Social robot navigation in environments with different population density

Jialu Yao

Manufacturing engineering, the University of British Columbia, Vancouver, BC, V6N3H8, Canada

tino0101@student.ubc.ca

Abstract. In the field of modern robotics, robots need not only to avoid obstacles, but also to meet the comfort of pedestrians. In this paper, two methods of robot navigation: CP-SFM and reinforcement learning are introduced. CP-SFM model can analyse the repulsive force between the pedestrians and robot and generate a pedestrian-friendly path to the destination. Reinforcement learning method create a crowd-human interaction and use deep learning framework to train the robot learning human-robot and human-human interaction. These methods are also applied to the simulation environment and the results show the success of them.

Keywords: Modern robot. Reinforcement Learning, Simulation

1. Introduction

As robots become the mainstream of the future, they are able to perform more and more tasks and are becoming more integrated into society. Therefore, robot navigation should not only avoid obstacle, cruise on a given trajectory, but also need a comfortable human-robot interaction which means human should be comfortable with space and the application of the robot when the robots are close to human and safety. Recently, most modern algorithms have been applied to robot but most of their functions are solving one-way robotic problem *i.e.,* navigating robot to one destination without considering obstacle that suddenly appears and do not achieve better human-robot interaction.–

To ensure that the robot is safe for the society, collision avoiding is vital for robot navigation. To avoid collision basically there are two main types of way: reaction based [1], and trajectory based [2]. For, reaction-based method, the robot only responds to its local environment like changing the speed and direction based on map that has been generated at any given time like reciprocal velocity obstacle (RVO) that the robot can detect the environment and adjust its speed and direction to avoid collision [3]. For instance, household sweeper robot which can detect the wall and sweep on a designated track, for some movable obstacles like cats or human, they cannot predict and avoid these obstacles in dynamic environments [4]. However, trajectory-based method can predict other performance of entities (human) such as going straight or turning left, and the robot can take action to avoid collision. The robot can analyse the purpose of pedestrians like going straight, turning left and real-time commands are given to the robot to avoid collision.

Some early models and algorithms laid the foundation for the future development of human-robot interaction. For instance, a different of Dynamic Window Approach (DWA) which extended optimization of navigation assignments is introduced [5]. DWA can help robot generate a better trajectory by constraining the speed and acceleration of the robot to avoid collision.

Another method is reinforcement learning (RL) and robots can learn how to deal with dynamic environment from the original observation or command by applying this method to robot [6]. By training the robots several times, they can learn and complete specified commands and policies. Inverse reinforcement learning (IRL) is the inverse method of RL, and the robot applied this method can learn and predict the human motion by observing people walking through the street [7]. Maximum entropy deep inverse reinforcement learning (MEDIRL) based on IRL is introduced to learn pedestrian trajectory and the human navigation behaviour is captured by deep neural network (DNN) using a nonlinear reward function [8]. In this paper, two methods of human-robot interaction on robot navigation will be compared and the result will be discussed in the following part.

2. Problem Formulation

Social robot navigation is a new navigation method for robot that the robot can not only avoid collision but also ensure the comfort of the pedestrians. The state of robot and human like speed and position can be obtained. Based on these conditions, different method can help the robot predict the motion of human and generate the best path to the destination. The main objective of robot navigation is how to train or tell the robot to predict the motion of pedestrians and avoid them.

2.1. Social force model

Social force model is used to predict the movement of human and can be separated to three essential properties: the acceleration in the direction of movement, proper distance between human and other obstacles including other pedestrians and a term modelling attractive effect [9].

The repulsive force is the force that keep a certain distance between pedestrian and obstacles and can be divided into two parts: circular repulsive force and collision repulsive force [10]. The formula of collision repulsive force is derived below:

$$f_{ij}^{rep} = \sum_{j \in Q_h}^{Q_h} A_h \frac{v_i}{t_i} \cdot e^{\frac{-d_{ij}}{B_h}} \cdot \frac{d_{ij}}{|d_{ij}|} \tag{1}$$

Where f is the collision repulsive force, Q_h means human entity, A_h is the interaction force intensity, d_{ij} is the distance between different entities, i represents robot observer and j represents perceived human, B_h is the range of interaction, v_i is the robot velocity and t_i is the minimum time. If the angle between the relative speed and position vector is smaller than radians, possible collisions may be determined, assuming linear motion at a constant speed or when the time derivative of the angle is approaching zero [9].

The formula of circular repulsive force is:

$$f_{ij}^{rep} = \sum_{j \in Q_o}^{Q_o} A_0 \cdot e^{\frac{-d_{ij}}{B_h}} \cdot \frac{d_{ij}}{|d_{ij}|} \tag{2}$$

Where f is the circular repulsive force, Q_o is obstacle entity. Unlike collision repulsive force, circular repulsive force only considers the normal distance between different entities instead of considering the speed and time.

The Attractive force is the force that can drive the robot toward a navigation goal, and it can be written as:

$$f^{att}(t) = m \cdot \frac{1}{r} \cdot \left(v^0(t) - v(t)\right) \tag{3}$$

Where v^0 is the multiplication of speed and the unit vector of desired direction, m is robot mass, r is relaxation time and v is the actual velocity of the robot.

2.2. Reinforcement learning

Another approach is considering the robots move around several of n people. Thus, this problem can be divided into many small problems in a reinforcement learning framework [6,11]. For every human or robot, the velocity v is separated as v_x and v_y and the position p is separated as p_x and p_y, the state of robot can be described by s_t, a matrix of iv_t can be used to display the status of human at time t. The union condition for robot navigation is described as $s_t^{jn} = [s_t, \tau v_t]$. The return value can be written as:

$$\pi^* \cdot s_t^{jn} = argmaxR\left(s_t^{jn}, a_t\right) + \mu^{t \cdot v_{pref}} \int_{s_{t+\Delta t}^{jn}} P\left(s_{t+\Delta t}^{jn}, a_t, s_{t+\Delta t}^{jn}\right) V^* \cdot \left(s_{t+\Delta t}^{jn}\right) ds_{t+\Delta t}^{jn} \tag{4}$$

$$V^*(s_t^{jn}) = \sum_{t=t}^{T} \gamma^{t \cdot v_{pref}} R_t(s_t^{jn}, \pi^* \cdot s_t^{jn}) \tag{5}$$

Where v_{pref} is preferential speed, at is the command of action, $R_t(s_t^{jn}, a_t)$ is the reward received at time t, the discount factor γ is between 0 and 1, $P(s_{t+\Delta t}^{jn}, a_t, s_{t+\Delta t}^{jn})$ is the transitional probabilities after receiving the action command among Δt [11].
The final reward equation is defined in [11]:

$$R(s_t^{jn}, a_t) \begin{cases} -0.25 & if\ d_t < 0 \\ -0.1 + \dfrac{d_t}{2} & else\ if\ d_t < 0.2 \\ 1 & else\ if\ p_t = p \\ 0 & otherwise \end{cases} \tag{6}$$

Where d_t is the minimum distance between different entities (human or robot) during the period Δt. As the minimum distance changes, the equation of reward function changes.

3. Method

CP-SFM and reinforcement learning method are introduced in the following. These methods can help the robot achieve human-robot interaction. CP-SFM is the model that can analyse the repulsive force and generate a pedestrian-friendly path to the destination. Reinforcement learning method create a crowd-human interaction and used deep learning framework to train the robot learning human-robot and human-human interaction. The main difference between two method is the applied environment. Due to the limitation of analysing several forces, CP-SFM can be applied to the low or average population density. However, reinforcement learning can be applied to a very large population environment.

3.1. CP-SFM

The first method for robot navigation is to create a proactive social motion model (CP-SFM) [12]. Unlike SFM, a robot's command of the next time will be exported. The acceleration of the robot can be calculated by Newton's law where the total resultant force is the sum of all the attractive force and repulsive forces. Collision prediction force was introduced in [10] which uses circular force of the robot and the direction of human moving. The time $\mathrm{t_i}$ is the minimum time that the relative distance between pedestrians and robot, assuming it's straight motion. The contribution of every human can be written as:

$$\mathrm{f_{ij}}(\{\mathrm{d_{ij}}\}, \{\mathrm{v_{ij}}\}, \mathrm{v_i}) = \mathrm{A}\frac{\mathrm{v_i}}{\mathrm{t_i}} \cdot \mathrm{e}^{\frac{-\mathrm{d_{ij}}}{\mathrm{B_h}}} \cdot \frac{\mathrm{d_{ij}}}{|\mathrm{d_{ij}}|} \tag{7}$$

Another term $\mathrm{v_i}/\mathrm{t_i}$ is added into the equation to adjust the velocity of the robot when the collision is going to happen. The collision prediction force is shown in Eq. (1) and a brief figure of this formula is shown in Figure.1.

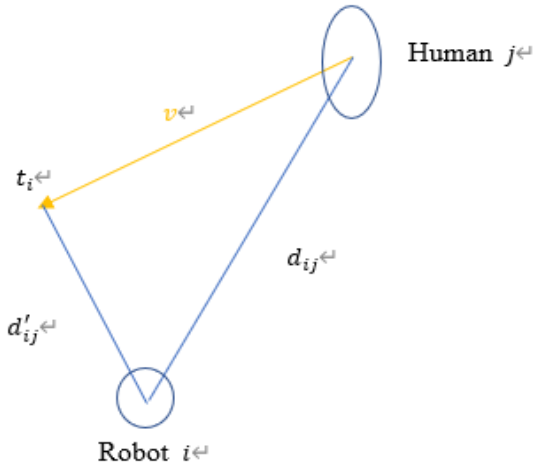


Figure 1. The graphical notation when the robot gets close to human

When a collision is imminent, the pedestrians should have the ability to predict the possible collisions and avoid them by moving away. When the robot is moving straightly to one destination, if the difference between the bearing angle θ which is the angle between speed v and the vector of distance d_{ij} is smaller than 45 degrees, the possible collision can be detected [10].

After that, a concrete robot motion model is established based on the vector of acceleration with a collection of movements that represents all potential directions. Thirteen high-level actions are defined in Fig.2 and each action has an angular difference of 15 degrees between the CP-SFM output and the robot angle.

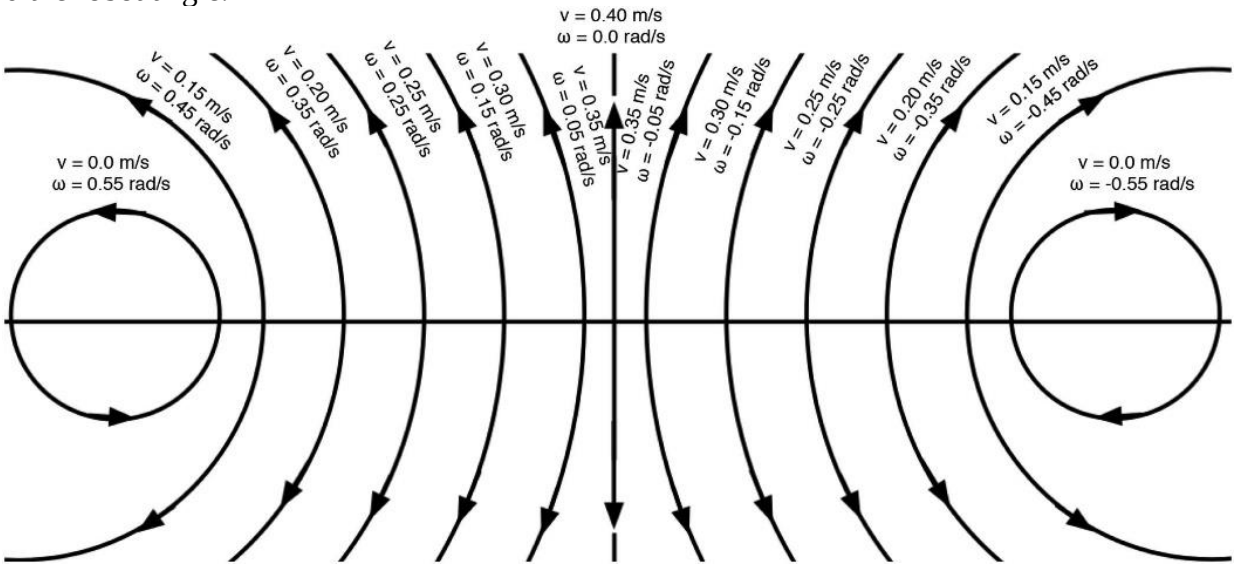


Figure 2. A set of angle velocities and speeds to control the acceleration

The simulation is done by setting a corridor-like environment in ROS Rviz simulator. The robot is commanded to navigate to a destination where pedestrians will pass by during simulation time. The CP_SFM node mentioned before is applied to this simulation.

Social Individual Index (SII) can give the measurement of human physical comfort by separating the space around a human into four parts. The relative movement of robot and pedestrians can be determined by Relative Motion Index (RMI) and these two indexes are used to determine whether a robot is successful in navigating through a crowd [13]. RMI increases as the robot and human approach at high speeds and high RMI means high probability of collision. Physical and psychological threshold values are defined to determine SII. When the distance to human equals 0.5m, SII is equal to psychological threshold value and when the distance to human equals 0.9m, SII is equal to physical value which is 0.14. As the distance between robot and human increases, SII decreases until 0 and the graph shows in Fig.3.

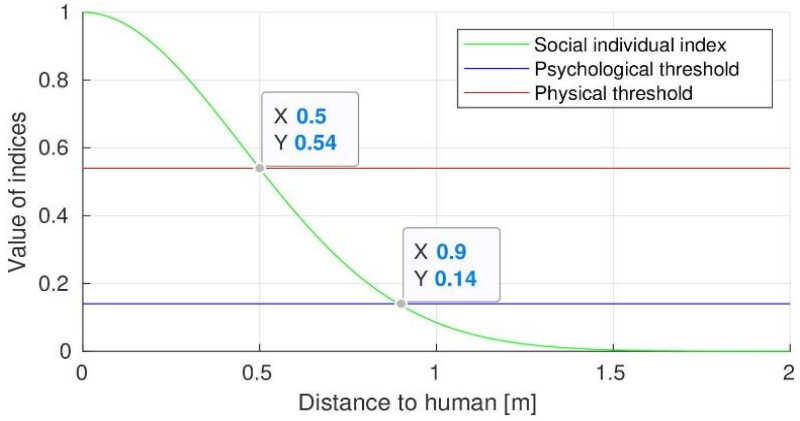


Figure 3. Social individual index (SII) vs distance to human [m]

There are several parameters in CP-SFM model, and the performance will be affected by them. These parameters can be set based on different experience, but the trend of the experience will be similar. The desired speed should not be too large so that the robot can change its speed stably. In this experience the strength of interpersonal communication A_h, range of human interaction B_h, relaxation time τ, obstacle repulsion strength A_o, obstacle repulsion range B_o and desired speed of robot A_o are 1.5, 4.9, 1.0,5.0,0.3,0.6 respectively [12].

Fig.4 shows the determination of SII and RMI values during 50 seconds (1200 simulation points). Both SII and RMI values are lower than thresholds for all pedestrians which means the robot can meet people's comfort level while avoiding pedestrians and keeping the proper distance from them.

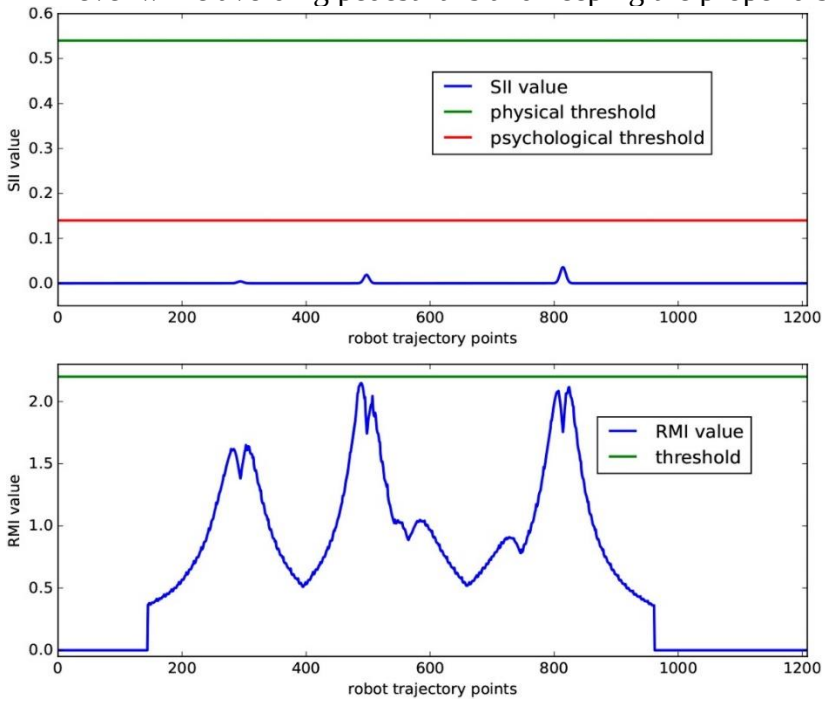


Figure 4. The result of social individual index (SII) and RMI while the robot is moving

3.2. Reinforcement learning

When a group of pedestrians gather, they will consciously observe the behavior of other pedestrians to interact indirectly like pedestrians will subconsciously avoid when there is an obstacle in front of them.

The robot centric parameterization mentioned in [11] shows that the robot's states s and pedestrian's states w_i can be represented as

$$s = [d_g, v_{pref}, v_x, v_y, r],$$

$$w_i = [p_x, p_y, v_x, v_y, r_i, d_i, r_i + r] \tag{8}$$

Where d_g is the distance between the robot and destination and d_i is the distance between the robot and the neighbor including other robots and human.

3.2.1. Interaction model

When the number of pedestrians is abundant, it would be too much work for the robot if each pedestrian interacts with the robot. Therefore, a paired interaction model is introduced [14] that can use local maps as a coarse-grained representation and can help robot achieve human-robot interaction.

A map constructor can be created if the size of neighbourhood is known.

$$M_i(a, b, :) = \sum_{j \in \infty N_i} \delta_{ab}[x_j - x_i, y_j - y_i] w'_j \tag{9}$$

The map size is $L * L * 3$, where L is the size of neighbor. δ_{ab} is an indictor function and it will change as the object's relative position change. When the position is coincided with (a, b), it equals to 1. w'_j represents the state of pedestrian j which contains horizontal and vertical velocities. N_i is the collection of neighboring pedestrians around the j^{th} human. Then a length vector e_i is defined by combining pedestrian state and M_i.

$$e_i = \rho_e(s, w_i, M_i; W_e) \tag{10}$$

Where ρ is function which is activated by ReLU and W_e is the weighting of the portfolio. After that the paired interaction function can be obtained:

$$h_i = \varphi_h(e_i; W_h) \tag{11}$$

Where φ is a fully connected layer with ReLU non-linearity and W_h is the weights of network.

3.2.2. Pooling model

The number of people can vary greatly in different scenarios, and in order for the robot navigation system to be used in more places, a pooling model is created by analyzing [15,16] to deal with different population densities. The embedding e_i can be replaced as attention score as shown below:

$$e_m = \frac{1}{n} \sum_{k=1}^{n} e_k \tag{12}$$

$$\alpha_i = \varphi_\alpha(e_i, e_m; W_\alpha) \tag{13}$$

Where e_m is a vector with fixed length embedding, derived by the average of all individuals. φ_α is an MLP with ReLU activations. The equation that represents all humans in a given place can be written as:

$$c = \sum_{i=1}^{n} soft\,max(\alpha_i)\, h_i \tag{14}$$

3.2.3. Planning model

A planning model is established to calculate the state value v using the previous parameters.

$$v = f_v(s, c; W_v) \tag{15}$$

Where f_v is similar like φ_α which is an MLP with ReLU activations.

3.2.4. Simulation setup

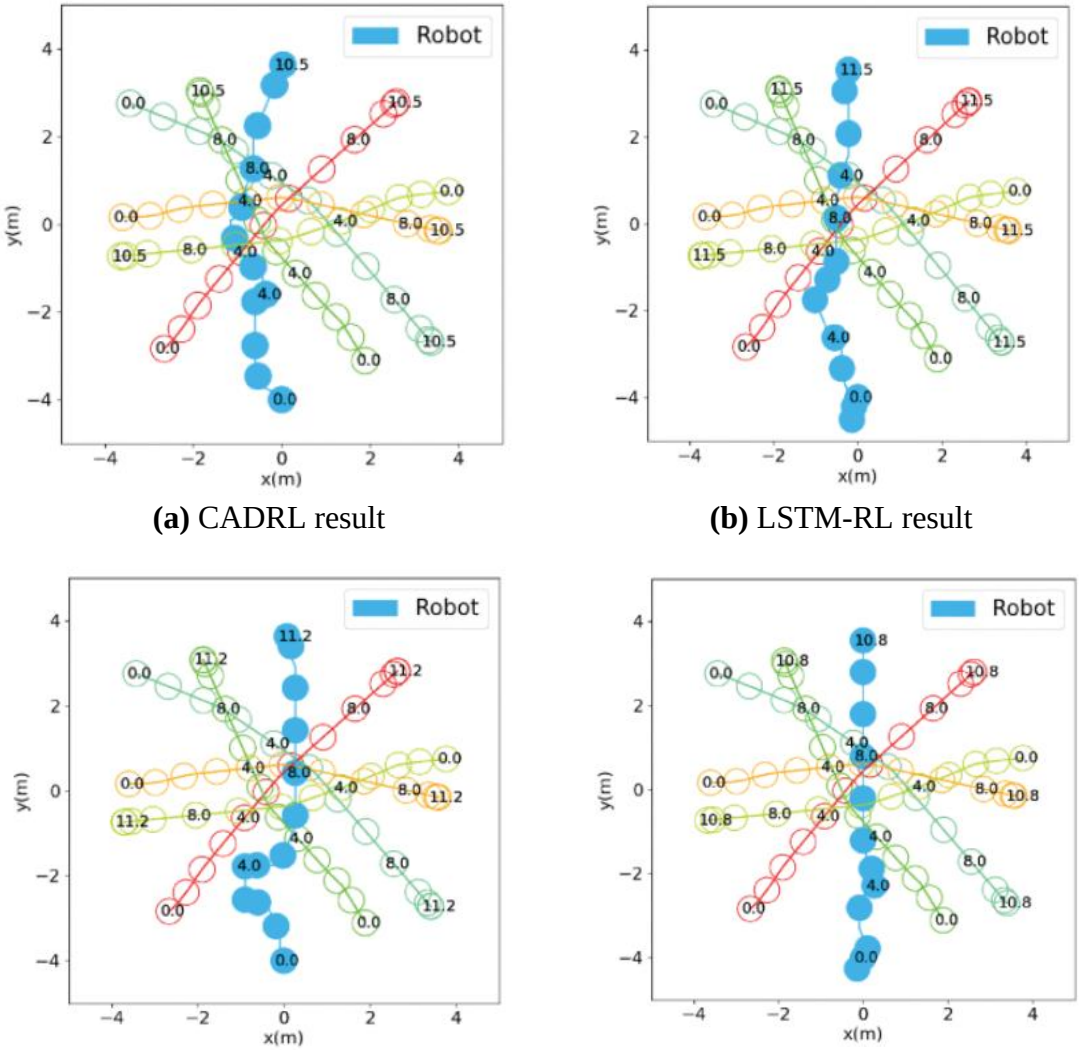
The simulation is finished using PyTorch [17] which is a branch of python and ORCA is used to train the simulated human [18]. The parameters can be set differently, in this experience, the functions ρ_e, $\varphi_h, \varphi_a, f_v$ are set as (150,100), (100,50), (100,100), (150,100,100) respectively. Gaussian

distribution is used to get the parameters of the simulated humans to introduce diversity behavior. The simulated pedestrians are randomly set on a circle with the radius 4m and the destination of them is on the other side of this circle.

CADRL [11]and LSTM-RL [19] are both based line method and they are used to compare the result. LM-SARL is the model that applied three models mentioned before and SARL is the model without using a local map [14].

3.2.5. Simulation result and analyze

The simulation result of four different models: CADRL, LSTM-RL, LM-SARL and SARL are shown in Fig.5 [14]. For CADRL, when the robot encounter pedestrians, it directly passes them without avoiding collision. For LSTM-RL, the robot slows down to avoid pedestrians, but the navigation time is quite long. SARL shows a better performance, though it hesitates at first time, it will generate a shorter path to the destination and avoid the crowd. The LM-SARL have built a trajectory considering both safe distance and time for navigation. Both SARL and LM-SARL results show that the reinforcement learning method can be applied to the robot and the robot can predict the pedestrians and generate the path to avoid collision.



(a) CADRL result **(b)** LSTM-RL result

(c) SARL result **(d)** LM-SARL result

Figure 5. Results of four different models

The circles represent human position in different time and different colour represents different human. CADRL shows that the robot will aggressively pass by pedestrians and LSTM-RL shows the robot will behave conservatively when meeting with pedestrians. On the contrary, SARL and LM-

SARL using deep learning framework can successfully design a shorter path while avoiding pedestrians.

4. Conclusion

In this work, two methods related to the robot navigation are discussed. CP-SFM navigates the robot by analysing the repulsive force and can generate a pedestrian-friendly path to the destination. Reinforcement learning method creates a crowd-human interaction and used deep learning framework to train the robot learning human-robot and human-human interaction. Because of limitations of analysing plenty of forces, CP-SFM is suitable for environment with low or average population density. However, deep learning framework can be applied to places with high population density like train stations. These two methods provide a reference for the subsequent research of robot navigation.

References

[1] Zieliński C, 1994 Reaction based robot control, Mechatronics 4 p 843-860

[2] Kumar A, Ojha A. and Padhy P.K. 2017 Anticipated trajectory based proportional navigation guidance scheme for intercepting high maneuvering targets. Int. J. Control Autom. 3 p 1351–1361

[3] Berg J, Lin M and Manocha D 2008 "Reciprocal velocity obstacles for real-time multi-agent navigation", *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA)* p. 1928-1935.

[4] Prassler E., Ritter A., Schaeffer C. et al. 2000 A Short History of Cleaning Robots. 9, 211–226.

[5] Forer S, Banisetty S B, Yliniemi L, Nicolescu M and Feil-Seifer D, 2018 "Socially-Aware Navigation Using Non-Linear Multi-Objective Optimization," pp. 1-9.

[6] Long P, Fan T, et al., 2017 "Towards Optimally Decentralized Multi-Robot Collision Avoidance via Deep Reinforcement Learning", *arXiv:1709.10082 [cs]*, .

[7] Pérez-Higueras N, Ramón-Vigo R, Caballero F, Merino L, 2014 Robot local navigation with learned social cost functions, in: Informatics in Control, Automation and Robotics (ICINCO), 11th International Conference on, 2, 618–625.

[8] Fahad M, Yang G and Guo Y, "Learning How Pedestrians Navigate: A Deep Inverse Reinforcement Learning Approach," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 819-826.

[9] Helbing D, Molnar P 1995 Social force model for pedestrian dynamics Phys. Rev. E 51 p. 4282

[10] Zanlungo F, Ikeda T, Kanda T 2011 Social force model with explicit collision prediction, EPL (Europhysics Letters) 93 P 68005.

[11] Chen Y, Liu M, Everett M and P J 2016 "Decentralized Non-communicating Multiagent Collision Avoidance with Deep Reinforcement Learning" *arXiv:1609.07845 [cs]* p 8.

[12] Kivrak H, Cakmak F, Kose h, Yavuz S, 2021 Social navigation framework for assistive robots in human inhabited unknown environments, Engineering Science and Technology, an International Journal, 24 p 284-298

[13] Truong X.T, Ngo T.D 2017 Toward socially aware robot navigation in dynamic and crowded environments: a proactive social motion model Trans. Autom. Sci. Eng., 14 p 1743-1760

[14] Chen C, Liu Y, Kreiss Sand, Alahi A, 2019 "Crowd-Robot Interaction: Crowd-Aware Robot Navigation with Attention-Based Deep Reinforcement Learning," 2019 International Conference on Robotics and Automation (ICRA) p 6015-6022,

[15] Liu Y, Sun C, Lin L and Wang X 2016 "Learning Natural Language Inference using Bidirectional LSTM model and Inner-Attention", *arXiv:1605.09090 [cs]*,

[16] Lin Z, Feng M, Santos C.N, Yu M, Xiang B, Zhou B, Bengio Y 2017 A Structured Self-attentive Sentence Embedding *arXiv:1703.03130 [cs]*,

[17] Paszke A et al 2017 Automatic differentiation in PyTorch

[18] Van den Berg J, Guy S J, Lin M and Manocha D, 2011 Reciprocal n-Body Collision Avoidance in Robotics Research ser. Springer Tracts in Advanced Robotics, Springer Berlin Heidelberg p 3-19

[19] Everett M, Chen Yand How J.P 2018 Motion Planning Among Dynamic Decision-Making Agents with Deep Reinforcement Learning *arXiv:1805.01956 [cs]*