Journal of Korea Robotics Society (2024) 19(1):092-097
https://doi.org/10.7746/jkros.2024.19.1.092 ISSN: 1975-6291 / eISSN: 2287-3961

분포형 강화학습을 활용한 맵리스 네비게이션

Mapless Navigation with Distributional Reinforcement Learning

짠 반 마잉[1] · 김 곤 우†
Van Manh Tran[1], Gon-Woo Kim†

Abstract: This paper provides a study of distributional perspective on reinforcement learning for application in mobile robot navigation. Mapless navigation algorithms based on deep reinforcement learning are proven to promising performance and high applicability. The trial-and-error simulations in virtual environments are encouraged to implement autonomous navigation due to expensive real-life interactions. Nevertheless, applying the deep reinforcement learning model in real tasks is challenging due to dissimilar data collection between virtual simulation and the physical world, leading to high-risk manners and high collision rate. In this paper, we present distributional reinforcement learning architecture for mapless navigation of mobile robot that adapt the uncertainty of environmental change. The experimental results indicate the superior performance of distributional soft actor critic compared to conventional methods.

Keywords: Mapless Navigation, Reinforcement Learning, Distributional Soft Actor Critic, Deep Learning

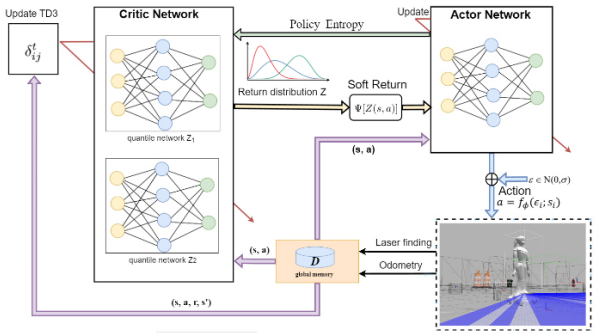
1. Introduction

Mapless navigation for mobile robots is one of the challenging problems for autonomous navigation tasks, where the robot finds collision-free paths in the unknown real-world. Besides, the conventional pipeline of map-based navigation including simultaneous localization and mapping (SLAM), and path planning algorithms is time-consuming, not feasible, and requires high computational costs for unstructured environments. To solve these issues, deep reinforcement learning (deep RL) models are considered as promising methods to navigate the robot in unknown scenarios without collision avoidance. Several studies [1,2] show the outstanding performance and high applicability, that deep learning models for mobile robots from virtual to real environments. Hence, applying the sim-to-real model to autonomous navigation tasks using RL algorithm in real-world scenarios should be thoroughly investigated and developed.

Recently, deep RL methods using actor critc architecture with entropy regularization have made significant strides in realm of robotics domains, including mapless navigation tasks[3]. Applying entropy regularization into actor critic networks has seen successes in continuous control tasks, which balance exploration and exploitation[4]. In another aspect of RL, the distributional RL has been developed successfully by considering the whole distribution of value function instead of the expected return in game environments[5].

A series of studies have been conducted to make agents get more insight and knowledge and also demonstrate the performance in the arcade game environments. By taking advantage of a distributional framework, Liu et al.[6] utilized the value of Conditional Value at Risk forecasting intrinsic uncertainty to perform drone navigation task. However, mapless navigation tasks are performed limited convincing results of real-time experiments using the distributional soft actor critic (DSAC)[7],

Received : Oct. 31. 2023; Revised : Nov. 24. 2023; Accepted : Nov. 24. 2023
※ This work was supported by the Technology Innovation Program (or Industrial Strategic Technology Development Program-ATC+) (20009546, Development of service robot core technology that can provide advanced service in real life) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea)
1. Graduate Student, Intelligent Systems and Robotics, Chungbuk National University, Cheongju, Korea (manhtran4321@gmail.com)
† Professor, Corresponding author: Intelligent Systems and Robotics, Chungbuk National University, Cheongju, Korea (gwkim@cbnu.ac.kr)



[Fig. 1] Mapless navigation framework based on DSAC

which take advantage of entropy term and distributional information of value function. The efficacy of DSAC for robot mapless navigation deserves more attention in real-world scenarios. Additionally, service mobile robots typically operate in chaotic real-world environments with human movement and unexpected obstacles, which differ from simulations and are prone to navigation errors leading to potential crashes.

The primary aim of this paper is to demonstrate and compare the effectiveness of deep RL networks in aspect of distributional RL compared with traditional RL for application of mobile robot navigation. Moreover, information randomness from complex environment is approximated by applying distributional value function to solve mapless navigation issues of mobile robot. Moreover, conditional value at risk (CvaR)[8] is applied to consider random uncertainty of the environments. The effectiveness of mapless navigation framework is proven on safety navigation task in simulation and real environments with limited field of view (FOV) of exteroceptive sensor (shown in [Fig. 1]).

The work is constructed as follow: the approach is presented in sention 2, which encompassed problem and our method. In section 3, the results related to simulation and real-world experiments are mentioned. Finally, the conclusion is summarized in section 4.

2. Mapless navigation framework

In this section, the problem formulation and proposed approach are described in detail.

2.1 Problem Formulation

Mapless sensor-level navigation is described as the policy trained in the simulation to drive a mobile robot in the real world. We formulate the autonomous navigation problem by using a Partial Markov Decision Process (POMDP), which consists of a tuple (S, A, P, R, O), where S presents state space $(s_t \in S)$, A is action space $(a_t \in A)$, P presents actions space, R present reward function, O is sensor observation.

State space: The state of the agent $s_t = [L_t, d_t, \alpha]$, where L_t is normalized laser readings, d_t is the displacement of the robot in polar coordinates, α is the deviation angle between the robot and the goal direction.

Action is two-dimensional vector, which includes linear velocity v and angular velocity w of that robot mapped with the input of model $a_t = (v, \mathrm{w})$.

Reward function: Since the optimal policy is affected intensively by reward signals, distinct behaviors of the agent are reshaped as the reward function. Based on the previous work[3], we restrengthen the reward function of the policy for collision-free navigation tasks, comprise of $r_g, \mathrm{r}_s, r_{rel}$ and r_{col}:

$$R(s_t) = r_g + \mathrm{r}_s + r_{rel} + r_{col} \tag{1}$$

P^t and P^{t-1} are the robot at the current and position and previous timestamp. r_g is reward of reaching the goal. The relative reward r_{rel} and safety reward r_s are defined as:

$$r_{rel} = c \parallel p^{t-1} - g \parallel - \parallel p^t - g \parallel + \frac{\pi - \parallel \alpha \parallel}{\pi} + \cos(\alpha) \tag{2}$$

$$\mathrm{r}_s = -k\left(1 - \frac{d_t}{2r_0}\right) \tag{3}$$

Where c, k are coefficients, r_0 is robot radius. The resparse reward when reaching to the target is $r_g = 500$ and $r_{col} = -600$ is given if the collisions happen.

2.2 Distribution Soft Actor Critic

The fundamental concept of Distributional SAC framework is to use random information to estimate value function within continuous action space. The advantage of DSAC is to take advantage of SAC while keep exploration based on random domain of value function. Existing distributional RL[5] algorithm rely on distributional Bellman equation. The value distribution can be defined as:

$$Z^{\pi}(s,a) \overset{D}{=} r(s,a) + \gamma Z^{\pi}(S',A') \tag{4}$$

Where $A \overset{D}{=} B$ denotes equal probability laws between those two random variables A and B, $\gamma \in [0,1)$ is discount factor, $Z^{\pi}(S',A')$ is random return given the next state-action (S',A'). $Z^{\pi}(s,a)$ is random distribution return and $r(s,a)$ is random reward. With maximum entropy RL of SAC[2], the soft action-value function of a DSAC policy is:

$$Z^{\pi}(s,a) := \sum_{t=0}^{\infty} \gamma^{t} [r(S_t,A_t) - \alpha \log \pi(A_t|S_t)] \tag{5}$$

Inheriting from the Actor-Critic network, the critic utilizes quantile fraction $\tau_1, \ldots \tau_N$ and $\tau_1^{'}, \ldots \tau_N^{'}$, $N' = 32$ is the number of quantile sampled separately, the loss of the critic:

$$L(s_t,a_t,r_t,s_{t+1}) = \frac{1}{N'} \sum_{i=1}^{N} \sum_{j=1}^{N'} \rho_{\tau_i}(\delta_t^{\tau_i,\tau_j'}) \tag{6}$$

The quantile regress loss and the temporal difference are defined as equation (7), (8) respectively.

$$\rho_{\tau}(x) = |\tau - \mathbf{1}_{x<0}| \min\{x^2, 2|x|-1\}/2 \tag{7}$$

$$\delta_t^{\tau_i,\tau_j'} = r_t + \gamma [Z_{\tau'}^{'}(s_{t+1},a_{t+1}) - \alpha \log \pi(a_{t+1}|s_{t+1})] - Z_{\tau}(s_t,a_t) \tag{8}$$

Where (s_t,a_t,r_t,s_{t+1}) is a transition of buffer. $\hat{Z}_{\tau}(s_t,a_t)$ is critic output, which estimate of the τ - quantile of distributional value function. The objective of policy is formulated as

$$J(\pi) = E_{s_t \sim D, \varepsilon_t \sim N}[Q(s_t, f(s_t,\varepsilon)) - \alpha \log \pi(f(s_t,\varepsilon|s_t))] \tag{9}$$

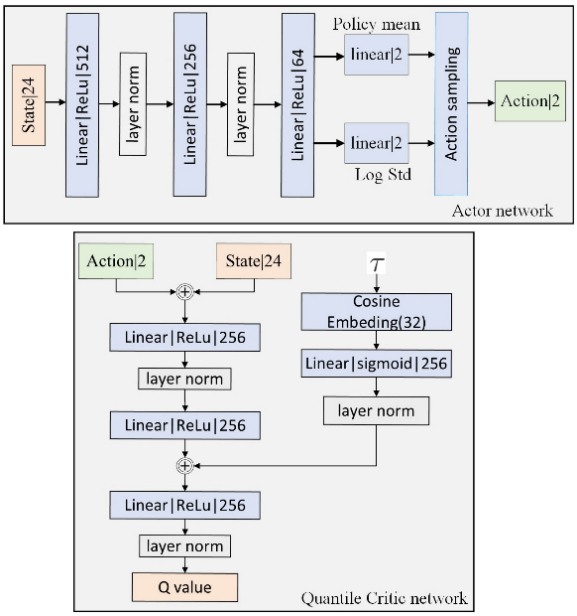
In order measure risk sensitive decision, CVaR[8] is applied to approximate quantile value under uncertainty with actions and rewards.

$$\phi^{CVaR}(\tau,\beta) := \beta\tau \tag{10}$$

Where $\beta \in (0,1)$ is the risk distortion coefficient, $\beta = 1$ give a risk-neutral policy.

2.3 Network architecture

The robot states are generated from laser readings from LiDAR and combined with relative location from odometry data. Based on[4], the architecture of the network comprises of 108 inputs of Actor networks including 105 laser scans from the YDLIDAR G6 sensor, deviation angles, and minimum range distance. After using a re-parameterized trick, the Actor network produces distribution of the bound angular and linear velocity. The Actor neural network is constructed by decreasing the network size over each layer, leading to optimize computation and keep the simple network and rich presentations The Critic network is present in [Fig. 2], which is incooperated with quantile[8].

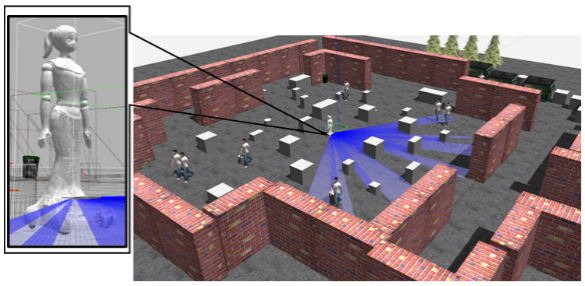


[Fig. 2] The structure of actor and critic network

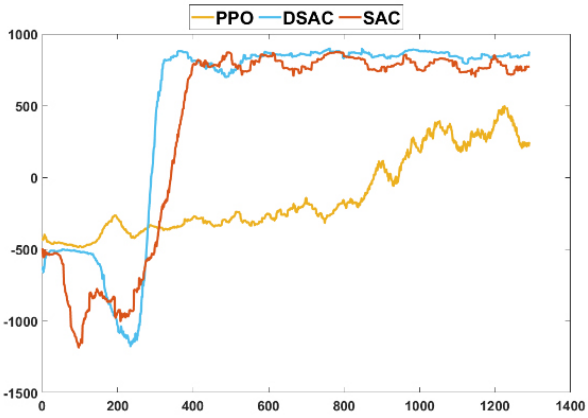
3. Experimental result

3.1 Simultion result

Both physical simulation and real-world environments are particularly utilized for training and testing Deep RL model in term of accelerating simulation time (shown in [Fig. 3]). The whole training process is conducted on a Gazebo Simulation with a 10Hz control frequency. The Simulation will reset (after 0.5s) when the robot's collision happens in the virtual environment The virtual environment is created by designing an environment with natural elements that closely resemble those found in the real world. Various obstacles of different sizes ensure the diversity and complexity of the test environment. To



[Fig. 3] The Gazebo simulation during training



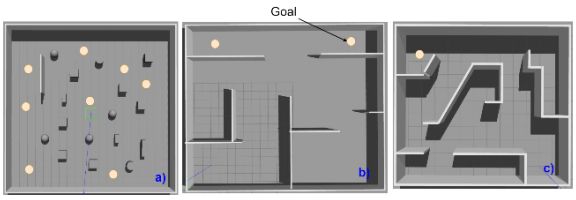
[Fig. 4] Training curve of SAC, PPO, DSAC algorithms according to hyper parameters in Table 1

[Table 1] Hyper parameter for training DSAC algorithm

Angular velocity (rad/s)	[-1.0,1.0]
Linear velocity (m/s)	[0.0,1.0]
Maximum steps per episode	1200
Buffer size	3.10^{6}
Batch size	3.10^{-4}
Learning rate	3.10^{-4}
Discount factor	0.99
Optimizer	Adam
Number of quantiles	32
Quantile fraction embedding size	64

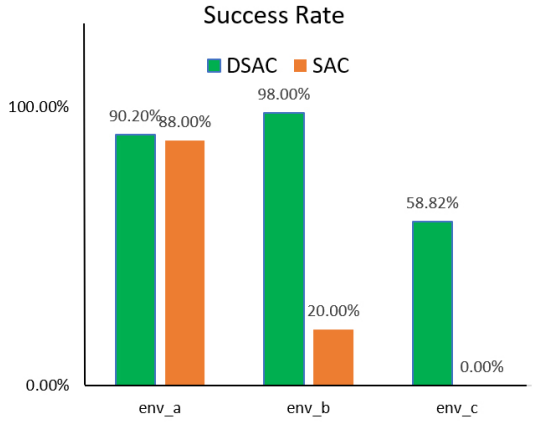
generalize the algorithm's performance, we trained the deep RL algorithm with the framework over service robot model for approximately 1200 episodes. With the acceleration time in simulation. The entire training process trained by DSAC tooks nearly 5 hours, significantly reducing the time[9].

To evaluate mapless navigation algorithms, the learning curve over training progress is depicted in [Fig. 4]. The maples navigation algorithm based on DSAC algorithm is compared with two state of the art algorithm SAC[4] and PPO[10]. It can be seen

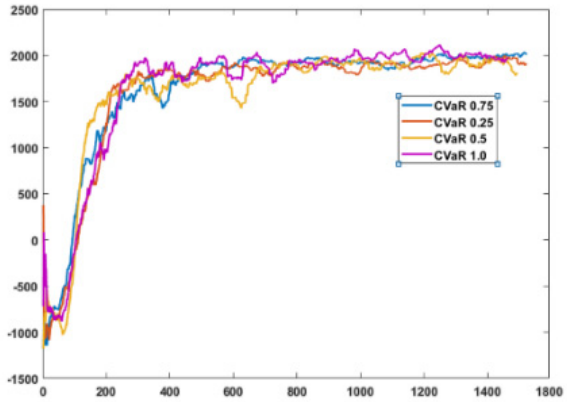


(a) (b) (c)

[Fig. 5] Testing in simulation environment including environment (a), environment (b), environment (c)

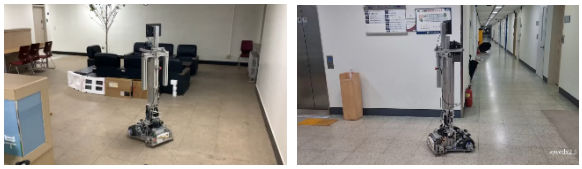


[Fig. 6] The success rate under test evaluation in the simulation



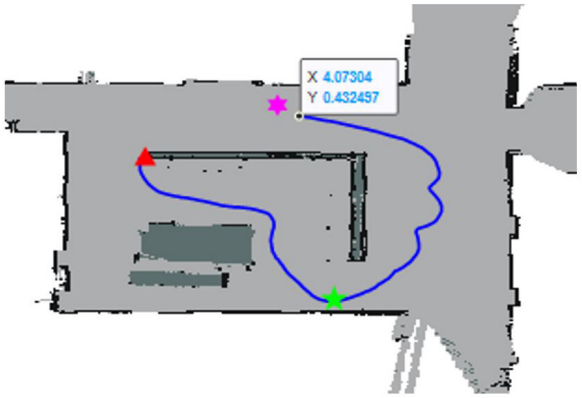
[Fig. 7] Comparison of risk averse policy with CVaR

that the speed of learning curve using DSAC get faster and also more stabilize than SAC and PPO. [Fig. 4] illustrates the highest average score of Distributional SAC among other SOTA algorithms. The hyper parameters for training and testing in the simulation are shown in [Table 1]. The deep RL model will find arbitrary goals which is set randomly in the virtual environment avoiding different shapes of object (shown in [Fig. 5]). The [Fig. 6] shows the success rate in three virtual environments environments. The risk-sensitive policy is changed slightly depending on confidence level β, the risk neutral policy performs better than others shown in [Fig. 7].



(a) (b)

[Fig. 8] Sim-to-real transfer in real-world scenario: (a) clutter environment, (b) corridor environment



(a)

(b)

[Fig. 9] Trajectory of the robot in real-world scenarios: (a) clutter environment, (b) corridor environment

3.2 Real-world experiment result

The mapless navigation using DSAC is performed in two real world scenarios. First, we evaluate the DSAC algorithm in the clutter environment illustrated in [Fig. 8], with two consecutive target positions. Then, corridor environment with human motions is performed to evaluate the collision avoidance ability of the DSAC algorithm. The YDLIDAR is equipped on the service robot to sensing the environment from -60° to 60°. The on-board computer was a mini-PC with CPU AMD Ryzen™ 7 5700U Processor. The trajectory result of clutter environment and corridor environment are shown in [Fig. 9]. The map information is installed to only assist localization visually in the real-world scenarios' evaluation[11]. The experiment video can be shown in https://youtu.be/CxFutv_RlkU.

4. Conclusion

This paper bridges the gap between simulation and the real world to demonstrate the efficiency of a deep RL model trained using the distributional SAC algorithm. In this work, we utilized low-cost LiDAR, which is susceptible to inaccurate sensing due to limited light in outdoor environments. This leads to the loss of the model's state input and imprecise localization. We consider this issue as a subject for future research.

References

[1] J. Jin, N. M. Nguyen, N. Sakib, D. Graves, H. Yao, and M. Jagersand, "Mapless navigation among dynamics with social-safety-awareness: a reinforcement learning approach from 2d laser scans," *2020 IEEE international conference on robotics and automation (ICRA)*, Paris, France, pp. 6979-6985, 2020, DOI: 10.1109/ICRA40945.2020.9197148.

[2] T. Fan, X. Cheng, J. Pan, D. Manocha, and R. Yang, "Crowd-dmove: Autonomous mapless navigation in crowded scenarios," *ArXiv*, Jul., 2018, [Online]. https://api.semanticscholar.org/CorpusID:49904993.

[3] L. Tai, G. Paolo, and M. Liu, "Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, Canada, pp. 31-36, 2017, DOI: 10.1109/IROS.2017.8202134.

[4] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *the 35th International Conference on Machine Learning*, pp. 1861-1870, 2018, [Online], https://proceedings.mlr.press/v80/haarnoja18b.html.

[5] W. Dabney, M. Rowland, M. Bellemare, and R. Munos, "Distributional reinforcement learning with quantile regression," *AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr., 2018, DOI: 10.1609/aaai.v32i1.11791.

[6] C. Liu, E.-J. van Kampen, and G. C. H. E. de Croon, "Adaptive Risk-Tendency: Nano Drone Navigation in Cluttered Environments with Distributional Reinforcement Learning," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, United Kingdom, pp. 7198-7204, 2023, DOI: 10.1109/ICRA48891.2023.1016032 4.

[7] J. Duan, Y. Guan, S. E. Li, Y. Ren, Q. Sun, and B. Cheng, "Distributional Soft Actor-Critic: Off-Policy Reinforcement Learning for Addressing Value Estimation Errors," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6584-6598, Nov., 2022, DOI: 10.1109/TNNLS.2021.3082568.

[8] W. Dabney, G. Ostrovski, D. Silver, and R. Munos, "Implicit quantile networks for distributional reinforcement learning," *the 35th International Conference on Machine Learning*, pp. 1096-1105, 2018, [Online], https://proceedings.mlr.press/v80/dabney18a.html.

[9] W. Zhu and M. Hayashibe, "A Hierarchical Deep Reinforcement Learning Framework With High Efficiency and Generalization for Fast and Safe Navigation," *in IEEE Transactions on Industrial Electronics*, vol. 70, no. 5, pp. 4962-4971, May, 2023, DOI: 10.1109/TIE.2022.3190850.

[10] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv:1707.06347*, 2017, DOI: 10.48550/arXiv.1707.06347.

[11] M. Labbé and F. Michaud, "Appearance-Based Loop Closure Detection for Online Large-Scale and Long-Term Operation," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 734-745, Jun., 2013, DOI: 10.1109/TRO.2013.2242375.

Van Manh Tran

2015 Hanoi University of Science and Technology

2020 VietNam-Korea Institute of Science and Technology (researcher)

2022~Present Department of Control and Robot Engineering, Chungbuk National University, Korea (Master degree)

Interests: Reinforcement learning, autonomous navigation, Sensor Fusion, path planning

Gon-Woo Kim

2008 Assistant Professor, Electronics and Control Engineering, Wonkwang University

2012 Assistant Professor, School of Electronics Engineering, Chungbuk National University

2014 Associate Professor, School of Electronics Engineering, Chungbuk National University

2021~Present Professor, Department of Intelligent Systems and Robotics, Chunbuk National University

Interests: Navigation, Localization, SLAM