Summer Reading 12: Learning Crowd-Aware Robot Navigation from Challenging Environments via Distributed Deep Reinforcement Learning

Social Robot Navigation Project @ Bot Intelligence Group

Paper:

Learning Crowd-Aware Robot Navigation from Challenging Environments via Distributed Deep Reinforcement Learning

Sango Matsuzaki1 and Yuji Hasegawa1

Summary:

Abstract

- This paper presents a deep reinforcement learning (DRL) framework for safe and efficient navigation in crowded environments.
- Here, the robot learns cooperative behavior using a new reward function that penalizes robot actions interfering with the pedestrian's movement.
- The authors propose a simulated pedestrian policy reflecting data from actual pedestrian movements.
- Furthermore, the authors introduce collision detection that considers the pedestrian's personal space to generate affinity robot behavior.

Introduction

- Navigating efficiently and safely as humans remains a challenge because pedestrians are moving obstacles, constantly making decisions and moving, thereby making their next move unpredictable.
- Also, interactions between pedestrians in crowded environments become synergistically more complex, making navigation more difficult.
- Recently, several deep learning (DL) methods have been studied to address this challenge. Crowd-aware navigation methods using DL are broadly categorized into:
    - Imitation Learning (IL)
        - IL methods optimize policies using extensive expert demonstrations.
        - IL methods require several times and techniques to collect and subsequently process the data.
        - Additionally, IL cannot be applied to robots of different sizes from people, and it is not suitable to control robots using learned policy.
            - Thus, a critical limitation when using robots that are larger than people, such as transport robots in factories.
    - Deep Reinforcement Learning (DRL)
        - DRL methods, however, create simulated environments that model pedestrians and robots, and obtain policy by interacting with the environment.
            - Therefore, robots of any size and mechanism learn to consider their characteristics.
- The following are the authors' main contributions:
    - (1) The authors present an environment that combines:
        - rewards for cooperation
        - collision detection using personal space
        - pedestrian policy reflecting actual pedestrian movement
    - (2) To efficiently explore this environment, the authors propose a training process using Ape-X.

        - (3) The authors’ navigation method is simulated and tested in actual crowded environments and demonstrates promising results. The authors refer to their navigation approach based on Ape-X, as NAX.

Related Work

- Navigation in Crowded Environments
    - Crowd-aware navigation methods can be divided into 2 categories:
        - Model-based methods
            - E.g. Social force: Social force models human interaction using repulsion and attraction.
            - In model-based methods, the appropriate parameters vary with the situation, including the population density or goal distance.
                - Therefore, finding a single parameter that is appropriate in crowded environments is difficult.
        - Learning-based methods
            - Learning-based methods have been actively studied with the development of DL, and methods using DRL have shown promising results.
    - These researches use a simulation environment to collect robot experiences and then, update the value function or policy using these experiences.
        - However, the policy of pedestrians in the simulation environment is the same as robots (reciprocal assumption), zero velocity, non-cooperative (straight to the goal), and ORCA (one and/or a combination of the above), so the simulation does not reflect real pedestrians’ motion.
        - Therefore, when a value function or policy trained in this simulation environment is run in a real environment, it can perform differently from the simulation.
    - Hence, the authors create an environment that reflects real pedestrians’ movements, and train state-action value function using the environment.

- Human Aware Reward Function
    - Navigating with pedestrians is an essential function for robots in crowded environments.
    - In a DRL framework, an acceptable policy can be obtained for pedestrians by penalizing actions that violate them.
        - Ex) SA-CADRL employed an approach to induce social norms by penalizing the three behaviors of overtaking, passing and crossing.
            - Due to the complicated human interaction in crowded environments, penalizing only these three behaviors is insufficient to correct a robot's disturbing behaviors.
        - Ex) A reward function that changes the safety radius of obstacles and moving obstacles like pedestrians to achieve a social manner was

    proposed. Further, a reward function that penalizes discomfort based on distance from others is also often used.
        - However, learning a social manner only from the positional relationship between obstacles and robots is difficult.
    - Thus, the authors propose a novel reward function penalizing robot actions that violate pedestrian movement, not only in specific scenes.
- Training Process for DRL
    - In a simulation environment reflecting the actual pedestrians behavior, navigating while not obstructing the surrounding pedestrians is challenging, thus efficient exploration is important to learn the optimal policy.
    - There is an issue of robots learning behavior specific only to certain environments (e.g. 10-agent environment, 2-agent environment, etc.)
    - Thus, the authors propose a training process allowing robots to efficiently explore the diverse number of agents while employing curriculum learning.

Approach

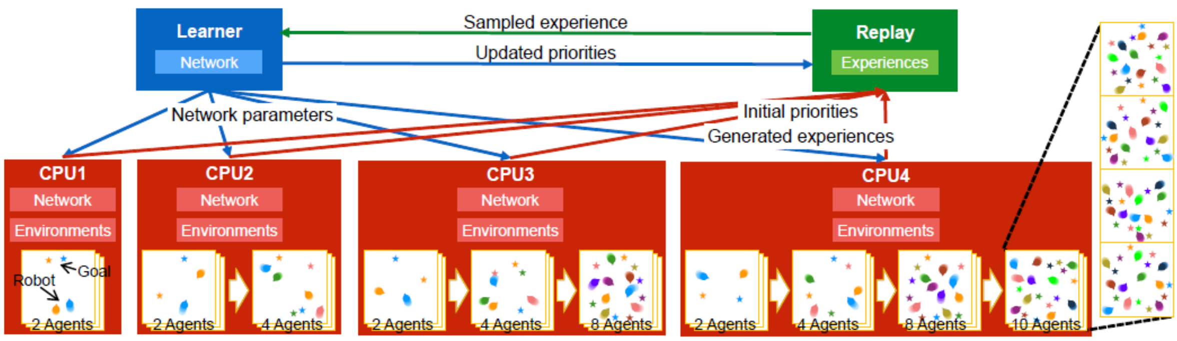
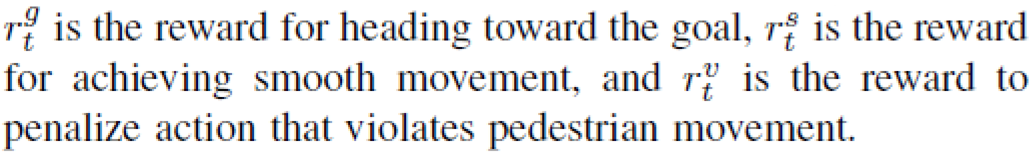
- Problem Formulation:
    - In this work, the authors developed crowd-aware navigation as a Partially Observable Markov Decision Process (POMDP).
    - There are 2 values: State of an agent & Action of the agent
        - State: consists of the state that is observable by all agents & hidden state that is observable only by the agent
        - Action: consists of translational velocities & rotational velocities
- Training Process:
    - The state-action value function q is trained using a method from Ape-X.
    - 

      Fig. 2: Overview of distributed learning, where multiple environments are created for each CPU (red), and the experiences of multiple robots are stored in the replay buffer (green). The stored experiences are prioritized and sampled, and the experiences are used by the learner (blue) to update the network parameters.
    - The authors employ a method, where multiple robots are placed in a single environment to efficiently gather more experiences.
    - Furthermore, the authors create multiple environments virtually within a single CPU to gather more experiences.
    - The authors propose a method of collecting a greater diversity of experiences by varying the number of agents in the final environment for each CPU.

- Realistic Simulated Pedestrians:
    - The authors used the method (creating several environments by
    - setting Social Force Model [SFM] as a policy for pedestrians is possible) to estimate SFM parameters from pedestrian flow data obtained in the previous work.
        - About Social Force Model (SFM):
            - *Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. Phys. Rev. E, 51:4282–4286, May 1995.*
- Learning Cooperative Behavior:
    - Reward function

$$r_t = r_t^g + r_t^s + r_t^v$$



r_t^g is the reward for heading toward the goal, r_t^s is the reward for achieving smooth movement, and r_t^v is the reward to penalize action that violates pedestrian movement.

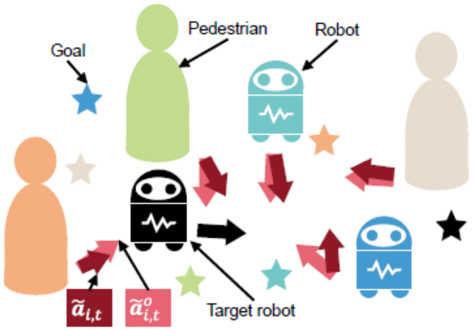


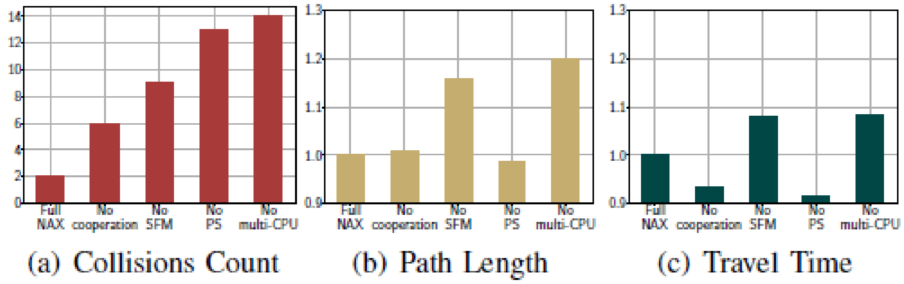
Fig. 3: Overview of calculation of rewards for cooperation. The black robot receives the reward. The reward is the sum of the difference between the actions of the surrounding agents mapped from the state including the black robot (dark red) and the actions of the surrounding agents mapped from the state without the black robot (light red) overall agents.

    -
    - Unlike previous studies, the authors did not set any rewards for collisions. Instead, the authors introduce a new collision detection method.
        - Humans have a personal space(PS), which is egg-shaped. Everyone holds, preserves, and updates the PS, and reacts when it is violated by others.
        - Previous studies used a perfect circle for collision detection, but by considering the PS for collision detection, more cooperative behaviors are learned, such as not crossing in front of pedestrians.
- Network Architecture:

    - The authors chose GCN derivative because it is adaptable, even if the number of pedestrians changes, the input dimension does not become large and the features can be convolved.
    - The authors considered relationships between nodes.

Experiments

- Ablation Studies
    - The authors ablated (removed) 4 different variables one at each time for ablation studies.
        - 4 different variables that were ablated are:
            - Cooperation
            - SFM
            - Egghead shape Personal Space (PS)
            - multi-CPU



(a) Collisions Count (b) Path Length (c) Travel Time

    - Fig. 5: Results of our full NAX and other parameters. The path length and elapsed time are divided by those of Ours, respectively.

Conclusion

- The authors proposed the NAX framework using DRL (Deep Reinforcement Learning) to achieve crowd-aware robot navigation equivalent to that of humans in the constructed scenarios.
- The authors created an environment that combines rewards for cooperation, collision detection using PS, and pedestrian policy using SFM (Social Force Model).
    - The authors proposed a training process using Ape-X to efficiently train the state-action values in this environment.