"Never in your way": Improving Human-Aware Robot Navigation with Target Prediction

Haritha Muralidharan
Electrical and Computer Engineering
Cornell University
Ithaca, USA
hm535@cornell.edu

Matthew Einhorn
Computer Science
Cornell University
Ithaca, USA
me263@cornell.edu

Yoyo Tsung-Yu Hou
Information Science
Cornell University
Ithaca, USA
th588@cornell.edu

***Abstract*—As robots become more commonplace at home and at workplace environments, it becomes necessary to guide their navigation behavior such that human expectations of socially acceptable distances and comforts are maintained. In the presence of humans, robots are expected to navigate around them in a manner that ensures that the humans are not required to change their course to avoid collision. This paper presents a multi-path trajectory prediction model based on the planning-based approach for social robot navigation proposed by Kollmitz et al. Our approach tries to identify the most likely final destination of the human an predicts his trajectory based on the destination. It then attempts to navigate around the human through dynamic social cost layers. The algorithm was tested out in simulation and through physical user testing.**

I. Introduction

Robots are increasingly incorporated in our working and living environments. The co-existence of robots and humans in the same space provides opportunities for better collaboration, but it also raises new problems. For example, movement coordination is a problem that has drawn much attention in the field of Human-Robot Interaction (HRI) [1]. While a robot moves to execute assigned tasks, it often comes across with other humans in the same environments. For the robot to function properly, it must know how to avoid collision by yielding the way to the human, detouring, or just waiting until the human has passed. Besides collision avoidance, the robot must also take humans' feelings into account. For example, turning sideways the last second before hitting an upcoming human may be a successful move to avoid collision, but it also causes threatening and unpleasant feelings, and makes the humans difficult to infer the robot's intention.

Many studies have investigated human-aware navigation of robots. Specifically, Kollmitz et al. have addressed human-aware navigation which accounts for the movement of human. In their model, they included a time-dependent cost map that represented the prediction of future human behavior. This model significantly improved the robot's performance by allowing the robots to react to human's future positions. However, their future prediction is based on the assumption that humans are always going in straight line, which in many cases is not optimal, especially in environments with obstacles.

We propose an improvement over the original algorithm with target prediction. The movement of people is not random, nor is it always straight. Instead, people generally moves toward certain targets in a space. By taking in the target information, we aim at improving the prediction of future human position, yielding a more accurate cost map, and thus improving the robot's navigation behavior.

II. Related Work

Human-robot interactions require robots to maintain a socially acceptable distance to ensure human comfort. Studies also show that humans prefer robots that behave predictably, by slowing down or stopping, when encountering a human, over behaviors where the robot makes large path deviations.

Early works in robot navigation in collaborative workspaces treated humans as static obstacles to be navigated around [2], but did not account for human motion over time. Other works accounted for the dynamic behavior of humans, but did not take their social preferences into account ([3], [4]).

Kuderer et al., proposed to mimic human behavior through observing and learning. While they were successful in producing a model that was able to acceptably avoid collision in crowded public spaces, their navigation model did not fit the expectations of a obedient or devoted robot in the home or at the workspace.

A layered dynamic social cost model was proposed by [1]. Their algorithm combines time dependent search-based path planning with dynamic social costs. The social costs are modeled as a Gaussian function based on the predicted trajectories of the human. In their paper, they established that the model was able to exhibit behaviors similar to human-human interactions, such as slowing and stopping upon collision with a fellow human.

In this paper we propose to extend the Kollmitz et al. algorithm through better path prediction. The original algorithm used a simple constant velocity prediction to estimate the future positions of the human. We augment the trajectory prediction through probabilistic reasoning, while keeping Kollmitz's original social cost calculation methods. Our goal is to identify if the robot is still able to perform well when the human's target is known with greater or less certainty.

III. Human Aware Navigation Framework

A. Target Prediction

In a collaborative space, humans often have specific targets that they move towards (e.g to the lamp, or to the workbench). It is reasonable to assume that the robot has knowledge of the locations of these static targets.

When a human is moving in the workspace, it is reasonable to expect that he is moving towards one of the known static targets.

In our model, the robot has knowledge of all possible final targets of the human, although it does not know *which* specific target the human is heading towards.

Given the locations of the targets and the current position of human, however, the robot is can predict the which target the human is most likely heading towards.

B. Probability Model

In this paper, we model the world to have three possible human targets - i.e. every time the human is moving, she is likely headed towards one of these three goals. The probability that the human is moving to Goal i at any given time is given by the following relationship:

$$P(H=i) = \frac{\Delta {d_i}^2}{\sum_{j=1}^{N} \Delta {d_j}^2}$$

where H represents the unknown target of the human, N represents the total number of goals, i represents the current goal being evaluated, and Δd_i represents the distance moved by the human *in the direction of goal i* since the last iteration. If the human moves away from a goal i, then

$$P(H=i) = 0.0$$

In summary, the probability that the human's target is i is the squared speed at which he moves towards i. Unlike the Kollmitz paper, which only predicts one path per human in the workspace, this algorithm will generate multiple paths per human, with each path indicating a trajectory to a known target. Each path is weighted by the probability of H being that particular goal.

C. Social Cost Model

Each layer in the social cost is calculated as in Kollmitz et al.'s algorithm. In each layer, the cost is based on a Gaussian distribution with amplitude A, and different standard deviations σ_x and σ_y along a person's front and side. The Gaussian distribution is displaced from the person's center by Δx and Δy, humans are more sensitive in the areas directly in front of them. A non-traversable radius of r_0 determines the physical space occupied by the human, and indicates the "forbidden" areas for robot navigation.

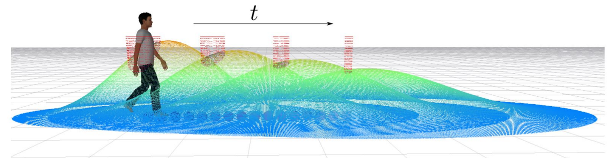

Fig. 1. Social cost for a single path decaying linearly over time.

D. Dynamic Cost Maps

Kollmitz chose to keep a single amplitude that decayed linearly with time as shown in Figure 1. The costs decay with time because the human's trajectory becomes more unpredictable over time.

Along with the linear decay, we weigh the paths by the path probability as described in Section II-B. This is achieved by simply scaling the amplitude of the Gaussian function (A) by the probability of that path. This is visualized in the cost map diagrams in Figure 2.

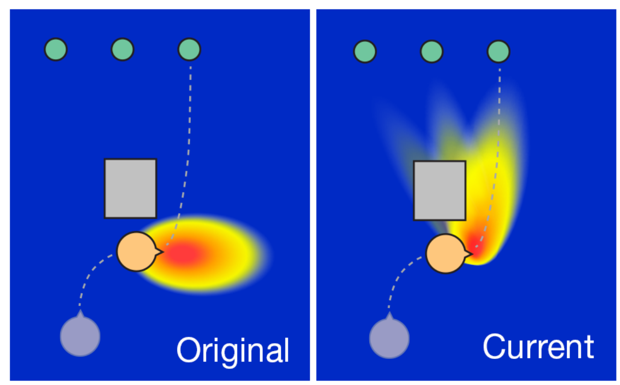

Fig. 2. Left: cost map using single path prediction, as proposed by Kollmitz, and Right: cost map with multiple paths, each weighted by the probability that its goal is the human's target.

E. Planner and Optimization

The dynamic cost maps are fed to a global planner which identifies a sequence of sub-goals for the robot to traverse in order to achieve its final goal. The planning is carried out in a time-dependent manner in order to ensure that the sub-goals are reachable within the time interval Δt given the robot's current pose, velocity, and its kinematic and dynamic constraints.

The sub-goals are fed to a local planner which optimizes the robot's path taken to reach the sub-goal via an A* search. The local planner also takes into consideration the static cost maps from obstacles in the workspace.

A timeout is also implemented as a safety net, in case the robot was unable to find a path within a reasonable amount of time. Both the global and the local planners are implemented using the ROS Navigation package.

IV. Evaluation Method

For testing our algorithm, we ran a lab user study with real human participants and a physical robot to evaluation the performance of our new algorithm.

A. Participants

Eight participants (6 males, 2 females) volunteered to participate in this study without compensation in a university in upstate New York. Their ages ranges from 25 to 34. All the participants have some knowledge about robotics.

B. Testing Environment

1) Simulation: The algorithms were first tested in the Gazebo simulation environment using relevant ROS packages. ROS Visualization (RViz), a 3D visualizer for displaying sensor data and state information was used to set the target location for the robot, for both the Gazebo simulation, and the physical testing.

2) Robot: The experimental validation of the proposed algorithm was done using a Clearpath Jackal. The computation was done locally on a laptop computer, and the final velocity commands were sent to the Jackal over ROS topics (Figure 3).

Fig. 3. Clearpath Jackal used in experimental setup.

3) Testing Environment: Accurate human localization was performed through a Vicon camera system. The space to conduct the experiments was limited, due to the requirements of the Vicon system, and therefore the physical space was smaller than that in the Gazebo simulation.

In Figure 5, the orange cones represent the locations through which neither the humans, nor the robots can traverse. The red cups in the distance represent the targets towards which the humans will navigate. The Vicon localization system is mounted to the ceiling.

C. Design

This was a within-subject, 2-condition (our new algorithm v.s. Kollmitz et al.'s original algorithm) experiment. The order of the condition is randomized for each participants. After

Fig. 4. Testing environment used for user testing

each condition, participants were asked to answer an online questionnaire to report the perceived intelligence, naturalness, responsiveness, interference, and number of collision (defined as ”when you feel like the robot gets in your way and severely interferes your movement”) with the robot.

D. Procedure

Participants were told the whole procedure first, and then they did two conditions consecutively (the order of the conditions was randomized between participants). For each condition, there were three trials. Participants always started at the starting point (see the blue cross in Figure 5), while the robot started at the left, middle, and the right (see the red circles in Figure 5).

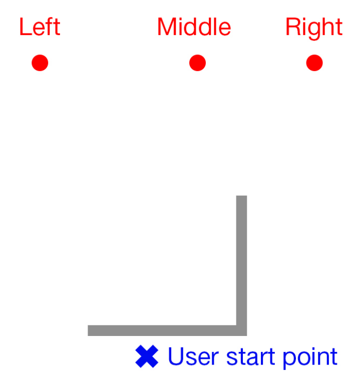

Fig. 5. Testing environment configuration. The three red circles are the goals that the human may head towards. The gray L shape is an obstacle that the human or robot may not cross. The blue cross is the location from which the human starts. The robot starts from one of the goals and heads towards the user's starting point.

The participants were told to walk ”through the path as you naturally would” to navigate to one of the goals, without crossing the obstacle, the gray wall in Fig. 5, which is represented by cones in our testing environment (Fig. 4). At the same time, we instructed the robot to navigate toward the user's starting point. By doing this, we make sure that the robot and the participant will encounter midway, and therefore we can test the behavior of the robot. The participant navigated from the start point to the left point in the first trial, returned to the start point, did the second trial which is navigation toward the middle point, and repeated again to the right point to conclude three trials of a condition.

After three trials, the participants did a brief survey on Qualtrics asking about the dependent variables, and then repeat the whole process under the second condition. The participants were then interviewed briefly and then debriefed about this study.

V. Findings

We used paired t-tests to analyze our data. Although the subjective ratings in our studies used a Likert scale, which is theoretically ordinal data, it is accepted that if the descriptions of the Likert scales were standard, the data can be analyzed as if they are interval data.

Our result showed that there were no significant difference between these two condition, on either of the 5 dependent variables.

A. Perceived Intelligence

The perceived intelligence did not differ significantly (Figure 6) between the original algorithm (M=3.5, SD=1.20) and our new algorithm (M=3.13, SD=0.83), t(7) = 0.70, p = .50.

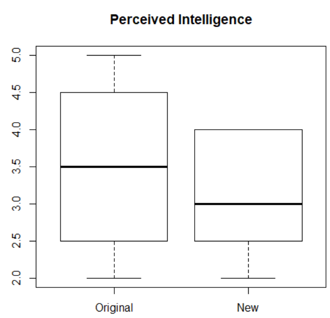

Fig. 6. Perceived intelligence did not differ significantly between conditions.

B. Perceived Naturalness

The robot was perceived as more natural with our new algorithm (M=3.13 SD=0.64) compared to the original algorithm (M=3.0, SD=0.95), but the difference did not reach a significant level, t(7) = -0.31, p = .76 (Figure 7).

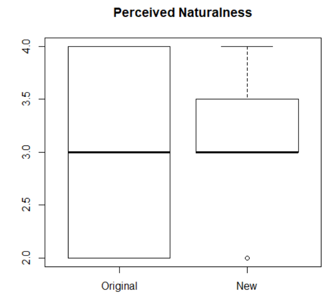

Fig. 7. The perceived naturalness is higher with our proposed new algorithm, but the difference was not significant.

C. Perceived Responsiveness

The perceived responsiveness did not differ significantly (Fig 8) between the original algorithm (M=3.25, SD=1.39) and our new algorithm (M=3.25, SD=0.71), t(7) = 0, p = 1.0 (Figure 8).

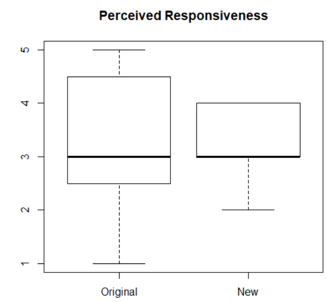

Fig. 8. Perceived responsiveness did not differ significantly between conditions.

D. Perceived Interference

The perceived interference is lower with our new algorithm (M=2.62, SD=0.92) compared to the original algorithm (M=2.75, SD=1.04), but they did not differ significantly, t(7) = 0.26, p = .80 (Figure 9).

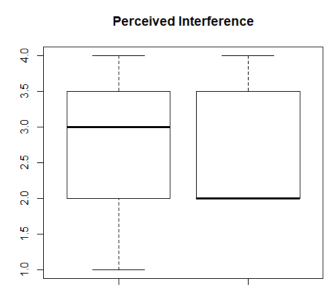

Fig. 9. The new algorithm resulted in lower interference, but the difference was not significant.

E. Number of collision

The number of collisions did not differ significantly (Fig 10) between the original algorithm (M=1.0, SD=0.76) and our new algorithm (M=1.25, SD=0.71), t(7) = -0.68, p = .52 (Figure 10).

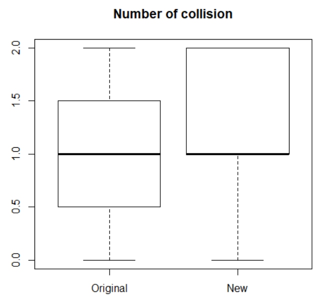

Fig. 10. No significant difference was found in the number of collision. However, the new algorithm did result in a higher mean.

VI. Discussion

Overall, our user testing showed that despite some promising trend in higher perceived naturalness and less interference, there was no significant difference between our proposed new algorithm and the original algorithm by Kollmitz et al.. We believe that there are several reasons that might explain this result.

A. Testing Environment

The lack of space in the testing environment proved to be detrimental in evaluating the difference between the two algorithms. Through the RViz tool, we observed that there was high qualitative overlap between the paths identified by either method. The ”corridor” through which the robot and the human had to navigate was often too narrow for observable changes between the single and multi-path prediction approaches. Also, we might be able to observe more difference in a more complex environment where users make turns frequently, because the new algorithm considers the goals while the original one often makes incorrect predictions around turns when the humans local path diverges from their global path.

We also observed that the execution of the path prediction algorithms (both the original, and the proposed) was very slow, requiring the users to walk very slowly to provide sufficient time for robot to execute its planning program. But even at slow speeds, the algorithm was often unable to find good paths and was very sensitive to the human's path and obstacles. This could be attributed to the algorithm not being optimized for real-time performance. For proper validation, the algorithms must parallelized for efficiency, such that the robots can avoid the humans even at regular speeds. Better planning algorithms should also be investigated.

B. Parameter Tuning

The parameters (such as the maximum amplitude of the Gaussian, and the number of predictions at each time-step) used for simulation and physical testing were mostly the default parameters recommended by Kollmitz et al. with some tuning. For the multi-path prediction to work well, it might need further tuning for the algorithm and the environment.

C. Human Path Prediction

We used the exponentailly weighted human velocity to estimate a direct path to each goal. For better performance, we can use a $A*$ planner to plan a path to each goal from the current human's position, which would better predict the human's path and potentially improve the robots ability to collaberativly navigate. However, estimating a plan for the human's path may only work for very simple environments or with way points and the forward time cost prediction would need to be tuned to decay appropriately, otherwise any changes in the human's velocity would continously change the prediction negating any benefit. It is also possible that estimating the human's path beyond a few frames is too noisy to be useful and thereofre $A*$ would not provide any benefit.

Despite the insignificant result in our user studies, we believe that in a larger space or with more precise tracking and planning, goal-directing prediction will improve the navigation of robots in a collaborative space. The movement of human in a space is not random, and usually has a pattern that is similar to eye movement: long fixation on certain hot spots with short and quick movements between these fixations. Current technology, such as machine learning, potentially makes identifying these hot spots easy. By utilizing this information, a robot may be able to better predict the future behavior of human, and therefore improves its navigational behavior.

VII. Conclusion

In this study, we proposed an algorithm to improve a robot's social navigation by incorporating information of human's navigational goals. Our user study result was not significant probably due to the smaller testing space. For future work, we suggest testing in a larger collaboration space, use better robot navigation algorithms and investigate more sophisticated human path prediction and parameter tuning.

References

[1] M. Kollmitz, K. Hsiao, J. Gaa, and W. Burgard, “Time dependent planning on a layered social cost map for human-aware robot navigation,” in *2015 European Conference on Mobile Robots (ECMR)*, Sept 2015, pp. 1–6.

[2] R. Kirby, R. Simmons, and J. Forlizzi, “Companion: A constraint-optimizing method for person-acceptable navigation,” in *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*, Sept 2009, pp. 607–612.

[3] M. Gller, F. Steinhardt, T. Kerscher, J. M. Zllner, and R. Dillmann, “Proactive avoidance of moving obstacles for a service robot utilizing a behavior-based control,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2010, pp. 5984–5989.

[4] M. Bennewitz, W. Burgard, and S. Thrun, “Adapting navigation strategies using motions patterns of people,” in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 2, Sept 2003, pp. 2000–2005 vol.2.

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