Target Driven Mobile Robot with Collision Avoidance

Ming-Jyun Hong, Chien-An Chen & Joy-Eveline Leeuw

Problem

Automatic service robots can be used in many situations to assist human beings, for example for dull or repetitive tasks. It is important, that the robot can reach a assigned goal destination in an unknown and dynamic environment while avoiding any collision.

Solution

This work presents a simulation of a mobile robot that can achieve such kind of tasks. The robot can move based on the 2-wheeled kinematic model and detect the environemnt with laser and sonar sensors. It learns how to reach its goal location without collision by using reinforcement learning.

Reinforcement Learning

We modeled a mobile robot and implemented reinforcement q learning to acquire the best policy [1][2].

$$Q_{t+1}(s_t, a_t) \leftarrow Q_t(s_t, a_t) + \alpha_t \cdot \left(R_{t+1} + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t)\right)$$

State [Orientation, Laser, Sonar]

The state is encoded by three elements: the direction towards the target, laser data and sonar data.
The robot needs the first element to be guided towards the target and the last two elements to detect any approaching obstacles.

Laser

Five laser rays pointing into different directions detect the distance between robot and obstacles in its range. Each laser's range is splitted into 5 discrete segments to tell the distance to an obstacle.

Sonar

Because the scanning range of the lasers are limited and the robot needs to detect obstacles approaching from both left and right sides, we installed sonar sensors on each side to ensure safety.

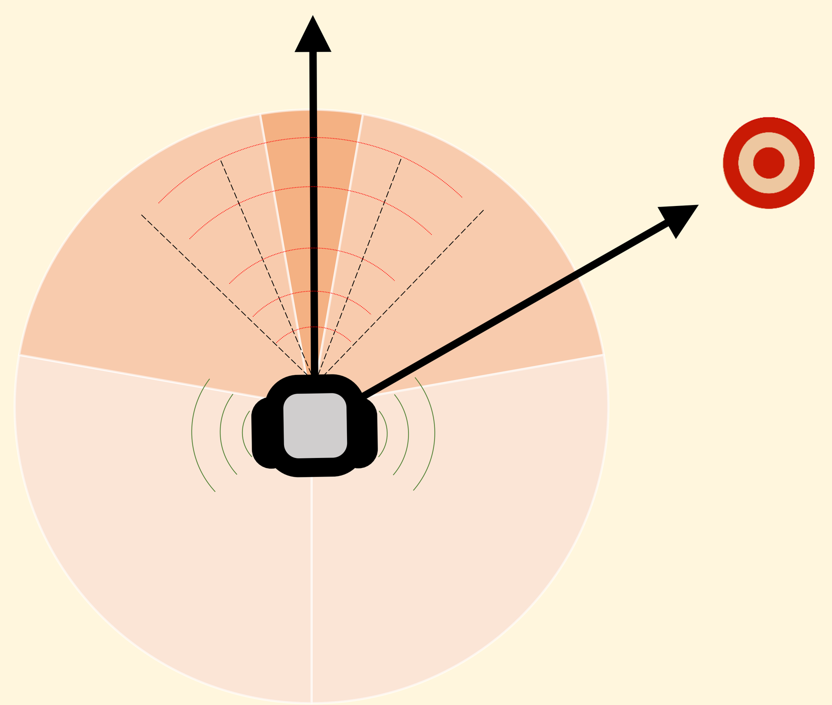

Fig1. Modeling of the robot and sensors

Orientation

We divided the 360° surrounding of the robot into 5 sections. The target will simply fall into one of these sections depending on the direction.

Action [Turn left, Turn right, Go straight]

We modeled the movement of the robot with a simple formula for 2-wheeled kinematics, which predicts how the robot's position will change based on velocities for the left and right wheel [3].

$$x(t) = x_0 + \frac{b(V_R + V_L)}{2(V_R - V_L)}\left[\sin\left(\frac{(V_R - V_L)t}{b} + \theta_0\right) - \sin\theta_0\right]$$

$$y(t) = y_0 - \frac{b(V_R + V_L)}{2(V_R - V_L)}\left[\cos\left(\frac{(V_R - V_L)t}{b} + \theta_0\right) - \cos\theta_0\right]$$

Reward

We defined the following four situations and gave the robot its corresponding reward in every sampling time.

- Collision (-100) : Robot collides with an obstacle
- Alive (+1) : Robot does not collide with any obstacles
- Safety Bonus (+2) : Robot does not detect any obstacles
- Direction Bonus (0~+5) : Positive if facing the right direction towards the target, negative otherwise

Results

Fig.2 shows that the alive time of the robot increases after about thousand trials are made. Every time the robot makes a move, it will learn new information from different situations and finally get a good policy to stay alive and reach randomly appearing targets. Also, Fig.3 shows that the robot can reach more and more targets due to the ability to avoid obstacles and drive towards targets.

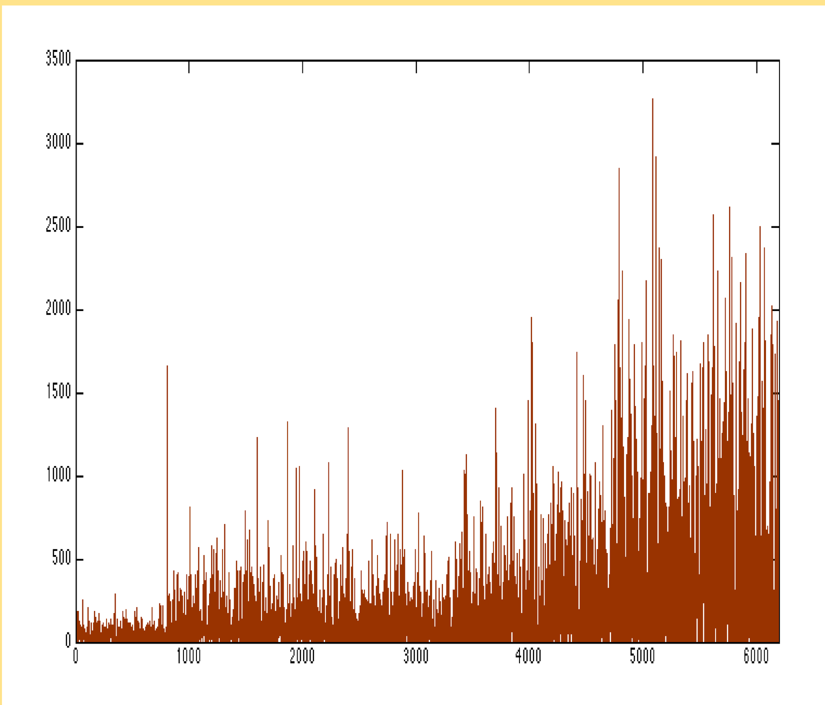

Fig2. Alive time vs Number of trials

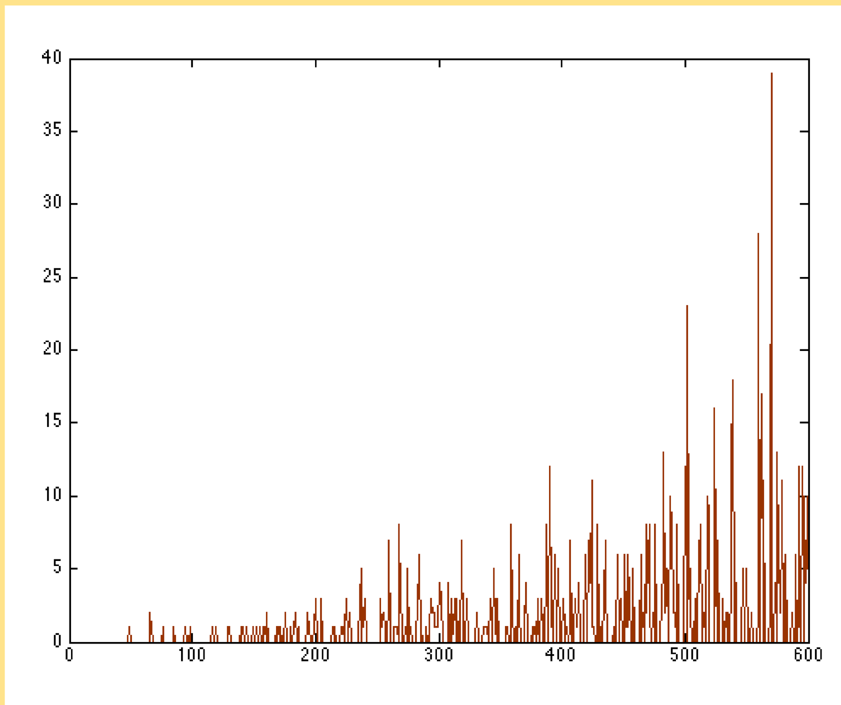

Fig3. Number of reached targets vs Number of trials

Future works

- Increase training efficiency by considering the symmetry
- Work on a real robot!

References

[1] Stuart J. Russell; Peter Norvig (2010). Artificial Intelligence: A Modern Approach (Third ed.). Prentice Hall. p. 649.
[2] Alexander L. Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L. Littman. Pac model-free reinforcement learning. In Proc. 22nd ICML 2006, pages 881–888, 2006.
[3] Reinforcement Learning: An Introduction. Richard Sutton and Andrew Barto. MIT Press, 1998.