Learning Human-Aware Robot Navigation from Physical Interaction via Inverse Reinforcement Learning

Marina Kollmitz[1] Torsten Koller[1] Joschka Boedecker[1] Wolfram Burgard[1,2]

***Abstract*— Autonomous systems, such as delivery robots, are increasingly employed in indoor spaces to carry out activities alongside humans. This development poses the question of how robots can carry out their tasks while, at the same time, behaving in a socially compliant manner. Further, humans need to be able to communicate their preferences in a simple and intuitive way, and robots should adapt their behavior accordingly. This paper investigates force control as a natural means to interact with a mobile robot by pushing it along the desired trajectory. We employ inverse reinforcement learning (IRL) to learn from human interaction and adapt the robot behavior to its users' preferences, thereby eliminating the need to program the desired behavior manually. We evaluate our approach in a real-world experiment where test subjects interact with an autonomously navigating robot in close proximity. The results suggest that force control presents an intuitive means to interact with a mobile robot and show that our robot can quickly adapt to the test subjects' personal preferences.**

Fig. 1. A person communicates her personal space preference by pushing the navigating robot away from her (left). Based on this interaction, the robot adapts its navigation behavior and learns to keep a greater distance to people (right).

I. Introduction

Robots that share their workspaces with people need to consider human comfort and safety to be tolerated and accepted. In addition, robots operating in public spaces may encounter people with little experience with robots. In this work, we investigate physical interaction for communicating robot navigation preferences. Physical gestures like pushing or guiding are familiar from our everyday experience, like pushing a shopping cart or guiding an elderly person by the arm. Thus, physical interaction has great potential for intuitive human-robot communication.

Imagine a delivery robot operating in a hospital. The robot has to ensure the safety and comfort of patients, hospital staff, and visitors when operating in their proximity. At the same time, the robot has a task-related objective of transporting items from one location to another as quickly as possible. Both objectives may contrast each other, and a reasonable trade-off is required to do both tasks well.

To balance navigation objectives, path planning is often formulated as an optimization problem, where the objectives are represented as costs. Traditionally, path planning aims to find the shortest or fastest path, but additional social costs have been formulated for keeping appropriate interaction distances [1, 2, 3], avoiding to pass behind a person [1], or preferring one side for passing [2]. Adjusting the cost function parameters for the desired robot behavior is not straight forward since it can depend on many factors. The preferred interaction distance, for example, is influenced by the task and role of the robot [4], the person's gender and familiarity with robots [5], and the appearance [6, 7] and speed [8] of the robot.

In this paper, we propose to learn the parameters of the navigation cost function through physical human-robot interaction. To this end, we formulate a cost function that balances social space preferences and the desire to reach a goal location. During autonomous operation, people can correct the behavior of a navigating robot by pushing it along the trajectory they would prefer, as depicted in Fig. 1. We regard the corrected trajectories as expert demonstrations of the desired robot behavior and use maximum entropy inverse reinforcement learning to adapt the cost function parameters accordingly. As a result, the robot learns to balance task objectives and social constraints, allowing it to travel on socially compliant paths.

While kinesthetic teaching has been primarily researched for robot manipulators [9], this is the first work in the socially compliant robot navigation context. Our approach does not presume any particular skills or experience regarding robot control. Furthermore, since the interaction does not require an external control device, the robot can refine its navigation behavior over time and continuously adapt to the people it interacts with. In experiments and a user study with our mobile robot Canny, we confirm that the robot can improve its navigation behavior based on force feedback and that this method of interaction is easy and intuitive. Our code and experiment data are available at `https://github.com/marinaKollmitz/learning-nav-irl`.

*This work has been partially supported by the German Federal Ministry of Education and Research (BMBF), contract number 01IS15044B-NaRKo.

[1]Department of Computer Science, University of Freiburg, Germany. {kollmitz, kollert, jboedeck, burgard}@informatik.uni-freiburg.de

[2]Toyota Research Institute, Los Altos, USA

II. Related Work

Various approaches have been proposed to represent human preferences during navigation. Pacchierotti et al. [10] developed a passing module that controls the signaling and lateral passing distance between the robot and people in a corridor passing setting. Other works model navigation preferences of people as costs for navigation to cover a broader range of navigation scenarios. Kirby et al. [2] modeled social space requirements and a tendency to pass a person on the right side as additional costs for path planning. Aspects of comfort, safety, and visibility are included as additional navigation constraints in work by Sisbot et al. [1].

Tuning the cost function parameters for the desired robot behavior for different robots, environments, and tasks is not straight forward. Instead, various approaches learn robot navigation behavior via human demonstrations or observations. Trautman and Krause [11] and Luber et al. [12] learned human-like navigation from top-view pedestrian scenes to plan socially acceptable paths among humans. Kuderer et al. [13] optimized joint collision avoidance models, where all agents are expected to cooperate during navigation, via inverse reinforcement learning. To this end, they observed avoidance trajectories of people frequently passing each other in an open area. Ziebart et al. [14], as well as Bennewitz et al. [15], used observations from people walking inside office environments to learn human path prediction models for hindrance-free robot navigation.

The approaches presented above all learned navigation behavior from observing humans. However, robots are not necessarily expected to behave human-like around people [16]. In contrast, Lichtenthäler et al. [17] proposed an "inverse Oz-of-Wizard" approach where people teleoperated the robot in a path crossing scenario to demonstrate how they expected the robot to behave. Similarly, Kim and Pineau [18] collected demonstrations via teleoperation in crowded navigation scenarios to learn a cost function using inverse reinforcement learning. They used a cost function with binary features, while ours is continuous and explicitly models personal space. Herman et al. [19] investigated inverse reinforcement learning for navigation scenarios where both the cost function parameters and the state transition dynamics are unknown. In their work, humans provided demonstrations by controlling a simulated mobile robot in a populated hallway scenario.

While it is possible to directly demonstrate how a robot should behave in a given situation through teleoperation, the demonstrators are not actively involved in the interaction. They have to anticipate the preferences of the people the robot interacts with. In our approach, people can directly demonstrate their own intent via physical interaction. Furthermore, since no external control device is required, the robot can continue to learn from demonstrations over time.

Physical interaction has been frequently employed for teaching via demonstration, often in the context of force-sensitive manipulators [20, 21]. Our work is inspired by Bajcsy et al. [22], who corrected the behavior of their robot during autonomous task execution. Similar to their work, we want to use physical interaction to enable humans to correct the current behavior of the robot according to their preferences. While they focus on tabletop tasks with a robot manipulator, our goal is to find an appropriate model for human preferences in the robot navigation domain.

So far, force-sensitive mobile robots have mostly responded to physical interaction in a reactive manner. Walking helper systems, like the ones presented by Sabatini et al. [23] and Spenko et al. [24], commanded a velocity based on force feedback but did not actively navigate by themselves. Khatib [25] presented an approach for load sharing, where people manipulated heavy objects in cooperation with mobile manipulators. Hirata et al. [26] also considered cooperative object transportation where users guided mobile robot helpers along a pre-planned trajectory. Load sharing policies for cooperative transportation have been investigated by Lawitzky et al. [27]. In their work, the task completion time and the user effort could be reduced when robots proactively worked towards the goal instead of reacting passively to user input. Later, Lawitzky et al. [28] proposed to learn the motion paths for cooperative transport of heavy objects via physical human-robot interaction to provide proactive assistance. In this paper, we also want the robot to adapt its behavior based on physical interaction instead of passively reacting to user inputs. The robot considers the user-adapted trajectories as demonstrations of its desired behavior and adapts its navigation cost function to match the demonstrations.

III. Learning From Physical Interaction

This section presents our approach to learning navigation behavior from physical interaction. Sec. III-A briefly introduces our force-sensitive robot Canny and explains how people can demonstrate their desired robot behavior. Sec. III-B gives an overview of the inverse reinforcement learning technique we employ to learn from the resulting demonstrations. Finally, Sec. III-C introduces our navigation reward function to model the navigation task. Note that while the term cost function is prevalent for path planning, we will, in the following, use the term reward function that is common in the (inverse) reinforcement learning context.

A. Demonstrating Desired Robot Behavior

Our robot Canny uses a 6-DoF force-torque sensor between its omnidirectional mobile base and its solid shell to perceive interaction forces [29]. During autonomous navigation, the robot commands a navigation velocity $\mathbf{v}_{\text{nav}} = (v_{\text{nav},x}, v_{\text{nav},y})^T$ to follow its navigation path. The robot further overlays external command velocities $\mathbf{v}_{\text{com}}$ to include user feedback. The resulting robot velocity is

$$\mathbf{v}_r = \mathbf{v}_{\text{nav}} + \mathbf{v}_{\text{com}}. \tag{1}$$

To integrate force feedback, an interaction force $\mathbf{F} = (F_x, F_y)^T$ is translated to a proportional velocity command,

$$\mathbf{v}_{\text{com},\mathbf{F}} = k \cdot \mathbf{F}, \tag{2}$$

where k (in $[\mathrm{s\,kg^{-1}}]$) is the proportionality factor. Other input modalities can be integrated to adapt the robot behavior, following Eq. 1.

B. Learning Human-Aware Navigation from Demonstrations

We see the user-adapted paths as demonstrations of the desired robot behavior and use Inverse Reinforcement Learning (IRL) to adapt the reward function to produce similar behavior without human interaction. We model the robot navigation task as a Markov Decision Process (MDP) $M = (S, A, T, r)$ with states S, actions A, a transition model T, and a reward function r. The 10×10 cm^2 grid cells of the navigation map are the MDP states S, and the actions A are traversing to adjacent map cells in an eight-connected fashion. We assume a deterministic transition model $s' = T(s, a)$, i.e., executing action a from state s always results in the same next state s'. Finally, we are given a reward function r_θ with unknown parameters θ and a set D of demonstrations $\tilde{\tau}_i = (s_0, a_0, \ldots, a_{N_i-1}, s_{N_i})_i$, $i = 1, \ldots, |D|$, of varying lengths N_i. The goal is to recover the parameters θ^* that best explain this set of given demonstrations.

In our work, we learn from human demonstrations, and we cannot assume that humans always act optimally when performing a task. Our approach therefore follows the Maximum Entropy IRL method of Ziebart et al. [30], which processes noisy-optimal demonstrations in a probabilistic fashion. Furthermore, it solves the inherent IRL ambiguity – that many reward functions may be optimal for given demonstrations – by choosing the distribution that remains maximally uncertain beyond matching the expert demonstrations.

For a given reward function r_θ, the Maximum Entropy principle assumes that trajectories τ are distributed according to

$$P(\tau \mid \theta) = \frac{1}{Z(\theta)} \exp(R_\theta(\tau)), \tag{3}$$

where $R_\theta(\tau)$ is the sum of rewards along the trajectory τ, and the partition function $Z(\theta) = \sum_\tau \exp(R_\theta(\tau))$ normalizes the distribution. While the original Maximum Entropy IRL work [30] assumed a reward function that is linear in θ, Wulfmeier et al. [31] showed that the formulation is also suitable for general non-linear, sufficiently smooth reward functions, which we adopt in this work.

To find the reward parameters θ^* that best explain the demonstrated behavior, Maximum Entropy IRL maximizes the log-likelihood of the demonstrations,

$$\theta^* = \arg\max_\theta L(\theta) = \arg\max_\theta \sum_{\tilde{\tau} \in D} \log p\left(\tilde{\tau} \mid \theta\right), \tag{4}$$

which can be optimized using gradient-based techniques. The gradient with respect to the reward parameters,

$$\nabla_\theta L(\theta) = \sum_{\tilde{\tau} \in D} \frac{\partial R_\theta(\tilde{\tau})}{\partial \theta} - |D| \sum_\tau p_\theta(\tau) \frac{\partial R_\theta(\tau)}{\partial \theta}, \tag{5}$$

requires to sum over all trajectories τ, which is intractable in large discrete or even continuous state spaces. Instead, we can unroll the trajectories to state-action pairs to obtain the tractable representation,

$$\begin{aligned} \nabla_\theta L(\theta) &= \sum_{\tilde{\tau} \in D} \sum_{(s,a) \in \tilde{\tau}} \frac{\partial r_\theta(s,a)}{\partial \theta} \\ &- |D| \sum_{t=0}^{\infty} \sum_{(s,a)} p_\theta(a \mid s) p_{\theta,t}(s) \frac{\partial r_\theta(s,a)}{\partial \theta}, \end{aligned} \tag{6}$$

with unknowns $p_\theta(a \mid s)$ and $p_{\theta,t}(s)$. In practice, we replace the infinite time horizon by a predefined or adaptively chosen finite horizon T. The term $p_\theta(a \mid s)$ represents the stochastic policy inducing the distribution over trajectories in Eq. 3. Note that the stochastic policy is different from the optimal policy we would obtain through solving the MDP using the standard Bellman equations. Instead, it was shown in [14] that $p_\theta(a \mid s)$ can be obtained by solving a "softened" version of the Bellman update, given by

$$Q^{\sim}(s,a) = r_\theta(s,a) + V^{\sim}(T(s,a)), \tag{7}$$

$$V^{\sim}(s) = \log \sum_a \exp(Q^{\sim}(s,a)), \tag{8}$$

which can be solved using standard value iteration. The desired stochastic policy is then obtained via,

$$p_\theta(a \mid s) = \exp(Q^{\sim}(s,a) - V^{\sim}(s)), \tag{9}$$

i.e., the probability of choosing an action is proportional to the expected exponentiated future rewards.

Given the stochastic policy $p_\theta(a \mid s)$, we can now obtain the distribution over states $p_{\theta,t}(s)$ at time steps $t = 0, \ldots, T$ by propagating the policy through the state space. Starting from an initial distribution over states $p_0(s)$, all future state distributions can be calculated via

$$p_{\theta,t+1}(s') = \sum_{(s,a)} \mathbb{1}_{\{s' = T(s,a)\}} p_\theta(a \mid s) p_{\theta,t}(s), \tag{10}$$

$\forall s' \in S$, where $\mathbb{1}_{\{\cdot\}}$ denotes the indicator function.

Once the gradient is calculated, we can use standard (stochastic) optimization techniques to solve for the reward function parameters. Since we have to solve Eqs. 6-10 for every iteration of the optimization, calculating the gradient amounts to solving an MDP in every step. While this can become computationally infeasible, the state-action space in the 2D path planning domain is typically small enough for exact solution methods [30, 14, 19], like in this work.

C. A Reward Function for Socially Compliant Navigation

In order to learn social navigation from demonstrations, our robot first requires a general parametric reward function $r_\theta(s,a)$ that explains socially compliant behavior. This function should reflect the desire to reach a predefined goal state s_g, to avoid obstacles, and to acknowledge the personal space of a person standing at a given state s_p.

To model the goal reaching behavior of the robot, we first plan the shortest path, which we denote by $\bar{\tau}$, from the start state s_0 to the goal state s_g. We then penalize the deviation from this nominal path and add a step reward term $r_{\text{step}}(s,a) = ||s - s'||_2$, where $s' = T(s,a)$ is the next state,

weighted by a factor c_0. The goal reaching reward is then given by

$$r_{\mathrm{g}}(s,a) = c_0 \cdot r_{\mathrm{step}}(s,a) + a_{\mathrm{g}} \cdot \min_{\bar{s}\in\bar{\tau}} ||s-\bar{s}||_2^2, \qquad (11)$$

with an additional weight a_{g} on the distance to the nominal path[1].

We consider static obstacles $S_{\mathrm{obs}} \subset S$ as a subset of the state space that needs to be avoided at all times. Since we assume that our dynamics are deterministic, we can simply assign a large penalty term for being in an obstacle state, i.e., $r_{\mathrm{obs}}(s) = -\infty \cdot \mathbb{1}_{\{s\in S_{\mathrm{obs}}\}}$ without introducing conservatism.

To reflect the personal space requirements of a person, we use a squared-exponential reward function,

$$r_{\mathrm{p}}(s) = a_{\mathrm{p}} \cdot \exp(-\sigma_{\mathrm{p}}||s-s_{\mathrm{p}}||_2^2), \qquad (12)$$

parametrized by a weight coefficient a_{p} and scaling coefficient σ_{p} to model the steepness or width of the personal space reward. This type of distance function is well-known to accurately describe the personal space in social navigation [1, 2].

In the terminal state s_g, we set all rewards to zero, i.e., $r_{\mathrm{g}}(s_{\mathrm{g}},a) = r_{\mathrm{p}}(s_{\mathrm{g}}) = 0$. Finally, the overall reward function is given by the sum of the individual reward terms,

$$r_\theta(s,a) = r_{\mathrm{g}}(s,a) + r_{\mathrm{p}}(s) + r_{\mathrm{obs}}(s), \qquad (13)$$

and the parameter set is given by $\theta = \{c_0, a_{\mathrm{g}}, a_{\mathrm{p}}, \sigma_{\mathrm{p}}\}$.

IV. Experiments

We conducted experiments with test persons to investigate whether force control is suitable for interacting with mobile robots and whether the robot can successfully learn human-aware navigation behavior with our model. Sec. IV-A explains the experiment design; the two experiment tasks are evaluated in Sec. IV-B and IV-C, respectively. Finally, Sec. IV-D evaluates the generalization capabilities of the learned reward function.

A. Experiment Design

We recruited thirteen test subjects (5 female, 8 male, aged 21 - 63). Eight participants had a technical background (2f, 6m), and six had previous experiences with robots (2f, 4m). None of the participants were involved in this research project or had prior knowledge of our research aim. The experiments took place in the hallway of our building at the faculty of engineering with our force-sensitive mobile robot Canny. Each experiment run lasted about $45\,\mathrm{min}$, during which the test subjects performed two sets of tasks with the robot. In one set, the participants controlled the robot via force feedback; they used a joystick in the other. We randomized the set order and gave the test subjects some time to familiarize themselves with the respective control method at the beginning of each set.

The joystick experiments serve as a baseline to compare the ease of use and intuitiveness of physical interaction against a more common control method. We further use them as a reference for the true desired robot behavior, since the participants can influence the entire robot trajectory with the joystick, while physical interaction requires the robot within reach. The joystick experiments, however, only provide an idealized and theoretical comparison, because the requirement to carry a controller renders joystick interaction infeasible in practical applications beyond an experimental setup.

[1]We note that, since the function is differentiable wrt. the parameters c_0, a_{g}, the non-differentiability of the max-operator does not pose a difficulty here.

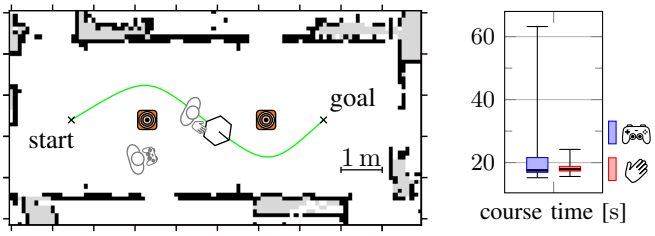

Fig. 2. For the obstacle course task, the participants controlled the robot past two traffic cones as indicated by the sketched green line (left). The box plot (right) shows the median task completion times and the interquartile ranges; the whiskers mark the minimum and maximum completion times.

B. Obstacle Course Task

We first performed a calibration task to compare the ease of use and intuitiveness of both control modalities without the robot learning or moving autonomously. For this task, the participants controlled the robot through an obstacle course, as sketched in Fig. 2 (left). All robot motion was commanded by the participants, and we limited the robot velocity to $0.4\,\mathrm{m\,s^{-1}}$. Each participant performed one obstacle course run per control modality. Fig. 2 (right) summarizes the task completion times with the joystick and with force control. Note that we did not inform the participants that we timed their runs because we wanted them to behave naturally. Both modalities exhibit similar median course times, but the joystick times show a much higher variance. Three participants took exceptionally long with over $30\,\mathrm{s}$ to drive the robot with the joystick. In contrast, no participant took longer than $25\,\mathrm{s}$ to finish the task with force control. The results indicate that force control can facilitate the interaction with mobile robots in populated spaces.

C. Passing Task with IRL

The passing task tests force and joystick feedback for interacting with an autonomously navigating robot. The participants stood in the center of the hallway and guided the robot past themselves, either via joystick or by pushing on its shell. We instructed the participants to adapt the robot path like they would prefer it to autonomously drive past them. The robot navigated with a velocity of $v_{\mathrm{nav}} = \{v_{\mathrm{nav},x}, v_{\mathrm{nav},y}\}$ and overlaid the user-commanded control velocity v_{com}, as described in Sec. III-A. The forward motion $v_{\mathrm{nav},x}$ was kept constant at $0.4\,\mathrm{m\,s^{-1}}$. The robot further aimed to follow its navigation path by commanding a sideways velocity $v_{\mathrm{nav},y}$ proportional to the distance to the path but limited to $|v_{\mathrm{nav},y}| \leq 0.2\,\mathrm{m\,s^{-1}}$. Before the actual passing task,

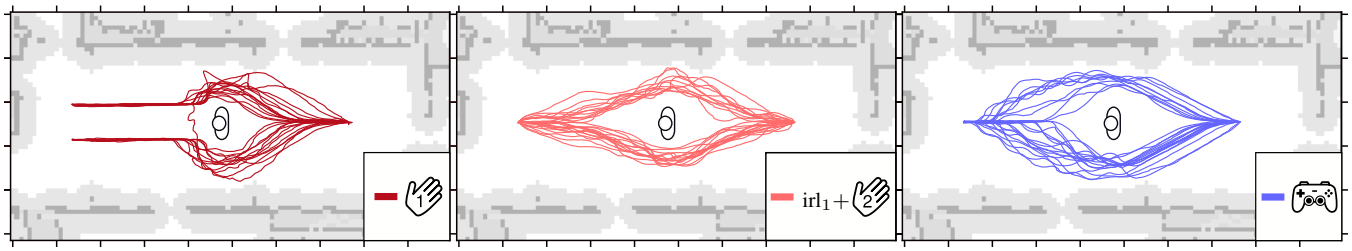

Fig. 3. Robot paths adapted via user feedback in the passing task. Left: initial paths, adapted through force feedback. Middle: paths from learned reward function after the first IRL run, adapted through force feedback. Right: initial paths, adapted through joystick feedback.

the participants performed two practice tasks to familiarize themselves with controlling the autonomously navigating robot. Firstly, the robot moved on a straight line from start to goal, and the participants freely deviated it from its path to test the behavior. Secondly, we arranged four traffic cones as obstacles and asked the participants to guide the navigating robot around them to ensure they were able to control the robot along their desired trajectory.

For both control modalities, the robot initially navigated autonomously from the start to the goal without considering the person in the hallway. In the force feedback set, the robot iteratively adapted its navigation behavior by optimizing the navigation reward function presented in Sec. III-C via IRL. We performed two force feedback passing cycles with two passing runs each. For the first two runs, the robot started with a small offset of $0.4\,\mathrm{m}$ to the left and right, respectively, to facilitate the interaction. The robot then performed one online IRL run to optimize the reward function parameters according to the corrected trajectories. We then performed another cycle with two passing runs, where the robot paths were generated from the learned model by sampling from the stochastic policy (Eq. 9). Again, the participants could correct the paths according to their preferences. We finally retrained the reward parameters with a second IRL run on the last two trajectories.

To find the reward parameters θ^*, we used the Adam [32] optimizer with an L2 regularization term, as suggested by Wulfmeier et al. [31]. We set the learning rate to 0.1 and ran the optimization until the log-likelihood of the demonstrations converged, which took under $2\,\mathrm{min}$ for 100 to 250 iterations on the experiment laptop. To speed up the training, we initialized the reward function with parameters we found during prior tests in simulation and refined the parameters with every IRL cycle.

In the joystick set, the participants performed two passing runs that serve as a reference for the true desired behavior without reachability constraints. We did not apply our learning method to the joystick trajectories because we assume that the unaltered trajectories are a better approximation of the true desired robot behavior than a learned model that may introduce a bias. Fig. 3 visualizes the joystick and force feedback trajectories from all test subjects.

Once the participants performed all tasks in the set, we asked them to rate the control method on a 5-point Likert scale questionnaire, summarized in Fig. 4. Both the force

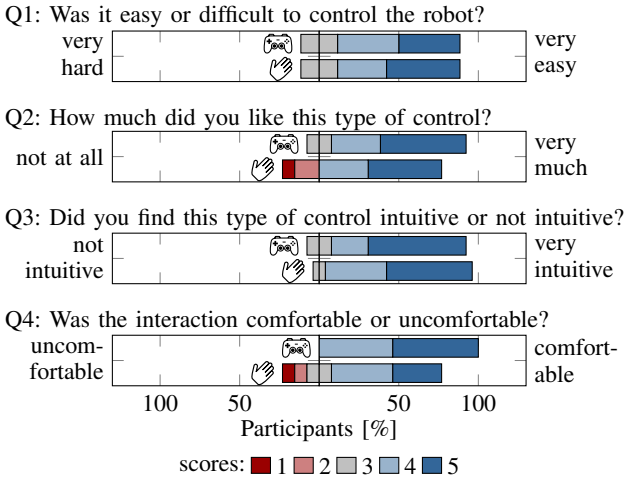

Fig. 4. Participant ratings after the passing task. Bars left of the center show negative, bars on the right show positive responses.

control and the joystick modality received very positive ratings for the ease of controlling the robot and the intuitiveness. The majority of the participants liked both the joystick and the force control and felt comfortable during the interaction. Three participants did not like the physical interaction (rating ≤ 2), and two felt uncomfortable (rating ≤ 2). In additional comments on the questionnaires, two participants stated that they did not like touching the robot and preferred to control it without physical contact. One participant wrote that she felt uneasy when the robot directly approached her, not knowing how it would react to her interaction. Fortunately, since the robot can learn from the interaction, it can correct such kind of behavior after only a few interactions. One participant stated that she found the joystick control more fun. Nonetheless, based on the mostly positive answers, we think that force control has great potential for interacting with a navigating robot. However, it is a very novel concept, and some people might need more time to get used to interacting with a robot in general, and especially via touch.

To validate the learned navigation behavior, we finally presented three different paths to the participants. The robot navigated past them without their interaction on

- a path generated from the leaned reward function by sampling from the stochastic policy (Eq. (9)),
- a replay of one of the trajectories the participant had commanded with the joystick,

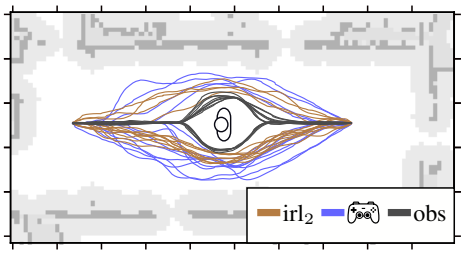

Fig. 5. Final trajectories presented to the test subjects: paths from the learned model after the second IRL run, joystick path replays, and obstacle avoidance paths.

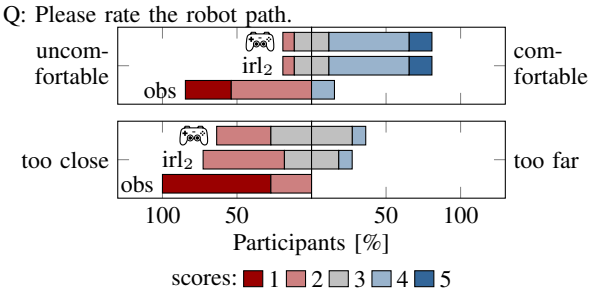

Fig. 6. Participant ratings of the final trajectories.

- and a path with just obstacle avoidance.

The resulting paths are presented in Fig. 5. We randomized the order of the three trajectories and did not tell the participants how the paths were generated. After the replay of the three paths, we asked the participants to rate the three paths in a third questionnaire. The questionnaire results are presented in Fig. 6. The majority of the participants rated the comfort of both the joystick path and the path from the learned model positively. In contrast, the obstacle avoidance path received very negative ratings. All participants rated the obstacle avoidance path as too close, and many criticized on the questionnaire that the robot indicated much too late that it would avoid them.

Interestingly, despite the positive comfort ratings, many participants rated the joystick path and the path generated from the learned reward function as (rather) too close. This effect is visible for both the joystick path and the path from the learned model, but more prominent in the latter. We were surprised by this finding, especially since the participants fully commanded the joystick trajectories themselves, and we always confirmed that the joystick passing runs reflected their preferences before saving them. One possible explanation is that participants might tolerate a smaller distance to the robot when they know that they can influence its behavior. Once they cannot interact, they prefer a larger distance.

Unfortunately, it is not possible to directly compare the learned rewards to the ground truth rewards because humans cannot directly quantify their reward preferences. Instead, we compare relevant path properties to confirm that the learned navigation behavior accurately follows the demonstrations. To this end, we compare the path lengths, the minimum distances to the participants, and the areas under the paths

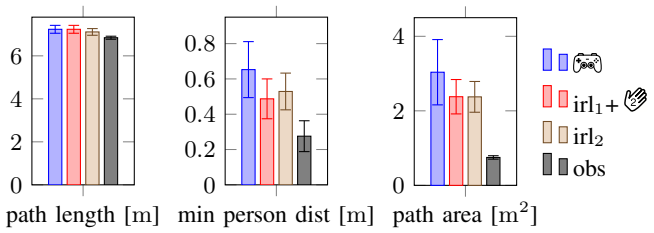

Fig. 7. Properties of the final trajectories: path length, minimum distance between the robot shell and the center of the person, and area under the path. The bar plots show the mean property value and the standard deviation for the joystick feedback paths, the force-adapted paths in the second passing cycle, the paths from the learned model, and with obstacle avoidance (obs).

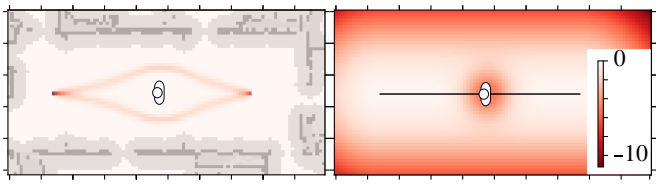

Fig. 8. Propagated policy from learned combined model (left) and state reward of the combined model with the nominal direct path (right).

from the start to the goal, i.e., the area between the nominal direct path and the commanded trajectory. The area under the path indicates how early the robot started the passing maneuver. Fig. 7 presents the path properties from the path replay experiment and the force feedback paths from the second passing cycle on which the reward functions were optimized. We can see that the force feedback paths from the final passing cycle exhibit very similar properties to the paths from the learned reward function. This indicates that the reward function can capture important path properties and that the robot can successfully learn to mimic the demonstrated behavior. We can further see that the force feedback paths and the paths from the learned model both pass closer to the participants than the joystick paths. One explanation is that the participants could not guide the robot further away from where they were standing. However, we do not think this is the reason here since we allowed the participants to step towards the robot if they wanted. From our observations, we instead think that the participants tolerated slightly unpleasant behavior before they intervened, even if they would have preferred the robot to behave differently. We do, however, acknowledge that the limited reach presents a drawback of our work. In future work, we plan to include other cues to address this limitation, such as a person's gaze or her/his own evasive maneuvers.

Finally, to visualize the navigation behavior learned during the passing experiments, we trained the parameters of our reward function with the last two force feedback trajectories from all participants. Fig. 8 shows the propagated policy and the learned state reward.

D. Generalization to New Environments

To verify that the learned reward function generalizes to new navigation situations, we employed it for two new

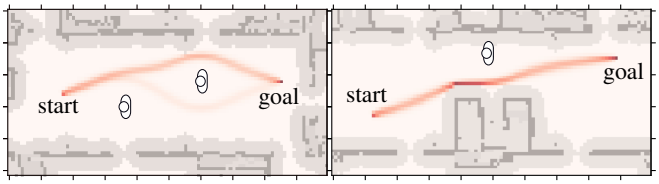

Fig. 9. Propagated policy from learned combined model, applied to two unseen navigation scenarios.

simulated scenes. In the first scene, the robot navigates in the same hallway, but two people are present. In the second, we tested the navigation in a different part of the hallway with one person standing next to a large obstacle. This time, the robot has to pass closer to the person because of the limited space. The results in Fig. 9 suggest that the learned navigation function can represent both new scenarios.

V. Conclusion

We introduced a novel approach for teaching robots social navigation behavior among humans via physical interaction and using inverse reinforcement learning. Through real-world experiments with human test subjects, we demonstrated that controlling the robot via pushing is a viable means of communicating navigation preferences and that the robot can learn to adjust its behavior accordingly. In future work, we plan to include additional cues like gaze or human path deviations to train our model for situations where the robot is not in reach of the people it interacts with. Furthermore, we want to investigate whether our current model is sufficient in new navigation contexts, or if multiple models should be learned for different scenarios. Finally, we plan to extend our model to handle dynamic situations with moving people.

References

[1] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Siméon, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, vol. 23, no. 5, 2007.

[2] R. Kirby, R. Simmons, and J. Forlizzi, "Companion: A constraint optimizing method for person-acceptable navigation," in *Int. Symp. on Robot and Human Interactive Communication (RO-MAN)*, 2009.

[3] D. V. Lu and W. D. Smart, "Towards more efficient navigation for robots and humans," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[4] Y. Kim and B. Mutlu, "How social distance shapes human-robot interaction," *Human-Computer Studies*, vol. 72, no. 12, 2014.

[5] L. Takayama and C. Pantofaru, "Influences on proxemic behaviors in human-robot interaction," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[6] J. Mumm and B. Mutlu, "Human-robot proxemics: Physical and psychological distancing in human-robot interaction," in *Int. Conf. on Human-Robot Interaction (HRI)*, 2011.

[7] J. T. Butler and A. Agah, "Psychological effects of behavior patterns of a mobile personal robot," *Autonomous Robots*, vol. 10, no. 2, 2001.

[8] D. Shi, E. Collins, A. Donate, X. Liu, B. Goldiez, and D. Dunlap, "Human-aware robot motion planning with velocity constraints," in *Int. Symp. on Collaborative Technologies and Systems (CTS)*, 2008.

[9] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annu. Rev. of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, 2020.

[10] E. Pacchierotti, H. I. Christensen, and P. Jensfelt, "Evaluation of passing distance for social robots," in *Int. Symp. on Robot and Human Interactive Communication (RO-MAN)*, 2006.

[11] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[12] M. Luber, L. Spinello, J. Silva, and K. O. Arras, "Socially-aware robot navigation: A learning approach," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[13] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation," in *Robotics: Science and Systems (RSS)*, 2012.

[14] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[15] M. Bennewitz, W. Burgard, and S. Thrun, "Adapting navigation strategies using motions patterns of people," in *Int. Conf. on Robotics and Automation (ICRA)*, 2003.

[16] K. Dautenhahn, S. Woods, C. Kaouri, M. L. Walters, K. L. Koay, and I. Werry, "What is a robot companion - friend, assistant or butler?" in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2005.

[17] C. Lichtenthäler, A. Peters, S. Griffiths, and A. Kirsch, "Social navigation – identifying robot navigation patterns in a path crossing scenario," in *Int. Conf. on Social Robotics (ICSR)*, 2013.

[18] B. Kim and J. Pineau, "Socially adaptive path planning in human environments using inverse reinforcement learning," *Social Robotics*, vol. 8, no. 1, 2016.

[19] M. Herman, T. Gindele, J. Wagner, F. Schmitt, C. Quignon, and W. Burgard, "Learning high-level navigation strategies via inverse reinforcement learning: A comparative analysis," in *Advances in Artificial Intelligence*, 2016, pp. 525–534.

[20] B. Akgun, M. Cakmak, J. W. Yoo, and A. L. Thomaz, "Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective," in *Int. Conf. on Human-Robot Interaction (HRI)*, 2012.

[21] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal, "Learning objective functions for manipulation," in *Int. Conf. on Robotics and Automation (ICRA)*, 2013.

[22] A. Bajcsy, D. P. Losey, M. K. O'Malley, and A. D. Dragan, "Learning robot objectives from physical human robot interaction," in *Conf. on Robot Learning (CoRL)*, 2017.

[23] A. Sabatini, V. Genovese, and E. Pacchierotti, "A mobility aid for the support to walking and object transportation of people with motor impairments," *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[24] M. Spenko, H. Yu, and S. Dubowsky, "Robotic personal aids for mobility and monitoring for the elderly," *Trans. on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 3, 2006.

[25] O. Khatib, "Mobile manipulation: The robotic assistant," *Robotics and Autonomous Systems*, vol. 26, no. 2-3, 1999.

[26] Y. Hirata, T. Takagi, K. Kosuge, H. Asama, H. Kaetsu, and K. Kawabata, "Map-based control of distributed robot helpers for transporting an object in cooperation with a human," in *Int. Conf. on Robotics and Automation (ICRA)*, 2001.

[27] M. Lawitzky, A. Mortl, and S. Hirche, "Load sharing in human-robot cooperative manipulation," in *Int. Symp. on Robot and Human Interactive Communication (RO-MAN)*, 2010.

[28] M. Lawitzky, J. R. Medina, D. Lee, and S. Hirche, "Feedback motion planning and learning from demonstration in physical robotic assistance: differences and synergies," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[29] M. Kollmitz, D. Büscher, T. Schubert, and W. Burgard, "Whole-body sensory concept for compliant mobile robots," in *Int. Conf. on Robotics and Automation (ICRA)*, 2018.

[30] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Conf. on Artificial Intelligence (AAAI)*, 2008.

[31] M. Wulfmeier, P. Ondruska, and I. Posner, "Maximum entropy deep inverse reinforcement learning," *arXiv preprint arXiv:1507.04888*, 2015.

[32] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. on Learning Representations (ICLR)*, 2015.