


RESEARCH ARTICLE

Safe and socially compliant robot navigation in crowds with fast-moving pedestrians via deep reinforcement learning

Zhen Feng , Bingxin Xue, Chaoqun Wang and Fengyu Zhou

School of Control Science and Engineering, Shandong University, Jinan, Shandong, China

Corresponding author: Fengyu Zhou; Email: zhoufengyu@sdu.edu.cn

Received: 5 November 2022; **Revised:** 6 November 2023; **Accepted:** 9 January 2024;

First published online: 26 February 2024

Keywords: Navigation; mobile robots; human safety and comfort; social robotics; collision theory; deep reinforcement learning

Abstract

Safe and socially compliant navigation in a crowded environment is essential for social robots. Numerous research efforts have shown the advantages of deep reinforcement learning techniques in training efficient policies, while most of them ignore fast-moving pedestrians in the crowd. In this paper, we present a novel design of safety measure, named Risk-Area, considering collision theory and motion characteristics of different robots and humans. The geometry of Risk-Area is formed based on the real-time relative positions and velocities of the agents in the environment. Our approach perceives risk in the environment and encourages the robot to take safe and socially compliant navigation behaviors. The proposed method is verified with three existing well-known deep reinforcement learning models in densely populated environments. Experiment results demonstrate that our approach combined with the reinforcement learning techniques can efficiently perceive risk in the environment and navigate the robot with high safety in the crowds with fast-moving pedestrians.

1. Introduction

Over the last few decades, the application scenarios of mobile robots have gradually expanded from isolated static space to social space shared with human beings, such as hospitals, shopping malls, and canteens. In these crowd scenarios, humans frequently change their states, including moving direction, speed, acceleration, etc., to avoid collisions with nearby humans and obstacles. While these behaviors may appear random, they are in fact dynamic adjustments in response to the ever-changing environment [1–4]. This dynamic nature of human behaviors makes robot navigation in a crowded environment more complex than in an environment without humans and leads to the Freezing Robot Problem [5–7]. The robot cannot find a feasible path and falls into an oscillating or stopping situation. In order to navigate in dense crowds safely and socially compliantly, robots need to understand human behaviors and obey their cooperative rules [8–11].

Previous work attempts to jointly plan feasible paths for all the agents [6,12] or predict the future trajectories of the humans before planning [13–15]. However, these methods suffer from the stochasticity of human behaviors and high computational cost in dense crowd scenarios. In recent years, deep reinforcement learning technologies have made significant progress in solving social navigation problems [16–18]. These methods train computationally efficient policies that implicitly encode the interactions and cooperations among agents. Despite the recent progress, navigation in the crowd with fast-moving pedestrians is not well investigated. In real-world applications, pedestrians in sparse environments or with emergency issues usually have random moving patterns in the crowd. As a result, the safety of existing models remains a concern.

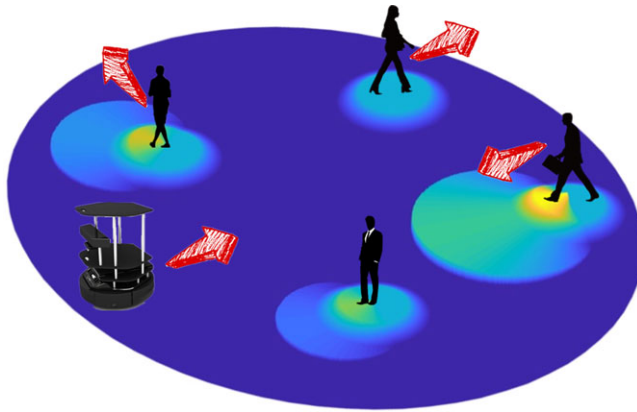


Figure 1. Safe and socially compliant robot navigation in a crowded environment considering Risk-Areas around each human based on real-time relative positions and velocities of agents.

We present a safety measure Risk-Area for safe robot navigation in densely populated environments as briefly shown in Fig. 1. The main contributions of this article are summarized as follows.

1. We introduce a dynamic model and analysis of the collision process between robots and pedestrians in-plane, providing a basis for assessing collision risk.
2. We incorporate human behavior and collision theory into the design of the Risk-Area, enhancing the understanding and implementation of socially aware navigation strategies.
3. We conduct extensive experiments and evaluations of our approach and state-of-the-art method in challenging simulation and real-world environments.

The rest of this paper is organized as follows. Section 2 provides the background and the problem formulation of the robot navigation in the crowd scenarios. Section 3 provides details of the Risk-Area design for the navigation problem. Section 4 presents the simulation and real-world experiment results. We conclude this paper in Section 5.

2. Background

2.1. Related work

Traditional methods attempt to tackle the robot navigation problem in crowded environments through well-engineered rules. A pioneer work develops social force model that defines attractive and repulsive forces to describe human interactions and has been successfully applied in both simulation and real-world environments [20–22]. Reciprocal velocity obstacles (RVO) [23] and optimal reciprocal collision avoidance (ORCA) [24] are velocity-based breakthrough algorithms in multi-agent collision avoidance. They consider joint collision avoidance under reciprocal assumptions to achieve collision-free navigation. Interacting Gaussian process [25] uses individual Gaussian process to model the trajectory of each agent and proposes an interaction potential term for interaction. Nevertheless, the above methods rely heavily on handcrafted design of interaction model, thus having difficulty generalizing complex human-robot interaction scenarios.

Learning-based methods focus on training strategies from human interactions to obtaining appropriate behaviors. For example, imitation learning approaches learn policies directly from expert demonstrations and map various inputs such as depth images, raw lidar data, and occupancy maps to robot actions [26–28]. Inverse reinforcement learning captures human cooperation features from human interactions via the maximum entropy method [29–31]. However, the outcomes of these methods are closely relevant to the quality of demonstrations.

Over the last few years, reinforcement learning has achieved great progress in solving real-time navigation problems in crowded environments. Long et al. [32] present a decentralized multi-robot collision avoidance algorithm, selecting actions directly from raw sensor data to achieve collision-free navigation. Chen et al. [33] develop a value network that encodes the estimated time to the goal based on the joint state of the agents. Subsequently, this work is extended by integrating social norms into reward functions in ref. [34]. In order to overcome the limitation of fixed observation size, Everett et al. [35] adopt the long short-term memory to encode agent states into a fixed-length vector. In ref. [36], attention mechanism is proposed to perceive the importance of humans in the crowd by modeling human-robot and human-human interactions, significantly improving the robots' decision-making ability. Thereafter, Chen et al. [37] utilize graph convolutional networks to extract the potential interaction features. The preceding studies succeed in capturing the cooperations and interactions of human behaviors. Nevertheless, the performance of the current models will degrade as the crowd complexity increases. Especially, most of existing models are overly optimistic about the speed of the robot and pedestrians. In order to achieve safe and comfortable navigation in social environments, the speed of mobile robots has to be significantly slower than pedestrians [1,38–40]. To address the issue, Samsani et al. [19] predict the constrained action space around the human based on the current human velocity and formulate Danger-Zones (DZ) for the robot. However, this method does not fully consider the relative motion among the agents. In this study, we design Risk-Areas around each human as safety measures according to collision theory and the real-time relative movements.

2.2. Problem formulation

We consider a situation where a robot moves in a crowd of n humans and reaches its goal without any collision. This task can be formulated as a sequential decision-making problem in a reinforcement learning framework [33]. It is assumed that humans do not avoid or intentionally hinder the robot during navigation. For each agent (human or robot), the position $[p_x, p_y]$, velocity $[v_x, v_y]$, and radius r are observable to the others. The agent's intended goal $[g_x, g_y]$, preferred speed v_{pref} , and heading angle θ are only aware by itself. At every time step, the robot can observe its full state \mathbf{s}_r and the observable state \mathbf{s}_{h_o} of other humans. The state of the robot is represented as $\mathbf{s}_r = [p_x, p_y, v_x, v_y, r, g_x, g_y, v_{pref}, \theta]$. The observable state of the i -th human is defined as $\mathbf{s}_{h_o}^i = [p_x^i, p_y^i, v_x^i, v_y^i, r^i]$. A robot-centric frame is employed to make the state representation more universal, in which the origin is set at the current position of the robot $\mathbf{p}_r = [p_x, p_y]$, and the x -axis points to the goal position $\mathbf{g}_r = [g_x, g_y]$. After transformation, the states \mathbf{s}_r and $\mathbf{s}_{h_o}^i$ can be rewritten as follows:

$$\begin{aligned} \mathbf{s}_r &= [d_g, v_x, v_y, r, v_{pref}, \theta] \\ \mathbf{s}_{h_o}^i &= [p_x^i, p_y^i, v_x^i, v_y^i, r^i, d^i, r^i + r], \end{aligned} \tag{1}$$

where $d_g = \|\mathbf{p}_r - \mathbf{g}_r\|$ indicates the distance from the robot \mathbf{p}_r to the goal \mathbf{g}_r , and $d^i = \|\mathbf{p}_r - \mathbf{p}_h^i\|$ denotes the distance from the robot \mathbf{p}_r to the i -th human \mathbf{p}_h^i . The joint state of the system at time t for robot navigation is defined as $\mathbf{s}_t^j = [\mathbf{s}_{rt}, \mathbf{s}_{h_o,t}^1, \mathbf{s}_{h_o,t}^2, \dots, \mathbf{s}_{h_o,t}^n]$.

The robot is deemed to adjust its velocity \mathbf{v}_{rt} promptly in light of the action instruction \mathbf{a}_t . It is expected to find an optimal policy π^* to maximize the expected reward:

$$\begin{aligned} \pi^*(\mathbf{s}_t^j) &= \operatorname{argmax}_{\mathbf{a}_t \in \mathbf{A}} R(\mathbf{s}_t^j, \mathbf{a}_t) + \gamma^{\Delta t \cdot v_{pref}} \\ &\int_{\mathbf{s}_{t+\Delta t}^j} P(\mathbf{s}_t^j, \mathbf{a}_t, \mathbf{s}_{t+\Delta t}^j) V^*(\mathbf{s}_{t+\Delta t}^j) d\mathbf{s}_{t+\Delta t}^j, \\ V^*(\mathbf{s}_t^j) &= \sum_{t^* = t}^T \gamma^{t^* \cdot v_{pref}} R_{t^*}(\mathbf{s}_{t^*}^j, \pi^*(\mathbf{s}_{t^*}^j)), \end{aligned} \tag{2}$$

where $R(\mathbf{s}_t^j, \mathbf{a}_t)$ is the reward received at time t , \mathbf{A} is the action space, $\gamma \in (0, 1)$ is a discount factor, V^* is the optimal value function, and $P(\mathbf{s}_t^j, \mathbf{a}_t, \mathbf{s}_{t+\Delta t}^j)$ is the transition probability from time t to $t + \Delta t$. The preferred velocity v_{pref} is used as a normalization term in the discount factor for numerical reasons.

Algorithm 1. Deep V-learning

- 1: **Input:** demonstration training set D
 - 2: **Output:** value network V
 - 3: Initialize value network V with demonstration D
 - 4: Initialize target value network $\hat{V} \leftarrow V$
 - 5: Initialize experience replay memory $E \leftarrow D$
 - 6: **for** $episode = 1, \dots, N$ **do**
 - 7: Initialize random joint state \mathbf{s}_0^{in}
 - 8: **repeat**
 - 9: With probability ϵ select random action \mathbf{a}_t , otherwise $\mathbf{a}_t = \arg \max_{\mathbf{a}_t \in \mathbf{A}} R(\mathbf{s}_t^{in}, \mathbf{a}_t) + \bar{\gamma} V(\mathbf{s}_{t+\Delta t}^{in})$,
 where $\bar{\gamma} = \gamma^{\Delta t \cdot v_{pref}}$, $\mathbf{s}_{t+\Delta t}^{in} = \text{propagate}(\mathbf{s}_t^{in}, \mathbf{a}_t)$
 - 10: Store transition $(\mathbf{s}_t^{in}, \mathbf{a}_t, r_t, \mathbf{s}_{t+\Delta t}^{in})$ in E
 - 11: Sample random minibatch of transitions $(\mathbf{s}_j^{in}, \mathbf{a}_j, r_j, \mathbf{s}_{j+1}^{in})$ from E
 - 12: Set TD target $y_j = r_j + \gamma^{\Delta t \cdot v_{pref}} \hat{V}(\mathbf{s}_{j+1}^{in})$
 - 13: Update value network V by gradient descent
 - 14: **until** terminal state $\mathbf{s}_t^{in} = \mathbf{s}_{t_{end}}^{in}$ or $t \geq t_{max}$
 - 15: Update target network $\hat{V} \leftarrow V$
 - 16: **end for**
 - 17: **return** V
-

2.3. Training strategy of the value network

The training procedure of the value network is outlined in Algorithm 1. At first, the value network model is trained with imitation learning using demonstration knowledge generated by ORCA (line 3-5) and subsequently trained by temporal-difference (TD) learning method with standard experience replay and fixed target network techniques (line 6-16) [41]. In line 9, the ϵ -greedy strategy is applied to select actions. The next state $\mathbf{s}_{t+\Delta t}^{in}$ is obtained by querying the true value from the environment to mitigate the issue of system dynamics in training. During deployment, we use a simple linear model to predict the motion of the agents in Δt , which has shown good accuracy on small time scales [42].

To tackle the problem (2), the value network model needs to approximate the optimal value function V^* to maximize the expected reward. The reward design is critical for achieving safe and socially compliant robot navigation [43]. Previous work on this track has not fully considered the risk in densely populated environments. In this paper, based on collision theory and real-time relative movements, we present Risk-Areas around each human as a safety measure.

3. Methodology

We outline the central idea of the Risk-Area in this section. First, we examine the human behavior and propose an impulse model of the human-robot collision. Then, we give the precise formulation and geometry of Risk-Area and discuss how the Risk-Area guides the robot to navigate in a safe and comfortable way.

3.1. Characteristic of the human and robot motion

Understanding the motions of both humans and robots is one of the bases of Risk-Area’s design. Humans’ navigation manners are closely related to a series of parameters, such as age, mood, and weight. According to previous work, the average walking speed of pedestrians is about 1.4 m/s [38], and in some situations might go up to 2.5 m/s [40]. In comparison, robots navigate with human-provided strategies,

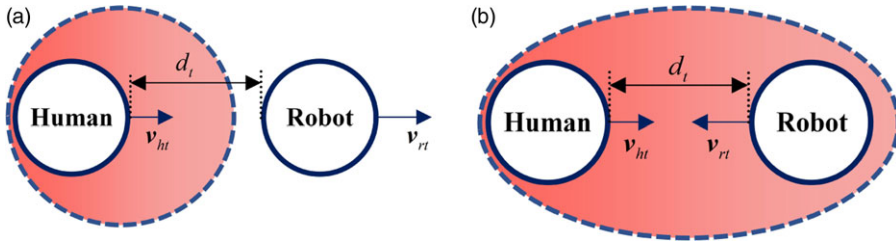


Figure 2. Two typical navigation scenarios. (a) A low-risk situation where the robot and the pedestrian move in the same direction. (b) A high-risk situation where the robot and the pedestrian move face to face.

and the kinematic patterns of robots can be artificially determined. From the perspective of psychological effect, the approaching speed of mobile robots exceeding 1 m/s causes obvious discomfort to humans [1,39]. Moreover, the maximum speed of the common indoor mobile robots such as turtlebot2 [44] and PR2 [45] is usually within 1 m/s. Due to the inertia, both robots and humans tend to stay in their original motion and posing more threats in this direction. It is not enough to model the risk only with the individual state. For example, in Fig. 2, the human state in (a) and (b) is the same while the collision risk is quite different. Therefore, the design of the collision risk should give more consideration to the relative movement of humans and robots.

3.2. Impulse model of the human-robot collision

The reward of the navigation system should reflect the aim of achieving safe and socially compliant robot navigation [43]. Apart from the reaching target reward, the robot’s collision and other risky behaviors should be punished. Previous methods in refs. [19] and [35] give every collision the same penalty without considering the relative movement of the agents, which does not correspond with the actual situation and influences the robot to judge the importance of different pedestrians. To address the issue, we take collision theory [46] and the relative motion among agents into account and design Risk-Area. After considering the mass and speed of common mobile robots and humans, the case of the robot colliding with the human in-plane is mainly discussed in this paper. Some typical collision assumptions [47] are made as follows:

Assumption 1. The duration of contact is enough short and the interaction forces are high.

Assumption 2. The effects of friction and other forces can be omitted.

Assumption 3. The masses of the human and the robot during the collision are constant.

The collision response contains the compression and expansion phases, as shown in Fig. 3. At the beginning of the compression phase, the bodies of the robot and the human start contacting and have a positive approaching speed v_a to compress each other. After compressing, the approaching speed v_a becomes zero and the velocity of both the human and the robot in the direction of the line of impact is the same. A period of expansion is followed during which the bodies of the human and the robot try to regain their original shape. At the end of the expansion phase, the bodies of the human and the robot separate with a negative approaching speed v_a . The collision response has the relationship as follows:

$$\begin{bmatrix} m_r & m_r \\ m_h & m_h \\ m_r & m_r \\ m_h & m_h \end{bmatrix} \begin{bmatrix} \mathbf{v}_n & \mathbf{v}_n & \mathbf{v}_{rn\tau} & \mathbf{v}_{hn\tau} \\ -\mathbf{v}_m & -\mathbf{v}_{hm} & -\mathbf{v}_n & -\mathbf{v}_n \end{bmatrix} = \begin{bmatrix} I_1 \\ -I_1 \\ I_2 \\ -I_2 \end{bmatrix}, \tag{3}$$

$$e = \frac{I_2}{I_1}, I = I_1 + I_2,$$

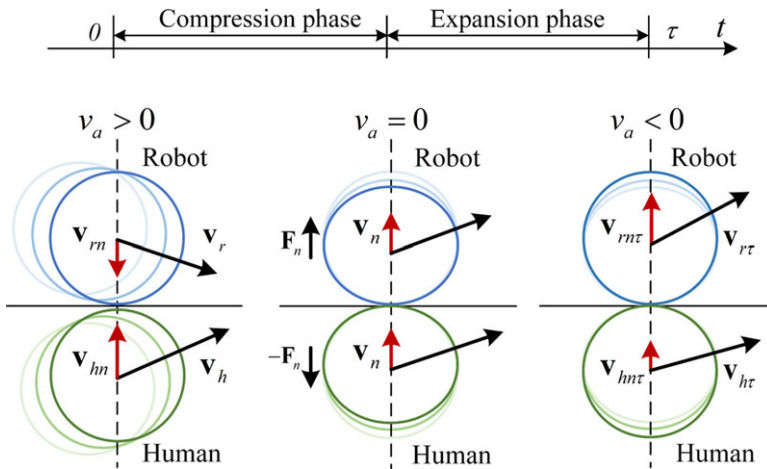


Figure 3. The collision response between the mobile robot and the human in-plane with $m_h \gg m_r$. The robot collides with the human and then is bounced away.

where m_r and m_h are the masses of the robot and the human, \mathbf{v}_{rm} and \mathbf{v}_{hm} are the normal velocity of the robot and the human at the beginning of the compression phase, \mathbf{v}_n is the normal velocity of the human and the robot at the end of the compression phase, $\mathbf{v}_{rm\tau}$ and $\mathbf{v}_{hm\tau}$ are the normal velocity of the human and the robot at the end of the expansion phase, I_1 and I_2 are the impulses in the compression phase and expansion phase, and e is the coefficient of restitution with the range $e \in [0, 1]$, which is related to the materials. The impulse I in the collision can be represented as follows:

$$I = (1 + e) \frac{m_r m_h (\|\mathbf{v}_{rm} - \mathbf{v}_{hm}\|)}{m_r + m_h} \tag{4}$$

Formula (4) reveals that the collision impulse is proportional to the relative approaching speed $v_a = \|\mathbf{v}_{rm} - \mathbf{v}_{hm}\|$. From the perspective of reducing collision loss, the robot should avoid approaching the humans fast.

3.3. Penalty formulation of the risk-area

Due to the influence of sensor accuracy, control error, and other factors, robots cause high collision risk when they get too close to humans. Thus, we combined the position penalty P_t^p into the design of the Risk-Area with the formulation as follows:

$$P_t^p(\mathbf{s}_t^j, \mathbf{a}_t) = \begin{cases} 0.1 & \text{if } d_m < 0, \\ 0.1(1 - \frac{d_m}{D_p}) & \text{else if } d_m < D_p, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where d_m is the minimum separation distance between the robot and the human within a duration $[t - \Delta t, t]$, and D_p is a threshold distance, which is set to 0.2 m.

According to the analysis in section III-B, the impulse of the collision is a linear function with respect to the human-robot relative approaching speed. Besides, fast approaching speed also causes discomfort to the human [39]. Therefore, the relative approaching speed of the robot and the pedestrian is applied as the velocity penalty of the Risk-Area to reflect the risk in the environment. Let \mathbf{v}_{ht} denote the velocity of the human at time t . In practice, the relative approaching speed v_{at} at time t can be calculated as follows:

$$v_{at} = \frac{\mathbf{p}_{ht} - \mathbf{p}_{rt}}{\|\mathbf{p}_{ht} - \mathbf{p}_{rt}\|} \cdot (\mathbf{v}_{rt} - \mathbf{v}_{ht}), \tag{6}$$

where \mathbf{p}_{rt} and \mathbf{p}_{ht} denote the position of the robot and human at time t . When the time scale Δt is small enough, we can assume the velocity of the robot and the human is constant. The form of the velocity penalty P_t^v during the time period $[t - \Delta t, t]$ is defined as follows:

$$P_t^v(\mathbf{s}_t^{jn}, \mathbf{a}_t) = \begin{cases} \frac{0.1 \cdot v_{at}}{v_{rmax} + v_{hmax}} & \text{if } v_{at} > 0 \text{ and } d_t < D_v, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where v_{rmax} and v_{hmax} are the maximum speed of the robot and the human, respectively, d_t is the separation distance between the robot and the human at time t , and D_v is a threshold distance with the form as follows:

$$D_v = v_{at} \cdot m_v + D_p, \tag{8}$$

where m_v is an adjustable parameter related to the agents' motion, which is set to 0.35 in this paper. The Risk-Area penalty function is the sum of the velocity penalty and the position penalty in the following form:

$$P_t(\mathbf{s}_t^{jn}, \mathbf{a}_t) = P_t^p(\mathbf{s}_t^{jn}, \mathbf{a}_t) + P_t^v(\mathbf{s}_t^{jn}, \mathbf{a}_t). \tag{9}$$

When navigating in the crowd, the robot will calculate Risk-Areas of the neighbors and try to avoid them. The penalty term of the reward function has been given in (9). Besides, the robot will be awarded when reaching the goal within a limited time. The final reward function has the following form:

$$R_t(\mathbf{s}_t^{jn}, \mathbf{a}_t) = \begin{cases} 1 & \text{if } \mathbf{p}_{rt} = \mathbf{g}_r, \\ -P_t(\mathbf{s}_t^{jn}, \mathbf{a}_t) & \text{else if Risk-Area,} \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

3.4. Geometry of the risk-area

Since the Risk-Area penalty function is composed of position penalty and velocity penalty, the geometry of the Risk-Area is also divided into two parts: position Risk-Area and velocity Risk-Area. With the center of the human as the pole and the moving direction of the pedestrian relative to the robot as the polar coordinate axis, the polar coordinate system is established, as shown in Fig. 4(a). The geometry of the position Risk-Area is a regular circle of radius $r_p = r_h + D_p$. The boundary point $P = [\rho, \theta]$ of the velocity Risk-Area needs to meet the conditions in (7) and has the formula as follows:

$$\begin{aligned} \rho &= m_v \cdot \|\mathbf{v}_{hrt}\| \cdot \cos \theta + D_p + r_h, \\ -\frac{\pi}{2} &< \theta < \frac{\pi}{2}, \end{aligned} \tag{11}$$

where $\mathbf{v}_{hrt} = \mathbf{v}_{ht} - \mathbf{v}_{rt}$ is the velocity of the human relative to the robot at time t , θ is the angle between OP and the reference axis, and ρ is the boundary distance of velocity Risk-Area in the direction of θ . Based on the penalty functions in (5) and (7), the contours of the Risk-Area are shown in Fig. 4(b).

Considering the flexible human motion ability and uncertainties in the environment, we do not narrow the scope of the Risk-Area as the relative speed increases. Furthermore, the velocity Risk-Area leaves a large penalty space in the direction of the pedestrian relative to the robot so that it is more likely to receive a penalty when approaching the human fast. When the robot moves into a Risk-Area, it will get punished. For example, in Fig. 4(c), Robot 1 is punished by position Risk-Area, Robot 2 and Robot 3 are not punished, and Robot 4 is punished by velocity Risk-Area. By introducing relative approaching velocity, the Risk-Area is able to perceive risk in the environment and navigate the robot safely and efficiently.

Proposition 3.1. *Defining the direction of the Risk-Area as the unit vector $\mathbf{n} = \frac{\mathbf{v}_{hrt}}{\|\mathbf{v}_{hrt}\|}$, then the impulse of the collision is always maximized in the direction of the Risk-Area.*

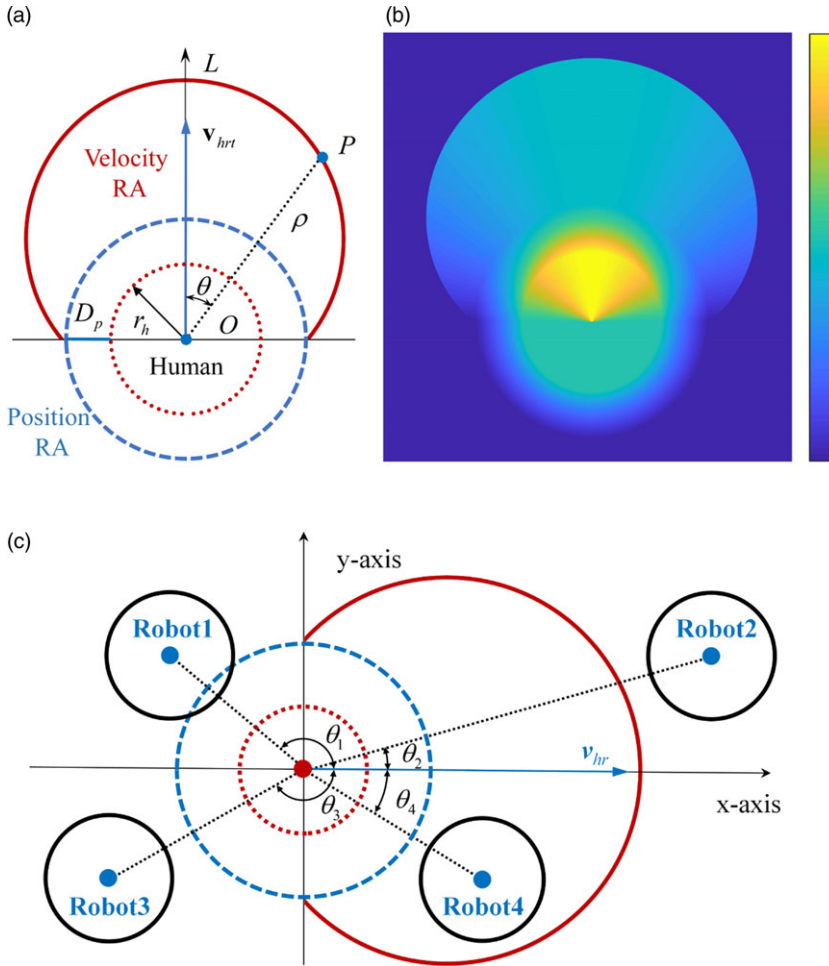


Figure 4. Given the relative velocity of the robot and the human, the geometric representation of the Risk-Area. (a) Geometry of Risk-Area. (b) Contours of Risk-Area. (c) The punishment of the robots at different positions.

Proof. Consider $\mathbf{p}_r = [l, \varphi] \in \mathcal{S}_{RA}$, $\varphi \neq 0$ with collision happen. According to (4), the impulse I_{pr} has the following form:

$$I_{pr} = (1 + e) \frac{m_r m_h \|\mathbf{v}_{hrt}\| \cos\varphi}{m_r + m_h} = k \|\mathbf{v}_{hrt}\| \cos\varphi, \tag{12}$$

where k is a positive constant. If the collision happens in the Risk-Area direction, the impulse is $I_n = k \|\mathbf{v}_{hrt}\| > I_{pr}$. This result implies that the direction of the Risk-Area indicates the maximum collision loss. \square

Proposition 3.2. In any direction of the velocity Risk-Area, the maximum boundary distance is proportional to the impulse of the collision.

Proof. Consider point $B = [l_B, \gamma] \in \mathcal{S}_{vRA}$ is on the boundary of the velocity Risk-Area. According to (4), the impulse of the collision is $I_B = k \|\mathbf{v}_{hrt}\| \cos\gamma$. The boundary distance $l_B = m_v \|\mathbf{v}_{hrt}\| \cos\gamma + D_p + r_h$ satisfies the condition in (11). Then, l_B can be represented by I_B as $l_B = \frac{m_v}{k} I_B + (D_p + r_h)$, where $\frac{m_v}{k}$ is constant. This result reveals that the robot motion with the risk of high collision loss is preferentially punished in the Risk-Area. \square

Proposition 3.3. Assuming the motion of the agent within a time interval Δt can be approximated as constant velocity and $\Delta t < m_v$, it is a sufficient condition for safe robot navigation that the robot does not enter the Risk-Area at any control moment $t_k \in \{t_0, t_1, \dots, t_{end}\}$.

Proof. For any given control moment t_k and any pedestrian i , the position of the robot satisfies $\mathbf{p}_{r_{t_k}} \notin \mathcal{S}_{RA}$. Therefore, the distance between the robot and pedestrian i fulfills the condition $d_{t_k} > D_v \geq D_p > 0$, indicating that a collision does not occur at time t_k .

For any time within the control interval $\forall T \in (t_k, t_{k+1})$, the following holds. When $v_{at_k} \leq 0$, as shown by Robot 3 in Fig. 4(c), we have $d_T \geq d_{t_k} > 0$, indicating the absence of a collision at time T . On the other hand, when $v_{at_k} > 0$, as shown by Robot 2 in Fig. 4(c), with the progression of the approach motion, the angle θ becomes larger or remains 0, and the approaching speed $v_{at} = v_{hrT} \cos \theta$ reaches its maximum at t_k within the interval (t_k, t_{k+1}) . Then, the distance between the robot and pedestrian i at time T satisfies $d_T \geq d_{t_k} - v_{at_k}(T - t_k) > v_{at_k}m_v + D_p - v_{at_k}\Delta t > 0$, indicating the absence of a collision at time T . This conclusion demonstrates that the Risk-Area ensures the safety of navigation. \square

Proposition 3.4. In the Risk-Area, the gradient of the velocity penalty function with respect to the position of the robot is always perpendicular to that of the position penalty function.

Proof. When the robot enters the Risk-Area without collision at time t , the position penalty P_t^p and velocity penalty function P_t^v can be expressed by \mathbf{p}_{rt} and \mathbf{p}_{ht} as follows:

$$P_t^p = c_p \left(1 - \frac{1}{D_p} (\|\mathbf{p}_{ht} - \mathbf{p}_{rt}\| - r - r_h) \right), \tag{13}$$

$$P_t^v = c_v \left(\frac{\mathbf{p}_{ht} - \mathbf{p}_{rt}}{\|\mathbf{p}_{ht} - \mathbf{p}_{rt}\|} \cdot (\mathbf{v}_{rt} - \mathbf{v}_{ht}) \right), \tag{14}$$

where c_p and c_v are positive constants. For each human, we assume that the position is static relative to the world coordinate system so as to analyze the influence of the robot’s position change on the risk penalty. Then, the partial derivative of P_t^p and P_t^v with respect to \mathbf{p}_{rt} are as follows:

$$\nabla P_t^p = \frac{c_p(\mathbf{p}_{ht} - \mathbf{p}_{rt})}{D_p \|\mathbf{p}_{ht} - \mathbf{p}_{rt}\|}, \tag{15}$$

$$\begin{aligned} \nabla P_t^v &= \frac{c_v(\mathbf{v}_{at} - (\mathbf{v}_{rt} - \mathbf{v}_{ht}))}{\|\mathbf{p}_{ht} - \mathbf{p}_{rt}\|} \\ &= \frac{-c_v \mathbf{v}_{at\perp}}{\|\mathbf{p}_{ht} - \mathbf{p}_{rt}\|}, \end{aligned} \tag{16}$$

where $\mathbf{v}_{at\perp}$ denotes the component of velocity \mathbf{v}_{rt} perpendicular to \mathbf{v}_{at} . It is obvious that ∇P_t^p has the same direction with \mathbf{v}_{at} . The derivative ∇P_t^v is perpendicular to ∇P_t^p . \square

According to (2), the collision avoidance policy $\pi^*(\mathbf{s}_t^{in})$ of the robot is inevitably affected by the current reward $R_t(\mathbf{s}_t^{in}, \mathbf{a}_t)$. The robot tends to move towards the direction of fast attenuation of punishment to maximum the final reward. Formula (16) shows that the position penalty decays most quickly in the direction of $-\nabla P_t^p$, which encourages the robot to move away from the human. However, the direction of $-\nabla P_t^p$ is deviating from the target in some cases as shown in Fig. 5. In comparison, the gradient $-\nabla P_t^v$ is perpendicular to that of the position penalty, which encourages the robot to bypass the human effectively.

4. Experiments and results

4.1. Simulation environment setup

The simulation environment is built on Python with the PyTorch library. Two existing methods, position-based reward function [33] and DZ reward function [19], are implemented as baselines. First, we use 3 k episodes of ORCA demonstrations to preprocess the model with imitation learning. Then, the model

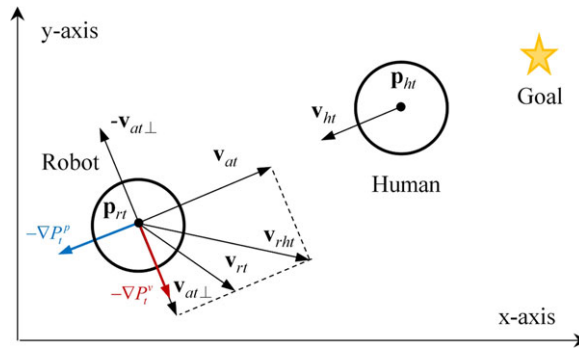


Figure 5. A situation in which a robot is punished for being in a Risk-Area.

is trained 10 k episodes with the RL method. The RL parameters include a learning rate of 0.001 and a discount factor γ of 0.9. All the agents in the simulation are assumed as circles with a diameter of 0.6 m. The minimum and maximum speed of pedestrians during crossovers is 0.5 m/s and 2.5 m/s [40, 48]. Still, after considering safety, we set the maximum speed of humans as 3 m/s in our experiments. To fully evaluate the effectiveness of the proposed method, we look into two robot settings: holonomic robot setting and non-holonomic robot setting, along with three distinct pedestrian crossing scenarios: circle crossing scenario, square crossing scenario, and group crossing scenario. In the non-holonomic robot setting, the robot’s action space is allowed in 16 directions evenly spaced between 0 to $\frac{\pi}{2}$ with five exponentially spaced speeds between 0 to 1.0 m/s resulting in 80 actions. The humans in the simulation navigate following ORCA policy. In the holonomic robot setting, the robot can move in 16 directions evenly spaced between 0 to 2π with five exponentially spaced speeds between 0 to 1.0 m/s. In the circle crossing scenario, the initial positions of the humans are on a circle of radius 6 m, and the goals of the humans are set exactly opposite to the initial position. In the square crossing scenario, the initial positions of humans are set along any pair of opposite sides of a square with a side length of 6 m, while their destinations are generated randomly on the opposite side of the initial positions. In the group crossing scenario, several static groups of 2-5 individuals are generated within the environment, while the remaining pedestrians are generated to engage in circle crossing motion. In the first two crowd scenarios, the number of humans is set to 5, while in the group crossing scenario, the number of humans ranges from 5 to 10. In the square crossing scenario, the radius of humans is set between 0.3 m and 0.5 m, while in the other crossing scenarios, the radius of humans is uniformly set to 0.3 m. The robot in all settings is set invisible to the human.

We employ 10 k sets of pedestrian trajectories generated by the ORCA as training data, similar to refs. [19] and [36]. These datasets encompass detailed information, such as pedestrian positions, velocities, and target destinations, ensuring the effective learning and inference capabilities of our models. Additionally, we subjected the robot to testing in 500 different scenarios. Performance measures include Success Rate, Collision Rate, Navigation Time, Danger Frequency, and Minimum Separation Distance. *Success Rate* (SR) is the ratio of the number of experiments that the robot reaches its goal to the number of total experiments. *Collision Rate* (CR) is the ratio of the number of experiments that the robot collides with humans to the number of total experiments. *Navigation Time* (NT) is the average time for the robot to reach its goal. *Danger Frequency* (DF) is the ratio of the number of times that the robot gets too close to the human ($d_m < 0.2m$) to the total time of navigation. *Minimum Separation Distance* (MSD) is the average minimum separation distance when the robot gets too close to the human.

4.2. Performance comparison with different RL methods

The safety of the navigation process is the most important in evaluating the navigation performance of the robot in crowded environments. To test the performance of different reward functions, we select

Table I. Navigation results of non-holonomic robot in three different crossing scenarios. The cell data represent the experimental results in circle/square/group crossing settings, respectively.

Method	SR	DF	MSD	CR	NT
CADRL	0.78/0.71/0.80	0.38/0.35/0.28	0.09/0.08/0.11	0.22/0.28/0.20	12.80/12.50/12.31
CADRL-DZ	0.82/0.73/0.78	0.18/0.22/0.21	0.07/0.12/0.10	0.16/0.24/0.11	14.36/14.20/14.09
CADRL-RA	0.88/0.85/0.90	0.12/0.12/0.08	0.14/0.15/0.17	0.11/0.15/0.07	13.04/12.84/12.59
LSTM_RL	0.74/0.57/0.63	0.25/0.32/0.31	0.10/0.07/0.05	0.25/0.37/0.23	14.90/14.11/15.72
LSTM_RL-DZ	0.76/0.66/0.71	0.25/0.27/0.14	0.06/0.08/0.11	0.24/0.34/0.16	14.14/13.82/16.74
LSTM_RL-RA	0.82/0.73/0.80	0.15/0.14/0.08	0.15/0.15/0.19	0.17/0.26/0.08	13.95/13.63/13.85
SARL	0.90/0.81/0.83	0.16/0.22/0.21	0.09/0.07/0.04	0.10/0.16/0.09	14.81/14.70/16.60
SARL-DZ	0.94/0.88/0.85	0.07/0.11/0.03	0.11/0.07/0.08	0.05/0.09/0.06	14.55/14.14/15.13
SARL-RA	0.97/0.93/0.92	0.04/0.06/0.01	0.14/0.15/0.15	0.03/0.05/0.04	14.00/13.49/13.52

three existing well-known deep reinforcement learning methods, including collision avoidance with deep reinforcement learning (CADRL) [33], long short-term memory with reinforcement learning (LSTM_RL) [35], and social attention with reinforcement learning (SARL) [36] as the base training models. All the reinforcement learning models are trained with four different reward functions: position-based reward function, DZ reward function, RVO reward function, and RA reward function proposed in this paper. To facilitate a comprehensive comparison of the model performance, we also test the ORCA algorithm in the experiment. However, it should be noted that the original ORCA is designed for holonomic robots. In this paper, we conduct the ORCA experiment within the holonomic robot setting to maintain consistency with its original scope.

The experiment results of the non-holonomic robot are listed in Table I. The bold data indicate the best results. In the three reinforcement learning models, SARL performs best in both safety and task accomplishment by introducing the attention mechanism and aggregating multi-agent information. From the perspective of reward functions, the models trained with the position-based reward function show poor safety and task accomplish capability because the velocity between the robot and the human is neglected. As a result of taking pedestrian actions into account, the DZ method achieves a better safety performance than the position-based reward function in terms of danger frequency and collision rates. Compared with the previous methods, the Risk-Area incorporates the relative motion of agents into the design of the Risk-Area, and the models combined with the proposed reward function achieve superior performance in terms of safety and task accomplishments. In particular, SARL-RA achieves the best performance of all methods. The collision rate and the danger frequency are the lowest, and the success rate and minimum separation distance are significantly improved compared with that of the previous methods. Moreover, the navigation time listed in the table shows the model trained by the proposed reward function navigates the robot in an efficient way. The experimental results obtained from three different crowd settings are similar, and our method consistently achieves better results in both task accomplishment and safety metrics.

The navigation performance in the holonomic robot setting is shown in Table II. Due to the fact that the robots in the environment are set invisible to pedestrians, the ORCA algorithm violates the reciprocal assumption, resulting in poor navigation performance. The success rate and safety metrics in various scenarios are lower than those of reinforcement learning methods. The reinforcement learning model SARL still outperforms CADRL and LSTM_RL in both security and task accomplishment. Furthermore, all the three deep reinforcement learning models trained with the proposed reward function have better performance. In the three different crowd scenarios, the average success rate of CADRL-RA is 12.3% and 9.0% higher than that of CADRL and CADRL-DZ, respectively. Moreover, the average minimum separation distance and collision rate of CADRL-RA are improved compared to that of the previous reward functions. For LSTM_RL, the model trained with the proposed reward function also achieves higher

Table II. Navigation results of holonomic robot in three different crossing scenarios. The cell data represent the experimental results in circle/square/group crossing settings, respectively.

Method	SR	DF	MSD	CR	NT
ORCA	0.77/0.65/0.78	0.57/0.54/0.46	0.08/0.09/0.08	0.23/0.35/0.22	12.17/12.65/12.20
CADRL	0.85/0.76/0.86	0.47/0.44/0.34	0.10/0.10/0.12	0.15/0.24/0.10	13.02/12.71/12.60
CADRL-DZ	0.88/0.81/0.88	0.17/0.10/0.15	0.06/0.12/0.11	0.07/0.12/0.10	16.88/16.08/14.71
CADRL-RA	0.95/0.94/0.95	0.10/0.09/0.07	0.16/0.17/0.16	0.05/0.05/0.04	13.43/13.30/12.74
LSTM_RL	0.84/0.81/0.79	0.29/0.26/0.20	0.08/0.08/0.09	0.11/0.12/0.11	14.58/14.15/14.61
LSTM_RL-DZ	0.89/0.83/0.81	0.27/0.24/0.16	0.06/0.07/0.07	0.08/0.12/0.13	15.00/14.32/14.06
LSTM_RL-RA	0.94/0.87/0.93	0.07/0.06/0.05	0.16/0.16/0.17	0.05/0.11/0.05	13.84/13.56/13.64
SARL	0.95/0.88/0.90	0.27/0.29/0.28	0.09/0.07/0.04	0.05/0.12/0.07	14.36/14.66/15.04
SARL-DZ	0.96/0.93/0.91	0.09/0.12/0.04	0.09/0.03/0.10	0.04/0.04/0.05	14.11/13.64/13.95
SARL-RA	0.99/0.97/0.99	0.05/0.03/0.01	0.16/0.14/0.19	0.01/0.02/0.01	13.35/12.98/13.02

safety performance while maintaining less navigation time than the previous methods. As for SARL, SARL-RA reaches the best performance among all the algorithms by achieving the highest safety and task accomplishment. The success rate of SARL-RA is higher than that of SARL and SARL-DZ, and the danger frequency is significantly decreased. In the group crossing scenarios, the danger frequency of SARL-RA achieves 0.01, which are the lowest among all models. In general, our approach demonstrates superior performance compared to the state-of-the-art method, Danger Zone, across various metrics. It achieves an average increase of 7.2% in success rate, while reducing the collision rate by 4.7% and the danger frequency by 52.4% in all the experiments. These results confirm the effectiveness of our approach in mitigating collision risks and enhancing overall navigation performance.

Fig. 6 shows the non-holonomic robot navigation results of the three reward functions based on the SARL algorithm in three different crossing scenarios. For fairness, the test cases in the same crossing scenario have the same human trajectories. In Fig. 6(a), SARL adopts aggressive avoidance behavior and attempts to move in front of Human 2, leading to the dangerous situation at 8.75 s. Similar situations also arise in Fig. 6(d) and Fig. 6(g), where SARL exhibits risky navigation behavior during interactions with mobile humans. DZ method pays too much attention to the forward direction of the fast-moving pedestrians while ignoring the possible risk around. In Fig. 6(b), the robot following SARL-DZ tries to move behind Human 2 but gets too close to the human side. Moreover, the DZ method converges to the original position-based penalty when dealing with interactions with stationary pedestrians, leading to situations where the robot gets too close to static pedestrians, as illustrated in Fig. 6(h). In comparison, SARL-RA selects an appropriate collision avoidance strategy and keeps a safe distance from all humans. It is because our approach establishes Risk-Area based on the collision theory and relative movement of the robot and pedestrians while taking into account all directions of human-robot interaction. According to (16), the introduction of relative speed in the Risk-Area also contributes the robot to learning effective collision avoidance actions, thus ensuring navigation efficiency. Experiment results reveal that the proposed reward function of the Risk-Area along with the SARL algorithm exhibits safe and socially compliant navigation.

4.3. Performance comparison in different dense crowds

In actual cases, the number of humans in the environment is changeable. To test the safety of the model, we further compare the navigation performance of our reward function with the baselines in the different dense crowds. The experiments are conducted based on the models with the best performance in Section 4.2, including SARL, SARL-DZ, and SARL-RA. The human number of the crowd is set to $N \in [7, 9, 11, 13, 15]$. In the non-holonomic robot setting, with the crowd density increasing, all three

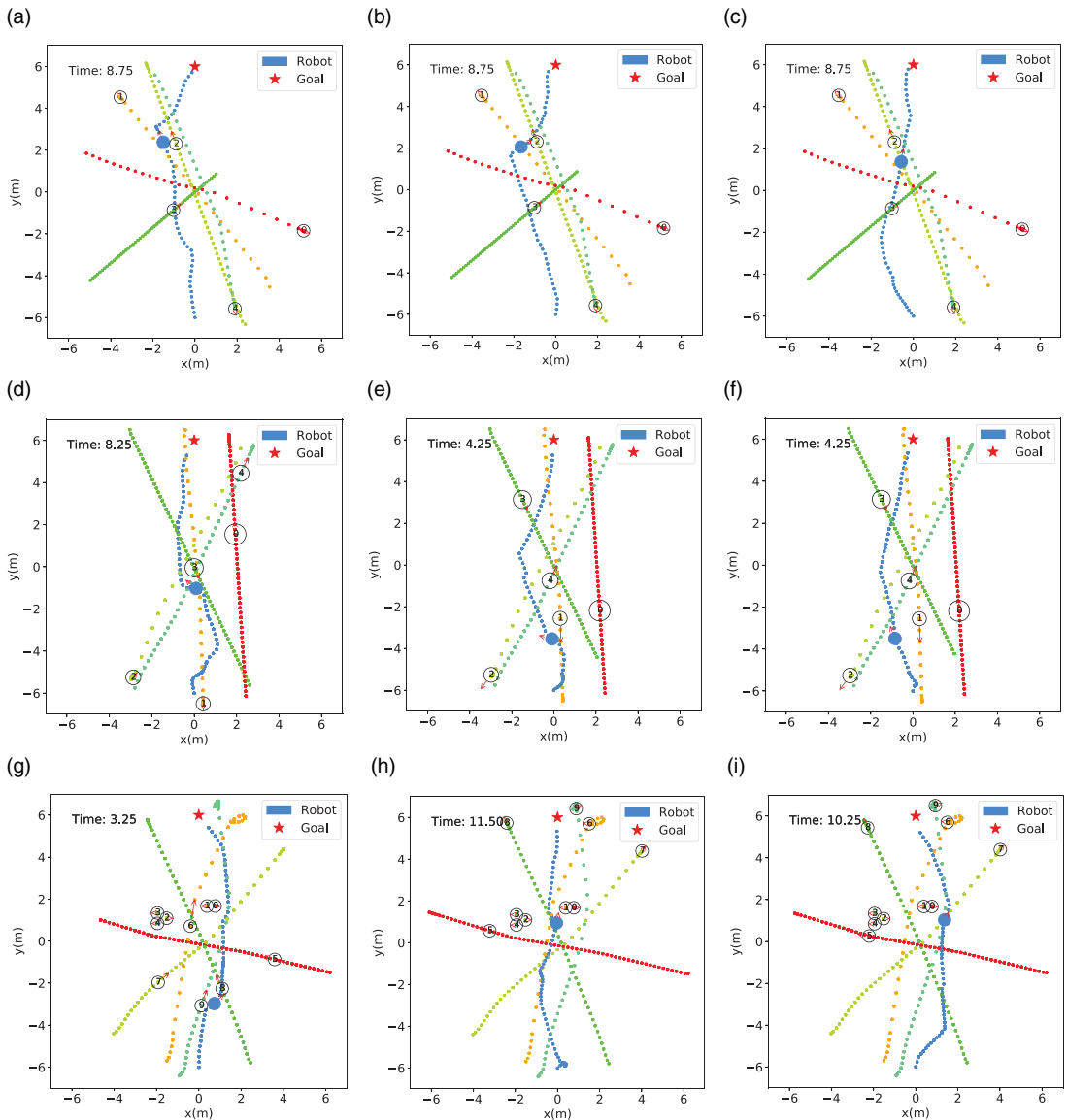


Figure 6. Comparison of the three different reward functions based on SARL in three different crossing settings. The hollow circles represent the pedestrians, and the red arrows indicate the velocities of the agents. (a) SARL(circle). (b) SARL-DZ(circle). (c) SARL-RA(circle). (d) SARL(square). (e) SARL-DZ(square). (f) SARL-RA(square). (g) SARL(group). (h) SARL-DZ(group). (i) SARL-RA(group).

models' task accomplishment and safety decrease, as shown in Fig. 7(a)-(c). The SARL trained with the position-based reward function has the fastest performance degradation. SARL-DZ achieves a better task accomplishment in the different dense crowds than SARL. In comparison, SARL-RA achieves the best performance, maintaining the highest success rate and minimum separation distance. The danger frequency of SARL-RA is also the lowest in all the scenarios. Experiment results show that the proposed reward function improves the safety of the model in different crowd density scenarios.

The experiment results in the holonomic robot setting are shown in Fig. 7(d)-(f). Similar to the non-holonomic robot setting, the performance of SARL degrades fast as the human number in the environment increases. The success rate of SARL is only 62% when the human number reaches 15.

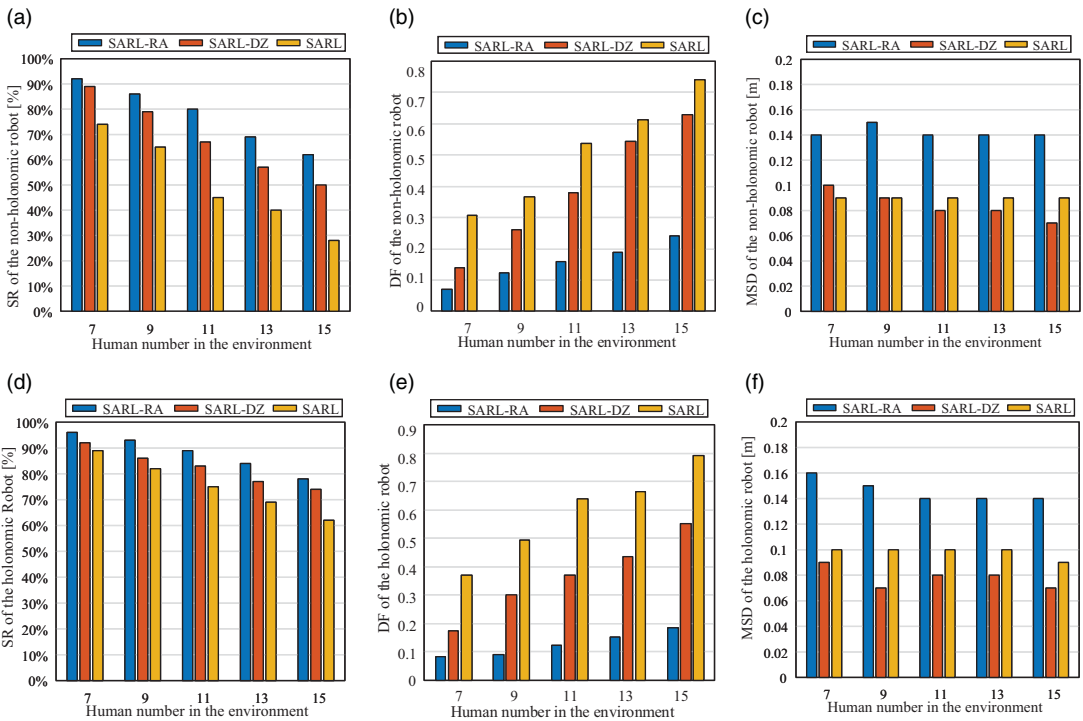


Figure 7. The navigation performance of SARL, SARL-DZ, and SARL-RA in the holonomic and non-holonomic robot settings with different crowd dense.

Benefiting from the consideration of human behaviors, SARL-DZ achieves a lower danger frequency than SARL. However, the minimum separation distance is not ideal due to the neglect of the relative movement among agents. It can be seen from the experiment that SARL-RA is least affected by the increase in crowd density. The model trained with our approach has the best performance in both success rate and safety. The experiment results reveal that the reward function is noteworthy to the navigation performance of the model. Moreover, by introducing relative movement and collision theory to represent the risk in the environment correctly, the model generalizes well in different dense crowd navigation scenarios.

Fig. 8 visualizes some typical results in holonomic robot setting for all the three models, including SARL, SARL-DZ, and SARL-RA. The methods with risk perception mechanisms have more efficient avoidance behaviors and reduce the freezing robot problem. The avoidance behaviors of SARL are aggressive, with a high frequency of danger occurring during traveling, as shown in Fig. 8(a) and (b). Moreover, in Fig. 8(c)-(d), SARL suffers from the freezing robot problem seriously, and the path has multiple diversions, which wastes time and causes more dangerous situations during navigation. Benefitting from the prediction of the human behaviors, SARL-DZ ameliorates the freezing robot problem, but unnatural and risky behaviors still occur during collision avoidance, as shown in Fig. 8(e). In contrast, SARL-RA makes the robot take clear paths to ensure safety and comfort instead of entering the human cluster since it considers the risk of all humans around based on the relative movement. In general, the proposed method, combined with the collective importance of the crowd of SARL, is able to avoid the freezing robot problem and has a better generalization ability in different dense crowd scenarios.

4.4. Real-world experiment

Besides the simulation experiments above, we also conduct real-world experiments to verify the effectiveness of the proposed method. We implement the SARL-RA policy with the best performance in the

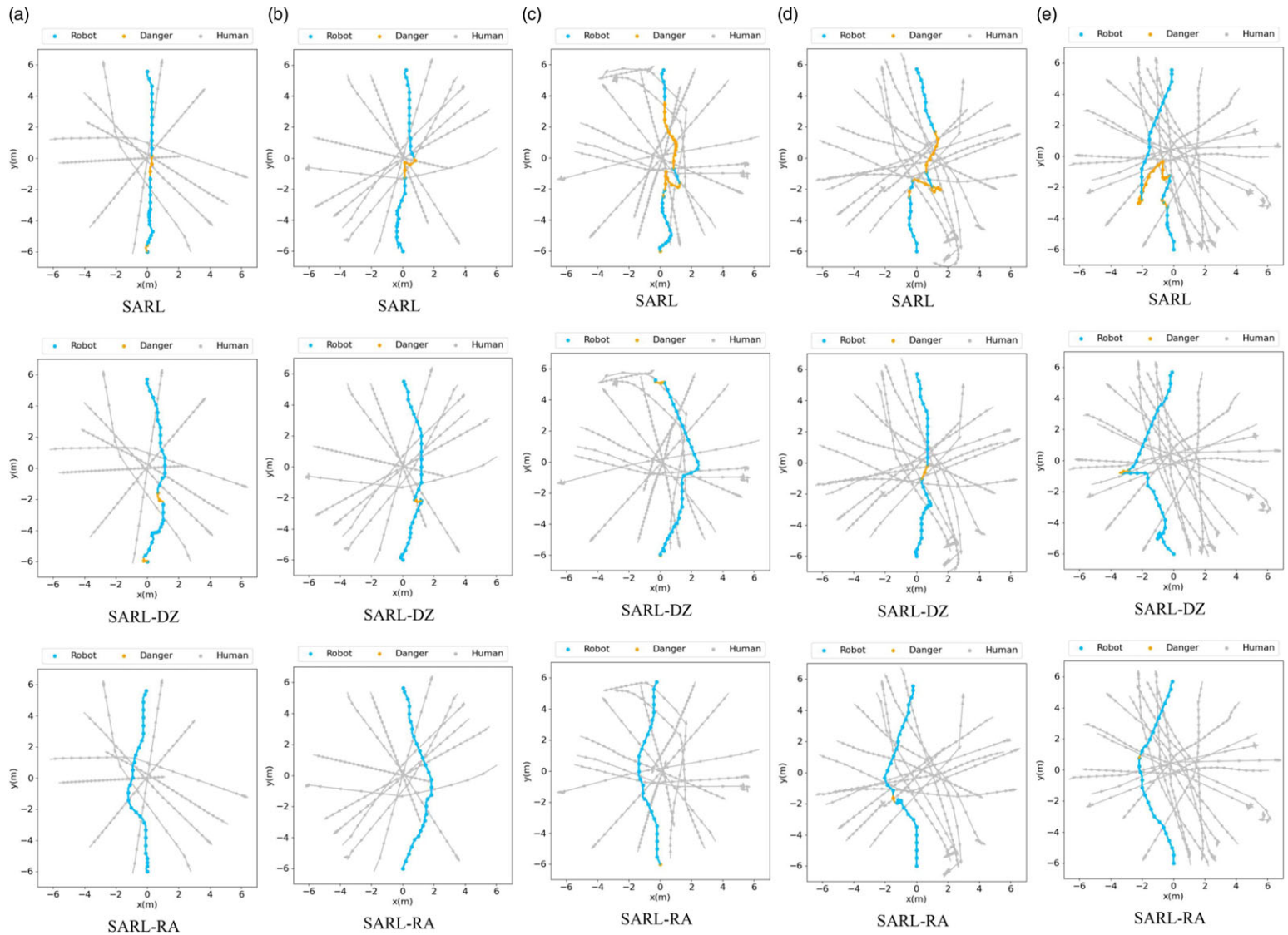


Figure 8. Comparison of the three different reward functions based on SARL algorithm in different dense crowds. The blue line represents the path of the robot without danger. The yellow line is the path that the robot gets too close to the humans. The gray line denotes the path of humans in the environment. (a) $N = 7$. (b) $N = 9$. (c) $N = 11$. (d) $N = 13$. (e) $N = 15$.

Table III. Navigation results of turtlebot2 in real-world environment.

Environment	Method	Result	Danger Count	Minimum Separation Distance	Navigation Time
circle	SARL-RA	suc	1	0.18	22.63
	SARL-DZ	suc	1	0.14	27.44
	SARL	suc	2	0.08	21.11
square	SARL-RA	suc	0	–	19.27
	SARL-DZ	suc	1	0.10	21.20
	SARL	col	–	–	–
group	SARL-RA	suc	0	–	21.65
	SARL-DZ	suc	0	–	20.71
	SARL	suc	0	–	31.46

non-holonomic robot setting simulation on a Turtlebot2 platform in a human-robot coexisting indoor environment. The robot is equipped with a SLAMTEC RPLIDAR A3 for pedestrian detection, and the action commands are generated by an IRU mini PC installed with Ubuntu 16.04 and ROS Kinetic. The positions and velocities of the robot and pedestrians are obtained using the ROS package *amcl* and *leg_detector*, respectively. The crossing scenarios in the real environment are similar to that in the simulated environment, including circle, square, and group settings. However, in the real environment, after completing the same navigation tasks as in the simulated environment, pedestrians continue to move randomly based on their own intentions. Additionally, due to sensor delays, pedestrian movements exhibit higher uncertainty compared to the simulated environment.

The navigation results in real-world environments are presented in Table III. Among all the experimental cases, only SARL has one collision. Both the SARL-RA and SARL-DZ safely complete the navigation tasks. Our method has the fewest occurrences of dangers among all the algorithms while maintaining a favorable minimum separation distance. In terms of navigation time, due to the maximum speed limitation of the turtlebot2 robot, the navigation time in the real environment inevitably becomes longer than in the simulated environment. In comparison, SARL-RA achieves the shortest average navigation time in the navigation cases.

Fig. 9 shows the robot navigates with the crowd. The numbers in red are the participant numbers. The blue dashed lines with arrows indicate the trajectories of the participants. The purple dots indicate the trajectory of the robot. The green arrow represents the action command generated by the SARL-RA algorithm. The red arrow denotes the current pose of the robot. The white arrows stand for the people's velocity detected by the 2D lidar. The red dots are the local targets generated by the robot navigation frame. The specific navigation cases demonstrate the robot's effective avoidance of fast-moving pedestrians in the environment. As observed in Fig. 9(a) and Fig. 9(b), the robot proactively executes avoidance maneuvers for pedestrians approaching from the opposite direction and timely creates pathways for pedestrians moving in the same direction. Additionally, Fig. 9(c) reveals that the robot maintains a safe distance from stationary pedestrians and selects appropriate routes to evade them. Throughout the navigation scenarios, no collisions or obstructions to pedestrian movement are encountered. Experiments in the real world demonstrate that our Risk-Area method, combined with the collective importance of crowd of SARL, develops secure and socially compliant robot navigation.

5. Conclusions and future work

This paper proposes a safety mechanism named Risk-Area within reinforcement learning framework for safe socially aware robot navigation in fast-moving and densely populated environments. First, we establish the collision model of the human and the robot in-plane and formulate Risk-Areas around

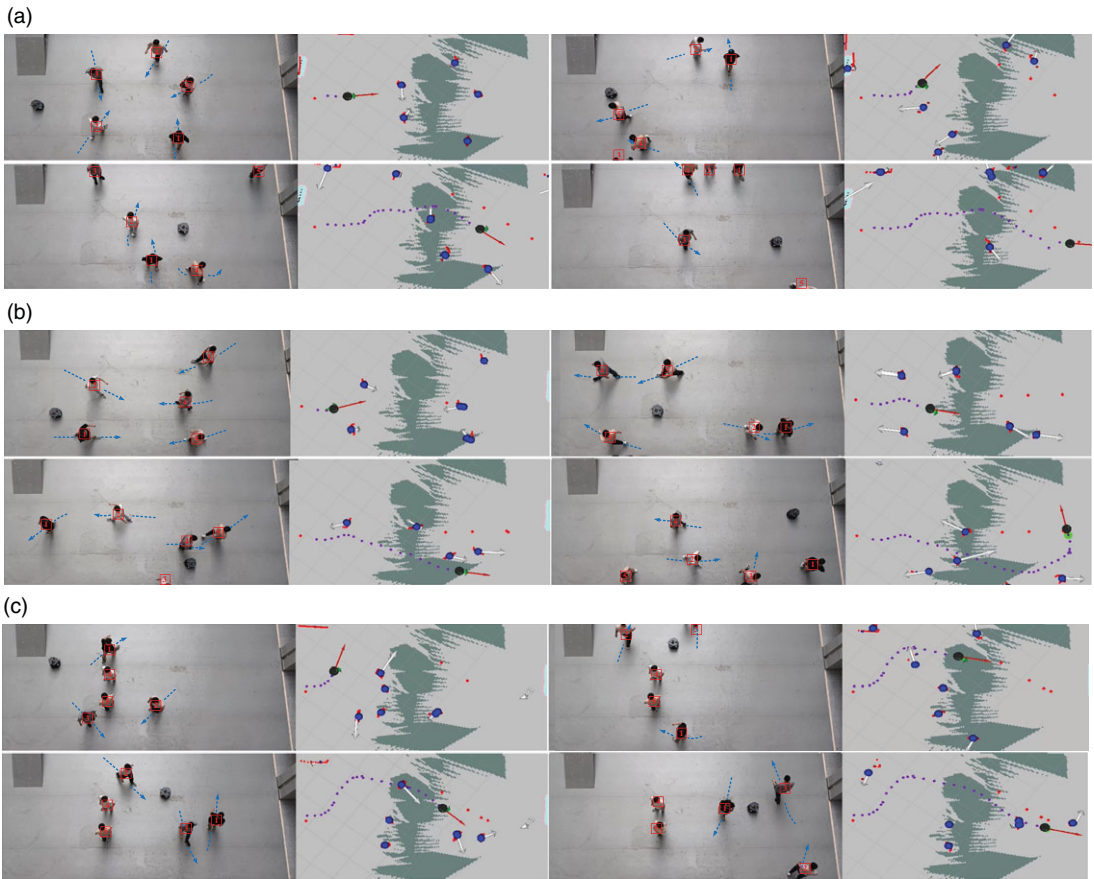


Figure 9. A Turtlebot2 robot navigates in the crowd with the SARL-RA policy. (a) Circle crossing scenario. (b) Square crossing scenario. (c) Group crossing scenario.

each human based on the relative movement of the agents. Then, the reward function and the geometry of the Risk-Area are established to ensure the safety of the crowd-robot navigation. After that, in the deep reinforcement learning framework, the robot is trained to understand the real-time human-robot interactions and not enter the Risk-Area. Experiment results confirm that our approach outperforms the state-of-the-art method in terms of safety and task accomplishments. In addition, by introducing the Risk-Area, the safety performance in different dense crowds of the model has also been improved. Finally, we deploy our method combined with SARL on a Turtlebot2 robot, which successfully navigates in real-world environments among fast-moving pedestrians.

While our approach has proven effective in enhancing safety in various crowded scenarios, the current risk representation based on collision theory lacks an expression for irregularly shaped moving objects, necessitating further research. Additionally, our adoption of a sparse reward function, which lacks rewards related to the robot's proximity to the goal, restricts improvements in navigation efficiency. In future work, we plan to 1) apply camera resources with depth, semantic labels, and optical flows into our system to resolve the limitation of only using 2D LiDAR information; 2) improve our reward function to incentivize more efficient navigation; 3) and extend our risk representation based on collision theory to more general environments involving pedestrians and other static obstacles.

Author contributions. ZF, BX, and CW conceived and designed the study. ZF and BX conducted the experiments and performed statistical analyses. ZF, BX, and CW wrote the article. CW and FZ reviewed and edited the manuscript.

Financial support. This work was supported in part by the Key R & D Project of Shandong Province under Grant (2023TZXD018), the Jinan “20 New Colleges and Universities” Funded Scientific Research Leader Studio under Grant (2021GXRC079), the Central Government Guiding Local Science and Technology Development Foundation of Shandong Province under Grant (YDZX2023122), and the Jinan City and University Cooperation Development Strategy Project under Grant (JNSX2023012).

Competing interests. The authors declare that they have no conflict of interest.

Ethical approval. None.

References

- [1] T. Kruse, A. K. Pandey, R. Alami and A. Kirsch, “Human-aware robot navigation: A survey,” *Robot Auton Syst* **61**(12), 1726–1743 (2013).
- [2] K. Charalampous, I. Kostavelis and A. Gasteratos, “Recent trends in social aware robot navigation: A survey,” *Robot Auton Syst* **93**, 85–104 (2017).
- [3] V. Malviya, A. K. Reddy and R. Kala, “Autonomous social robot navigation using a behavioral finite state social machine,” *Robotica* **38**(12), 2266–2289 (2020).
- [4] L. Yin and Y. Yin, “An Improved Potential Field Method for Mobile Robot Path Planning in Dynamic Environments,” **In**: 2008 7th World Congress On Intelligent Control and Automation (WCICA), Chongqing, China (IEEE, 2008) pp. 4847–4852.
- [5] T. Fan, X. Cheng, J. Pan, P. Long, W. Liu, R. Yang and D. Manocha, “Getting robots unfrozen and unlost in dense pedestrian crowds,” *IEEE Robot Autom Lett* **4**(2), 1178–1185 (2019).
- [6] P. Trautman and A. Krause, “Unfreezing the Robot: Navigation in Dense, Interacting Crowds,” **In**: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan (IEEE, 2010) pp. 797–803.
- [7] A. J. Sathyamoorthy, U. Patel, T. Guan and D. Manocha, “Frozone: Freezing-free, pedestrian-friendly navigation in human crowds,” *IEEE Robot Autom Lett* **5**(3), 4352–4359 (2020).
- [8] P. Bachiller, D. Rodriguez-Criado, R. R. Jorvekar, P. Bustos, D. R. Faria and L. J. Manso, “A graph neural network to model disruption in human-aware robot navigation,” *Multimed Tools Appl* **81**(3), 3277–3295 (2022).
- [9] K. Charalampous, I. Kostavelis and A. Gasteratos, “Robot navigation in large-scale social maps: An action recognition approach,” *Expert Syst Appl* **66**, 261–273 (2016).
- [10] K. Li, Y. Xu, J. Wang and M. Q.-H. Meng, “SARL: Deep Reinforcement Learning Based Human-Aware Navigation for Mobile Robot in Indoor Environments,” **In**: 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dali, China (IEEE, 2019) pp. 688–694.
- [11] X.-T. Truong and T.-D. Ngo, “To approach humans?: A unified framework for approaching pose prediction and socially aware robot navigation,” *IEEE Trans Cogn Develop Syst* **10**(3), 557–572 (2018).
- [12] M. Pfeiffer, U. Schwesinger, H. Sommer, E. Galceran and R. Siegwart, “Predicting Actions to Act Predictably: Cooperative Partial Motion Planning with Maximum Entropy Models,” **In**: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea (South) (IEEE, 2016) pp. 2096–2101.
- [13] M. Bennewitz, W. Burgard, G. Cielniak and S. Thrun, “Learning motion patterns of people for compliant robot motion,” *Int J Robot Res* **24**(1), 31–48 (2005).
- [14] P. Agarwal, S. Kumar, J. Ryde, J. Corso, V. Krovci, N. Ahmed, J. Schoenberg, M. Campbell, M. Bloesch, M. Hutter, M. Hoepflinger, S. Leutenegger, C. Gehring, C. David Remy, R. Siegwart, J. Brookshire, S. Teller, M. Bryson, M. Johnson-Roberson, . . . P. R. Giordano, Feature-based prediction of trajectories for socially compliant navigation. (2013). (MIT Press, USA).
- [15] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy and J. P. How, “Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns,” *Auton Robots* **35**(1), 51–76 (2013).
- [16] Z. Zhou, P. Zhu, Z. Zeng, J. Xiao, H. Lu and Z. Zhou, “Robot navigation in a crowd by integrating deep reinforcement learning and online planning,” *Appl Intell* **52**(13), 15600–15616 (2022).
- [17] L. Sun, J. Zhai and W. Qin, “Crowd navigation in an unknown and dynamic environment based on deep reinforcement learning,” *IEEE Access* **7**, 109544–109554 (2019).
- [18] Z. Hu, Y. Zhao, S. Zhang, L. Zhou and J. Liu, “Crowd-comfort robot navigation among dynamic environment based on social-stressed deep reinforcement learning,” *Int J Soc Robot* **14**(4), 913–929 (2022).
- [19] S. S. Samsani and M. S. Muhammad, “Socially compliant robot navigation in crowded environment by human behavior resemblance using deep reinforcement learning,” *IEEE Robot Autom Lett* **6**(3), 5223–5230 (2021).
- [20] D. Helbing and P. Molnar, “Social force model for pedestrian dynamics,” *Phys Rev E* **51**(5), 4282–4286 (1995).
- [21] G. Ferrer, A. Garrell and A. Sanfeliu, “Robot Companion: A Social-Force Based Approach with Human Awareness-Navigation in Crowded Environments,” **In**: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan (IEEE, 2013) pp. 1688–1694.
- [22] C. Yang, T. Zhang, L.-P. Chen and L.-C. Fu, “Socially-Aware Navigation of Omnidirectional Mobile Robot with Extended Social Force Model in Multi-Human Environment,” **In**: 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy (IEEE, 2019) pp. 1963–1968.

- [23] J. van den Berg, M. Lin and D. Manocha, "Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation," **In: IEEE International Conference on Robotics and Automation (ICRA) 2008, Pasadena, CA (IEEE, 2008)** pp. 1928–1935.
- [24] J. van den Berg, S. J. Guy, M. Lin and D. Manocha, "Reciprocal n-body collision avoidance," **In: Proceedings of the Robotics Research, Berlin, Heidelberg (Springer, 2011)** pp. 3–19.
- [25] P. Trautman, J. Ma, R. M. Murray and A. Krause, "Robot Navigation in Dense Human Crowds: The Case for Cooperation," **In: IEEE International Conference on Robotics and Automation (ICRA) 2013, Karlsruhe, Germany (IEEE, 2013)** pp. 2153–2160.
- [26] L. Tai, J. Zhang, M. Liu and W. Burgard, "Socially Compliant Navigation Through Raw Depth Inputs with Generative Adversarial Imitation Learning," **In: IEEE International Conference on Robotics and Automation (ICRA) 2018, Brisbane, Australia (IEEE, 2018)** pp. 1111–1117.
- [27] Q. Wu, X. Gong, K. Xu, D. Manocha, J. Dong and J. Wang, "Towards target-driven visual navigation in indoor scenes via generative imitation learning," *IEEE Robot Autom Lett* **6**(1), 175–182 (2021).
- [28] L. Qin, Z. Huang, C. Zhang, H. Guo, M. Ang and D. Rus, "Deep Imitation Learning for Autonomous Navigation in Dynamic Pedestrian Environments," **In: 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China (IEEE, 2021)** pp. 4108–4115.
- [29] H. Kretzschmar, M. Spies, C. Sprunk and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *Int J Robot Res* **35**(11), 1289–1307 (2016).
- [30] M. Kollmitz, T. Koller, J. Boedecker and W. Burgard, "Learning Human-Aware Robot Navigation from Physical Interaction via Inverse Reinforcement Learning," **In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA (IEEE, 2020)** pp. 11025–11031.
- [31] A. Konar, B. H. Baghi and G. Dudek, "Learning goal conditioned socially compliant navigation from demonstration using risk-based features," *IEEE Robot Autom Lett* **6**(2), 651–658 (2021).
- [32] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang and J. Pan, "Towards Optimally Decentralized Multi-Robot Collision Avoidance via Deep Reinforcement Learning," **In: 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia (IEEE, 2018)** pp. 6252–6259.
- [33] Y. F. Chen, M. Liu, M. Everett and J. P. How, "Decentralized Non-Communicating Multiagent Collision Avoidance with Deep Reinforcement Learning," **In: 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore (IEEE, 2017)** pp. 285–292.
- [34] Y. F. Chen, M. Everett, M. Liu and J. P. How, "Socially Aware Motion Planning with Deep Reinforcement Learning," **In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada (IEEE, 2017)** pp. 1343–1350.
- [35] M. Everett, Y. F. Chen and J. P. How, "Motion Planning Among Dynamic, Decision-Making Agents with Deep Reinforcement Learning," **In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain (IEEE, 2018)** pp. 3052–3059.
- [36] C. Chen, Y. Liu, S. Kreiss and A. Alahi, "Crowd-Robot Interaction: Crowd-Aware Robot Navigation with Attention-Based Deep Reinforcement Learning" **In: IEEE International Conference on Robotics and Automation (ICRA) 2019, Montreal, QC, Canada (IEEE, 2019)** pp. 6015–6022.
- [37] Y. Chen, C. Liu, B. E. Shi and M. Liu, "Robot navigation in crowds by graph convolutional networks with attention learned from human gaze," *IEEE Robot Autom Lett* **5**(2), 2754–2761 (2020).
- [38] R. W. Bohannon and A. W. Andrews, "Normal walking speed: A descriptive meta-analysis," *Physiotherapy* **97**(3), 182–189 (2011).
- [39] J. T. Butler and A. Agah, "Psychological effects of behavior patterns of a mobile personal robot," *Auton Robots* **10**(2), 185–202 (2001).
- [40] D. F. Hoyt and C. R. Taylor, "Gait and the energetics of locomotion in horses," *Nature* **292**(5820), 239–240 (1981).
- [41] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature* **518**(7540), 529–533 (2015).
- [42] A. Bera and D. Manocha, "Realtime Multilevel Crowd Tracking Using Reciprocal Velocity Obstacles," **In: 2014 22nd International Conference on Pattern Recognition (ICPR), Stockholm, Sweden (IEEE, 2014)** pp. 4164–4169.
- [43] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. (2018). (MIT Press, USA).
- [44] T. Bai, Z. Fan, M. Liu, S. Zhang and R. Zheng, "Multiple Waypoint Path Planning for a Home Mobile Robot," **In: 2018 Ninth International Conference on Intelligent Control and Information Processing (ICICIP), Wanzhou, China, (IEEE, 2018)** pp. 53–58.
- [45] C. M. Watts, P. Lancaster, A. Pedross-Engel, J. R. Smith and M. S. Reynolds. 2D and 3D Millimeter-Wave Synthetic Aperture Radar Imaging on a PR2 Platform. **In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea (South) (IEEE, 2016)** pp. 4304–4310.
- [46] R. M. Brach, *Mechanical Impact Dynamics: Rigid Body Collisions* (Wiley, New York, 1991).
- [47] R. M. Brach, "Rigid body collisions," *J Appl Mech* **56**(1), 133–138 (1989).
- [48] A. E. Minetti, "Chapter 5 - The three modes of Terrestrial locomotion," **In: Biomechanics and Biology of Movement (Human Kinetics, Champaign, 2000)** pp. 67–78.