2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Socially Aware Local Planning: a Dynamic Window-based Approach

Luca Mozzarelli[1], Matteo Corno[1] and Sergio Matteo Savaresi[1]

***Abstract*— As the interest in mobile robots navigation on public sidewalks increases, so does the attention given to local planners capable of navigating around humans in a socially acceptable way. This paper presents a socially aware version of the Dynamic Window Approach planner. The DWA is augmented with an additional cost function, which predicts pedestrian trajectories using the Social Forces Model. Scoring of the robot control action is achieved by weighting the disturbance the robot causes to pedestrians. The approach is validated in a simulation environment with realistic pedestrian motions, showing superior performance with respect to the original DWA as well as to a distance-based scoring method.**

I. INTRODUCTION

Interest in mobile robots traveling on sidewalks has been on the rise for the last few years, with the major use case being autonomous urban deliveries. In this kind of application, robots coexist with pedestrians, making human-robot interactions a common occurrence. It's thus essential for navigation algorithms to be capable of gracefully avoiding collisions with humans. As testified by recent surveys [1], the task is less trivial than it might seem : while overcautious behaviors can avoid collisions, more often than not they cause unnecessary stopping. Furthermore, for the robots to be socially accepted, the chosen avoidance maneuvers must feel natural and predictable, so that passers-by do not feel scared. Such challenges are caused by the peculiarities of pedestrian motion. Humans fall in the category of dynamic obstacles, just like cars and bicycles, but their response to the behavior of other agents in the scene is much stronger. For generic dynamic obstacles, for instance, kinematic models can produce trajectory predictions sufficiently accurate, at least for obstacle avoidance purposes. This kind of prediction, though, does not capture social interactions. Indeed, cooperation is a prominent feature of human navigation in populated environments. Consider, as an example, a human facing other pedestrians in a narrow passage. While a kinematic model would predict a collision independently of the action taken, the human expects its opponents, even if they are strangers, to leave them some space to pass. In some sense, the responsibility for the avoidance maneuvers is shared between the two engaged parties. It does not come as a surprise, then, that an ever-increasing number of works are focusing on local planners that specifically handle interactions with humans, so-called *Socially Aware Local Planners*.

This paper, fostering these efforts, presents a Dynamic Window Approach based socially aware local planner. Differently from most of the recent literature, though, our approach does not constrain the robot to imitate human motion. Rather, it makes the robot aware of human motion patterns, while maintaining an optimal selection process of the control actions. With this goal, we modify the Dynamic Window Approach [2] to include pedestrian trajectories predictions by employing the Social Forces Model [3]. The predictions are tightly integrated with a novel Social Cost Function, that assesses the disturbance caused to pedestrians by the robot. Validation is carried out in a simulated environment with custom pedestrian simulator.

The remainder of this paper is structured as follows. Section II presents a review of related works. The proposed approach is explained in Section III, while the simulator setup and the simulated validation results are presented in Section IV and in Section V, respectively.

II. LITERATURE REVIEW

While local planning has been the center of numerous research efforts since the early days of robotics, planners that explicitly handle humans are a relatively recent trend. This literature review focuses on this type of local planners, of which two major categories can be identified: learning-based and model-based methods. In the former group, [4] proposes a planning architecture composed of two Auto-Regressive Gaussian Processes. The first one is employed to predict the trajectories of nearby pedestrians given position measurements, while the second performs the actual planning by estimating the robot trajectory given the human trajectories as inputs. The prediction model was trained on a very simple potential field-based simulator, while the planner on data collected while a human was manually driving a simulated robot. [5] presents a detailed analysis of the "freezing robot problem" described in the introduction, highlighting how the unfreezing is only possible if the robot is capable of cooperating with humans. The authors then proceed to develop a planner based on Gaussian Processes. Differently from [4], prediction and planning are integrated into a single model: the robot trajectory is computed jointly with the trajectories of the humans in the scene, given their measured positions and the past robot trajectory. Note that the robot is considered exactly as a person in the proposed model, with the goal of obtaining human-like behaviors. A similar goal is pursued by [6], which applies Inverse Reinforcement Learning and trains the planner with data recorded in a crowd simulator. Finally, [7] reduces the problem to making the robot respect simple social rules, like passing on the right and overtaking on the left. The social rules are encoded in the reward function of a deep reinforcement learning algorithm, which

[1] The authors are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milano, Italy. Email: {luca.mozzarelli, matteo.corno, sergio.savaresi}@polimi.it

is trained for one-versus-one and multi-agent environments. While data-driven methods do seem to successfully learn to navigate human environments, their major drawback is the amount of time and resources needed for data collection and training. Indeed all of the cited works train their systems on simulators based on different kinds of models, rendering the development of pedestrian models necessary anyways. The only exceptions are [5], which however limits the study to a small indoor area that can be instrumented for data acquisition, and [7] which, requires the manual definition of the reward function for each social rule to be handled.

Among model-based approaches, two major classes can be identified: a potential fields class, and a social forces one. The former employs potential fields [8] to encode information about both human dynamicity and their proxemics [9], *i.e.* the areas around a person which, if occupied, can influence its behavior. The latter, to which this work belongs, makes use of the Social Forces Model (SFM) [3] which is one of the most popular approaches to model and predict human motion. The SFM models each agent in the environment as a point mass subject to an attractive force (towards a goal location) and repulsive forces from other agents and obstacles. More details will be discussed in Section III-B. Within the potential fields class, [10] proposes an approach in which the potential field is modeled to have high peaks in the direction of motion of the pedestrians. The control action is selected to make the robot move in the direction of the steepest descent, so away from the pedestrians. This formulation encodes just information about human motion, and does so through a kinematic model, not differently from how planners traditionally handle generic dynamic obstacles. [11] also employs a potential field as a cost function for the RRT planner [12]. In this case, high potential values representing human proxemics are used to discourage interactions with humans, while their future trajectories are taken into account by rendering the field time-varying within the planning window. Still, in this case too pedestrian positions are predicted based on a constant velocity and heading model. It is worth noting that [10] and [11] validate the approach in a kinematic simulator, which cannot highlight the peculiarities of human behavior. A different approach is proposed by [13], which creates "sensitive fields" around each pedestrian considering its speed and gaze direction. The fields are then considered as simulated hard obstacles by creating a virtual laser scan that includes their occupation.

Shifting the focus to the social forces class, a first example is [14], which employs the SFM to create a more natural global plan: the tentative global plan is obtained by the A* algorithm [15]. Then, the SFM is used to refine it by simulating the behavior a pedestrian would have in following it. This allows the robot to follow a human-like trajectory. The vast majority of the papers, apply the SFM to the robot directly at the local planner level. [16], for example, exploits the SFM in two ways. A first, custom, version is employed to predict human intentions, that is if the pedestrian wants to approach or avoid the robot. A second version is applied to the robot itself: by making it abide by the equations of a model describing human motion, the authors want to obtain a human-like, and thus socially acceptable, behavior. A very similar approach is used by [17], which was also adapted to follow a target person in [18]. Note that in these approaches in absence of obstacles, the SFM would make the robot move directly towards the goal. This makes the choice of the goal a critical task: the robot could otherwise get stuck in local minima. From a wider perspective, directly applying a model to guide the robot means not exploring the control space at all: the choice is already made at the moment of the model design. Of course, this may lead to suboptimal decisions, like the local minima mentioned above. To mitigate this issue, [19] introduces a complex, RRT-based anticipative kinodynamic planner. The idea is to select multiple goal points, randomly sampled around a preview point along the global plan. A trajectory and the required control actions are then simulated for each goal by applying the SFM from the robot pose. Finally, a multi-objective cost function is used to select the best trajectory.

From this brief literature review, a trend emerges: almost all approaches tackling navigation in presence of humans do so by imitating human behavior. The approaches that do not, typically employ kinematic models more suitable for generic dynamic obstacles than for pedestrians. Imitating human motion does indeed create more natural motion, but often requires "tricks" to adapt the models to a completely different way of locomotion. Furthermore, simply applying the models severely impacts the optimality of the motion planning algorithms and complex mechanisms are needed to recover it while maintaining their benefits. In this paper, we aim to explore a simpler solution, in which pedestrian motion models are used for prediction purposes only, while the optimality of the choice is maintained by design.

III. Socially Aware Dynamic Window Approach

A. Dynamic Window Approach

The Dynamic Window Approach [2] is one of the most popular motion planners, especially for differential drive robots. DWA is a receding horizon method based on a grid search optimization in the velocities space, with a preliminary step that reduces the search space. The reduction is performed by applying dynamic constraints, limiting the maximum and minimum speeds to those achievable within the prediction horizon T:

$$\begin{aligned} W_D = \Big\{ \langle v, \omega \rangle : \; & v_{min} \leq v \leq v_{max} \\ & \wedge \omega_{min} \leq \omega \leq \omega_{max} \\ & \wedge \bar{v} - a_{max}T \leq v \leq \bar{v} + a_{max}T \\ & \wedge \bar{\omega} - \dot{\omega}_{max}T \leq \omega \leq \bar{\omega} - \dot{\omega}_{max}T \Big\} \end{aligned} \tag{1}$$

where W_D is the dynamic window and $\bar{v}$ and $\bar{\omega}$ are the current speed and yaw rate, respectively. v_{min}, v_{max}, ω_{min}, ω_{max} represent the minimum and maximum velocities and yaw rates the robot can achieve, while a_{max} and $\dot{\omega}_{max}$ are the maximum linear and rotational accelerations. Scoring of the trajectories is done through multiple cost functions, which are linearly combined. The optimization problem is then defined as

$$\langle v^*, \omega^* \rangle = \underset{\langle v,\omega\rangle \in W_D}{\operatorname{argmin}} \; w_o J_o\left(v, \omega\right) + w_p J_p\left(v, \omega\right) \tag{2}$$

In this work, two cost functions are combined: an obstacles cost function J_o and a path cost function J_p, with their respective weights being w_o and w_p.

The cost functions simulate the robot trajectory in the prediction horizon assuming a constant velocity model. The obstacle cost function then discards all the trajectories colliding with obstacles, based on range data typically coming from a LiDAR sensor. The path cost function, instead, pushes the robot forwards and toward the global plan. This is achieved by assigning to each robot location a weight linearly proportional to the distance from the goal and a cost quadratic in the distance from the global plan. A representation of the path cost function is given in Fig. 1.

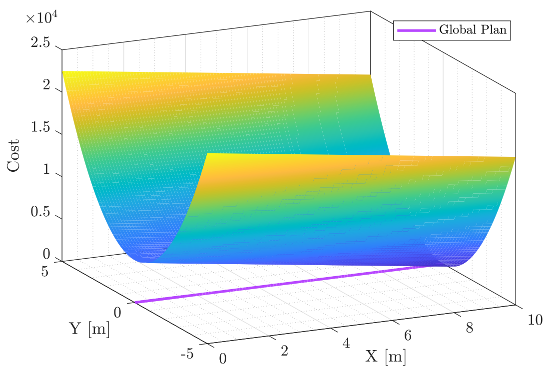

Fig. 1: DWA path cost function example

B. Social Forces Model

This paper proposes an additional cost function, dubbed “Social Cost Function” (SCF), which handles exclusively pedestrians. The idea of the SCF is to predict the humans’ trajectories using the SFM, scoring the robot control actions based on the disturbance caused to the pedestrians. The SFM, first introduced in [3], is by far the most popular model for pedestrian trajectory prediction. Each pedestrian is modeled as a point mass subject to a set of forces and is associated with a desired goal location. The goal attracts the pedestrian with a force given by

$$\boldsymbol{F}_{goal} = \frac{v^0 \boldsymbol{e} - \boldsymbol{v}}{\tau} \tag{3}$$

being v^0 the pedestrian’s desired speed, $\boldsymbol{e}$ the unit vector pointing from its position to the goal position, v its speed vector and τ a time constant. Obstacles exert a repulsive force instead. In this work, the original circular specification is used for static obstacles:

$$\boldsymbol{F}_o = A_o e^{-\frac{\|\boldsymbol{d}_o\|}{B_o}} \frac{\boldsymbol{d}_o}{\|\boldsymbol{d}_o\|} \tag{4}$$

being $\boldsymbol{d}_o$ the vector from the pedestrian position to the closest point of the obstacle and A_o, B_o two tuning parameters. For other pedestrians and the robot itself, the elliptical specification [20] is used. The force felt by pedestrian i due to the presence of agent j is then defined by equations (5).

$$\boldsymbol{F}_{ij} = W(\theta_{ij}) \cdot A_j e^{-\frac{b_{ij}}{B_j}} \cdot D_{ij} \tag{5a}$$

$$D_{ij} = \frac{\|\boldsymbol{d}_{ij}\| + \|\boldsymbol{d}_{ij} - \boldsymbol{y}_{ij}\|}{4b_{ij}} \frac{\boldsymbol{d}_{ij}}{\|\boldsymbol{d}_{ij}\|} + \frac{\boldsymbol{d}_{ij} - \boldsymbol{y}_{ij}}{\|\boldsymbol{d}_{ij} - \boldsymbol{y}_{ij}\|} \tag{5b}$$

$$b_{ij} = \frac{1}{2}\sqrt{(\|\boldsymbol{d}_{ij}\| + \|\boldsymbol{d}_{ij} - (\boldsymbol{v_j} - \boldsymbol{v_i}\Delta t)\|)^2 - \|\boldsymbol{y}_{ji}\|^2} \tag{5c}$$

$$\boldsymbol{y}_{ij} = (\boldsymbol{v}_i - \boldsymbol{v}_j)\,\Delta t \tag{5d}$$

$$W(\theta_{ij}) = \lambda_i + (1 - \lambda_i)\frac{1 + \cos\theta_{ij}}{2} \tag{5e}$$

Here, $W(\theta_{ij})$ is an anisotropy term dependent on the angle of approach θ_{ij}, which models the fact that people prefer a larger free space in front of them, but can accept closer obstacles on the back. The remaining terms are function of the distance vector $\boldsymbol{d}_{ij}$, the relative velocity $\boldsymbol{v}_i - \boldsymbol{v}_j$ and a few parameters (namely, A_j, B_j, λ_i and Δt). An example of the elliptical force field is shown in Fig. 2. In this paper,

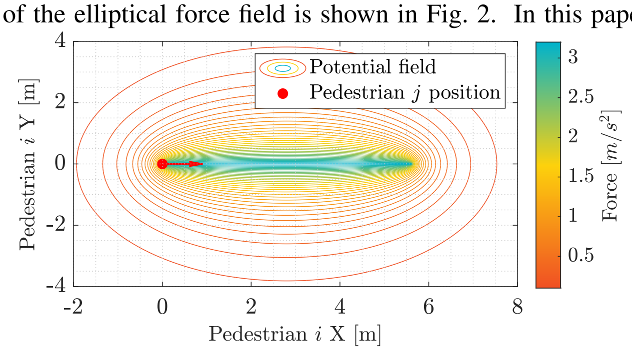

Fig. 2: Example of the shape of the elliptical force field felt by agent i due to the presence of agent j. The speeds of the agents are $\boldsymbol{v}_i = [-1\ 0]^T m/s$ and $\boldsymbol{v}_j = [1\ 0]^T m/s$, respectively

the force as formulated in (5) is applied for other humans, as well as the robot itself. Tuning of the A_j and B_j parameters is particularly critical: these define the attitude pedestrians present in reciprocal interactions and in interactions with the robot. Lower values of the parameters will lower the force felt by the pedestrian due to the presence of another agent, making closer encounters possible. Vice-versa, higher values will make the modeled pedestrian keep a distance. As shown in [21], one could say that high values represent a person aware of their surroundings, or scared of other agents, while lower values model unaware or “aggressive” people. To the extreme, with values close to zero the pedestrian model will not attempt any avoidance or collaboration, a case representative of a distracted person. Fig. 3 shows the effect of such parameters for three classes of pedestrians: unaware (or distracted: $A_j = 0.01$, $B_j = 0.92$), nominal (or balanced: $A_j = 2.98$, $B_j = 1.1$), and aware (or timid: $A_j = 2.0$, $B_j = 6.0$). For the purposes of this visualization, the pedestrians encounter a robot that does not attempt to avoid them.

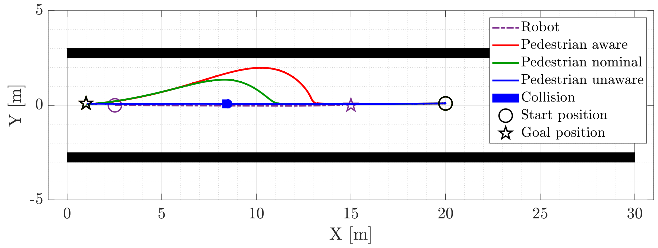

Fig. 3: Pedestrian avoidance of a non-cooperating robot in a wide corridor, as modeled by the SFM. Three classes of pedestrian attitudes are compared.

C. Social Cost Function

The goal of the Social Cost Function is to evaluate how much impact a given robot control action will cause to nearby pedestrians. With this target, each robot control action (v, ω) is simulated inside the DWA prediction horizon, similarly to what occurs in the obstacle and path cost functions. Each simulation, however, includes also the positions and velocities of the pedestrians, which are predicted according to the SFM equations outlined in Section III-B. It is important to note that, since the SFM models interactions between the agents on the scene, the pedestrian trajectories will depend on the robot one. For this reason, it is not possible, to perform just one simulation involving the pedestrians for all the robot trajectories. In this work, we compute a pedestrian simulation for each robot control action evaluated by the DWA. Then, the social score of each trajectory is computed as

$$J_s(v, \omega) = \frac{dt}{T} \sum_{t=0}^{T/dt} \sum_{i=1}^{N_{ped}} \boldsymbol{F}_{ir} \tag{6}$$

being T the DWA prediction horizon, dt the simulation time step, N_{ped} the number of pedestrians in the vicinity of the robot and $\boldsymbol{F}_{ir}$ the force felt according to the SFM (equation (5)) by pedestrian i due to the presence of the robot. In essence, the proposed SCF weights how much the pedestrian has to deviate from its ideal trajectory to stay at a comfortable distance from the robot. This ensures that the robot exploits the cooperation of the pedestrians only when strictly necessary (as presented in Section I), while it will tend to take more responsibility in the avoidance maneuver whenever possible. The SCF is added to the other cost functions of the DWA, modifying the cost minimized in equation (2) into

$$w_o J_o(v, \omega) + w_p J_p(v, \omega) + w_s J_s(v, \omega) \tag{7}$$

being w_s the linear combination term for the SCF J_s. Particular care needs to be taken when using the SCF: in order not to consider the pedestrians twice, they need to be removed from the point cloud fed to the obstacles cost function J_o.

It is worth mentioning that the parameters of $\boldsymbol{F}_{ir}$ as defined in (5) can be considered design parameters for the purposes of SCF implementation and need not be adapted to the actual behavior of the encountered pedestrian. This can be justified philosophically stating that the robot should treat all pedestrians equally, without leaving more space for some pedestrians and less for others. This approach comes with the additional benefit of trajectory predictability, which can be appreciated by timid pedestrians.

IV. SIMULATION SETUP

The proposed approach was implemented in the ROS [22] framework and was validated in a simulated corridor environment within the Gazebo simulator [23] set up to recreate a passing face-to-face scenario. The robot used for testing is a simulation model of Yape [24], [25], a two-wheeled inverted pendulum robot designed for last-mile delivery. It features a differential drive drivetrain and is fitted with a 16-layer LiDAR for obstacle perception. In order to validate against a realistic human-like behavior, a custom SFM-based pedestrian simulator was developed as a Gazebo plugin. Fig. 4 shows a snapshot of the simulation environment during a face-to-face passing test.

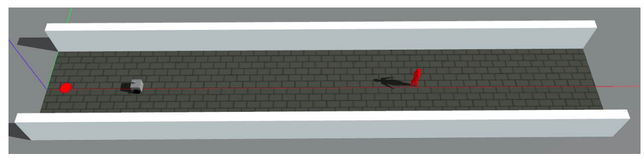

Fig. 4: Snapshot of the simulated environment: the differential drive robot is represented on the left, the pedestrian is on the right, with its goal represented by the red circle behind the robot.

The input data required by the DWA planner are:

- Pose of and velocity of the robot.
- LiDAR point clouds to feed the obstacles cost function.
- Positions, velocities, goal positions and SFM attitude parameters for each pedestrian, to feed the SCF.

The simulation environment allows to remove non-idealities from the algorithm testing by exploiting ground truth data for both localization and pedestrian tracking (positions and velocities). In real-world applications, this data would be provided by a localization algorithm like AMCL [26] and a target tracking algorithm [25]. To remove the pedestrians from the point cloud, they were simply made invisible to the LiDAR within the simulator. A similar result could be achieved by classifying the tracked obstacles and pre-processing the point clouds to remove all the points labeled as pedestrians. While ground truth data could be used for the SFM parameters as well, it is not realistic to accurately estimate them online in the real world. For this reason, robustness to the pedestrian attitude was extensively tested, as explained in Section V. Finally, ground truth data was also exploited to feed the cost function with pedestrian goal locations. While not trivial, online goal estimation is possible, as proven by [21].

V. VALIDATION RESULTS

The proposed planner is evaluated against the three pedestrian classes defined in Section III-B: unaware, nominal and aware. For each pedestrian class, the algorithm is tested for robustness, by varying the SFM parameters used for trajectory prediction, simulating a pedestrian class misclassification. Three algorithms are benchmarked:

- The DWA fitted with the proposed SCF.
- The original DWA as formulated in equation (2), which considers pedestrians as static obstacles.
- The DWA fitted with a Distance Cost Function (DCF), which employs the same SFM-based predictor of the SCF, but weights the robot trajectories based on the robot-pedestrian distance $\|\boldsymbol{d}_{ir}\|$:

$$J_d(v,\omega) = W_d \frac{1}{N_{ped}} \frac{dt}{T} \sum_{t=0}^{T/dt} \sum_{i=1}^{N_{ped}} \max\left(0, d^{max} - \|\boldsymbol{d}_{ir}\|\right) \tag{8}$$

being W_d a scaling factor and d^{max} is the distance threshold over which no weight is given to the robot-pedestrian interactions. The reasoning behind this second benchmarked cost function is to assess whether the SFM-based prediction alone can improve the performance of the local planner.

Fig. 5a shows the trajectories of the robot and the pedestrian in the case of a nominal simulated pedestrian attitude, with nominal parameters used for trajectory prediction. By comparing the trajectories of the original DWA (purple) with the two proposed cost functions, it is apparent that the SFM-based prediction is capable of reacting earlier to an approaching pedestrian. Indeed, the trajectory of the pedestrian facing the robot with the original DWA has to widen its trajectory more and in a sharper way. Moreover, comparing the Social and Distance cost functions one can note how the inclusion of a speed term within the weighting function further increases the smoothness of the avoidance maneuver. Fig. 5b presents the profile of the pedestrian speed, which shows a slowdown phenomenon in correspondence of the approach point. The severity of this braking maneuver is significantly reduced with the SFM-based prediction, especially when paired with the SCF. A reduced slowdown signals that the pedestrian was disturbed less and could continue towards its goal without altering its path too abruptly.

The lower impact is also represented by the synthetic metrics in Fig. 6. The top figures depict the average and maximum forces felt by the pedestrian around the approach point according to (5), while the latter quantifies the percentage slowdown from the pedestrian's desired speed. In each row, the first figure was obtained with an unaware simulated pedestrian, the central ones with a nominal pedestrian, and the right ones with an aware pedestrian. Within each figure, different colors (indexed on the X-axis) refer to the models used by the local planner to predict the pedestrian's motion. Over the three graphs of each row, all possible mismatches between real and assumed pedestrian models are evaluated, assessing the robustness of the planners to model uncertainties. In all evaluated cases, the SCF performs better than both the original DWA and the DCF, proving its robustness even to severe pedestrian misclassification. It is worth noting that, in the case of an unaware pedestrian, the original DWA does not manage to avoid the collision, reacting too late to the incoming pedestrian. On the contrary, the Socially Aware DWA takes full responsibility of the avoidance maneuver, widening the trajectory so much that the pedestrian speed is virtually unaffected. The DWA modified with the Distance Cost Function, instead, does manage to avoid the unaware pedestrian, but it is quite susceptible to errors in the prediction model: when the pedestrian is classified as nominal or aware, a collision cannot be avoided.

VI. CONCLUSIONS

This paper presented a socially aware version of the Dynamic Window Approach planner. The proposed planner is designed to handle interactions with humans in a smooth and socially acceptable way, by predicting their trajectories using the Social Forces Model and implementing a novel cost function. The Social Cost Function scores the trajectories based on how much disturbance they give to pedestrians by using the elliptical formulation of the Social Forces Model obstacles cost. The approach is validated in a simulation environment with realistic pedestrian motions, avoiding collisions both with collaborating and distracted pedestrians, even under in presence of severe model uncertainties. The approach shows superior performance with respect to the original DWA and a distance-based scoring function, which fails when facing non-collaborating pedestrians. Given the promising results, the experimental validation of the proposed algorithm is the natural next step and is currently underway.

REFERENCES

[1] C. Mavrogiannis, F. Baldini, A. Wang, D. Zhao, P. Trautman, A. Steinfeld, and J. Oh, "Core Challenges of Social Robot Navigation: A Survey," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 3, pp. 36:1–36:39, Apr. 2023.

[2] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Automat. Mag.*, vol. 4, no. 1, pp. 23–33, Mar. 1997.

[3] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, no. 5, pp. 4282–4286, May 1995.

[4] S. Choi, E. Kim, and S. Oh, "Real-time navigation in crowded dynamic environments using Gaussian process motion control," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong: IEEE, May 2014, pp. 3221–3226.

[5] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 335–356, Mar. 2015.

[6] P. Henry, C. Vollmer, B. Ferris, and D. Fox, "Learning to navigate through crowded environments," in *2010 IEEE International Conference on Robotics and Automation*. Anchorage, AK: IEEE, May 2010, pp. 981–986.

[7] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2017, pp. 1343–1350, iSSN: 2153-0866.

[8] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *1985 IEEE International Conference on Robotics and Automation Proceedings*, vol. 2, Mar. 1985, pp. 500–505.

[9] E. T. Hall, "A System for the Notation of Proxemic Behavior," *American Anthropologist*, vol. 65, no. 5, pp. 1003–1026, 1963.

[10] N. Pradhan, T. Burg, and S. Birchfield, "Robot crowd navigation using predictive position fields in the potential function framework," in *Proceedings of the 2011 American Control Conference*. San Francisco, CA: IEEE, June 2011, pp. 4628–4633.

[11] M. Svenstrup, T. Bak, and H. J. Andersen, "Trajectory planning for robots in dynamic human environments," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Taipei: IEEE, Oct. 2010, pp. 4293–4298.

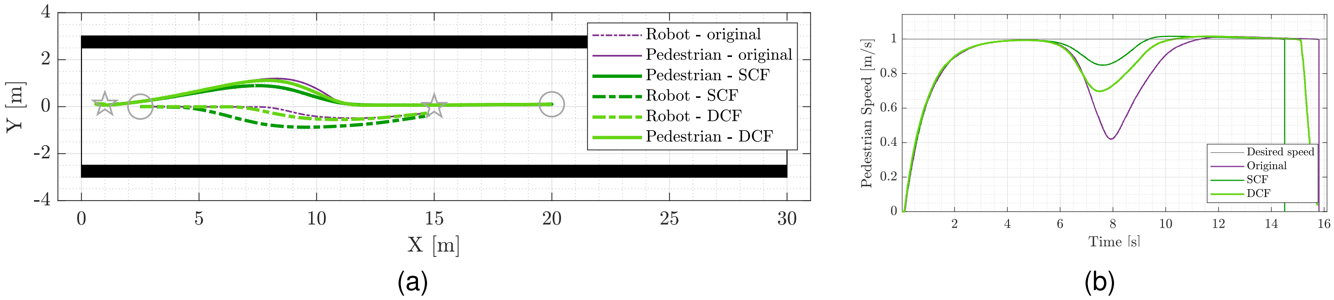

Fig. 5: Trajectories (a), and pedestrian speeds (b) for the simulations with nominal pedestrian attitude and nominal prediction parameters in the Social and Distance cost functions.

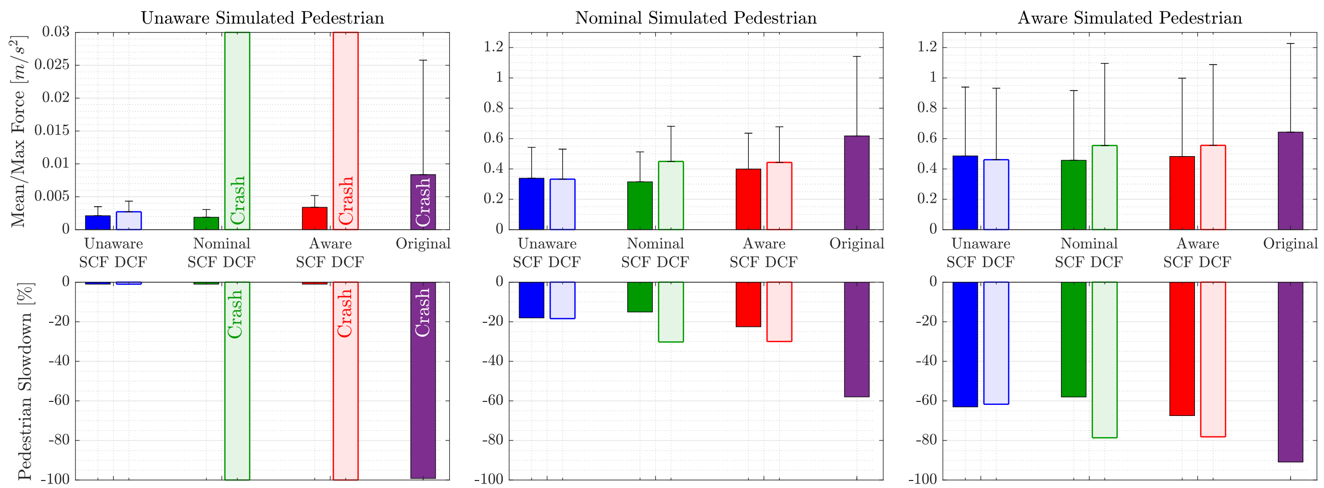

Fig. 6: Average and maximum forces felt by the simulated pedestrian due to the presence of the robot according to (5) (top) and pedestrian slowdown (bottom) in case of unaware (left), nominal (center) and aware (right) simulated pedestrian. Within each figure, the SCF and DCF are tested with unaware (blue), nominal (green), and aware (red) assumed behaviors.

[12] S. M. LaValle and J. J. Kuffner, "Randomized Kinodynamic Planning," *The International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, May 2001.

[13] C.-P. Lam, C.-T. Chou, K.-H. Chiang, and L.-C. Fu, "Human-Centered Robot Navigation—Towards a Harmoniously Human–Robot Coexisting Environment," *IEEE Trans. Robot.*, vol. 27, no. 1, pp. 99–112, Feb. 2011.

[14] A. Cosgun, E. A. Sisbot, and H. I. Christensen, "Anticipatory robot path planning in human environments," in *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. New York, NY, USA: IEEE, Aug. 2016, pp. 562–569.

[15] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, July 1968.

[16] P. Ratsamee, Y. Mae, K. Kamiyama, M. Horade, M. Kojima, and T. Arai, "Social interactive robot navigation based on human intention analysis from face orientation and human path prediction," *Robomech J*, vol. 2, no. 1, p. 11, Dec. 2015.

[17] G. Ferrer, A. Garrell, and A. Sanfeliu, "Social-aware robot navigation in urban environments," in *2013 European Conference on Mobile Robots*. Barcelona, Catalonia, Spain: IEEE, Sept. 2013, pp. 331–336.

[18] ——, "Robot companion: A social-force based approach with human awareness-navigation in crowded environments," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Tokyo: IEEE, Nov. 2013, pp. 1688–1694.

[19] G. Ferrer and A. Sanfeliu, "Anticipative kinodynamic planning: multi-objective robot navigation in urban and dynamic environments," *Auton Robot*, vol. 43, no. 6, pp. 1473–1488, Aug. 2019.

[20] D. Helbing and A. Johansson, "Pedestrian, Crowd and Evacuation Dynamics," in *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, Ed. New York, NY: Springer, 2009, pp. 6476–6495.

[21] G. Ferrer and A. Sanfeliu, "Behavior estimation for a complete framework for human motion prediction in crowded environments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 5940–5945, iSSN: 1050-4729.

[22] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source Robot Operating System," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009, p. 6.

[23] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, Sept. 2004, pp. 2149–2154 vol.3.

[24] F. Parravicini, M. Corno, and S. Savaresi, "Robust State Observers for Two Wheeled Inverted Pendulum under wheel-slip," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. Auckland, New Zealand: IEEE, Oct. 2019, pp. 1525–1530.

[25] ——, "Extended target tracking for autonomous street crossing," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 15 440–15 445, 2020.

[26] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo localization for mobile robots," in *Proceedings 1999 IEEE International Conference on Robotics and Automation*, vol. 2. Detroit, MI, USA: IEEE, 1999, pp. 1322–1328.