Safe and Efficient Autonomous Navigation in the Presence of Humans at Control Level

Klaus Buchegger[1], George Todoran[1], and Markus Bader[1]

Vienna University of Technology, Karlsplatz 13, Vienna 1040, Austria

Abstract. In order to enable mobile robots to navigate autonomously in an environment shared with humans, special considerations are necessary to ensure both safe and efficient navigation. This work presents a predictive, human-aware motion controller, based on the Robot Operating System (ROS), which optimizes the vehicle trajectory at the control level with a high update rate. Predicting future positions of persons allows the system to optimize a trajectory around those predictions, yielding a sequence of motor controls for a smooth executed motion. The improvements were statistically evaluated using simulation runs in terms of travel duration, path length, and minimum distance to persons along the path. This way, we are able to show that our new motion controller performs significantly better in the presence of humans than a controller without human-awareness.

Keywords: autonomous navigation, human-aware, local planner, human space, robotics, control level

1 Introduction

In human-aware robot navigation, a driver-less vehicle has to navigate through an environment with humans. In order to establish a system which is safe and efficient, human motions must be predicted and integrated into the navigation system. Such an integration can be done on a discreet planning level or on the control level. The difference in the resulting trajectory can be seen in in Fig. 1.

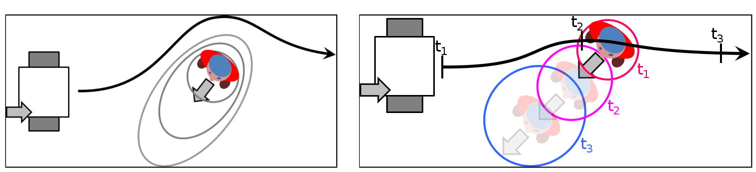

Fig. 1. Left, a classical planning approach on the discreet path planning level using a single cost-map to circumnavigate humans. On the right, our new approach using a time-dependent cost function to optimize the vehicles trajectory on the control level, resulting into a shorter motion.

On the planning level, a single cost-map can be used to define a safety region around a detected person's current position. The robot then plans a trajectory around this stationary safety region to avoid a collision. Depending on the movement of the person, either away from or closer to the planned trajectory, this can result in trajectories with large, unnecessary detours, or even cause a crash. By reasoning about future positions of a person, the robot can plan efficient trajectories that possibly pass through areas which were initially occupied by a person, but will be free once the robot reaches those areas.

The scientific contribution of this work is to present a novel navigational planner, that predicts human movements and avoids collisions on the control level. The main advantage of performing these actions on the control level is the high update rate (10-50 Hz). However, classical control level strategies (e.g. DWA [2]) are only capable of handling simple tasks, like immediate collision avoidance. Our novel approach is, nevertheless, able to plan trajectories for complex, dynamic situations, similar to those handled by a higher level approach proposed by Kollmitz et al. [5]. Since predictions of human walking paths cannot be perfect, the planner has to be able to react to prediction errors. Due to the high update rate of the control level, the robot can react faster to imperfect predictions by replanning the trajectory 10-50 times per second. Additionally, our approach does not require large, multi-layered cost-maps to keep track of a person's future positions, as the positions and according safety regions are directly computed for every evaluated time step.

In this work, we present the results of our human-aware robot navigation approach in multiple showcases. Section 3 provides an overview of our approach, and in Section 4 we discuss in detail how our robot handled different scenarios, compared to a non-human-aware implementation. Our new approach was implemented by enhancing the MPN framework developed by Todoran et al. [8] for the Robot Operating System (ROS)[1]. This implementation was tested in a simulation environment, using Gazebo[2], and statistically compared to the previous state of the framework, which treated every person as a static obstacle. Finally, in Section 5 we conclude our work by summarizing the results and highlighting the advantages of our approach compared to others.

2 Related Work

Recent publications show progress in developing human-aware mobile robots. Kollmitz et al. [5] proposed a navigational planner which uses a social cost field around a detected person and the prediction. The approach uses at least two layers for planning. A discreet planning layer, which implements the new time-dependent person costs, and the control layer, which computes the final motion commands. The integration of time-dependent person cost maps allows an A* algorithm to determine the best path around moving persons. These time-dependent cost maps are computationally complex to update and to evaluate,

[1] ROS: http://www.ros.org/
[2] Gazebo: http://gazebosim.org/

therefore the algorithm is only able to run at a update rate of 2 Hz. Kollmitz et al. [5] stated that the integration of such cost-maps for the control layer would make sense but it is, due to the computational complexity, not feasible. However, we show with our proposed work, by using cost functions and not discreet cost-maps, human prediction can be integrated on the control layer at update rates of up to 50 Hz. In [6], Kostavelis et al. proposed a navigational planner, that predicts human movement towards certain points of interest in the environment using a D* algorithm. Along the predicted path, costs are assigned in a single time-independent cost-map layer. In their work, they are mostly concerned about collision avoidance. Efficient robot paths are of minor importance, as their model is only predicting locations where the person could be at any time, and not whether the person actually leaves a location and therefore frees space for the robot to pass through. Chen et al. [1], recently presented a human-aware planner, based on deep reinforcement learning. In contrast to their approach, we formulated an explicit, transparent movement prediction model with parameters adjustable at run-time and without going through a learning phase again. Other publications not only focus on the avoidance of collisions with humans, but also on replicating human-like behavior, in a way that persons will not be irritated by the robot's presence. Moussaid et al. [7], for example, discovered, that depending on the culture, people have a generally preferred side on which they pass other persons. By knowing such preferences, the robot could be designed to behave as people will expect.

3 Human-Aware Approach

Fig. 1 (left) shows, that the typical navigational approach plans a path around a person by estimating the possible positions of that person in an ellipse, expanded in the person's moving direction. By actually predicting the person's movement for future timestamps (t_1-t_3), the robot can more accurately estimate the position of the person, and thus choose a more efficient trajectory, as shown in Fig. 1 (right).

Our approach consists of three core elements, which enable the robot to plan trajectories that account for human movements. We implemented all three steps at the control level of our planner, allowing for fast update rates of 10-50 Hz. At first, the movements of detected persons were predicted, by assuming people in general choose similar paths. Therefore, our prediction follows paths many people have been observed walking along. With this prediction, the robot can estimate the future position of a person for any point in time. In the second step, possible trajectories towards a predefined goal were checked for validity. For every point in time, the future position of the robot had to satisfy a required safety distance to the future positions of the detected persons. And finally, the trajectories were evaluated with the cost function, which contains a social cost field around persons, favoring trajectories that passed on a person's left side.

For the prediction, we recorded the paths of persons walking through the environment in a map. For every cell of the map we counted how many people walked through this cell. With these recordings we could predict a person to walk towards areas which have seen a high occupancy in the past, using an explicit model. For every pose of a person, an area in front of that person is evaluated to determine a predicted change in orientation. An attracting potential is assigned to each cell in that area, scaled by the occupancy count, such that the combined potentials represent the most likely movement direction. The person is then predicted to turn towards that movement direction within one second, while walking with assumed constant velocity.

The safety distance was formulated as an inequality constraint in the framework, such that the Euclidean distance to the closest person had to be at least 0.7 m (with an allowed error margin of 0.1 m), for every point along the trajectory. The distance of 0.7 m was chosen so the robot avoids entering the intimate space, defined as a 0.45 m wide circle around a person [4]. We considered forcing the robot to pass outside of the personal space (0.45 - 1.2 m), but then the robot was not able to find a trajectory in some scenarios (e.g. narrow hallway, with a person approaching the robot) even though the person could have been passed at a comfortable distance. Kollmitz et al. [5] similarly designed their planner to accept trajectories passing through a person's personal space.

For the preference to pass on a person's left side, the core cost function was modified. The cost of the existing framework is based on a velocity map, created using the fast marching method [3]. The velocity map represents how fast the goal can be reached from any position and following the slope results in the shortest path, avoiding obstacles but ignoring the robot's dynamics. For every trajectory, accounting for the robot's dynamics, the end point is optimized to be as far along the slope as possible, resulting in the lowest cost for the remaining path to travel. To favor trajectories that pass on a person's left side, we increased the cost in an elliptical area around the person, slightly offset to the person's right, similar to the social cost-field of Kollmitz et al. [5]. While this increased cost is only assigned to the person's current position, it was sufficient to shift the slope in the velocity map to the person's left side and therefore the planned trajectories also end on a point to the person's left side.

Besides the recorded person positions, our approach is based on an explicitly formulated model. Therefore, an advantage over designs based on neural networks is that all parameters can be adjusted quickly, either at run-time or at compile-time, without the need of a new learning phase.

4 Experimental Setup and Results

In our experiments, we simulated a Pioneer P3-DX robot and a walking person in different scenarios. For each scenario, we performed several runs with various walking speeds for the person, in order to observe the robot's behavior for consistency. The runs were done with the MPN framework as proposed by Todoran et al. [8], and with our improved version of the framework. We compared the performances in terms of travel duration, path length and the minimum distance to the person along the path. Additionally, we recorded velocity data of the robot to examine how *smooth* the robot moved - sudden stops or jerks might startle surrounding persons and reduce the predictability of the robot's motion.

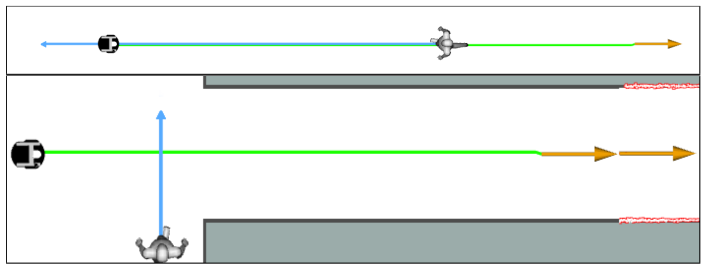

Fig. 2. Simulated test scenarios. In the approaching scenario (top) the robot and the person move towards each other, trying to pass. In the crossing scenario (bottom) the robot tries to enter a hallway while the person passes between the robot and the hallway entrance.

The scenarios are shown in Fig. 2. In the first scenario (top), the robot and the person started facing each other, with the goals each behind the other. The person was simulated to walk in a straight line from right to left, while the robot was instructed to find a path from left to right. For the second scenario (bottom), the person was simulated to walk from bottom to top, crossing the robot's path vertically, and the robot had to enter a hallway on the right. The robot's maximum movement speed was limited to 0.4 m/s and the simulated person walked with speeds ranging from 0.3 to 1.5 m/s. We recorded travel duration, path length and minimum distance to the person, from when the robot started moving, up until it passed the person's initial position.

For the first scenario, Fig. 3 shows, that both the old MPN framework (top) and our improved version (bottom) found paths that deviated to the side in order to let the person pass unhindered. However, without our social cost function, favoring passing a person on his/her left side, the robot randomly chose the passing side. With the improvement, the robot always took similar turns to the right. Fig. 4 shows a summary of the measurements of 30 simulation runs and one example velocity profile. While the path length was similar for both implementations, for the travel duration and the minimum distance to the person, our improved version performed significantly better. With our human-aware approach, the robot always kept the required minimum distance of 0.7 m with an allowed error margin of 0.1 m, and passed on the socially preferred side. We noticed that increased walking speed of the person resulted in longer travel durations and slightly shorter safety distance, as the robot had to perform a steeper evasive maneuver. The non-human-aware version of the framework couldn't satisfy the specified minimum distance to obstacles, and therefore the robot had to stop and wait for the person to pass in most of the runs. In the exemplary velocity profile, shown in Fig. 4, the braking can be seen after twelve seconds. The

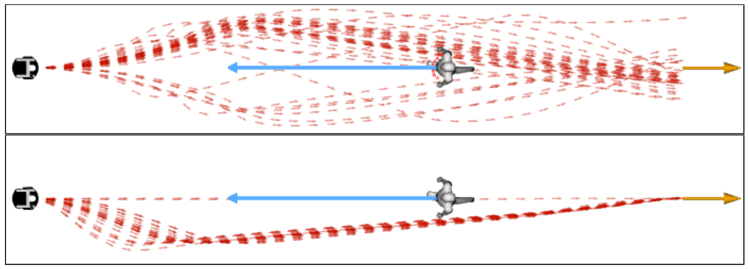

Fig. 3. Resulting paths in the first scenario where the robot and the person approached each other. The top image shows the non-human-aware approach, the bottom image shows our improved version. The robot is represented by the black model on the left, with its goal, the orange arrow on the right. The trajectories are shown by sequences of small red arrows. The person is represented by the gray figure in the center, and the predicted path by the blue arrow pointing leftward.

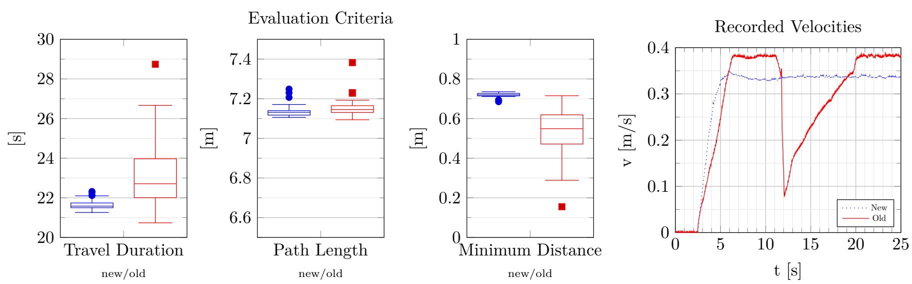

Fig. 4. Comparison of measurements of travel duration, path length, minimum distance and velocity profile of the old implementation to our new implementation in the approaching scenario.

plot for our improved version show, on the other hand, that no sudden stops or jerks were necessary, as the robot smoothly accelerated and passed the person.

In the crossing scenario, the person started in front and to the right of the robot, and the robot tried to drive through a hallway entrance. The resulting paths for both approaches can be seen in Fig. 5.

In the old version, the robot tried to pass in front of the person by turning left, while the new approach always selected a path that passed behind the person. The spirals in the old approach resulted from a fast walking speed. As the robot tried to pass in front of the person it had to detour further and further, until it reached the point where turning around and passing behind the person was necessary to enter the hallway. For slower walking speeds, the robot started turning left, but then the person moved too close, so the robot had to stop until the person passed. A summary of the recorded measurements is presented in Fig. 6. Again, the path length were similar, but our new approach performed better in terms of travel duration and distance to the person. In the velocity plot

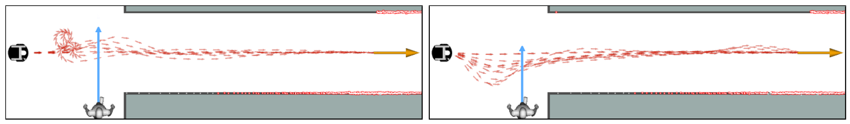

Fig. 5. Resulting paths in the second scenario where the person crossed in front of the robot. The left image shows the non-human-aware approach, the right image shows our improved version. The robot is represented by the black model on the left with its goal, the orange arrow on the right. The trajectories are shown by sequences of small red arrows. The person is represented by the gray figure on the bottom, and the predicted path by the blue arrow pointing upward.

for the old version it can be seen, that after six seconds the robot violated the safety distance, and therefore had to stop. For our new approach the velocity plot shows that the robot accelerated slower, allowing the robot to smoothly keep the safety distance while passing behind the person.

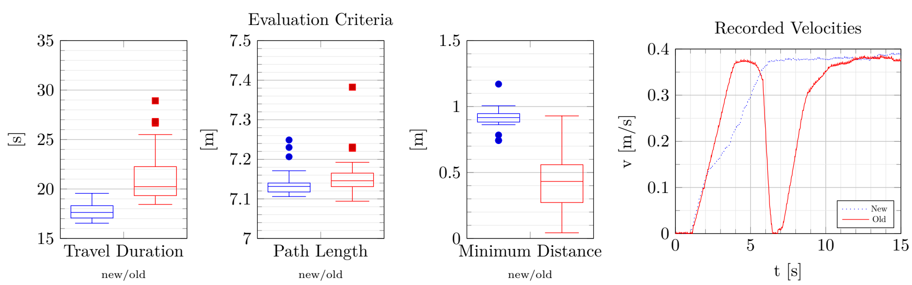

Fig. 6. Comparison of measurements of travel duration, path length, minimum distance and velocity profile of the old implementation to our new implementation in the crossing scenario.

In this scenario we noticed that for our new version, the travel duration was not connected to the path length. The optimizer found two different solutions for an efficient trajectory towards the goal. Either the trajectory detoured to the right, to actively evade the person, or a straight line with limited acceleration was chosen. This trade-off between speed and path length resulted in very similar travel durations, and therefore similar costs.

5 Conclusion

In this work, we demonstrated the benefits of making a navigational planner human-aware. By accounting for a person's movements and predicting future positions, our planner is able to find efficient trajectories that safely avoid collisions on the control level. In addition to collision avoidance with humans, our approach also allows the robot to consistently pass approaching persons on a preferred side, which makes it more predictable for humans. Our enhancements of the MPN framework brought significant improvements for the tested scenarios, as they prevented the need for sudden stops. Unlike the approach of Kostavelis et al. [6], our planner predicted the persons to move away from their initial position, and therefore allowed the robot to plan paths through initially occupied areas. The results were similar to those of Kollmitz et al. [5], who proposed a higher level planner, with an update frequency of 2 Hz. By integrating the human-awareness into the control level of the motion planner, we are able to achieve much higher update frequencies of 10-50 Hz, allowing fast reactions to unforeseen situations. For future work, the approach still has to be tested outside of the simulation, in a real world environment. In addition, as our explicit prediction model is evaluated for every future position of the robot, human-robot interactions could be implemented as well. For example, the person could then be predicted to slightly turn away from the robot's trajectory.

Acknowledgement

The research leading to these results has received funding from the Austrian Research Promotion Agency (FFG) according to grant agreement No. 854865 (TransportBuddy) and No. 855409 (AutonomousFleet)

References

1. Chen, Y. F., Everett, M., Liu, M., How, J. P.: Socially aware motion planning with deep reinforcement learning. CoRR, abs/1703.08862, 2017.
2. Fox, D., Burgard, W., Thrun, S.: The dynamic window approach to collision avoidance. In: IEEE Robotics Automation Magazine, 4(1):23–33, Mar 1997.
3. Gómez, J. V., Álvarez, D., Garrido, S., Moreno, L.: Fast methods for eikonal equations: an experimental survey. CoRR, abs/1506.03771, 2015.
4. Hall, E.: The Hidden Dimension: Man's Use of Space in Public and Private. Doubleday anchor books. Bodley Head, 1969.
5. Kollmitz, M., Hsiao, K., Gaa, J., Burgard, W.: Time dependent planning on a layered social cost map for human-aware robot navigation. In: Proc. of the IEEE European Conference on Mobile Robots (ECMR), Lincoln, UK, 2015.
6. Kostavelis, I., Kargakos, A., Giakoumis, D., Tzovaras, D.: Robot's workspace enhancement with dynamic human presence for socially-aware navigation. In: Computer Vision Systems, pages 279–288, Cham, 2017.
7. Moussaïd, M., Helbing, D., Garnier, S., Johansson, A., Combe, M., Theraulaz, G.: Experimental study of the behavioural mechanisms underlying self-organization in human crowds. In: Proceedings of the Royal Society of London B: Biological Sciences, 276(1668):2755–2762, 2009.
8. Todoran, G., Bader, M.: Expressive navigation and local path-planning of independent steering autonomous systems. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4742–4749, Oct 2016.