PRTIRL Based Socially Adaptive Path Planning for Mobile Robots

Zhiyu Ding[1] · Jie Liu[1] · Wenzheng Chi[1] · Jiankun Wang[2] · Guodong Chen[1] · Lining Sun[1]

Abstract

In recent years, learning-based methods have been increasingly studied to combine social norms with mobile robot navigation, such as social proxemics, distance to pedestrians, passing ways, etc. In these challenging tasks, existing approaches suffer from the non-homotopic path problem, resulting in a significant difference between the demonstration path and the generated path. To address this problem, we propose a nocel socially adaptive path planning framework for mobile robots incorporating a non-homotopic path penalty strategy with the Rapidly-exploring random Trees Inverse Reinforcement Learning (RTIRL) called PRTIRL. First, we propose a homotopic feature extraction method for the demonstration path and the generated path. Next, a series of path features along with the homotopic feature are fed to the Inverse Reinforcement Learning (IRL) framework. A non-homotopic path penalty strategy is proposed to learn homotopy features and improve the learning process. In the learning process, the Optimal Rapidly-exploring Random Trees (RRT*) algorithm is adopted to generate paths by applying the learned features in the sampling process. Inspired by the Turing test, a new metric is provided to evaluate the socially adaptive navigation capability. We evaluate the proposed method with both comparative experiments and questionnaire analysis and demonstrate that it outperforms the state-of-the-art approach.

Keywords Socially Adaptive Path Planning · IRL · Non-homotopic Path Optimization · RRT*

1 Introduction

Nowadays, mobile service robots have been successfully applied to various environments, such as airports, hospitals and supermarkets. Currently, most mobile robots are only designed with optimizations related to task execution efficiency, including how to quickly generate the shortest path, shorten the navigation time, and ensure the control constraints of the robot, and hardly consider the compliance with social norms. The path generated by the robot is usually quite different from that of pedestrians, which makes it difficult

Zhiyu Ding and Jie Liu have contributed equally to this work

✉ Wenzheng Chi
  wzchi@suda.edu.cn

1 Robotics and Microsystems Center, School of Mechanical and Electric Engineering, Soochow University, Suzhou 215021, China

2 Department of Electronic Engineering, The Chinese University of Hong Kong, N.T., Hong Kong SAR, China

for people to understand the behavior of the robot. Pedestrians often need to accommodate the movement of robots and lack a natural and comfortable interactive experience. As shown in Fig. 1, two pedestrians are talking as they walk forward, and the robot has several optional paths with the current target. In this scenario, people tend to choose path 1 to minimize the disturbance to pedestrians, so path 1 is used as the demo path. Although path 2 can reach the destination faster, it ignores the feelings of pedestrians during the interaction process, and does not respect the social rules of human beings. There may be many social-adaptive paths, we record some of them for learning to extract the inherent characteristics of social adaptation. Moreover, the demonstration paths are generated by volunteers, and we find that paths that are not homotopy to the demonstration path may be shorter, but are mostly not socially compliant with human behaviors. Therefore, designing a socially adaptive navigation method and improving the naturalness of the navigation are of great significance for mobile robots.

In order to improve the social adaptability of robots, some scholars have introduced Proxemics [1], an achievement in the field of social psychology, into the field of human-robot motion interaction, combining pedestrian-related features
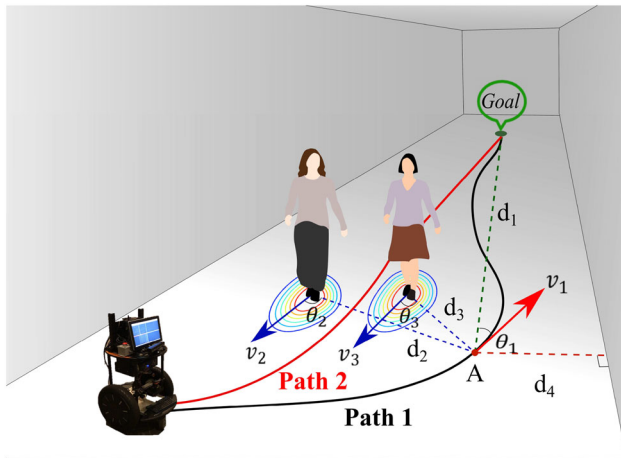
Fig. 1 Schematic diagram of non-homotopic paths and robot-scenario interaction features

with motion planners through hard programming. The methods based on Proxemics combine the influence of intimacy between people on social distance with the path planning so as to protect the personal space of pedestrians. Except for the artificial features, some other social norms learning method are also proposed for socially adaptive navigation, such as the learning from demonstration methods [2]. Pérez-Higueras et al. [3] proposed a method of learning social navigation from the demonstration path, combining IRL [4] and RRT* [5] to learn the cost function through the offline demonstration path. The learned feature weights are applied to the real-time path planning of RRT*, so as to provide robots with social awareness in the robot navigation process. This work provides a way for the robot to navigate like a human and makes the robot with certain socially adaptive capability, however, still suffers the non-homotopic path problem.

The definition of homotopic class of paths is given in Definition 1.

Definition 1 (Homotopy Class of Trajectories [6,7]) Two trajectories with the same start and the same end are said to be in the same **Homotopy Class** if one can be smoothly deformed into the other without intersecting obstacles. Otherwise they belong to different homotopy classes.

In this work, we propose to solve the non-homotopic path problem based on the framework of [3] combined with a non-homotopic path penalty strategy. The rest of the paper is organized as follows: We first introduce the related work in Sect. 1.1. The problem statement of the paper is shown in Sect. 2. Section 3 explains the details of the proposed Penalty based Rapidly-exploring random Tree Inverse Reinforcement Learning (PRTIRL) algorithm. The experimental results are reported in Sect. 4. We draw conclusions at the end in Sect. 5.

1.1 Related Work

Recently, scholars have launched a series of studies on robot navigation in the human-robot integration environment to ensure the comfort of pedestrians and improve the naturalness and smoothness of the generated path, which can be roughly divided into four categories [8,9]: path planning based on pedestrian motion predictors; path optimization based on known dynamic obstacles; path planning with hand-designed social variables; and learning-based socially adaptive path planning.

The first kind of methods combines the path planning with the motion prediction of pedestrians so as to avoid the collision with pedestrians. Foka and Trahanias [10] proposed a Polynomial Neural Network to predict the movement of pedestrians. Ziebart et al. [11] proposed the maximum entropy inverse optimal control to train a motion predictor for pedestrians. Garzón et al. [12] collected the trajectories of a large number of pedestrians in a specific scenario and adopt the Gaussian process to predict the motion of pedestrians. These methods usually depend on a large training data set and cannot guarantee their generalization to untrained scenarios.

The second kind of methods focus on the path optimization assuming that the precise poses of the dynamic obstacles are given. It explores a reactive path planner [13–15] without predictions for pedestrians, and rely on the information obtained by the sensor to quickly plan a new path during the navigation process. The sampling-based path planning algorithms, such as the RRT* algorithm [5], have been widely applied in this area for its computational savings and its flexibility in avoiding obstacles. Typically, the RRT* algorithm uses cumulative Euclidean distance as a cost function, and takes the shortest cumulative distance of nodes as an optimization target. However, this kind of methods pays more attention to the real-time performance of the reactive planning when encountering dynamic obstacles and does not consider the motion interaction with pedestrians.

To improve the social adaptability of robots, some scholars have proposed to start with the social norms of pedestrians and the research results of pedestrian psychology to analyze the human-robot motion interaction mechanisms during the robot navigation process. Helbing and Molnar [16] proposed a Social Force Model based on a multi-particle self-driving system, in which the path planning is influenced by the force between pedestrians and the force between pedestrians and obstacles. Based on the Proxemics, Huang et al. [17] presented an oval-shaped General Comfort Space model. Lam et al. [18] proposed an egg-shaped Human Sensitive Field model. Chi et al. [19] designed the Personal Comfort Function model, which replaces the previous deterministic model with a flexible function and comprehensively considers factors such as height, width, density and movement speed of

pedestrians in the environment. These models are still based on a hand-designed social variables.

The specific implementation methods of the learning-based socially adaptive path planning methods can be divided into the following two groups. The first one directly learns policies by using supervised learning, also known as inverse optimal control [20]. This learned strategy is susceptible to environmental noise, such as robot wheel wear and slippage, which will cause the drift of the optimal policy [21]. By contrast, IRL based methods directly use the cost function learned from demonstration paths for planning, which are more robust to environmental changes. In [22–24], the maximum entropy algorithm are adopted to learn pedestrian behavior model and predict social norms compliant path. The Bayesian inverse reinforcement learning is also usually adopted for the sampled trajectory learning [25,26]. Shiarlis et al. [27] proposed to replace Markov Decision Process (MDP) with a progressive optimal planner RRT* in the inverse reinforcement algorithm based on the maximum margin, which avoids accurate modeling of the environment and makes it more suitable for high-dimensional state space. Pérez-Higueras et al. [3] combined IRL based on Maximum entropy with RRT* planner, which reduces the weight of the calculation and optimizes the performance of IRL in high-dimensional complex spaces. The learning-based method provides a way for the robot to navigate like a human and makes the robot with certain socially adaptive capability. However, few work proposed to solve the non-homotopic path problem in the learning process. Among them, the concept of homotopy was originally used to judge whether the path between two points can be transited in topology. Werner et al. [28] propose to use the topological concept of homotopy in order to decide if two routes should be considered equivalent or alternative. Hernandez et al. [29] use homotopy classes to constrain and guide path planning algorithms. Homotopic path planners can improve the quality of the solutions of their respective non-homotopic versions with similar computation time while keeping the topological constraints.

In this work, we propose to solve the non-homotopic path problem by integrating a non-homotopic path penalty strategy with the IRL method. A data set of demonstration paths from volunteers is firstly collected in different scenarios. A homotopic feature extraction algorithm is proposed for the contrast between the demonstration path and the generated path. A PRTIRL framework is proposed for the learning of the socially adaptive features, in which a non-homotopic path penalty strategy is proposed for the backpropagation of the homotopy features and accelerate the learning process. The RRT* planner is adopted here for the path generation by using the learned socially adaptive features.

The contributions of our work are summarized as follows:

- A PRTIRL based socially adaptive path planning framework for mobile robots;
- A homotopy feature extraction algorithm for two paths;
- A non-homotopic path penalty strategy for the backpropagation of the homotopy features; and
- A new metric to evaluate the socially adaptive navigation capability.

2 Problem Statement

Let $\mathcal{X} \in \mathbb{R}^n$ be the state space. Let $\mathcal{X}_{obs}$ be the obstacle space and $\mathcal{X}_{free} = \mathcal{X} \backslash \mathcal{X}_{obs}$ be the obstacle-free space. Given the initial state of the robot as x_{init} and the goal region as $\mathcal{X}_{goal}$, the path planning problem is defined to find a path $\sigma : [0, 1] \mapsto \mathcal{X}_{free}$, where the initial state $\sigma(0) = x_{init}$, and the terminal state $\sigma(1) \in \mathcal{X}_{goal}$.

Let Σ be the set of all nontrivial paths. The optimal path planning problem is then formally defined as the search for the feasible path σ^* that minimizes a given cost function, $c : \Sigma \mapsto \mathbb{R} \geq 0$.

$$
\begin{aligned}
\sigma^* &= \underset{\sigma \in \Sigma}{\arg\min} \, c(\sigma) \\
s.t. \ \sigma(0) &= x_{init}, \\
\sigma(1) &\in \mathcal{X}_{goal}, \\
\sigma(t) &\in \mathcal{X}_{free}, \ \forall t \in [0, 1].
\end{aligned} \tag{1}
$$

The socially adaptive path planning aims to find a path that is as similar as possible to the path planned by humans. Let σ_{human} be a path generated by human. The socially adaptive path planning problem is defined as:

$$
\begin{aligned}
\sigma_{social}^* &= \underset{\sigma \in \Sigma}{\arg\min} \, ||(c(\sigma) - c(\sigma_{human}))|| \\
s.t. \ \sigma(0) &= x_{init}, \\
\sigma(1) &\in \mathcal{X}_{goal}, \\
\sigma(t) &\in \mathcal{X}_{free}, \ \forall t \in [0, 1].
\end{aligned} \tag{2}
$$

In order to obtain a socially adaptive path, the cost function used in the path planning process needs to consider elements related to social rules, such as the distance to obstacles, the distance to pedestrians, the passing ways, etc. Herein, let $f(x) = [f_1(x), f_2(x), ..., f_N(x)]^T$ denote the elements in the cost function and then the cost function can be written as:

$$
\begin{aligned}
c(x) &= \sum_{j=1}^{N} w_j f_j(x) \\
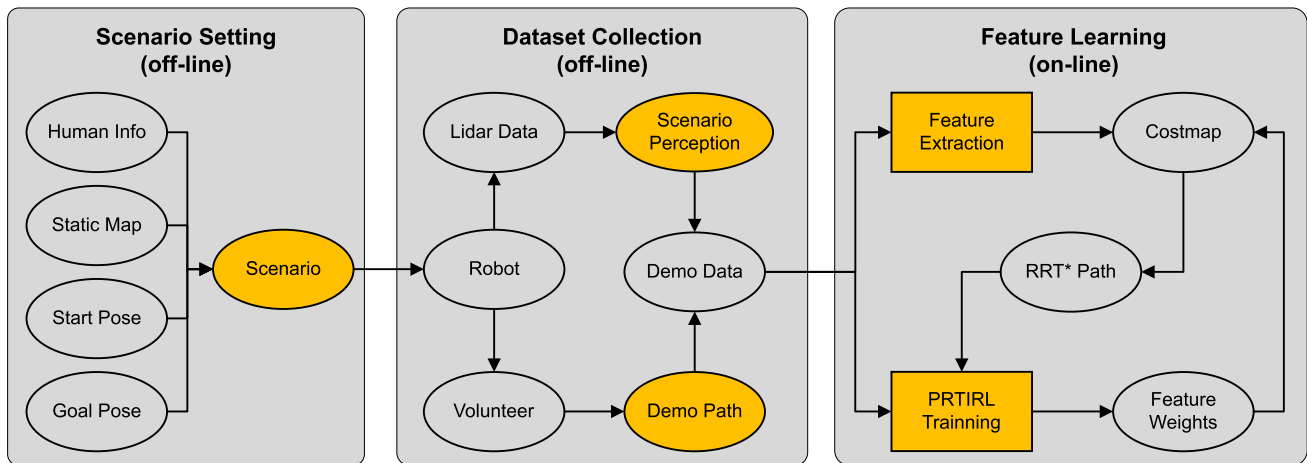&= w^T f(x).
\end{aligned} \tag{3}
$$

Fig. 2 Outline of our approach

Meanwhile, $f(x) = [f_1(x), f_2(x), ..., f_N(x)]^T$ represents the interaction feature between the robot and the scene and $w = [w_1, w_2, ..., w_N]^T$ represents the feature weights. Features and feature weights jointly determine the cost function of the path planner, and later IRL is used to measure and learn the importance of each feature(feature weight). Among these elements, whether the planned path is in the same homotopy class of the path generated by human is one of the key factors. As shown in Fig. 1, the path 1 and the path 2 are in the different homotopy class. For the clarity of expression, we represent paths belonging to different homotopy classes as non-homotopic paths. In the human-robot coexistence environment, although non-homotopic paths are feasible and even have shorter path lengths, they may violate social rules, such as interrupting the people in conversation. In this work, we propose a PRTIRL framework to solve the non-homotopic path problem.

3 Algorithms

In Fig. 2, we depict the outline of our approach, which includes three stages: scenario setting, data set collection, and feature learning. In the scenario setting stage, we construct different human-robot motion interaction scenarios by placing obstacles and pedestrians and setting the starting and target poses. Herein, the information of the pedestrian includes the position and the direction. In the data set collection stage, demonstration paths are generated by manipulating the robot with a remote control handle by volunteers, during which the scenario information is recorded by a lidar. The demonstration path and its corresponding scenario information constitute the training data set. Then in the feature learning stage, the features of the scenarios from the

data set are extracted and are utilized for the construction of the costmap, which is used to guide the path generation by a RRT* planner. The RRT* path and the demonstration path are then fed to the PRTIRL module for the update of the feature weights. After the feature learning, our approach outputs a socially adaptive feature model, which is further utilized in the cost function during the path planning process. In this section, we will focus on introducing the feature learning process.

3.1 PRTIRL

In this work, we propose a PRTIRL scheme to optimize the feature weights in the cost function of the path planning. The general framework of PRTIRL is shown in the Fig. 3 and its pseudocode is shown in Algorithm 1. In the training process, a cost function is initialized for the node connection of the RRT* algorithm. When the initial pose and the goal pose of the robot are given, a path is generated in real time by using the RRT* algorithm, which is denoted as the RRT* path. Meanwhile, another path, namely the demonstration path, is manually generated by the volunteers through the remote control. Then, we compare the characteristics of the RRT* path with the demonstration path. In this work, we propose to use two groups of features to describe a path, namely the global features and the local features, which will be explained in Sect. 3.2. Correspondingly, a penalty function is proposed to deal with the difference of the proposed feature set. A weight update scheme for the cost function is proposed, especially for the global feature. The learning process is carried out through different scenarios until the weights of the cost function converge.

In Algorithm 1, we explain the PRTIRL algorithm in detail. First, the data set recorded in the first section is
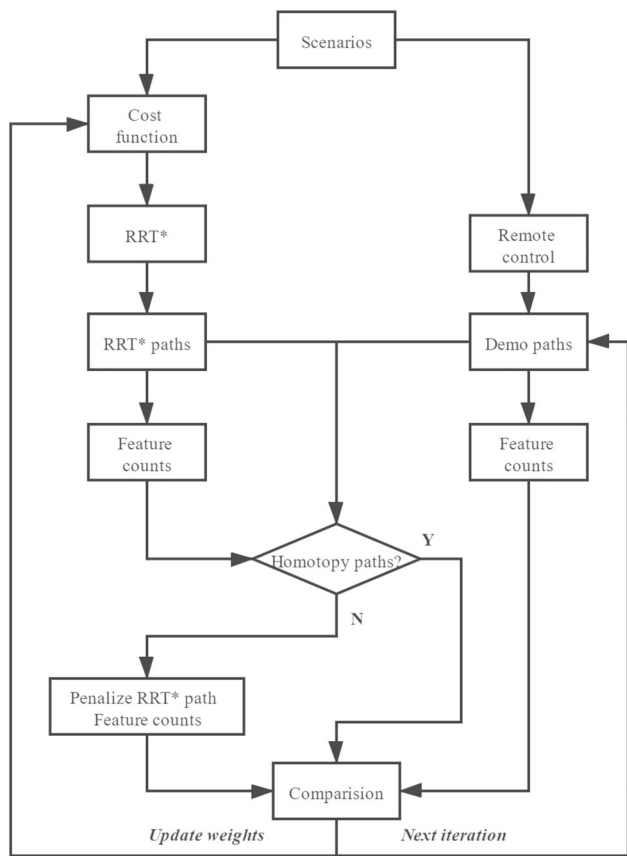
Fig. 3 The general framework of training process

Input: Demo paths $D = \{\zeta_{D1}, ..., \zeta_{DS}\}$ from S scenarios
Output: Features weights $w = [w_1, ..., w_J]^T$
1: **while** w not converge **do**
2:    **for all** $s \in S$ **do**
3:       **for** $repetitions$ **do**
4:          $\zeta_{pi} \leftarrow RRT^*(s, w)$
5:          $\boldsymbol{F_L} \leftarrow \boldsymbol{LocalFeature}(\zeta_{pi})$
6:          $\boldsymbol{F_{NH}} \leftarrow \boldsymbol{GlobalFeature}(\zeta_{pi}, \zeta_{Ds})$
7:          $f(\zeta_{pi}) \leftarrow \boldsymbol{Penalty}(f(\zeta_{pi}), \boldsymbol{F_L}, \boldsymbol{F_{NH}})$
8:       **end for**
9:       $\bar{f}^s_{RRT^*} \leftarrow (\sum_{i=1}^{repetitions} f(\zeta_{pi})/repetitions$
10:    **end for**
11:    $\bar{f}_{RRT^*} \leftarrow (\sum_{i=1}^{S} \bar{f}^i_{RRT^*})/S$
12:    $\nabla L \leftarrow (\bar{f}_{RRT^*} - \bar{f}_D)$
13:    $w \leftarrow UpdateWeights(\nabla L)$
14: **end while**
15: **return** w

3.2 Robot-scenario Interactive Features

In order to describe the characteristic of a path, two kinds of robot-scenario interaction features are extracted from the paths, including the local features and the global features. The local features describe the features of nodes that constitute the path. In this work, we adopt the following local features [3]:

- d_t, the distance between the node and the target position;
- d_o, the distance between the node and the nearest obstacle;
- d_p, the distance between the node and pedestrians; and
- θ_p, the angle between the movement direction of the node and pedestrians.

Herein, we denote the local feature set as $F_L = \{d_t, d_o, d_p, \theta_p\}$. The local feature focus on the interactive characteristics between the local node with the scenario. The corresponding feature weights $w = [w_1, w_2, w_3, w_4, w_5]^T$ are listed as follows .

- w_1, the weight of feature d_t;
- w_2, the weight of feature d_o;
- w_3-w_5, the weight of features of the interaction between robots and pedestrians d_p and θ_p.

The larger the feature weight, the more important the feature is.

As shown in Fig. 4, the enclosed area formed by two non-homotopic paths with the same starting point and target point contains at least one obstacle inside, such as pedestrians or walls, etc. In terms of socially adaptive characteristics, the non-homotopic path of pedestrians often does not conform to human behavior and causes discomfort of pedestrians. However, the local features can not
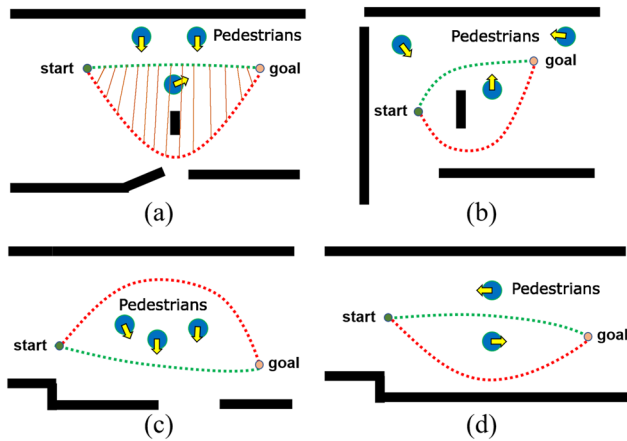
imported as input, including interactive scene, starting point, target point, demonstration path and other information. During the training process, the feature weights w are initialized for the node connection of the RRT* algorithm. The data set contains a total of S training scenarios. For each training scenario s, the RRT* path algorithm is used to plan *repetition* paths (Line 3,4). We compare each RRT* path with the demonstration path, and obtain the local features F_L and global features F_{NH} (Line 5,6) respectively. The local features are the sum of the features in the socially adaptive feature model. The calculation method of the global features is shown in Algorithm 2. Then, We combine the global features and local features, optimize the path feature sum through the penalty function (Line 7, seen in Sect. 3.3) and calculate the average path feature sum of all RRT* paths ($\bar{f}_{RRT^*}$) in S training scenes (Line 9-11). The difference between the average path feature sum of all RRT* paths and the average path feature sum of all demonstration paths ($\bar{f}_D$) is utilized as the gradient (Line 12) to update the feature weights with the help of gradient descent algorithm (Line 13) until the weights converge and return to the weights (Line 14-15).

Fig. 4 The non-homotopic feature extraction. The green dotted line and the red dotted line indicate two paths, respectively. Orange solid line constitutes the detection line segment between the node pairs of the two paths. (Color figure online)

Algorithm 2 Homotopic Feature Extraction.

Input: Arbitrary two paths $\zeta_D = \{x_{d1}, ..., x_{dn}\}$ and $\zeta_P = \{x_{p1}, ..., x_{pm}\}$
Output: The non-homotopic feature F_{NH}
1: $F_{NH} \leftarrow False$
2: **for** $k \leq K$ **do**
3:    $i_k \leftarrow k * n/\Delta$
4:    $j_k \leftarrow k * m/\Delta$
5:    $l_k \leftarrow line(x_{di_k}, x_{pj_k})$
6:    **for** $s = 0, ..., steps$ **do**
7:       $x_{current} \leftarrow steer(l_k, s)$
8:       **for** $p = 0, ..., P$ **do**
9:          **if** $dis(x_{current}, x_p) < r$ **then**
10:             $F_{NH} \leftarrow True$
11:             **return** F_{NH}
12:          **end if**
13:       **end for**
14:    **end for**
15: **end for**
16: **return** F_{NH}

---

describe the homotopic property. From a global perspective, we propose a homotopic feature to describe the path and focus on detecting the non-homotopic features caused by pedestrians.

In Algorithm 2, we explain the non-homotopic path feature extraction method in detail. To extract the non-homotopic feature of the two target paths, we firstly segment each path with K nodes (Line 3,4). The corresponding k^{th} nodes from two paths respectively, namely $< x_{di_k}, x_{pj_k} >$, form a pair of nodes, where $k = 1, 2, \cdots, K$. Then we discrete the line into $steps$ segments between these two nodes and steer along this line. The $steer$ function indicates that the coordinates of the points on the line segment are extracted sequentially along the line segment l_k. If the line intersects the pedestrian area, then the non-homotopic feature is set as true, otherwise it is set as false. In the line 8-9 of Algorithm 2, x_p indicates the position of the p-th pedestrian in the scenario and r represents the threshold of offending pedestrian space, P represents the total number of pedestrians in the scenario.

In the tree construction process of the RRT* algorithm, we only adopt the local feature set to formulate its cost function for the node on the tree, which is defined as follows:

$$c_{RRT^*}(x) = \sum_{j=1}^{W} w_j F_L(x, j). \tag{4}$$

With the initialized cost function, a RRT* path can be generated with given start pose, goal pose and the specific scenario. Therefore, the inverse reinforcement learning process is to compare the demo path and the RRT* path and optimize the feature weights for the cost function of the RRT* algorithm.

3.3 Non-homotopic Penalty based Dissimilarity Measure

In this section, in order to measure the dissimilarity of between the demo path and the RRT* path with respect to the aforementioned interactive features, a non-homotopic penalty based function is proposed, which is defined as follows:

$$f(\varsigma, w) = \sum_{i=1}^{N-1} \frac{F_L(x_i) + F_L(x_{i+1})}{2} ||x_{i+1} - x_i|| \tag{5}$$

$$+ F_{NH} * c * \underbrace{\sqrt{max(\frac{length(\varsigma_D)}{\sum_{i=1}^{N-1} ||x_{i+1} - x_i||} - 1, 0)}}_{non-homotopic\ penalty:\ f_{penalty}(\varsigma)}. \tag{6}$$

Herein, x_i denotes the ith node on the RRT* path ς, F_L denotes the local feature and F_{NH} denotes the non-homotopic feature. c is the penalty parameter from the non-homotopic feature to the local features. $length(\varsigma_D)$ represents the length of demonstration path and the length of RRT* path is calculated by $\sum_{i=1}^{N-1} ||x_{i+1} - x_i||$. If the global feature F_{NH} is True, then we set the penalty value based on the difference between the demonstration path and the planned path. Otherwise, we follow the original training method to train the feature weights. In addition to local features, the proposed penalty based function focuses on reflecting the difference on the non-homotopic feature.

An illustration of the non-homotopic penalty function is shown in Fig. 5. Herein, ς_0 denotes the Euclidean distance between the start pose and the goal pose, which is also the minimum value of the path length. ς_D represents the length of the demonstration path, thus $\varsigma_D \geq \varsigma_0$. The horizontal axis represents the length of the RRT* path and the verti-
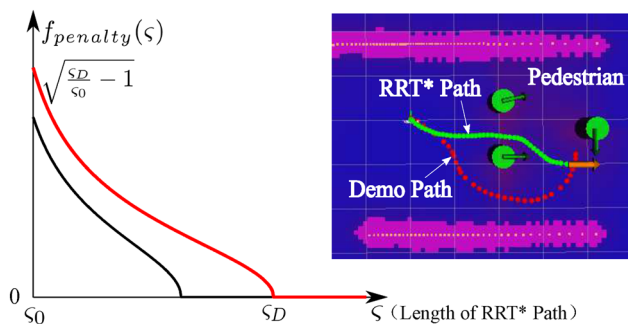
Fig. 5 The non-homotopic penalty function

Fig. 6 Dataset recording. The black in the figure represents static obstacles, the red dots on the edges of the obstacles represent Laserscan data of lidar, and the red dots represent PointCloud2 data of lidar. (Color figure online)

cal axis shows the penalty value. In the interval from ς_0 to ς_D, the penalty function decreases monotonically with respect to ς, otherwise the penalty function is equal to zero. This function describes the non-linearity characteristics of the non-homotopic path well. Comparing with the demonstration path, non-homotopic paths are usually with shorter path lengths. The main reason is that the attraction of the target exceeds that of the socially adaptive factors. Therefore, a higher penalty value is set when the length of the RRT* path is much shorter than that of the demonstration path and the penalty value decreases quickly when the length of the planned path approaches the demonstration path. No additional penalty presents when the length of the planned path is longer than that of the demonstration path in this case, the difference between the planned path and the demonstration path can be reflected well by the local features.

3.4 Simulated Robot Platform

With the proposed cost function, the difference of the feature expectations of the planned path and the demonstration path can be calculated among different repetitions in various scenarios. The gradient descent method is adopted to update the feature weights until the weights converge. This finishes the feature learning process. After that, the learned feature weights can be utilized in the RRT* path planning process, as presented in Equation (5).

4 Experiments

The mainstream evaluation methods of robotic socially adaptive navigation capabilities are generally based on Proxemics and objective analysis of experimental data. We are inspired by the Turing test [30] and incorporate human subjective initiative into the evaluation method. The Turing test was proposed by Alan Mathison Turing, which refers to the random question of the testee through some devices when the tester is separated from the testee (a person and a machine). After multiple tests, if the machine made the average par-
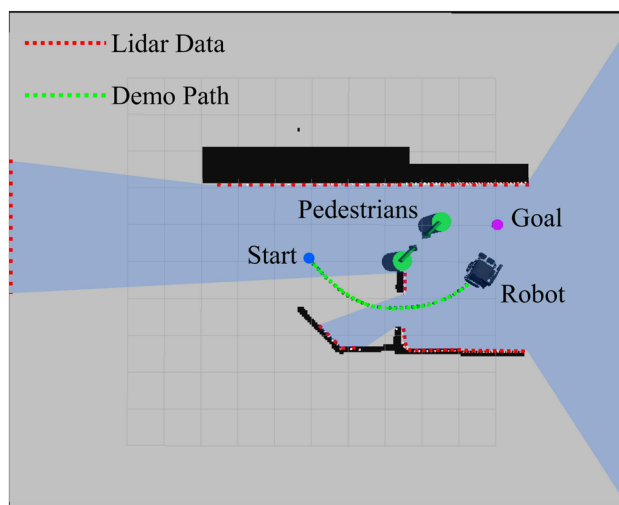
ticipant make more than 30% false positives, the machine passed the test and was deemed to have human intelligence. Herein, we also use the same method to test whether the path planned by the robot is recognized by people. Combining objective data and evaluation, the evaluation method of robot's socially adaptive navigation ability is optimized. The experimental results of these two aspects both show that our method has achieved better results.

In order to verify the feasibility of the proposed algorithm in human aware navigation, we build a simulation experiment framework. ROS [31], as a robot platform with good reusability, is welcomed by many companies and researchers in related fields. This paper uses ROS simulation software stage, 3D visualization tool *rviz* and related plugins to build a simulation environment, so as to realize the construction of static maps and the generation of demonstration paths. The robot model uses robotic wheelchair [32] equipped with lidar, as shown in Fig. 6. In order to lighten the navigation dataset, lidar is used as the only sensor source in this work to observe the surrounding scenario. The lidar sensor installed in front of the robot has a scanning range of 360° and a maximum detection distance of 10*m*.

4.1 Navigation Scenario and Dataset

We complete the design of the human-robot integration scenario and the recording of the navigation data set in the built simulation experiment environment. The related technologies are shown below. Firstly, a static obstacle map is drawn and published to *rviz*. Then the start and goal are set on the map, using the stage simulation platform to import the robot pose and pedestrian pose, and visualizing it in *rviz*. Then
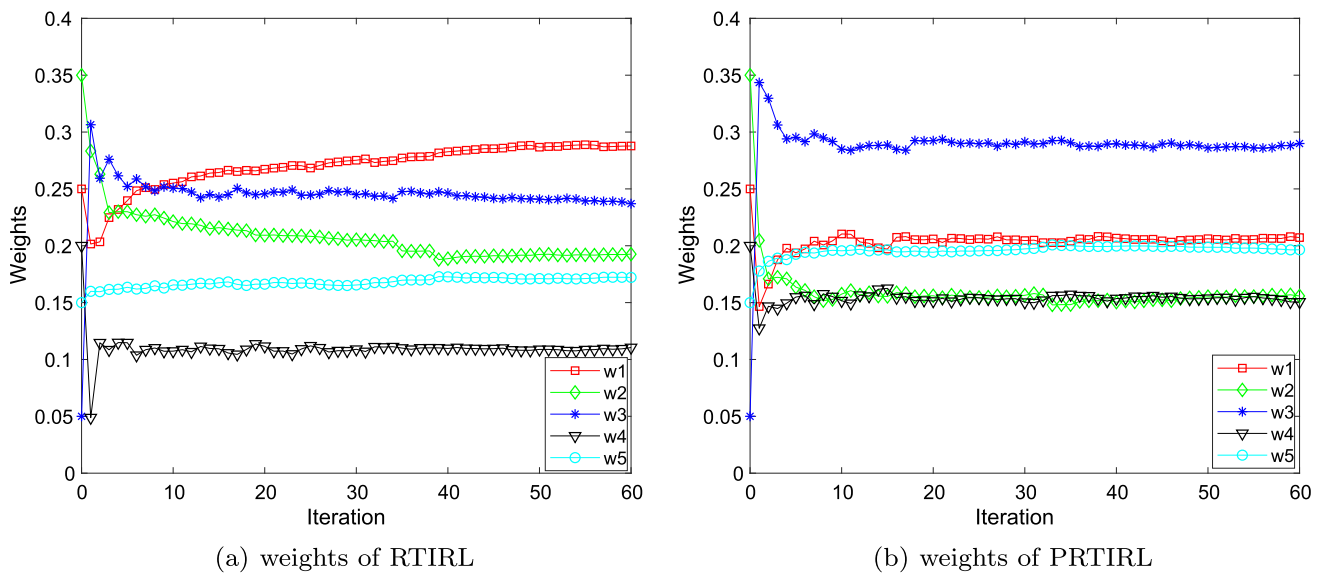
(a) weights of RTIRL

(b) weights of PRTIRL

Fig. 7 Evolution of the weights during the learning iterations. (a) Weights trained by RTIRL. (b) Weights trained by PRTIRL

we control the robot through the remote control to generate demonstration path in the designed scenario. The demonstration path is composed of a series of discrete nodes with time stamp information. After that, the sensor data (static obstacle information), demonstration path, pedestrian pose, start, and goal are packed into a dataset file.

4.2 Objective Evaluation Metric

Figure 7 respectively shows the convergence of the weights corresponding to the features mentioned in Sect. 3.2 along the iterations of the RTIRL and PRTIRL. It can be found from the figure that as the number of iterations proceed, both methods can make weights converge, but the final convergence results are different. Compared with PRTIRL, the final convergence weights of RTIRL is more biased towards $w1$ (the distance to goal), and the pedestrian-related weights do not converge correctly. Therefore, as shown in Fig. 8, the planned paths in some scenarios cause potential interference to the normal movement of pedestrians.

We selected two metrics for objective evaluation: dissimilarity and Homotopic rate.

Dissimilarity is numerically approximately equal to the area of a closed area composed of two paths (demonstration path and planned path) with the same start and goal calculated according to the Riemann integral, and the difference between the two paths is vividly and visually compared. The outgoing path is closer to the demonstration path. The smaller the dissimilarity, the more similar the planned path is to the demonstration road.

As shown in Fig. 8, pedestrian 1 will be blocked by the path (green) planned by the robot during the process of
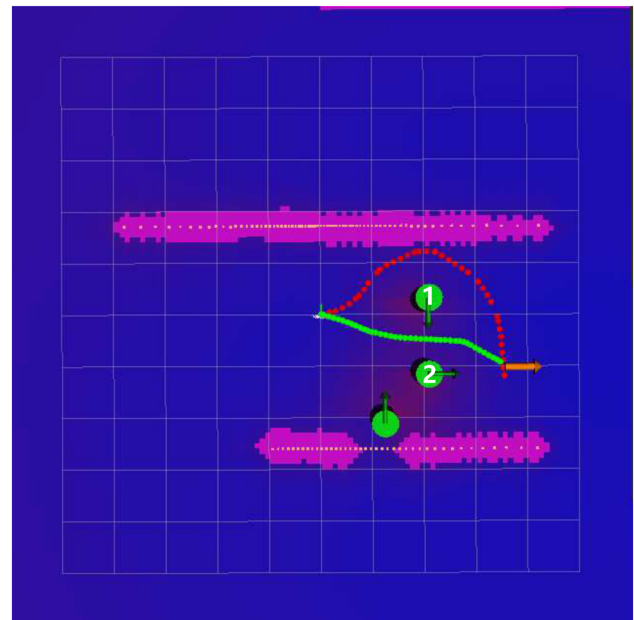

Fig. 8 The relationship between path homotopic and navigation effect.The red discrete points in the figure represent the demonstration path, and the green discrete points represent the path generated by RTIRL. The green column indicates the pedestrian, and the green arrow indicates the direction of the pedestrian. The orange arrow indicates goal. (Color figure online)

walking to pedestrian 2, interrupting the approach of the two and interfering with the pedestrian's original trajectory. In this situation, the pedestrian needs to sacrifice psychological comfort to compromise the robot's movement. In comparison, although the demonstration path is long, it can ensure that pedestrians' actions are not disturbed, and ensure the pedestrian's physical safety and psychological comfort.
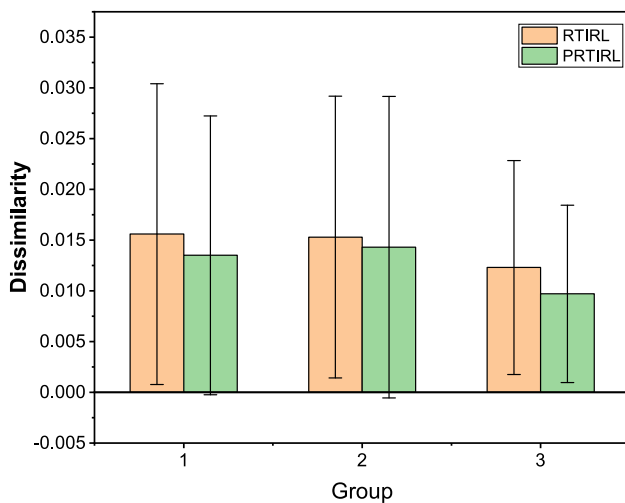
Fig. 9 Dissimilarity for the 3 groups

Therefore, we determine that the planned path similar to the path in Fig. 8 is a manifestation of the failure of socially adaptive navigation, and whether the path homotopic also fits this feature. Thence we use the homotopic rate of the path to represent the degree of anthropomorphism of the planned path on the other hand.

The data set contains 25 sets of scenarios, corresponding to 25 demonstration paths, and the area of all scenarios is $12*12m^2$. In order to comprehensively compare the generalization of the scheme, cross-validation is adopted. 15 groups are used as the training set training feature weights, and 10 groups are used as the verification set to analyze the path dissimilarity and the homotopic rate of the planned paths. A total of three groups of experiments are done. The dissimilarity and homotopic rate obtained through the verification set are shown in Fig. 9 and Table 1. It can be concluded that by introducing non-homotopic detection and penalty function in the training process, the weights corresponding to pedestrian-related features are adjusted in a targeted manner, enentually making the path planned by our method more suitable for the demonstration path and improving the homotopic rate to a considerable degree.

Overview

A questionnaire is designed divided into four different tests to complete the evaluation method of the robot's social navigation capabilities from the perspective of human psychology. 33 participants coming from place [1] were invited to participate in this questionnaire, completing all stages of the questionnaire individually. Finally, 30 of these samples were effectively recycled with an average age of 25.15 and standard deviation of 1.29. The average time used to finish the questionarie is 11.8min. In order to increase the representativeness and diversity of the questionnaire, the

---

[1] School of Mechanical and Electric Engineering, Soochow University

research directions of the 30 participants range from traditional mechanical engineering lacking robot experience to the field of mobile robots classified into industrial robot group, divided into 5 groups (see Table 2). There are only two participants in the direction of mobile robots related to the thesis, ensuring the objective and fairness of the questionnaire survey.

4.3 Subjective Evaluation Metric

The questionnaire has gone through four stages. The general scheme of the questionnaire is presented in Fig. 10. Table 3 lists the studies that we have conducted in questionarie along with the objectives of each stage.

First, several priori questions related to daily behavior of participants were proposed to record the issues that participants pay attention to during daily social interaction. Second, the behavior habits and concerns recorded in **Stage 1** are verified according to the interaction path selected by participants in **Stage 2**. The degree of correlation between the two stages determines the validity of the questionnaire. In the third stage of questionarie, the anthropomorphic confidence of path generated based on PRTIRL is investigated by conducting a Turing-like test. Last but not least, we utilize a Likert scale to evaluate the superiority of our proposed algorithm in comparison with the state-of-the-art planning algorithm.

Stage 1-Collect Behaviors In this section, to collect the participants' daily behaviors on the interaction with other people and issues for robots in human-robot interaction, we proposed several qualitative questions with relative results in Table 4. It can be qualitatively analysed that in order to avoid embarrassment, 73.33% of participants choose to detour when facing a scenario with a dense flow of people. Different participants may have diverse ways of thinking, herein dissimilar strategies will also be adopted. Obviously, it is too arbitrary to judge the validity of the questionnaire on this basis, and the second part must be combined to jointly determine whether it can be effectively recycled. It is also clear from Table 4 that all participants account that robots need to fully respect human behavior and habits in human-robot interaction. In addition, the precautions for robot navigation in such scenes are proposed from different angles, such as the distance between pedestrians and robot, the angle between pedestrians and robot, the length of path planned by robot and the relative speed of pedestrians and robot.

Stage 2-Comparision between Paths Generated by A* and Demo Paths In order to verify the quality of the questionnaire, we combined the collection of daily behavior in **Stage 1** with the figures of scenarios to comprehensively check the credibility of the questionnaire. We designed six groups of representative interactive scenarios, each of which

Table 1 Dissimilarity means and standard errors committed by RTIRL and PRTIRL

	RTIRL			PRTIRL		
Group No	1	2	3	1	2	3
Dissim	0.0156	0.0153	0.0123	0.0135	0.0143	0.0097
Stddev	0.0148	0.0139	0.0105	0.0137	0.0149	0.0087
Stderror	0.0025	0.0022	0.0017	0.0023	0.0023	0.0014
Homotopic Rate	77.5%	60%	80%	90%	87.5%	95%

Table 2 Distribution of participants' research directions

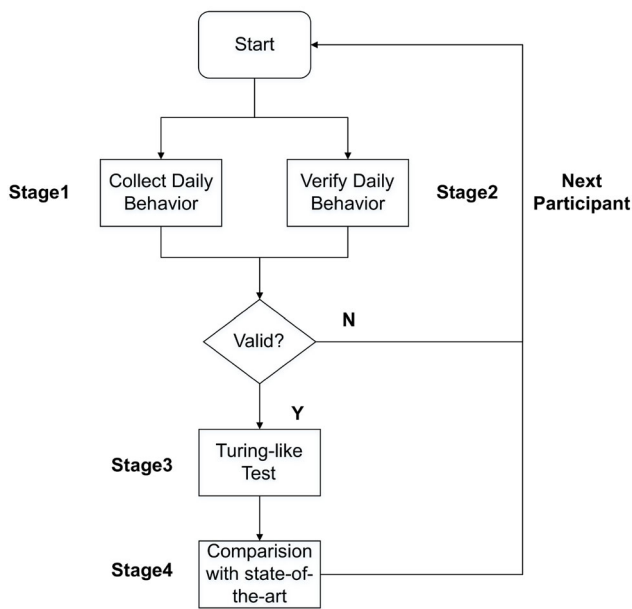| Mechanical Engineering	Materials Engineering	Industrial Robot	Medical Robot	Microsystems
7	9	7	4	3

Fig. 10 The process of subjective experiment

consists of two pictures and two paths (see in Fig. 11). In the interactive scenario shown in Fig. 11, the paths generated by A* are compared with the artificially set demonstration

paths, and the unit length of each grid is 1m. A question is proposed for participants based on Fig. 11:

In the following scenario, which path best suits your behavioral habits?

Then, the results of the two stages of the survey were combined to verify the validity of the questionnaire. If a participant chooses to cross the crowd in **Stage 1**, then the participant should prefer to choose the path generated by the A* algorithm in **Stage 2**, which means that the personal comfort is not taken into account by the participant in social interaction. If the participant chooses the demo path in**Stage 2**, it means that the investigation results of the two stages are inconsistent, and the questionnaire is regarded as invalid. Conversely, if a participant chooses to bypass the crowd in **Stage 1**, then the participant should prefer to choose the demo path in **Stage 2** as a personal habit; if the participant chooses the path generated by the A* algorithm in **Stage 2**, it means that the investigation results of the two stages are inconsistent, and the questionnaire is also regarded as invalid.

According to this method, out of 33 questionnaires, 3 were excluded, and 30 sample papers were effectively recovered. In the six designed scenarios, the percentages of participants choosing demonstration paths are shown in Fig. 12, indicating that the absolute majority of participants feel

**Table 3** List of experiments conducted and their objectives

Experiments	Objectives
Stage 1: Investigation of daily habits	1-To collect behavior of participants Qualitatively 2-As one of the assessment methods of the validity of the questionnaire
Stage 2:Comparison between the path generated by A* planner and the demonstration path	1-To validity behavior of participants Qualitatively 2-As one of the assessment methods of the validity of the questionnaire
Stage 3:Turing-like test	To calculate the anthropomorphic confidence of the navigation trajectory quantitatively
Stage 4:Comparision between paths based on RTIRL and PRTIRL	To highlight the effect of PRTIRL in solving non-homotopic path optimization quantitatively

Table 4 Collection of behavioral habits and precautions for human-robot interaction

Questions	Results
Question 1: When faced with a scene with a lot of people, (a)go around from a place with fewer people; (b)go straight through the crowd	(a)73.33%; (b)26.67%
Question 2: Do you think robots need to respect people's daily behaviors and habits during human-robot interaction?	(a)100%; (b)0%
Question 3: What factors do you think robots need to consider in the process of human-robot interaction?	1.distance between pedestrians and robot 2.angle between pedestrians and robot 3.length of path planned by robot 4.relative speed of pedestrians and robot
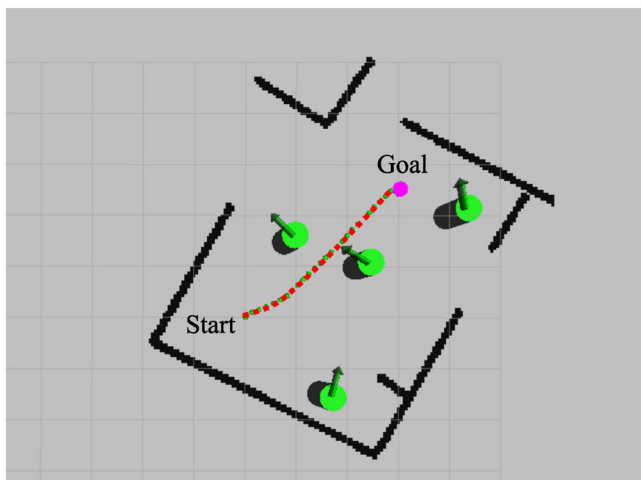
(a) A*
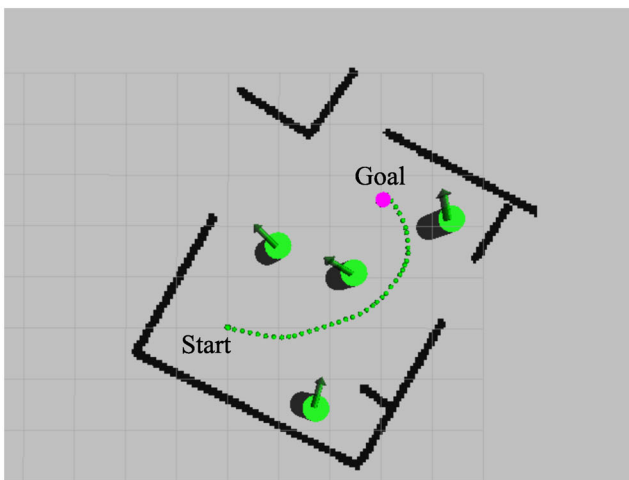
(b) demo path

Fig. 11 Comparison between the path generated by A* and the demonstration path. The path is marked by green discrete nodes, goal point is marked in red, the other end of the path is the starting point. The green column indicates the pedestrian, and the green arrow represents the direction the pedestrian is facing. (Color figure online)
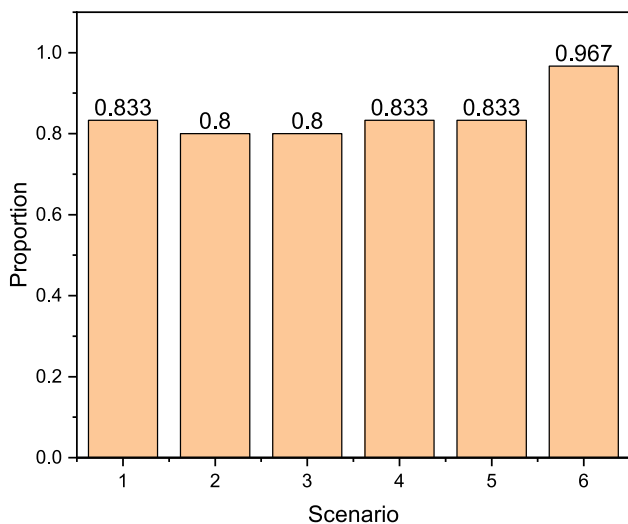

Fig. 12 Proportion of selecting demonstration path

more friendly to demonstration paths. Although the paths planned by A* planner can reach the goal faster, it ignores the interaction of pedestrians, which are typically rejected by participants. In certain scenarios, participants tend to be subjective in thinking. Thus the existance of certain fluctuations is also resonable, as long as satisfying the validition requirements of the questionnaire.

Stage 3-Turing-like Test: Assessment of Anthropomorphic Confidence of Path Generated based on Our Frame Inspired by the Turing test, the demonstration paths and the paths planned by robot based on PRTIRL are mixed together in a certain proportion to form a navigation trajectory data set of 16 scenes. The probability that the robot path is mistaken as a demonstration path is calculated by Eq. (7), which is numerically equal to the anthropomorphic confidence(ac) of the navigation path. One of scenarios and path are shown in Fig. 13. A question is proposed for participants based on Fig. 13:
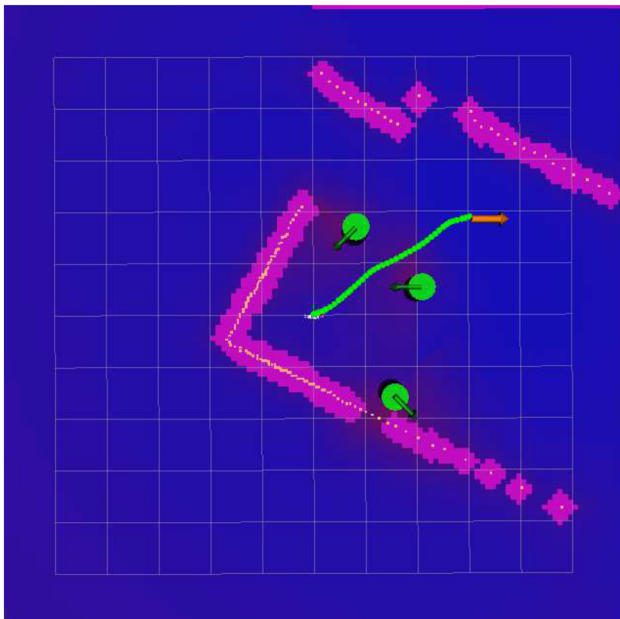
Fig. 13 Turing Test. The annotations are the same as in Fig. 8. The barrier expansion layer is marked in pink. The orange arrow indicates goal

Which path do you think the path in the graph is more likely to belong to?
(A) demonstration path; (b) robot path

The final anthropomorphic confidence of social navigation scheme based on PRTIRL is 70.77%, as far as we know, this evaluation method for assessing the quality of social navigation is the first to be proposed, indicating that our method can behave socially adaptive during human-robot interaction.

$$ac = \sum_{i=1}^{S}\sum_{j=1}^{N} \frac{mis(ij)}{n_{PRTIRL} * N} \tag{7}$$

$$mis(ij) = \begin{cases} 1 & \text{if robot path mistaken} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where S is the number of total scenarios, N is the number of participants, $mis(ij)$ represents whether the path based on

Table 5 The relationship between behavioral compliance and score

Strongly agree	Agree	Common	Disagree	Strongly disagree
5	4	3	2	1

PRTIRL is mistaken as demonstartion paths, and N_{PRTIRL} is the number of paths based on PRTIRL in scenarios.

Stage 4-Comparision between Paths based on RTIRL and PRTIRL In order to verify the effect of social navigation after the introduction of non-homotopic optimization, a total of 9 scenes were designed, and each scene uses the trained results of RTIRL and PRTIRL to guide RRT* to generate paths (see Fig. 14). The participants are asked to score the path in the scene according to their behavior habits, the items were evaluated on a 5-point Likert scale (see Table 5). A question is proposed for participants based on Fig. 14:

Which path do you think is more in line with your behavior in the next group of figures? please rate them separately

The score is sorted out by Eq. (9):

$$score_{kind} = \sum_{i=1}^{S}\sum_{j=1}^{N} \frac{P(ij)_{kind}}{S * N}. \tag{9}$$

PRTIRL-kind=1, RTIRL-kind=0; $P(ij)_{kind}$ indicates the path score, referring to Table 5. The final average scores of RTIRL and PRTIRL are shown in Table 6. According to the scores of 30 participants on the two paths in 9 scenes, we choose the average score of the two paths in each scene as the measurement scale (Fig. 15). In most scenarios, the paths generated by our method are more popular with participants, especially when non-homotopic paths appear in RTIRL, participants will give very low scores according to the likert table. After introducing non-homotopic detection and penalty function, PRTIRL will fully respect the comfort space of pedestrians and avoid generating a planned path that is not homotopic with the demonstration path, so it is more recognized by participants.
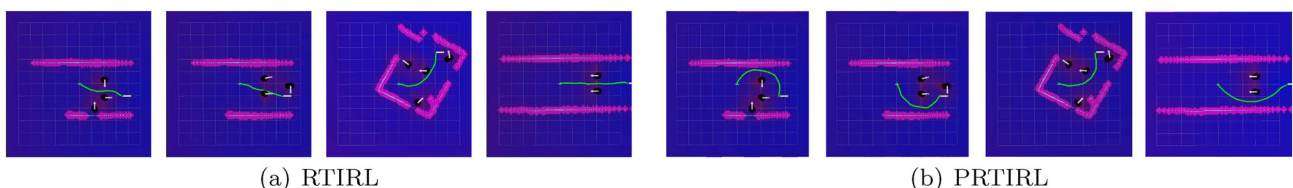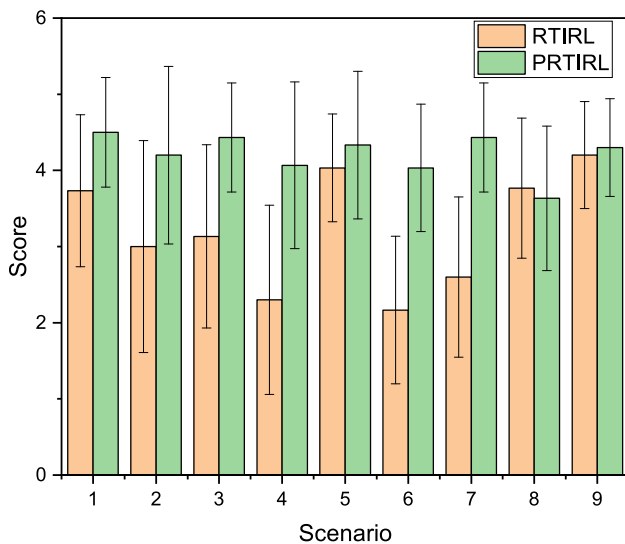

(a) RTIRL   (b) PRTIRL

Fig. 14 Comparison between the paths generated by RTIRL and PRTIRL. Part of the labels is the same as Fig. 8

Table 6 The scores of RTIRL and PRTIRL

RTIRL	PRTIRL
3.21	4.36

Fig. 15 Average score of RTIRL and PRTIRL in all 9 scenarios

5 Conclusions

We propose Penalty Rapidly-exploring random Trees Inverse Reinforcement Learning (PRTIRL), which is used to guide robots to learn navigation behaviors with social adaptability from demonstrations. Using RRT as a path planner, through Inverse reinforcement learning the cost function of RRT*, the problem of robot navigation in high-dimensional and complex human-robot interaction scenarios can be quickly and effectively solved. We introduced non-homotopic detection and penalty functions in the inverse reinforcement learning module to improve the performance of the algorithm in complex human-robot integrating scenarios and increase the generalization of behavior.

In addition, the objective data and subjective investigation are combined to optimize the evaluation method of the robot's socially adaptive navigation capability. The results show that the path generated by our method in the interactive scenario with denser traffic and narrower space can be closer to the demonstration path and achieve satisfactory results.

References

1. Hall ET, Birdwhistell RL, Bock B, Bohannan P, Diebold AR, Durbin M, Edmonson MS, Fischer JL, Hymes D, Kimball ST, W. La Barre, , J. E. McClellan, Marshall DS, Milner GB, H. B. Sarles, Trager GL, and Vayda AP (1968) "Proxemics [and comments and replies]," Current Anthropology, vol. 9, no. 2/3, pp. 83–108

2. Ambhore S (2020) "A comprehensive study on robot learning from demonstration," in 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)

3. N. Pérez-Higueras, Caballero F, and Merino L (2017) "Teaching robot navigation behaviors to optimal rrt planners," Int J Soc Robot

4. Ziebart BD, Maas AL, Bagnell JA, and Dey AK (2008) "Maximum entropy inverse reinforcement learning," in Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008

5. Karaman S, Frazzoli E (2010) "Incremental sampling-based algorithms for optimal motion planning," CoRR, vol. abs/1005.0416. [Online]. Available: arXiv:1005.0416

6. Chi W, Wang C, Wang J, Meng MQ (2019) Risk-dtrrt-based optimal motion planning algorithm for mobile robots. IEEE Trans Autom Sci Eng 16(3):1271–1288

7. Bhattacharya S, Kumar V, and Likhachev M (2010) "Search-based path planning with homotopy class constraints," in Third Annual Symposium on Combinatorial Search

8. Kim B, Pineau J (2015) "Socially adaptive path planning in human environments using inverse reinforcement learning," Int J Soc Robot, pp. 1–16, 01

9. Lee J, "A survey of robot learning from demonstrations for human-robot collaboration," CoRR, vol. abs/1710.08789, 2017. [Online]. Available: arXiv:1710.08789

10. Foka AF, Trahanias PE (2010) "Probabilistic autonomous robot navigation in dynamic environments with human motion prediction,". Int J Soc Robot 2(1):79–94

11. Ziebart BD, Ratliff ND, Gallagher G, Mertz C, Srinivasa SS (2009) "Planning-based prediction for pedestrians," in 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems

12. Garzón M, Fotiadis EP, Barrientos A, and Spalanzani A (2013) "Riskrrt-based planning for interception of moving objects in complex environments," Advances in Intelligent Systems & Computing, vol. 253

13. Otte M, Frazzoli E (2016) Rrt x: Asymptotically optimal single-query sampling-based motion planning with quick replanning. Int J Robot Res 35(7):797–822

14. Seder M, Petrovic I (2007) "Dynamic window based approach to mobile robot motion control in the presence of moving obstacles," in Proceedings 2007 IEEE International Conference on Robotics and Automation, pp. 1986–1991

15. Svenstrup M, Bak T, Andersen HJ (2010) "Trajectory planning for robots in dynamic human environments," in 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2010, pp. 4293–4298

16. Helbing D, Molnar P May (1995) "Social force model for pedestrian dynamics," Physical Review E, **51**(5), pp. 4282–4286. [Online]. Available: https://doi.org/10.1103/PhysRevE.51.4282

17. Huang K, Li J, Fu L (2010) Human-oriented navigation for service providing in home environment. Proceedings of SICE Annual Conference 2010:1892–1897

18. Lam CP, Chou CT, Chang CF, and Fu LC "Human-centered robot navigation–toward a harmoniously coexisting multi-human and multi-robot environment," in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on 2010

19. Chi W, Kono H, Tamura Y, Yamashita A, and Meng QH (2016) "A human-friendly robot navigation algorithm using the risk-rrt approach," in 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR)

20. Kalman RE (1963) UndeterminedWhen is a linear control system optimal? UndeterminedJoint Automatic Control Conference 1:1–15

21. Andreu-Perez J, Deligianni F, Ravi D, and Yang G (2018) "Artificial intelligence and robotics," CoRR, vol. abs/1803.10813, 2018. [Online]. Available: http://arxiv.org/abs/1803.10813

22. Ziebart BD, Ratliff N, Gallagher G, Mertz C, Peterson K, Bagnell JA, Hebert M, Dey AK, Srinivasa S (2009) "Planning-based prediction for pedestrians," in 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009, pp. 3931–3936

23. Henry P, Vollmer C, Ferris B, Fox D (2010) "Learning to navigate through crowded environments," in 2010 IEEE International Conference on Robotics and Automation, 2010, pp. 981–986

24. Agarwal P, Kumar S, Ryde J, Corso J, Krovi V, Ahmed N, Schoenberg J, Campbell M, Bloesch M, Hutter M, Hoepflinger M, Leutenegger S, Gehring C, Remy CD, Siegwart R, Brookshire J, Teller S, Bryson M, M. Johnson-Roberson, Pizzaro O, Williams S, Colonnese N, Okamura A, Dey D, Liu TY, Hebert M, J. A. Bagnell, Dogar MR, Hsiao K, Ciocarlie M, Srinivasa S, Douillard B, N. Nourani-Vatani, Roman C, Vaughn I, Inglis G, Dragan A, Gillula J, Tomlin C, Gilpin K, Rus D, Gioioso G, Salvietti G, Malvezzi M, Prattichizzo D, Goodrich M, Kerman S, Pendleton B, Sujit PB, Hartmann C, Boedecker J, Obst O, Ikemoto S, Asada M, Hauser K, Kanda T, Jaillet L, Porta JM, Jain A, Crean C, Kuo C, Bremen HV, Myint S, Joho D, G. D. Tipaldi, Engelhard N, Stachniss C, Burgard W, Julian B, S. L. Smith, Kazemi M, J.-Valois S, Pollard N, Kovac M, Bendana M, Krishnan R, Burton J, Smith M, Wood R, Kuderer M, Kretzschmar H, Sprunk C, Kuindersma S, Grupen RA, Barto AG, Kunz T, Stilman M, Kushleyev A, Kumar V, Mellinger D, Laffranchi M, Tsagarakis N, Caldwell D, Latif Y, Lerma CC, Neira J, Li M, Mourikis A, Lim S, Sommer C, Nikolova E, Liu L, Shell D, Long A, Wolfe K, Mashner M, Chirikjian G, Lysenkov I, Eruhimov V, Bradski G, Maddern W, Milford M, Wyeth G, Marthi B, Nakanishi J, Vijayakumar S, Olson E, Agarwal P, Paprotny I, Levey C, Wright P, Donald BR, Q.-C. Pham, Nakamura Y, Phillips M, Cohen B, Chitta S, Likhachev M, Qian F, Zhang T, Li C, Hoover A, Masarati P, Birkmeyer P, Pullin A, Fearing R, Goldman D, Rawlik K, Toussaint M, Robin C, Lacroix S, Rombokas E, Malhotra M, Theodorou E, Matsuoka Y, Todorov E, Sarid S, Xu B, H. Kress-Gazit, Schepelmann A, Geyer H, Taylor M, Sentis L, Petersen J, Philippsen R, Taylor C, Cowley A, Tellex S, Thaker P, Deits R, Simeonov D, Kollar T, Roy N, Vernaza P, Narayanan V, Wang H, Hu G, Huang S, Dissanayake G, Wang Z, Deisenroth MP, H. B. Amor, Vogt D, B. Schölkopf, Peters J, Wilcox R, Nikolaidis S, Shah J, Wolff E, Topcu U, Murray R, Yoo C, Fitch R, Sukkarieh S, Zarubin D, Ivan V, Komura T, Zelazo D, Franchi A, F. Allgöwer, H. Bülthoff, and Giordano PR (2013) Feature-Based Prediction of Trajectories for Socially Compliant Navigation, pp. 193–200

25. Imani M, Braga-Neto U (2018) "Control of gene regulatory networks using bayesian inverse reinforcement learning," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. PP, pp. 1–1, 10

26. Okal B, Kai OA (2016) "Learning socially normative robot navigation behaviors using bayesian inverse reinforcement learning," in IEEE International Conference on Robotics and Automation (ICRA)

27. Shiarlis K, Messias J, Whiteson S (2017) "Rapidly exploring learning trees," in 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 1541–1548

28. Werner M, Feld S (2014) "Homotopy and alternative routes in indoor navigation scenarios," in 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 230–238

29. Hernandez E, Carreras M, Ridao P (2015) A comparison of homotopic path planning algorithms for robotic applications. Robot Auton Syst 64:44–58

30. Turing AM (2009) Computing Machinery and Intelligence. Springer Netherlands, Dordrecht, pp 23–65

31. Quigley M, Gerkey B, Conley K, Faust J, Foote T, Leibs J, Berger E, Wheeler R, and Ng A (2009) "Ros: An open-source robot operating system," ICRA Workshop on Open Source Software, **3**, pp. 1–6, 01

32. Pineau J, and Atrash A (2007) "Smartwheeler: A robotic wheelchair test-bed for investigating new models of human-robot interaction." Proc IEEE, pp. 59–64

Zhiyu Ding received the bachelor degree in mechanical and electronic engineering from Nanjing Institute of Technology, Nanjing, China in 2018. He received the master degree with the School of Mechanical and Electrical Engineering, Soochow University. His research field includes mobile robot social navigation and motion planning.

Jie Liu received the bachelor degree in mechanical and electronic engineering from Nanjing Institute of Technology, Nanjing, China in 2018. He received the master degree with the School of Mechanical and Electrical Engineering, Soochow University. His research field includes mobile robot exploration and motion planning.

Wenzheng Chi received her B.E. degree in automation from Shandong University, Jinan, China, in 2013 and the Ph.D. degree in Biomedical Engineering from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2017. During which she spent six months at The University of Tokyo, Japan, as a Visiting Scholar. She was a Post-Doctoral Fellow with the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, from 2017 to 2018. She is currently an Associate Professor with the Robotics and Microsystems Center, School of Mechanical and Electric Engineering, Soochow University, Suzhou, China. Her research interests include mobile robot path planning, intelligent perception, human-robot interaction, etc.

Jiankun Wang received the B.E. degree in automation from Shandong University, Jinan, China, in 2015, and the Ph.D. degree from the Department of Electronic Engineering, Chinese University of Hong Kong, Hong Kong, in 2019. He is currently a Research Assistant Professor with the Department of Electronic and Electrical Engineering, Southern University of Science and Technology, Shenzhen, China. During his Ph.D. degree, he spent six months with Stanford University, Stanford, CA, USA, as a Visiting Student Scholar supervised by Prof. O. Khatib. His current research interests include motion planning and control, human robot interaction, and machine learning in robotics.

Guodong Chen was born in 1983. He received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2011. He is currently an Associate Professor with Soochow University. His research interests include robot vision and intelligent industrial robots.

Lining Sun was born in Hegang, Heilongjiang Province, China in January 1964. In 1993, Prof. Sun received his Ph.D. degree in Engineering from the Mechanical Engineering Department of Harbin Institute of Technology (HIT), and joined HIT after graduation. He is a National Outstanding Youth Fund Winner, Changjiang Scholar Distinguished Professor by the Ministry of Education. He formed an internationally competitive robotics team and made outstanding contributions to the creation and development of robot-related disciplines in China.