中国科学技术大学

USTC Robotics Lab

Robot Navigation with Map-Based Deep Reinforcement Learning

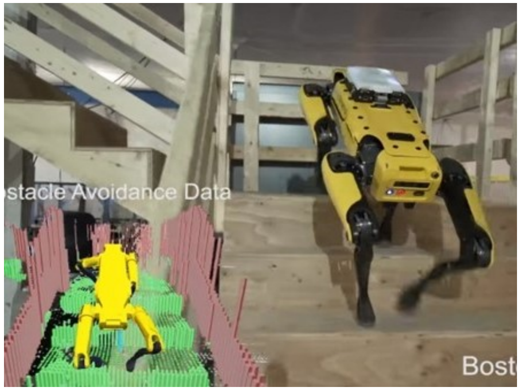
Guangda Chen, Lifan Pan, Yu'an Chen, Pei Xu, Zhiqiang Wang, Peichen Wu, Jianmin Ji* and Xiaoping Chen

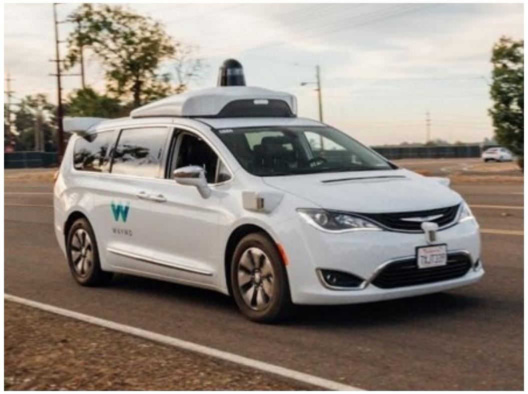
School of Computer Science and Technology,
University of Science and Technology of China

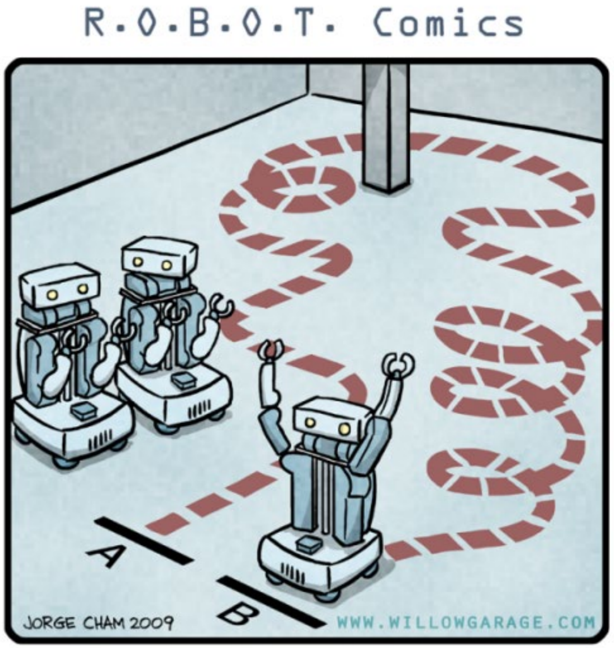
https://cgdsss.github.io/
cgdsss@mail.ustc.edu.cn

Robot Navigation

- Navigation is the basic ability of mobile robots.
- Navigation is widely used in all kinds of mobile robots, unmanned driving and drones.



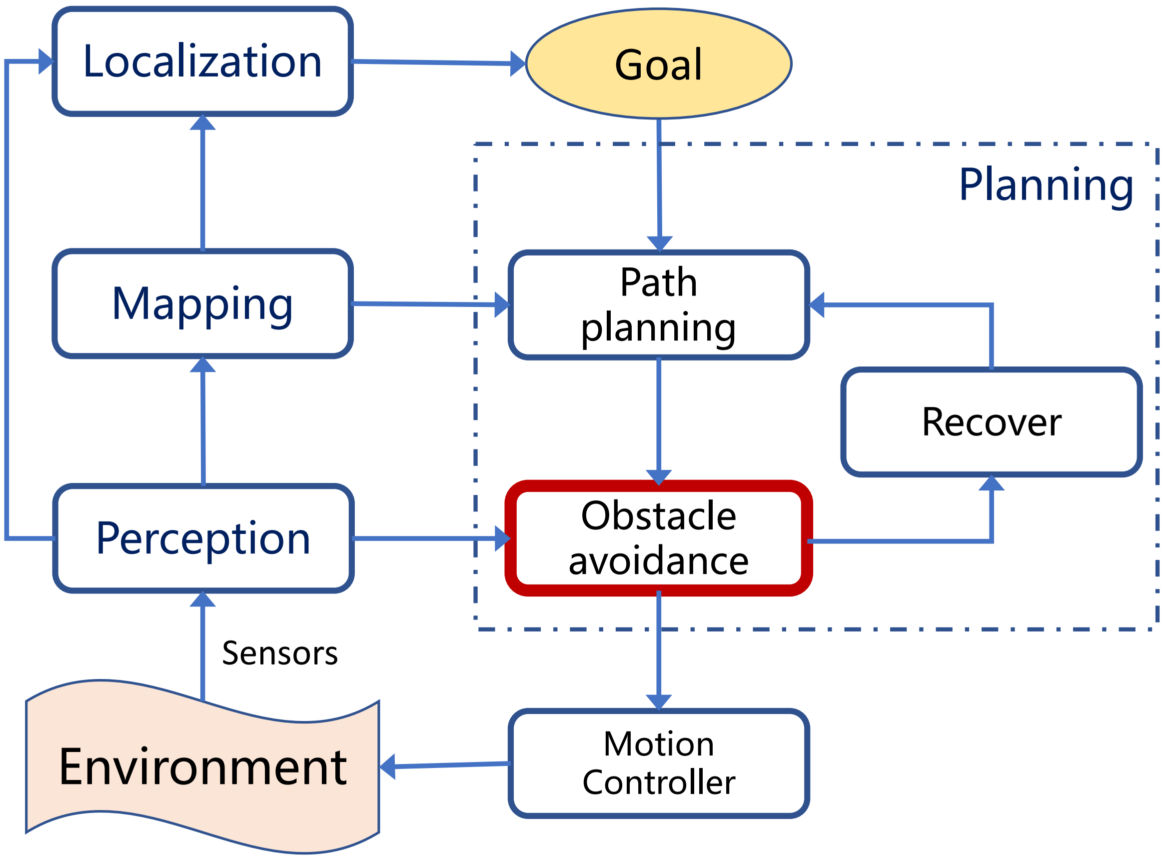




"HIS PATH-PLANNING MAY BE SUB-OPTIMAL, BUT IT'S GOT FLAIR."

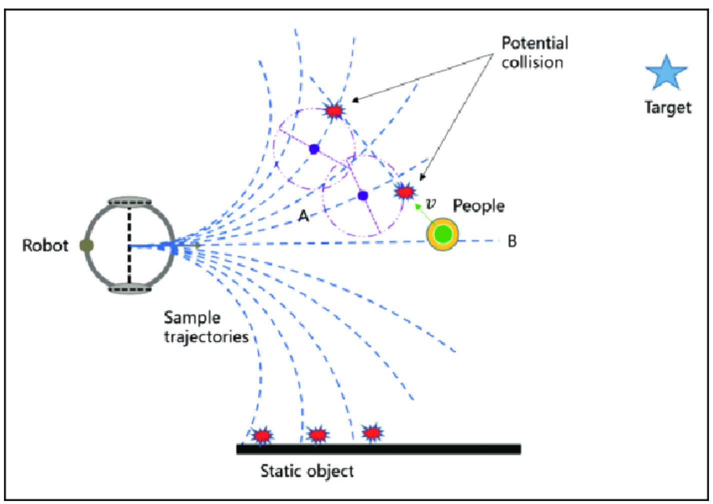
http://wiki.ros.org/navigation

Robot Navigation

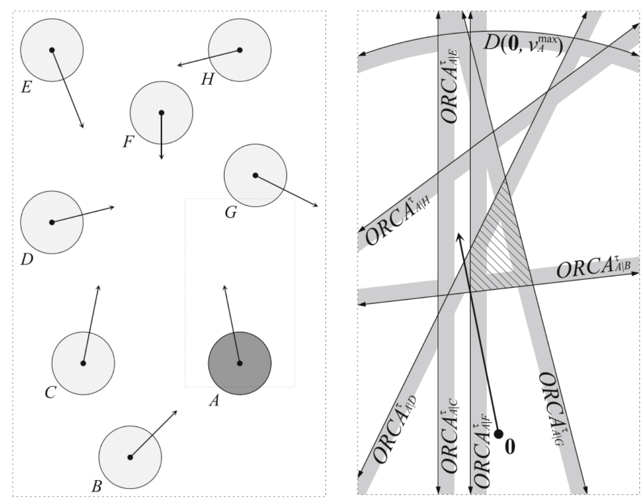


Traditional Collision Avoidance

- based on some assumptions that are not to be satisfied in practice
- may require a lot of computational cost
- many parameters that need to be tuned manually
- cannot learn from past experience automatically
- difficult to generalize well to unanticipated scenarios.



(a) DWA



(b) ORCA

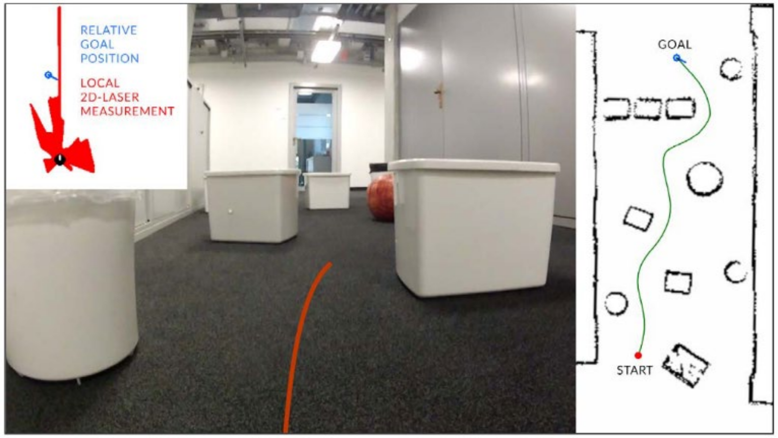
Supervised learning based OA

- ◆ require a massive manually labeled dataset
- ◆ the performance of learned models is largely limited by the strategy of generating training labels

(a) *RAL-15*

(a) Giusti, Alessandro, et al. "A machine learning approach to visual perception of forest trails for mobile robots." *IEEE Robotics and Automation Letters* 1.2 (2015): 661-667.

(b) Pfeiffer, Mark, et al. "From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots." ICRA-17.



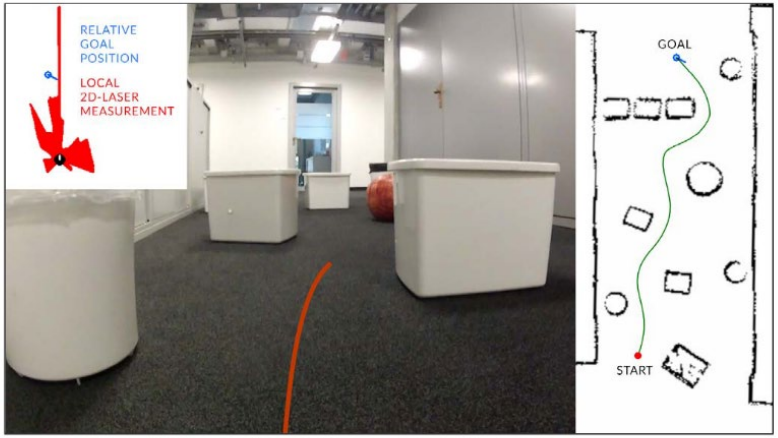
(b) *ICRA-17*

Supervised learning based OA

- require a massive manually labeled dataset
- the performance of learned models is largely limited by the strategy of generating training labels

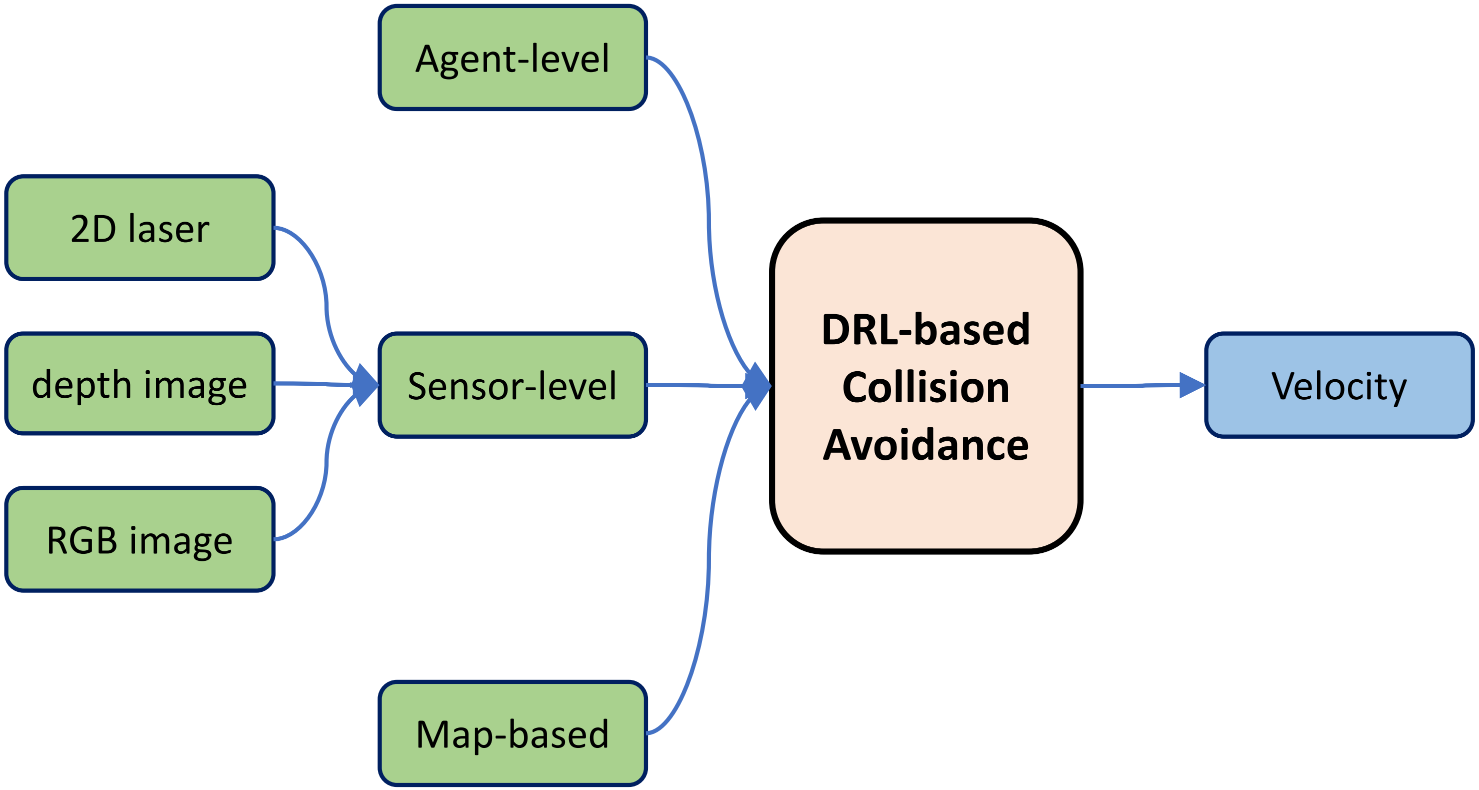
However, deep reinforcement learning (DRL) based methods learn from a large number of trials and corresponding feedback (rewards), rather than from labeled data.

(a) ***RAL-15***



(b) ***ICRA-17***

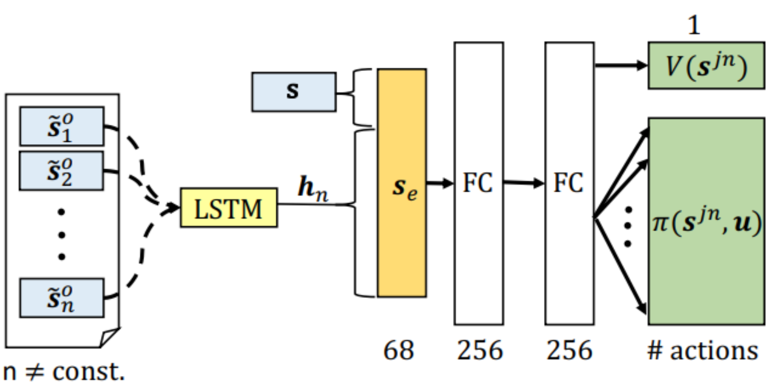
DRL-based Collision Avoidance



DRL-based Collision Avoidance

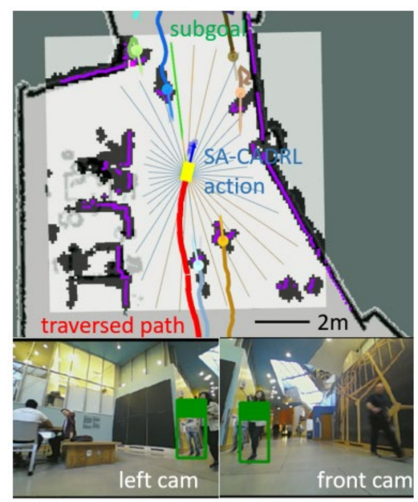
Agent-level

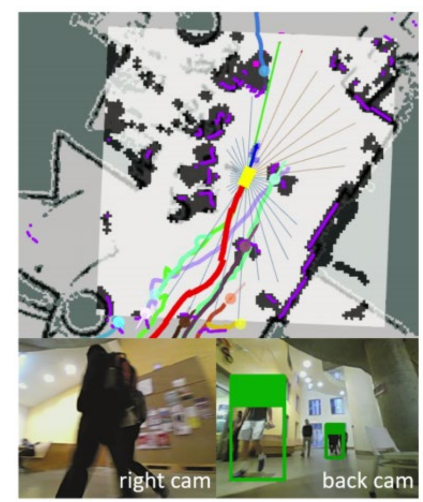
[1] Chen, Yu Fan, et al. "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning." *ICRA 2017*. IEEE.
[2] Chen, Yu Fan, et al. "Socially aware motion planning with deep reinforcement learning." IROS 2017.
[3] Everett, Michael, Yu Fan Chen, and Jonathan P. How. "Collision Avoidance in Pedestrian-Rich Environments with Deep Reinforcement Learning." arXiv preprint arXiv:1910.11689 (2019).



$$\mathbf{s} = [d_g,\ v_{pref},\ \psi,\ r]$$

$$\tilde{\mathbf{s}}^o = [\tilde{p}_x,\ \tilde{p}_y,\ \tilde{v}_x,\ \tilde{v}_y,\ \tilde{r},\ \tilde{d}_a,\ \tilde{r}+r]$$

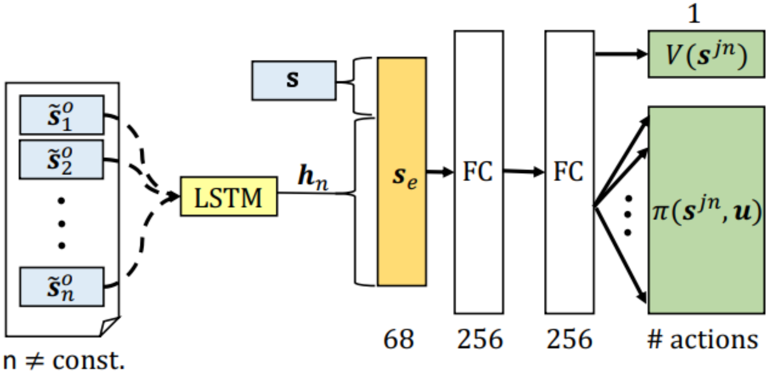




DRL-based Collision Avoidance

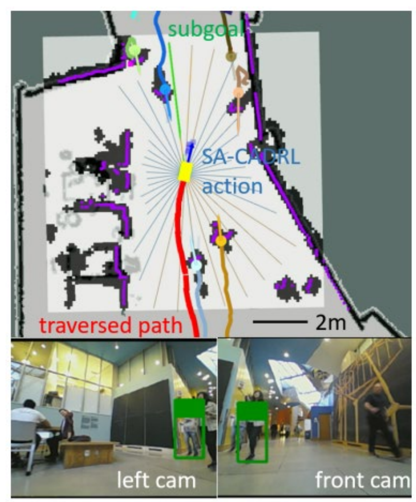
Agent-level

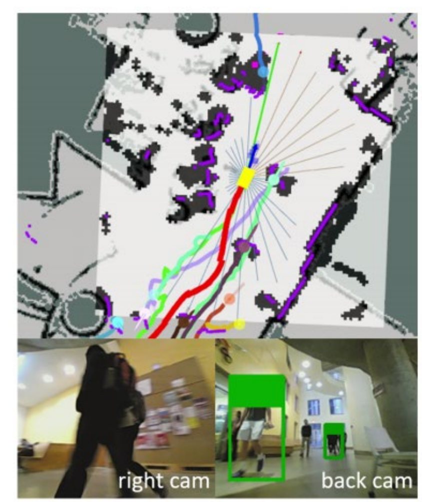
- requiring precise and complex front-end perception processing modules
- sensor type independence
- can be adapted to different front-end perception modules
- easy to design training simulation environment
- easy transfer to the real environment



$$\mathbf{s} = [d_g,\ v_{pref},\ \psi,\ r]$$

$$\tilde{\mathbf{s}}^o = [\tilde{p}_x,\ \tilde{p}_y,\ \tilde{v}_x,\ \tilde{v}_y,\ \tilde{r},\ \tilde{d}_a,\ \tilde{r}+r]$$

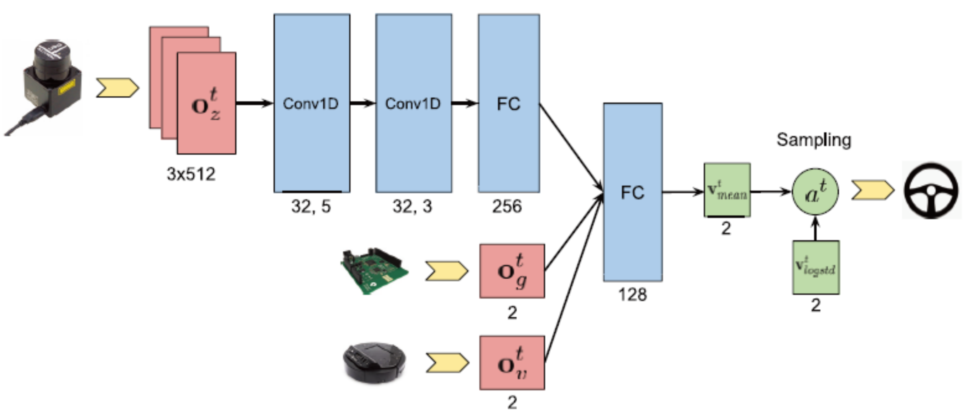


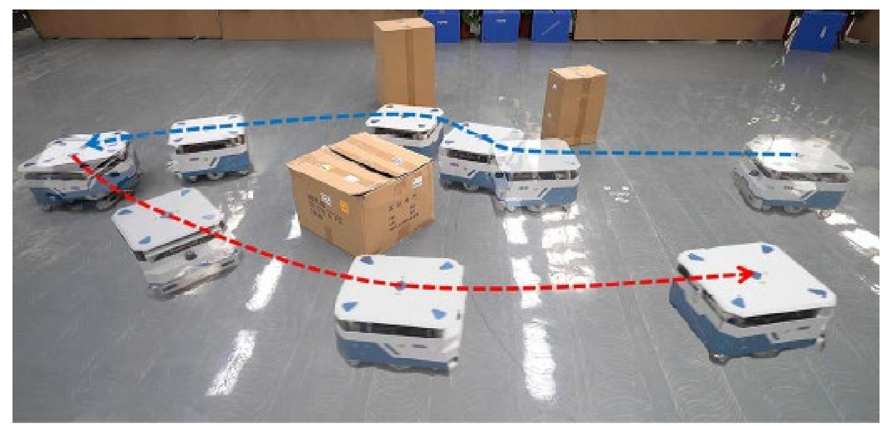


DRL-based Collision Avoidance

Sensor-level

[1] Long, Pinxin, et al. "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning." ICRA 2018. IEEE.
[2] Fan, Tingxiang, et al. "Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios." The International Journal of Robotics Research (2020).

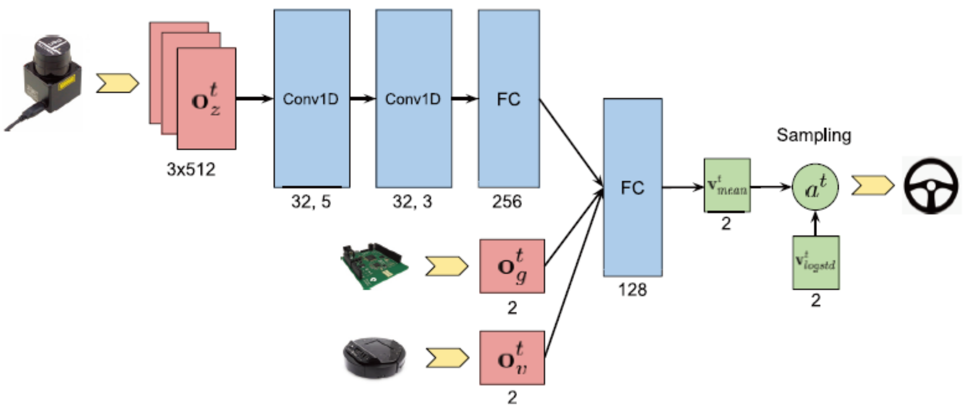


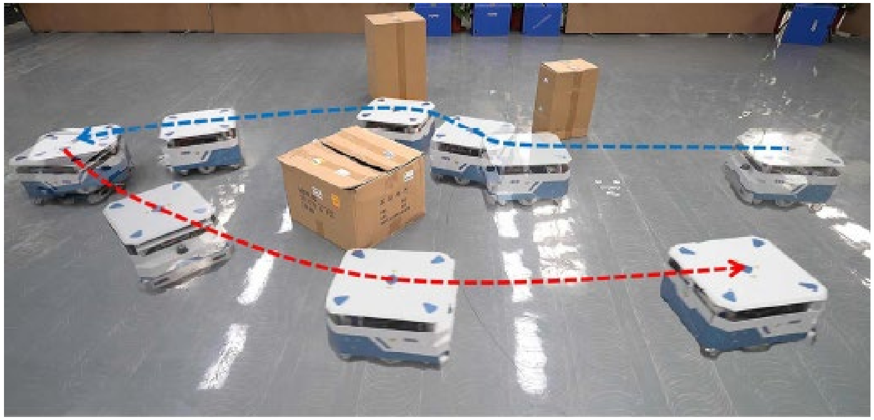


DRL-based Collision Avoidance

Sensor-level

- do not require a complex and time-consuming front-end perception module
- only restricted to specific sensors





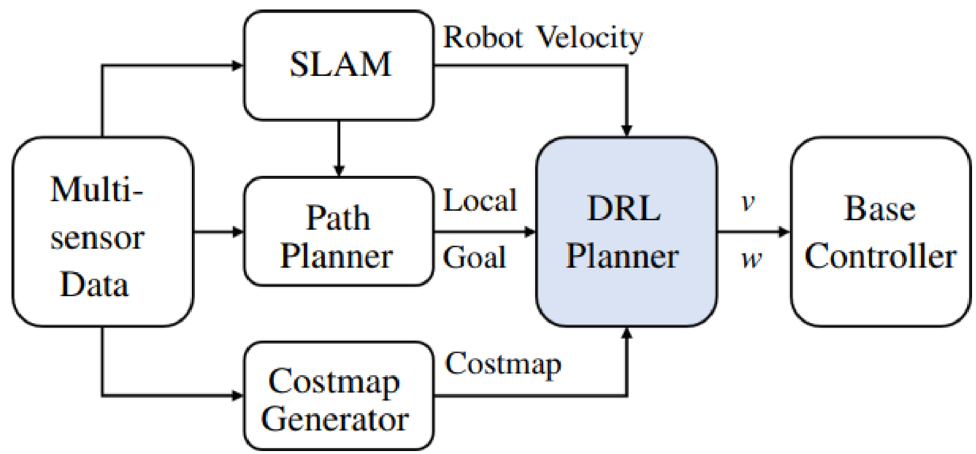
Robot Navigation with Map-Based DRL

- use the egocentric local grid map of a robot to represent the environmental information around it
- has the advantages of both agent-level and sensor-level methods
- adaptable to various sensors, easy to be trained in simulation environments, more robust to noisy sensor data, does not require robots' movement data and a precise and complex front-end perception procedure

$$\mathbf{M}_t = f_\lambda(s_t)$$

$$a_t = \pi_\theta(\mathbf{M}_t, g_t, v_t, \omega_t)$$

$$\underset{\pi_\theta}{\operatorname{argmin}} \mathbb{E}[t_g | a_t = \pi_\theta(\mathbf{o}_t)],$$



$$\mathbf{p}_t = \mathbf{p}_{t-1} + a_t \cdot \Delta t,$$
$$\forall k \in [1, N] : \|\mathbf{p}_t - (\mathbf{p}_{obs})_k\| > R]$$

Robot Navigation with Map-Based DRL

MDP: $M=(S,A,P,R,\gamma)$

The quality of policy $\pi(a|s)$ can be evaluated by Q-value :

$$Q^{\pi}(s,a)=\mathbb{E}^{\pi}[\sum_{t=0}^{\infty}\gamma^{t}R(s_t,a_t)|s_0=s,a_0=a]$$

Q-learning algorithm

$$Q^{*}(s_t,a_t)=R(s_t,a_t)+\gamma\max_{a_{t+1}}Q(s_{t+1},a_{t+1})$$

DQN loss function: $(y_t-Q(s_t,a_t;\theta'))^2$

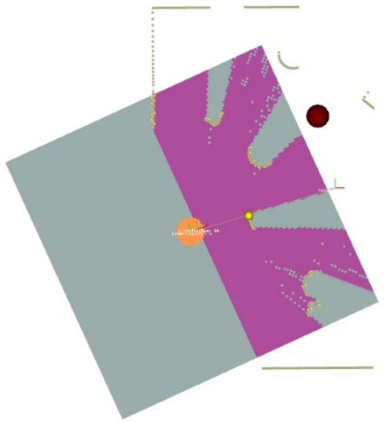
$$y_t=\begin{cases} r_t & \text{if episode ends} \\ r_t+\gamma\max_{a_{t+1}}Q(s_{t+1},a_{t+1};\theta') & \text{otherwise} \end{cases}$$

Double DQN

$$y_t=r_t+\gamma Q(s_{t+1},\operatorname*{argmax}_{a_{t+1}}Q(s_{t+1},a_{t+1};\theta);\theta')$$

Robot Navigation with Map-Based DRL

- Observation
  - grid maps
  - relative goal position
  - current velocity
- Action
  - angular velocity and linear velocity
  - a set of allowable velocity in discrete space
- Reward



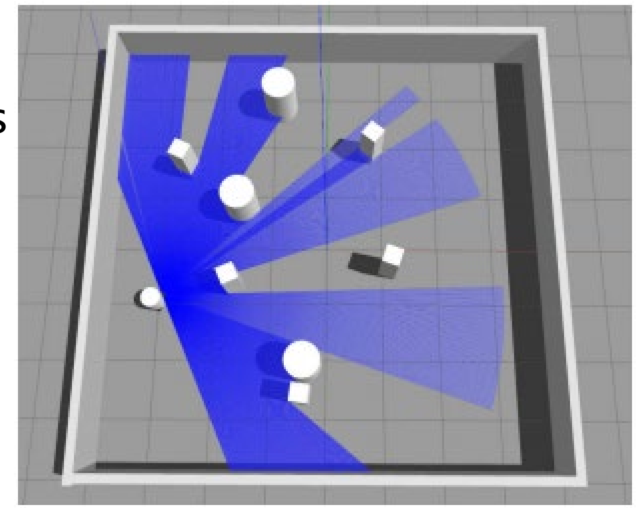
$$r_t = (r^g)_t + (r^c)_t + (r^s)_t$$

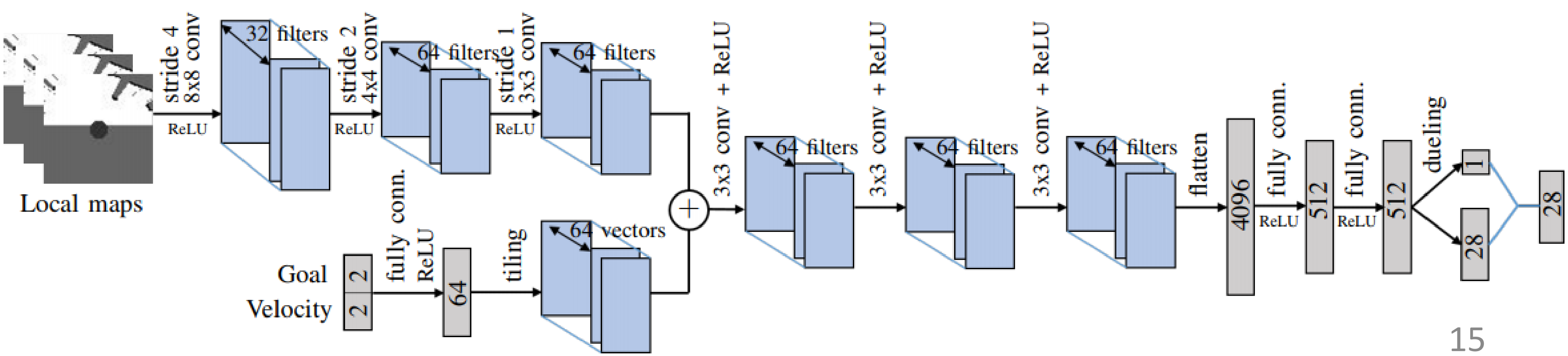
Reward Shaping:

$$(r^g)_t = \begin{cases} r_{arr} & \text{if } \|\mathbf{p}_t - \mathbf{g}\| < 0.2 \\ \varepsilon(\|\mathbf{p}_{t-1} - \mathbf{g}\| - \|\mathbf{p}_t - \mathbf{g}\|) & \text{otherwise} \end{cases}$$

Robot Navigation with Map-Based DRL

- Environment
  - Gazebo simulator
  - gradually increase the number of obstacles
  - the distance from the starting point to the target point gradually increases
- Training algorithm
  - Dueling DDQN
  - Prioritized experience reply
  - Curriculum Learning
- Network
  - a CNN-based deep convolutional neural network



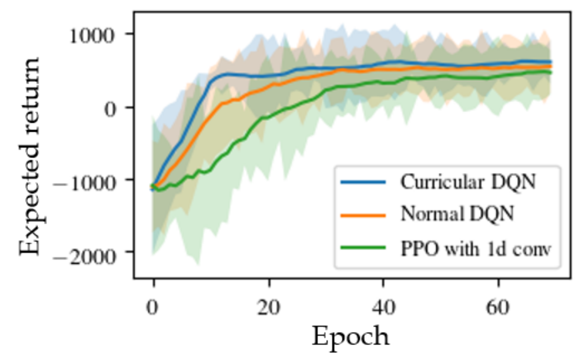


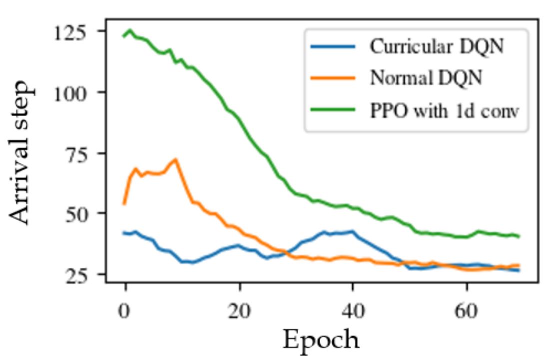
Experiments -- simulation scenarios

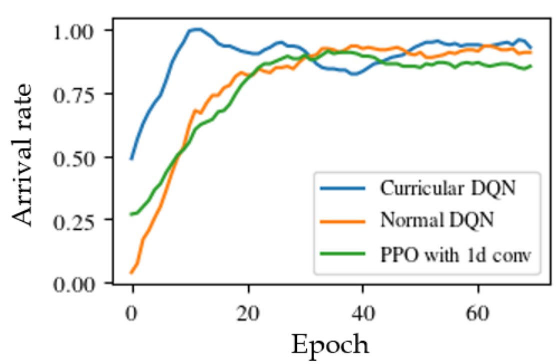
- **Expected return** E_r is the average of the sum of rewards of episodes.
- **Arrival rate** $\bar{\pi}$ is the ratio of the episodes in which the robot reaching the goals within a certain step without any collisions over the total episodes.
- **Arrival step** $\bar{s}$ is the average number of steps required to successfully reach the target point without any collisions
- **Average angular velocity change** $\nabla\omega$ is the average of the angular velocity changes for each step, which reflects the smoothness of the trajectory.

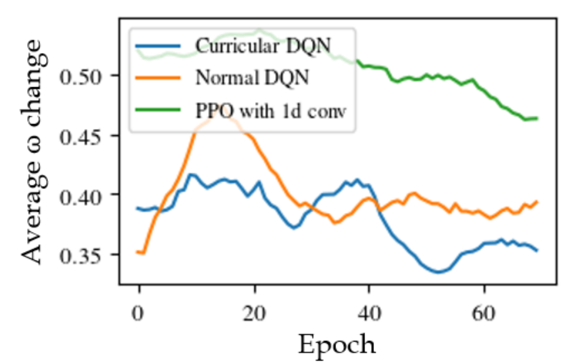
Comparative Policy：

- curricular DQN policy
- normal (non-curricular) DQN policy
- PPO with one-dimensional convolutional network







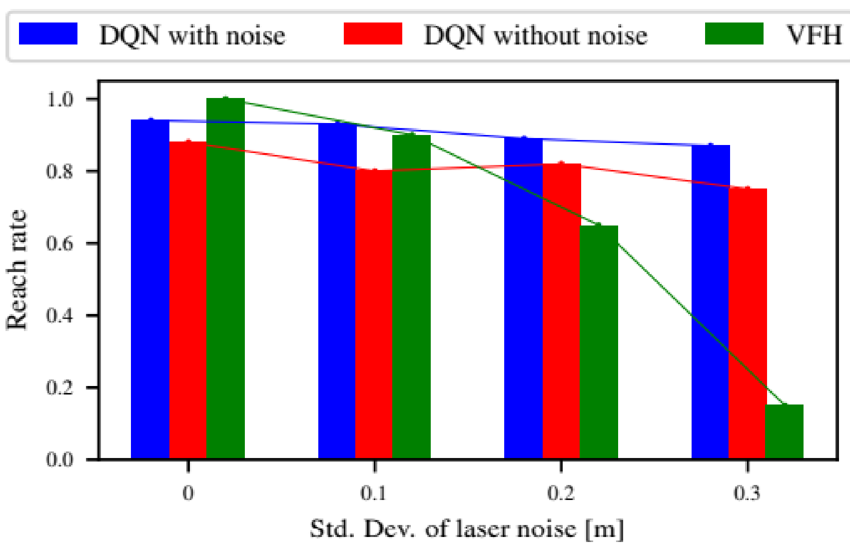


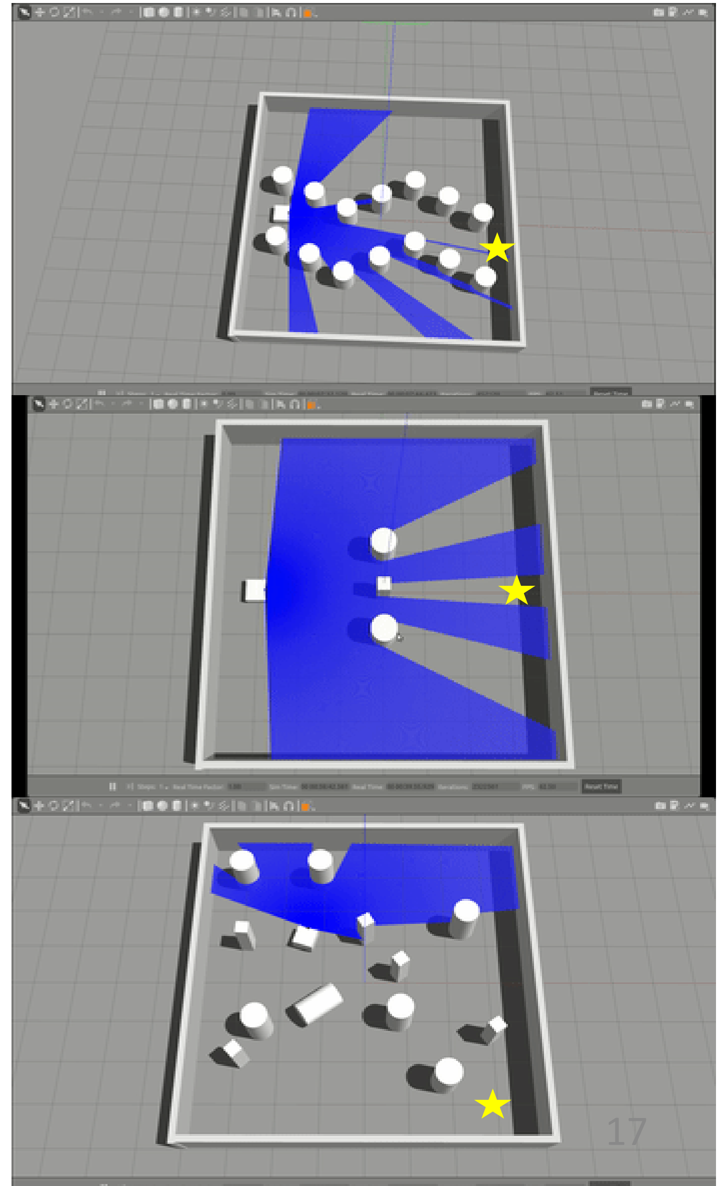
Experiments -- simulation scenarios

◆ Comparative experiments

Method	E_r	$\bar{\pi}$	$\bar{s}$	$\nabla\omega$
PPO with 1d conv	467.87	0.85	40.19	0.46
Normal DQN	547.43	0.91	27.76	0.39
Curricular DQN	**617.04**	**0.94**	**26.13**	**0.35**

◆ Robustness to noise





Experiments -- real-world

https://youtu.be/Eq4AjsFH_cU

- Differential wheel robot
- Hokuyo UTM-30LX laser
- i7-8750H CPU, NVIDIA 1060 GPU
- 6.0×6.0m and resolution 0.1m local grid map

- Complex static environment built by cartons
- Dynamic pedestrian environment
- Long-distance open lobby environment
- Long-distance corridor environment

CONCLUSIONS

- A model-free deep reinforcement learning method for mobile robot navigation with map-based obstacle avoidance, which directly maps egocentric local grid maps to an agents steering commands in terms of target position and movement velocity.
- Based on dueling double DQN with prioritized experience reply, and integrate curriculum learning techniques to further enhance our performance.
- Both qualitative and quantitative results show that the map-based obstacle avoidance method outperforms other related DRL-based methods in multiple indicators in simulation environments and is easy to be deployed to a robotic platform.
- Integrated our obstacle avoidance policy into the navigation framework for long-range navigation testing.

Extended work (https://cgdsss.github.io/)

Guangda Chen, Shunyi Yao et. al. Distributed Non-Communicating Multi-Robot Collision Avoidance via Map-Based Deep Reinforcement Learning[J]. *Sensors*, 2020, 20(17): 4836

Youtube: https://youtu.be/KOb1q23L7-U Bili: https://www.bilibili.com/video/BV12f4y1Q7cx/