

Towards Online Socially Acceptable Robot Navigation

Steven Silva¹, Dennys Paillacho², Nervo Verdezoto¹, and Juan David Hernández¹

Abstract—When robots move through social spaces (i.e., environments shared with people) such as museums and shopping centers, they must navigate in a safe and socially acceptable manner to facilitate their inclusion and adoption. Therefore, robots operating in such settings must be able not only to avoid colliding with nearby obstacles, but also to show socially accepted behaviors, e.g., by minimizing the disruption in the comfort zone of nearby people. While there are well known approaches for social robot navigation, they are mostly based on social force models, which suffer from local minima. Meanwhile, other robot navigation frameworks do not consider social aspects. In this paper, we present an online social robot navigation framework, which is capable of generating collision free and socially acceptable paths online in uncontrolled crowded environments. Our proposed framework employs a modified sampling-based planner together with a new social relevance validity checking strategy. To evaluate our approach, we have designed a simulated social space in which the Pepper robot can safely navigate in a socially accepted manner. We compare our approach with other two alternative solutions while measuring specific social navigation metrics.

I. INTRODUCTION

New and potential applications of service robots (e.g., a robot guide in hospitals or a waiter robot) require them to navigate around people in narrow and indoor dynamic environments [1] (see Fig. 1). Such applications pose additional challenges and considerations, such as maximizing human safety and comfort of nearby people [2]. Therefore, social robot navigation differs from other robot navigation cases in which the environment includes static and dynamic obstacles, but no social aspects must be considered when planning the paths that guide the robots (e.g., underwater exploration [3]).

In general, socially aware robot navigation considers comfort, naturalness and sociability [4], and it refers to robots that are capable of moving around while considering social aspects, for example, maintaining a comfortable distance to nearby people [5]. Kivrak et al. [6], for instance, describe a social compliant robot as one that moves around people while taking them into account as social entities, and not only as simple obstacles, thus requiring to minimize the discomfort of an individual or group of people. Therefore, since humans usually follow certain social conventions and behaviors, it is

The authors would like to acknowledge the support provided by Cardiff University's Centre for Artificial Intelligence, Robotics and Human-Machine Systems (IROHMS) operation C82092, part-funded by the European Regional Development Fund (ERDF) through the Welsh Government.

¹S. Silva, N. Verdezoto, and J.D. Hernández are with IROHMS in the School of Computer Science and Informatics at Cardiff University, UK. {SilvaS1, VerdezotoDiasN, HernandezVegaJ}@cardiff.ac.uk

²D. Paillacho is with the Research Development and Innovation Center of Computer Systems - CIDIS at ESPOL Polytechnic University, Ecuador. dpaillac@espol.edu.ec



Fig. 1. Start-to-goal query from the Start point (lower right side) and the Goal destination (upper left side).

important for robots to exhibit and respect such conventions and behaviors while navigating social spaces [7].

Prior work has tried to generate socially-aware navigation, and an important number of the existing approaches are based on the Social Force Model (SFM) that was introduced by Helbing et al. [8]. The SFM seeks to express the social human navigation behavior as a resultant force, which is calculated as the sum of repulsive and attractive forces. While objects and pedestrians exert repulsive forces depending on factors such as proxemics and gaze direction, a desired destination exerts an attractive force [9]. Each of the forces is dependent on a scalar factor (also known as force factor) that controls the force magnitude. The total resulting force is finally used to calculate the desired velocity vector, which corresponds to the motion that a person would follow.

Nevertheless, SFM-based approaches suffer from limitations that are well known in other potential-field-based planning approaches, e.g., the possibility of encountering local minima that do not correspond to the final goal, thus preventing the robot from reaching the intended destination [10]. In this paper, we present an online social robot navigation framework that is capable of solving requests to guide a robot from a start to a goal position (i.e., start-to-goal queries), while also providing social acceptable paths in low controlled and crowded social spaces. To do so, our framework seeks to minimize the potential discomfort to surrounding people. Our approach works in an online manner

and avoids collisions with surrounding obstacles (including moving people). Our framework is composed of three modules: 1) a world modeling module that provides information about the obstacles and the agents in the environment, 2) a planning module to generate collision-free and socially admissible paths, and 3) a differential control module that allows the robot to follow the calculated path.

In proposing this novel framework, this paper makes the following contributions. First, we present an online social robot navigation framework that is capable of solving start-to-goal queries in social spaces, which uses a cost optimization function that is based on the individual’s human comfort zone model, thus allowing to generate socially accepted robot motions. Second, we have proposed a social relevance validity checking strategy to reduce needless computation consumption by considering the limited sensors’ detection range. Lastly, we have tested and compared our approach in simulation against two alternative approaches, including the widely used SFM.

II. RELATED WORK

Social robot navigation approaches can be classified in two main categories: those that are based on the SFM, and those that use robot path planning approaches which formulate social criteria as optimization cost functions.

A. SFM-based Approaches

The originally proposed SFM can be extended by incorporating additional forces to obtain different desired behaviors, thus resulting in what is commonly referred to as Extended Social Force Model (ESFM). One example of ESFM was proposed by Kivrak et al. to model new collision prediction repulsive force based on the agents’ velocities, future distance, bearing angle and time of collision [11]. Other modifications of SFM include a combination of ESFM and Hybrid Reciprocal Velocity Obstacle (HRVO) [12], which considers the desired output velocity to avoid collisions [13]. This latter approach also allows considering groups of people and human-objects interactions (e.g., a person interacting with a TV).

Similarly, Ferrer et al. [14] presented another ESFM that allows a robot to follow a person, where both navigate an outdoor environment among obstacles and other pedestrians. This approach includes an anisotropic factor that scales the interaction forces depending on the velocity and distance between the robot and nearby people. Their work makes use of a learning-based method to modify the ESFM’s parameters and force factors according to the environment.

Although prior works [6], [13], [14], [15] have proved to be effective in several cases, they inherit some common limitations from the originally proposed SFM. First, they can suffer from local minima induced by both obstacles and pedestrians [10]. Second, SFM-based approaches consider an unrealistic infinite range of perception to detect surrounding obstacles and people, thus assuming that every social agent in the scene can be tracked all the time, which is not the case

with a real-world robot. Lastly, this group of approaches depends on several parameters that are heuristically determined and cannot be easily generalized for different scenarios.

B. Robot Path Planning Based Approaches

Other common robot path planning approaches (e.g., search-based and sampling-based methods [16]) have been used to solve start-to-goal social robot navigation queries. Patompak et al. [17] proposed the Social Relationship Model (SRM), another ESFM, which takes into consideration people’s characteristics like motion velocity, gender as well as a social distance factor that is based on whether the individuals are related to each other or not. SRM results in a cost function that is used by a T-RRT algorithm [18] to generate socially acceptable paths. While the presented results show efficacy in solving the given start-to-goal queries, they limit their analysis to the path quality in terms of its length, without discussing other relevant metrics in social navigation. Furthermore, their experiments assume static environments, which is unrealistic in social spaces.

Similarly, Ngo et al. [19] uses a human interaction detection module, a personal space model and Truong’s Dynamic Social Zone (DSZ) [20] to establish a cost function. The resulting cost function is then used by a Dynamic Window Approach (DWA) planner [21] to generate socially acceptable paths. They also present two social physical safety metrics, the Collision Index (CI) and the Interaction Index (CII), which are used to assess their own approach.

While the previous works by Ngo et al. and Patompak et al. show simulation and experimental tests, the pedestrians involved are static and the spaces are uncrowded. Therefore, those approaches still have limitations in highly dynamic and indoor environments. Finally, the aforementioned approaches assume a fixed priori and complete knowledge of the surroundings, i.e., common obstacles in the environment do not change and all social agents’ state is known, which is not the case in real-world changing scenarios.

C. Social Robot Navigation Metrics

There are different metrics that can be used to estimate the human acceptance in social robot navigation. Some of those metrics are focused on aiming to maintain human comfort. Below we present some examples of such metrics:

1) *Total Number of Collisions*: represents collisions that the robot has with both people or any object. It can be presented as the amount of collisions per minute or total of collisions for a range of time [22].

2) *Social Individual Index (SII)*: this metric was firstly introduced by Truong et al. [13]. It measures the acceptable distance between the robot and a person. SII can be calculated as shown in Eq. (1), where (x_i^p, y_i^p) corresponds to the position of the human and (x_r, y_r) corresponds to the position of the robot. N is the number of humans near the robot, while $\sigma_0^p = \frac{d_c}{2}$, where d_c is suggested to be between [0.45-1.2 m] according to Hall’s personal space [23].

$$SII = \max_{i=1:N} \exp \left(- \left(\left(\frac{x_r - x_i^p}{\sqrt{2}\sigma_0^p} \right)^2 + \left(\frac{y_r - y_i^p}{\sqrt{2}\sigma_0^p} \right)^2 \right) \right) \quad (1)$$

3) *Relative Motion Index (RMI)*: this metric was introduced by Truong et al. [13], and it expresses the probability of a collision between the robot and a person. It considers the velocity and orientation of both the robot and people, and it has a maximum value when a robot and a human are moving towards each other at their highest velocity.

RMI is calculated as shown in Eq. (2), where β_i is the angle between the robot orientation and the vector projected from the robot to the human p_i . ϕ_i is the angle between the person orientation and the vector projected from the person to the robot. v_i^p and v_r are the velocities of the person p_i in reference to the main world frame, respectively.

$$RMI = \max_{i=1:K} \frac{2 + v_r \cos(\beta_i) + v_i^p \cos(\phi_i)}{\sqrt{(x_i^p - x_r)^2 + (y_i^p - y_r)^2}} \quad (2)$$

III. PROBLEM DEFINITION

The solution to the general start-to-goal robot path planning problem consists in finding a path p that connects a start and a goal configuration, q_{start} and q_{goal} , such that $p: [0, 1] \rightarrow \mathcal{C}$, such that $p(0) = q_{start}$ and $p(1) = q_{goal}$, and \mathcal{C} corresponds to robot's configuration space (C-Space) [24]. Social robot navigation has some aspects that differ from other path planning related problems, since it requires not to consider people surrounding the robot as simple dynamic obstacles. In this paper, we call this problem start-to-goal social robot path planning.

A. Definitions

Definition 1: A social agent is a social entity (i.e., a person), that is capable of expressing social behaviors, e.g., discomfort when interacting with their surroundings including other social agents or objects.

Definition 2: \mathcal{C} , is divided into free space (\mathcal{C}_{free}) and obstacle space (\mathcal{C}_{obs}), i.e., $\mathcal{C} = \mathcal{C}_{free} \cup \mathcal{C}_{obs}$ [24]. While \mathcal{C}_{obs} includes all the obstacles, in this work we would like to differentiate between social agents and other static and dynamic objects, i.e., $\mathcal{C}_{obs} = \mathcal{C}_{social_agents} \cup \mathcal{C}_{common_obs}$.

B. Social Robot Path Planning

In social robot path planning, the main objective is still to connect q_{start} and q_{goal} , but the solution path p must be not only collision free and feasible, but also socially accepted. In order to do so, we include a function $U(q_{start}, q_{goal}) \rightarrow \mathbb{R}$ that represents the discomfort of nearby social agents. Therefore, the optimal path P^* is given by:

$$P^*(q_{start}, q_{goal}) = \arg \min_{P(q_{start}, q_{goal}) \in \mathcal{C}_{free}} U(q_{start}, q_{goal}) \quad (3)$$

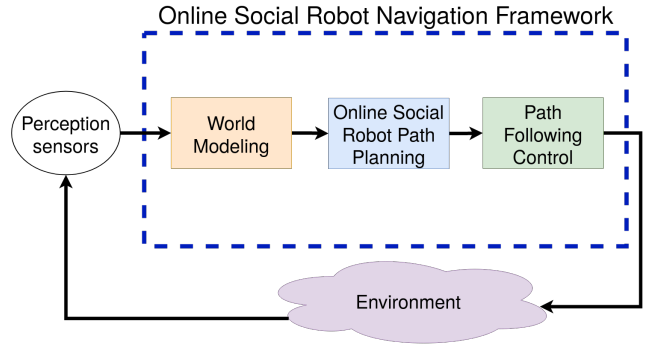


Fig. 2. Proposed framework for online social robot navigation (three modules inside the blue dashed square).

IV. ONLINE SOCIAL ROBOT NAVIGATION FRAMEWORK

In this section, we present a framework that seeks to solve the aforementioned social robot path planning problem. The framework consists of three functional modules: world modeling, online social robot path planning, and path following control.

A. World Modeling

World modeling refers to the representation of the robot's surroundings, which in the case of robot social navigation includes social agents, and other static and dynamic obstacles. The main reason for making this distinction (between social agents and other obstacles) is due to the necessity for providing enough flexibility when considering different human aspects (e.g., age, gender, disabilities, among others), which could potentially lead to further developing more advanced robot social behaviors. Therefore, the world modeling module will not only report all the obstacles (both social agents and other objects) for collision checking purposes, but also the required social agents' information for social validity checking purposes.

In the context of this work, we assume that the world modeling module represents obstacles with a volumetric representation given by the Octomap library [25] for collision checking purposes, and separately a social agent descriptor that includes the agent's position and velocity for social validity purposes.

B. Online Social Robot Path Planning

The online social robot path planning module is in charge of generating paths that are both collision-free and socially acceptable. In order to do so, this module iteratively calls a sampling-based tree planner (see Alg. 1), which is inspired by the RRT* algorithm [26], but it has been extended to solve the start-to-goal social robot path planning problem. Such a problem involves navigating highly dynamic social spaces, thus requiring the robot to continuously replan the path to ensure its validity. In order to do so, we propose to endow our sampling-based tree planner with two strategies that we have extended for the social navigation context.

Algorithm 1: Start2GoalSocialPathPlanning(q_{start}, q_{goal})

Input: q_{start} : Start configuration. q_{goal} : Goal configuration.**Output:** P^* : An array of configuration waypoints.

```
1 begin
2    $planner \leftarrow$ 
     SamplingBasedPlanner ( $SE2StateSpace$ )
3    $last\_best\_known\_solution \leftarrow \{\}$ 
4    $q_{new\_start} \leftarrow q_{start}$ 
5   while not  $stop\_condition$  do
6      $world\_model \leftarrow reqUpdatedWorldModel()$ 
7      $planner.updateWorldModel(world\_model)$ 
8      $planner.startFrom(last\_best\_known\_solution)$ 
9      $planner.solve(q_{new\_start}, q_{goal})$ 
10    if  $solution\_not\_found$  then
11       $P^* \leftarrow$ 
         $planner.getPartialSolution()$ 
12    else
13       $last\_best\_known\_solution \leftarrow$ 
         $planner.getSolution()$ 
14       $P^* \leftarrow last\_best\_known\_solution$ 
15    return  $P^*$ 
```

When the sampling-based tree is expanding to find a path towards the goal, we consider that any part of the tree that lies in unknown regions of \mathcal{C} is assumed to be both collision-free ($\notin C_{common_obs}$) and socially valid ($\notin C_{social_agents}$). This extends the *Opportunistic Collision Checking* [3] strategy to a more general *Opportunistic Validity Checking* strategy. The main objective of using this strategy is to avoid running unnecessary checking routines in areas that have not been explored by the robot.

Second, our modified planner *reuses the last best known solution* [3] (see Alg. 1 line 3) to rapidly fix paths that have become locally invalid. This strategy is particularly useful in cases in which the robot is incrementally acquiring information of a partially known and changing environment, as it is the case in the social navigation context. Furthermore, we extend this strategy to deal with social agents that can rapidly change their behavior which often leads to locally invalid paths, e.g., in situations where new social interactions among agents can make a path invalid, but where a valid path can be found nearby by avoiding the interacting agents.

1) *Social Comfort Optimization Cost Objective*: to generate socially acceptable paths for the motion of the robot, we have decided to use the Extended Personal Space Model (EPSM) [20] as the cost optimization function for our planning module. The EPSM is a Gaussian function presented

in Eq. (4) which expresses the individual human discomfort around a person p_i . In this function A^p and $(\sigma_0^{px}, \sigma_0^{py})$ correspond to the amplitude factor and the standard deviations of the Gaussian function, respectively.

$$f_i^p(x, y) = A^p \exp \left(- \left(\left(\frac{d \cos \theta - \theta_i^p}{\sqrt{2} \sigma_0^{px}} \right)^2 + \left(\frac{d \sin \theta - \theta_i^p}{\sqrt{2} \sigma_0^{py}} \right)^2 \right) \right) \quad (4)$$

where d , θ and θ_i^p are calculated as shown below, having (x_r, y_r) , (x_i^p, y_i^p) , θ_i^{pv} and θ_i^{ph} as the robot position, social agent position, velocity vector angle and the gaze angle respectively:

$$d = \sqrt{(x_r - x_i^p)^2 + (y_r - y_i^p)^2} \quad (5)$$

$$\theta = \text{atan2}((y_r - y_i^p), (x_r - x_i^p)) \quad (6)$$

$$\theta_i^p = \begin{cases} \theta_i^{pv} & \text{if } v_p^i > 0 \\ \theta_i^{ph} & \text{if } v_p^i = 0 \end{cases} \quad (7)$$

Finally, σ_0^{py} is modified depending on the robot's position around the agent as shown in Eq. (8), where v_i^p , f_v , f_{front} and f_{fov} are the social agent's velocity, the velocity factor, the front factor and the Field Of View (FOV) factor, respectively:

$$\sigma_0^{py} = \begin{cases} (1 + v_i^p f_v + f_{front} + f_{fov}) \sigma_0^{py} & \text{if robot is} \\ & \text{in agent FOV} \\ (1 + v_i^p f_v + f_{front}) \sigma_0^{py} & \text{if robot is not} \\ & \text{in agent FOV} \end{cases} \quad (8)$$

The EPSM is considered along the path length evaluation of our sampling based planner, such that the final cost of each state is computed as the integral of the social comfort with respect to the distance as defined in Eq. (9) and Eq. (10). As a result, our planning algorithm returns the path with the lowest found discomfort cost surrounding social agents.

$$U(q) = \max(f_i^p(q \rightarrow x_r, q \rightarrow y_r)) \quad (9)$$

$$Cost(q) = \int_0^q U(q) dq \quad (10)$$

2) *Social Relevance Validity Checking*: there is a limitation from the sensors that restricts the range of detection and tracking of social agents. To reduce computation consumption, we have defined a FOV for the robot (see Fig. 3), where only the agents inside the FOV are considered as part of C_{social_agents} , while the rest of the space is taken as free of social agents. The distance range D of the FOV, is increased by a ratio factor between the current and maximum speed of the robot.

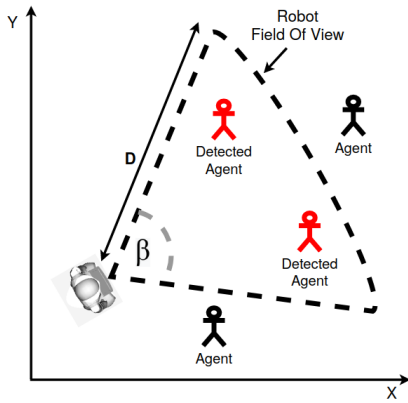


Fig. 3. Robot’s field of view, where β and D are the angle and distance range.

3) *Partial Solution*: in some cases the positioning of social agents can invalidate a current solution and avoid generating a new one. To prevent possible collisions when no new valid solution has been found (e.g., because the previous solution has become invalid due to a change in the surroundings), a partial solution (see Alg. 1 line 11), which corresponds to a portion of the previous solution that is still valid, is returned as the solution of our sampling-based path planner. By doing this, we enable the robot to move to the goal without colliding while our sampling-based planner finds another valid solution within the surroundings.

C. Path Following Control

This module corresponds to a base controller, which uses the path obtained from our sampling-based planner as inputs to move the robot using differential kinematics. In general the controller seeks to lower the yaw and distance error from the robot position to the configuration of the next waypoint. For the final waypoint, the robot decelerates and moves slower until the yaw and position error is less than a specified tolerance.

V. EXPERIMENTS AND RESULTS

To validate our proposed approach, we designed two case scenarios in which our framework was compared with two alternative approaches: the SFM and a RRT* based solution that optimizes the path length.

A. Simulation and Experimental Setting

In order to validate the different approaches, we conducted experiments in a simulated indoor office-like environment (see Fig. 1). In this scenario, a social robot that navigates the surroundings must deal with static obstacles (e.g., walls) and dynamic obstacles such as social agents that move around randomly to pre-defined waypoints.

The proposed framework has been implemented in ROS1¹ and the simulations conducted in Gazebo². In order to simulate social agents and their behavior, we have used a

¹<https://www.ros.org/>

²<http://gazebo.org>

customized version of *pedsim_ros*³ which has been modified to represent smoother pedestrians simulations by detecting and overcoming frozen agents that get trapped among objects in the surroundings due to local minima [10]. This paper is focused on the social robot navigation problem, particularly how to plan socially valid paths by obtaining the agents’ position from *pedsim_ros* as one of the inputs of the world modeling module.

In our validation, a simulated Pepper Robot must solve different start-to-goal queries required to safely navigate the environment. During the experiments, different metrics such as the SII, RMI, and the amount of collisions were gathered and analyzed. To better understand the experiments, the reader is referred to: <https://youtu.be/aZf27nthmX8>.

B. Case 1: Start-to-goal Navigation Query

In the first scenario, we simulated 10 social dynamic agents moving randomly between 13 different waypoints located in the environment. While the agents move around, they also consider the robot as an additional social agent (i.e. the social agents react to robot’s position and movement). In this situation, the main mission is to move the robot from the start position to the goal position as illustrated by Fig. 1.

The results from the metrics obtained are presented in Table I. For SFM, the success rate is none. The main reason is because of the presence of local minima in the zone of the start point generated by the surrounding walls.

Our approach has a much higher success rate than the original RRT* planner. Although both the social navigation metrics of SII and RMI are pretty similar, the successful attempts of the RRT* method only happen when the most favorable situation is presented, e.g., when the social agents are not in between the robot and the desired destination.

TABLE I

METRICS OBTAINED WITH EACH APPROACH IN CASE 1 (100 TESTS)

	Success rate	SII	RMI
		Mean (SD)	Mean (SD)
SFM	0%	-	-
RRT*	57%	0.030 (0.012)	0.881 (0.099)
Our framework	90%	0.030 (0.015)	0.881 (0.102)

Fig. 4 presents a graph of the SII measured during one complete experiment. While our framework generates less local maximum peaks, the graph of RRT* shows values higher than 0.5 which corresponds to possible collisions.

Something similar can be seen in Fig. 5, as the RRT* graph shows more abrupt and higher measurements than our approach. In addition, Fig. 5 shows how a general run of the RRT* planner could take more than double the time to complete the query mission even when best possible condition is presented.

C. Case 2: Start-to-goal Navigation Query with Social Agents in Close Proximity

We validated our approach with a second scenario as seen in Fig. 6, where we included a queue of social agents in

³https://github.com/sasilva1998/pedsim_ros

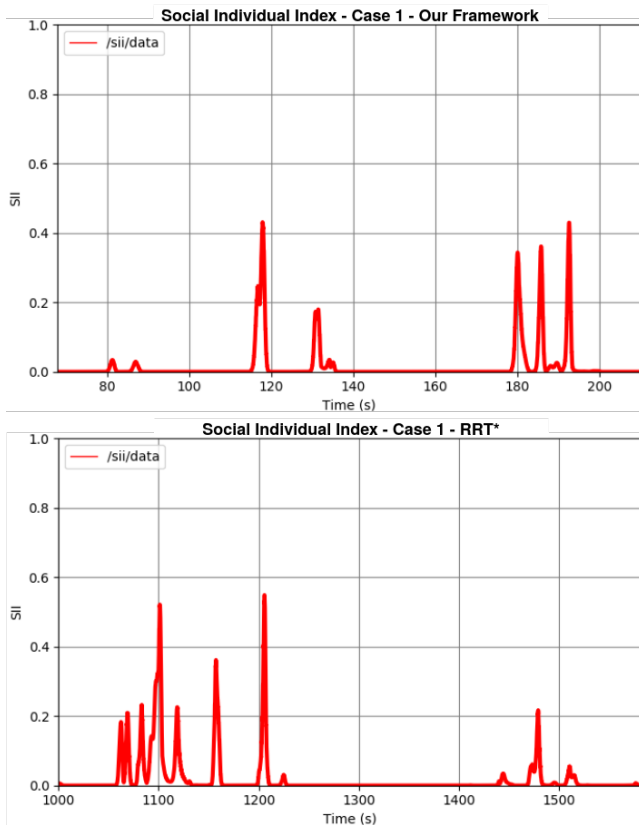


Fig. 4. SII graphs for our framework and RRT* in case 1.

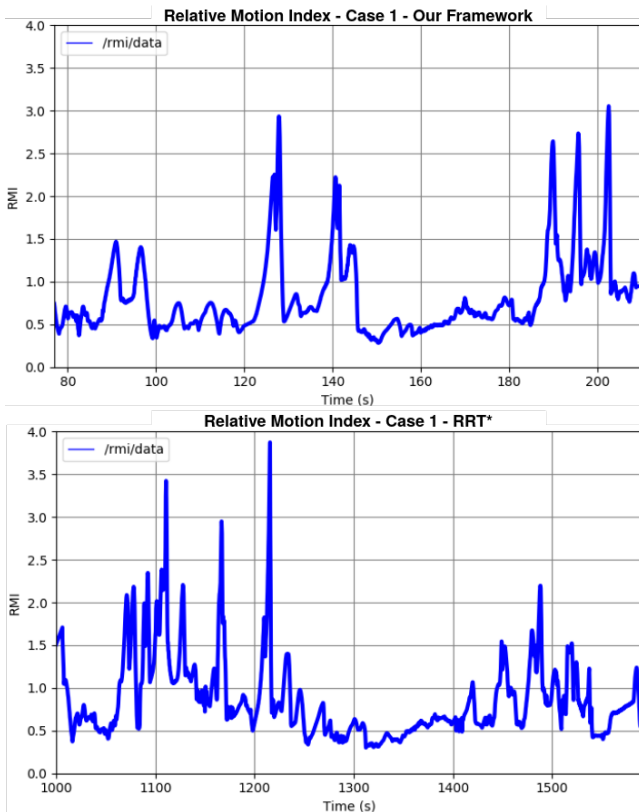


Fig. 5. RMI graphs for our framework and RRT* in case 1.

the narrow passage right before the goal (destination). This is a common scenario in social spaces like hospital halls or school buildings.



Fig. 6. Start-to-goal query from the Start point (lower side) and the Goal destination (upper side) including a static queue of agents.

Since the SFM depends on several factor values, for this specific situation we heuristically decided to test it with the Social Force Factor (SFF) with values of 22 and 27, for which the method had a significant change in its behavior. The results can be observed in Table II.

When using a $SFF = 22$, the SFM is successful in all cases. Yet by looking at these metrics in Table II, the SII and RMI values are considerably high. With this SFF, the forces generated by the social agents are not repulsive enough, hence highly increasing the probability of a collision.

When increasing the SFF above 27, no success was obtained in the query as a result of the local minima generated by the social agents. This latter means that the SFM can get the robot stuck not only in between obstacles but also between social agents, even when there is enough space within the corridor for the robot to reach the goal. For the original RRT* approach, a really low success rate is obtained from the experiments because of the highly dynamic environment. This approach struggles to find a collision free path with all the dynamic agents moving around discarding continuously a possible solution. In addition, RRT* needs replanning a path all over again, barely being able to solve the query. Our framework shows to have an acceptable success rate of 81% compared to the RRT* and the SFM. By looking at Table II, the social navigation metrics are quite low for the crowded and narrow corridor that is presented.

Figure 7 shows a graph of the measured SII. Our approach generates values that are barely higher than 0.5 and the graph is smooth without many changes. In the case of the SFM, the graph shows higher values than 0.7 certainly representing

collisions with the social agents. In addition, the SFM shows more local maxima than in our framework.

TABLE II
METRICS OBTAINED FOR EACH APPROACH IN CASE 2 (100 TESTS)

	Success Rate	SII	RMI	# Collisions
		Mean (SD)	Mean (SD)	Mean (SD)
SFM $SFF = 22$	100%	0.28 (0.01)	2.12 (0.09)	1.3 (0.70)
SFM $SFF = 27$	0%	-	-	-
RRT*	12%	-	-	-
Our framework	81%	0.15 (0.04)	1.51 (0.15)	0.11 (0.32)

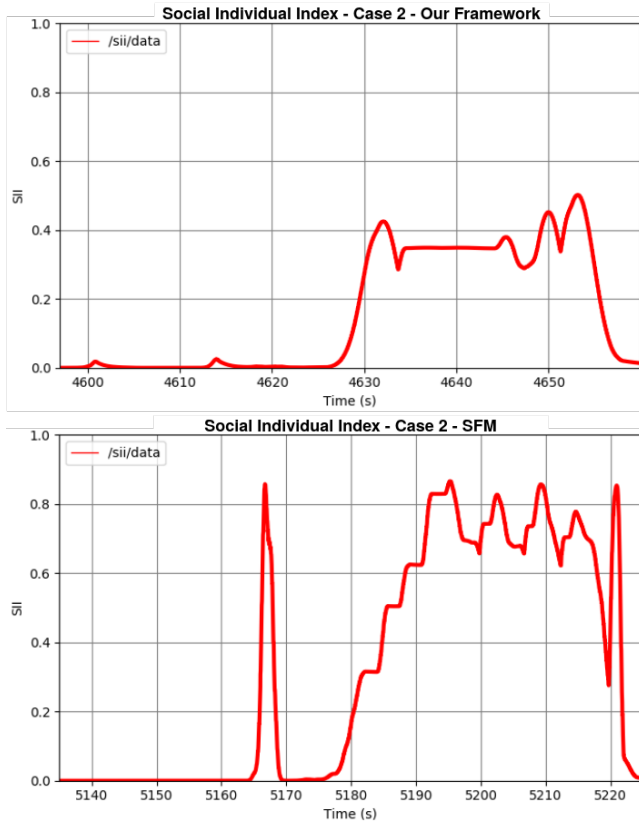


Fig. 7. SII graphs for our framework and the SFM with a $SFF = 22$ in case 2.

As illustrated in Fig. 8, our approach hardly reaches a value of 3.0 RMI. However, in the case of the SFM, the value is much higher than 6 RMI meaning that it is more likely to make the social agents uncomfortable with a higher chance of collision.

VI. DISCUSSION AND CONCLUSIONS

We proposed a flexible framework with three modules (world modelling, online social robot path planning, path following control), by extending a sampling-based planner, capable of generating socially acceptable paths for a robot in highly dynamic and crowded spaces. The framework considers a social cost optimization function based on the Gaussian distribution that models the social comfort space

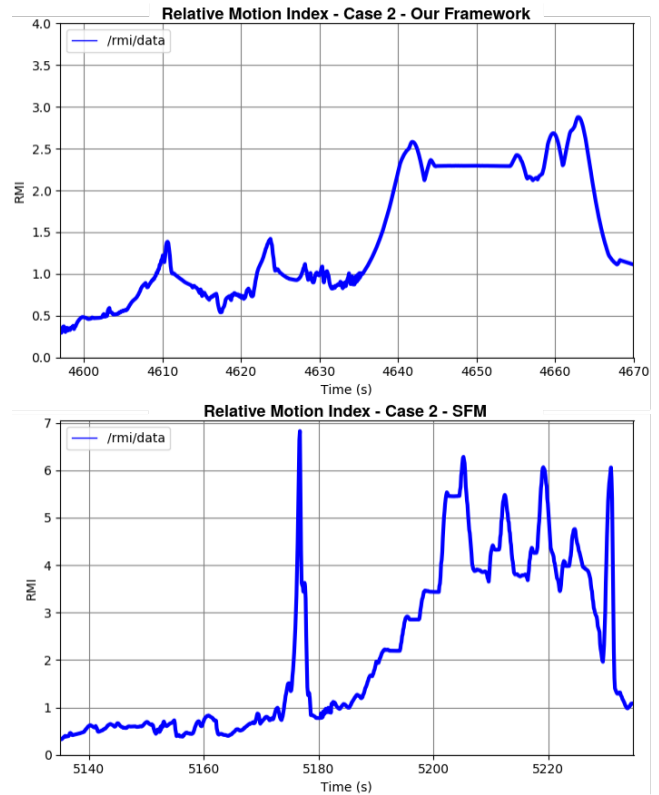


Fig. 8. RMI graphs for our framework and the SFM with a $SFF = 22$ in case 2.

around humans. The framework also includes the use of *Reusing the Last Best Known Solution* and the *Opportunistic Validity Checking* strategies which helped vastly for online path planning in social environments. We also proposed a new *Social Relevance Validity Checking* strategy, considers the robot's FOV to reduce unnecessary computation consumption from the interaction with social agents.

We tested the feasibility of our proposed approach through simulation and comparing with two other important approaches (RRT* and the SFM) in two complementary use case scenarios. The results from the social navigation metrics showed that our framework has an acceptable success rate in the experiments, while keeping the SII and RMI lower than the other two approaches, enabling the robot to work in dynamic crowded places reasonably. In contrast to SFM that requires several heuristically adjustments of its parameters, our framework has a more flexible and scalable structure since the world modeling can support more social interactions as input, e.g., a gaze tracing can be included as part of the cost optimization function without heuristic adjustments.

Our future work includes implementing a more complex social comfort model based on the DSZ model [20] to ensure a more holistic and acceptable navigation with groups and human-object interactions in social environments. Finally, we plan to conduct real-world experiments by using our proposed approach with a Pepper Robot while also extending the use case scenarios presented in this work.

REFERENCES

- [1] J. Cheng, H. Cheng, M. Q. Meng, and H. Zhang, "Autonomous Navigation by Mobile Robots in Human Environments: A Survey," in *2018 IEEE International Conference on Robotics and Biomimetics, ROBIO 2018*, 2018.
- [2] A. Bera, T. Randhavane, R. Prinja, and D. Manocha, "SocioSense: Robot navigation amongst pedestrians with social and psychological constraints," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-September, no. c, pp. 7018–7025, 2017.
- [3] J. D. Hernández, M. Moll, E. Vidal, M. Carreras, and L. E. Kavraki, "Planning feasible and safe paths online for autonomous underwater vehicles in unknown environments," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2016-Novem, pp. 1313–1320, 2016.
- [4] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, 2013.
- [5] J. Rios-Martinez, A. Spalanzani, and C. Laugier, "From Proxemics Theory to Socially-Aware Navigation: A Survey," *International Journal of Social Robotics*, vol. 7, no. 2, pp. 137–153, apr 2015. [Online]. Available: <http://link.springer.com/10.1007/s12369-014-0251-1>
- [6] H. Kivrak, F. Cakmak, H. Kose, and S. Yavuz, "Social navigation framework for assistive robots in human inhabited unknown environments," *Engineering Science and Technology, an International Journal*, no. xxxx, 2020. [Online]. Available: <https://doi.org/10.1016/j.jestch.2020.08.008>
- [7] A. K. Reddy, V. Malviya, and R. Kala, "Social cues in the autonomous navigation of indoor mobile robots," *International Journal of Social Robotics*, vol. 13, pp. 1335–1358, 9 2021. [Online]. Available: <http://link.springer.com/10.1007/s12369-020-00721-1https://link.springer.com/10.1007/s12369-020-00721-1>
- [8] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, pp. 4282–4286, 1995.
- [9] H. Kivrak, F. Cakmak, H. Kose, and S. Yavuz, "Waypoint based path planner for socially aware robot navigation," *Cluster Computing*, 2022. [Online]. Available: <https://doi.org/10.1007/s10586-021-03479-x>
- [10] J. Barraquand and J. C. Latombe, "Robot Motion Planning: A Distributed Representation Approach," *The International Journal of Robotics Research*, vol. 10, no. 6, pp. 628–649, 1991.
- [11] H. Kivrak, F. Cakmak, H. Kose, and S. Yavuz, "A multilevel mapping based pedestrian model for social robot navigation tasks in unknown human environments," in *International Conference on Innovations in Intelligent Systems and Applications (INISTA)*, 2020.
- [12] J. Snape, J. Van Den Berg, S. J. Guy, and D. Manocha, "The Hybrid Reciprocal Velocity Obstacle," *IEEE Transactions on Robotics*, vol. 27599, no. October, pp. 1–11, 2009.
- [13] X.-T. Truong and T. D. Ngo, "Toward socially aware robot navigation in dynamic and crowded environments: A proactive social motion model," *IEEE Transactions on Automation Science and Engineering*, vol. 14, pp. 1743–1760, 10 2017. [Online]. Available: <http://ieeexplore.ieee.org/document/8011466/>
- [14] G. Ferrer, A. G. Zulueta, F. H. Cotarelo, and A. Sanfeliu, "Robot social-aware navigation framework to accompany people walking side-by-side," *Autonomous Robots*, 2017.
- [15] G. Ferrer and A. Sanfeliu, "Proactive kinodynamic planning using the Extended Social Force Model and human motion prediction in urban environments," *IEEE International Conference on Intelligent Robots and Systems*, no. Iros, pp. 1730–1735, 2014.
- [16] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, May 2005.
- [17] P. Patompak, S. Jeong, N. Y. Chong, and I. Nilkhamhang, "Mobile robot navigation for human-robot social interaction," *International Conference on Control, Automation and Systems*, vol. 0, pp. 1298–1303, 1 2016.
- [18] L. Jaillet, J. Cortés, and T. Siméon, "Transition-based RRT for path planning in continuous cost spaces," *International Symposium on Combinatorial Search, SoCS 2008*, no. Section V, pp. 22–26, 2008.
- [19] H. Q. T. Ngo, V. N. Le, V. D. N. Thien, T. P. Nguyen, and H. Nguyen, "Develop the socially human-aware navigation system using dynamic window approach and optimize cost function for autonomous medical robot," *Advances in Mechanical Engineering*, vol. 12, pp. 1–17, 2020.
- [20] X. T. Truong and T. D. Ngo, "Dynamic social zone based mobile robot navigation for human comfortable safety in social environments," *International Journal of Social Robotics*, vol. 8, pp. 663–684, 2016.
- [21] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
- [22] J. Guzzi, A. Giusti, L. M. Gambardella, G. Theraulaz, and G. A. Di Caro, "Human-friendly robot navigation in dynamic environments," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2013, pp. 423–430. [Online]. Available: <http://ieeexplore.ieee.org/document/6630610/>
- [23] E. Hall, *The hidden dimension: man's use of space in public and private*. London, UK: The Bodley Head Ltd, 1966.
- [24] T. Lozano-Pérez, "Spatial Planning: A Configuration Space Approach," *IEEE Transactions on Computers*, vol. C-32, no. 2, pp. 108–120, 1983.
- [25] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [26] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.