively for small and large simulated disturbances). This phenomenon might result from that, for disturbances with larger amplitudes, the optimal controls for different wave profiles tend to be more similar. Thus, it is easier for RL to learn a control policy under larger disturbance amplitudes.

The 3D trajectories of the AUV subject to large simulated disturbances are compared among these algorithms in Fig. 7. The red ball region represents the AUV's maximum distance from the target during last 50 steps, called converged region. According to this region, we can see that the AUV is difficult to achieve satisfactory bounded response using either conventional controllers (RISE and DOBC) or classical RL policy (A2C). While the proposed DOB-Net can significantly reduce the converged region. Using the DOB-Net, the AUV can quickly navigate to the target and stabilize itself within a distance of 0.493m from the target thereafter. However, there is still an obvious difference between the DOB-Net results and the optimal trajectories under ideal conditions.

### C. Test Results on Collected Disturbances

Besides the simulated disturbances, we also use the collected wave and current disturbances (as shown in Fig. 4 (c))

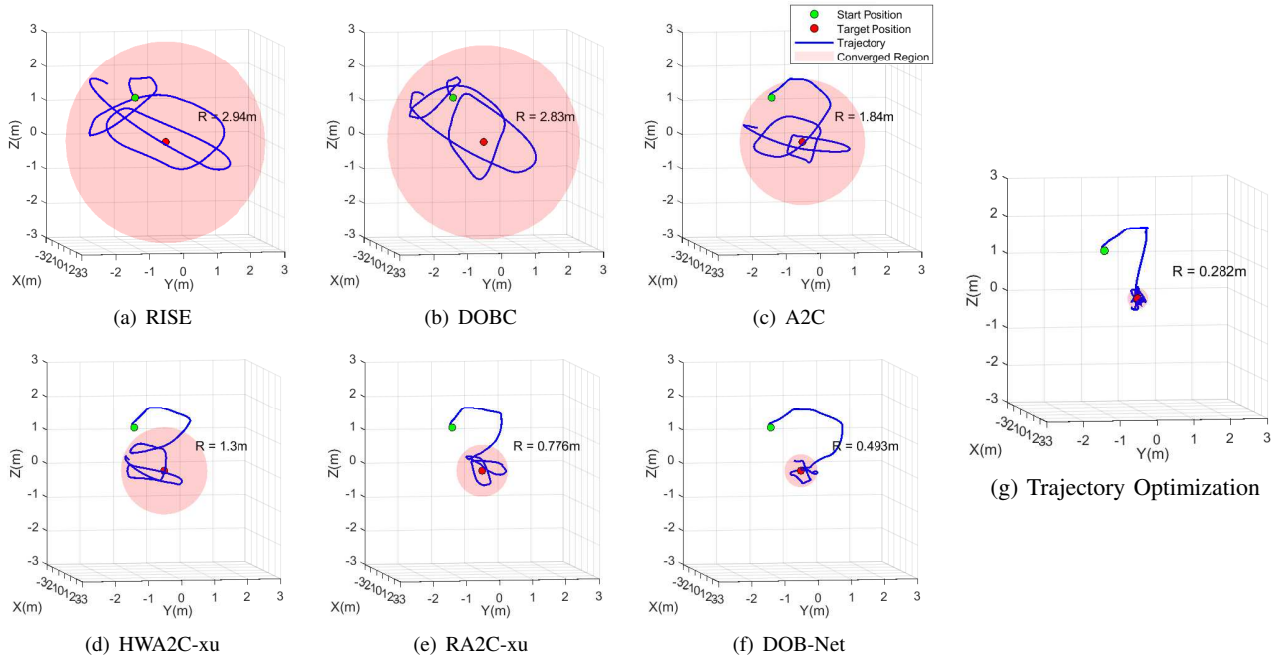


Fig. 7. 3D trajectories of the AUV with large simulated disturbances ( $R$  is the radius of the converged region). Note the trajectory optimization assumes that the disturbance dynamics is known in advance, thus provides the ideal performance.

for testing. Note the collected data is only used for testing, no retraining is required. As compared in Fig. 8 and Fig. 9, the DOB-Net still has satisfactory performance on real-world wave profiles and outperforms all the other algorithms, which proves the practical effectiveness of the DOB-Net. However, the converged region of all the algorithms become larger compared with using the simulated disturbances. The reason behind is that the collected disturbances are more diverse and complicated, thus have a wider range of amplitudes compared with using the simulated disturbances. The DOB-Net may not be capable of handling such unseen scenarios optimally. This gives rise to another research question, which is to deal with disturbances with a wide range of parameters based on training on small range of parameters. This may require the technique of transfer learning [30].

## VI. CONCLUSIONS & FUTURE WORK

This paper proposes an observer-integrated RL approach called DOB-Net, for mobile robot control problems under unknown excessive time-varying disturbances. A disturbance dynamics observer network employing RNNs has been used to imitate and enhance the function of conventional DOB, in order to produce the embedding of disturbance estimation

and prediction. A controller network is designed using the observer outputs as well as current state as inputs, to generate optimal controls. Multiple control and RL algorithms have been tested and compared on position regulation tasks using both simulated disturbances and collected disturbances, the results demonstrate that the proposed DOB-Net does have a significant improvement for the disturbance rejection capacity compared to existing algorithms.

Currently, the test disturbances are collected in a water tank using wave generator, we plan to seek for the disturbance data from open water environments with natural wave and current for further testing. Also, we have noticed that the performance of the DOB-Net is worse using the collected disturbances, due to its more complex and diverse dynamics. An interesting future work is to investigate the usage of transfer learning in dealing with real world wave and current disturbances. In addition, the deployment of this algorithm on real-world robotic systems requires future investigation, where the low sample efficiency of generic model-free RL might be a problem.

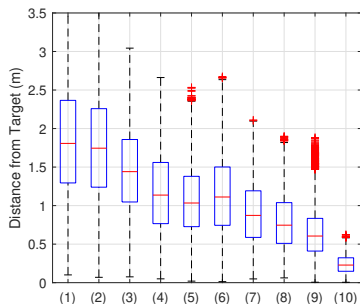


Fig. 8. Distribution of distance from target with collected disturbances.

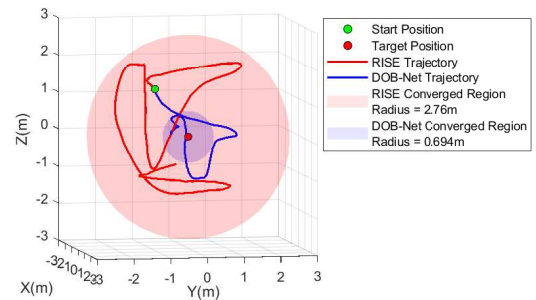


Fig. 9. 3D trajectories of the AUV with collected disturbances ( $R$  is the radius of the converged region).

## REFERENCES

- [1] G. Griffiths, *Technology and applications of autonomous underwater vehicles*. CRC Press, 2002, vol. 2.
- [2] T. Wang, W. Lu, and D. Liu, "A case study: Modeling of a passive flexible link on a floating platform for intervention tasks," in *2018 13th World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2018, pp. 187–193.
- [3] J. Woolfrey, D. Liu, and M. Carmichael, "Kinematic control of an autonomous underwater vehicle-manipulator system (auvms) using autoregressive prediction of vehicle motion and model predictive control," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4591–4596.
- [4] L.-L. Xie and L. Guo, "How much uncertainty can be dealt with by feedback?" *IEEE Transactions on Automatic Control*, vol. 45, no. 12, pp. 2203–2217, 2000.
- [5] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [6] W.-H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O'Reilly, "A nonlinear disturbance observer for robotic manipulators," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.
- [7] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-observer-based control and related methods—an overview," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1083–1095, 2016.
- [8] H. Sun and L. Guo, "Neural network-based dobc for a class of nonlinear systems with unmatched disturbances," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 2, pp. 482–489, 2016.
- [9] C. Edwards and S. Spurgeon, *Sliding mode control: theory and applications*. Crc Press, 1998.
- [10] W. Lu and D. Liu, "Active task design in adaptive control of redundant robotic systems," in *Australasian Conference on Robotics and Automation*. ARAA, 2017.
- [11] —, "A frequency-limited adaptive controller for underwater vehicle-manipulator systems under large wave disturbances," in *2018 13th World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2018, pp. 246–251.
- [12] H. Gao and Y. Cai, "Nonlinear disturbance observer-based model predictive control for a generic hypersonic vehicle," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 230, no. 1, pp. 3–12, 2016.
- [13] C. E. Garcia, D. M. Pretz, and M. Morari, "Model predictive control: theory and practice—a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.
- [14] U. Maeder and M. Morari, "Offset-free reference tracking with model predictive control," *Automatica*, vol. 46, no. 9, pp. 1469–1476, 2010.
- [15] S. Brahmabhatt and J. Hays, "Deepnav: Learning to navigate large cities," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3087–3096.
- [16] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.
- [17] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International Conference on Machine Learning*, 2015, pp. 1889–1897.
- [18] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, 2016, pp. 1928–1937.
- [19] S. Sæmundsson, K. Hofmann, and M. P. Deisenroth, "Meta reinforcement learning with latent variable gaussian processes," *arXiv preprint arXiv:1803.07551*, 2018.
- [20] L.-J. Lin and T. M. Mitchell, "Reinforcement learning with hidden states," *From animals to animats*, vol. 2, pp. 271–280, 1993.
- [21] T. Wang, W. Lu, and D. Liu, "Excessive disturbance rejection control of autonomous underwater vehicle using reinforcement learning," in *Australasian Conference on Robotics and Automation*, 2018.
- [22] M. Hausknecht and P. Stone, "Deep recurrent q-learning for partially observable mdps," in *2015 AAAI Fall Symposium Series*, 2015.
- [23] D. Wierstra, A. Foerster, J. Peters, and J. Schmidhuber, "Solving deep memory pomdps with recurrent policy gradients," in *International Conference on Artificial Neural Networks*. Springer, 2007, pp. 697–706.
- [24] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.
- [25] S. Waslander and C. Wang, "Wind disturbance estimation and rejection for quadrotor position control," in *AIAA Infotech@ Aerospace Conference and AIAA Unmanned... Unlimited Conference*, 2009, p. 1983.
- [26] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*. IEEE, 2018, pp. 7579–7586.
- [27] S. Li, J. Yang, W.-H. Chen, and X. Chen, *Disturbance observer-based control: methods and applications*. CRC press, 2016.
- [28] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
- [29] N. Fischer, D. Hughes, P. Walters, E. M. Schwartz, and W. E. Dixon, "Nonlinear rise-based control of an autonomous underwater vehicle," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 845–852, 2014.
- [30] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.