IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 May 2023, accepted 16 July 2023, date of publication 26 July 2023, date of current version 3 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3299178

RESEARCH ARTICLE

Quantitative Metrics for Benchmarking Human-Aware Robot Navigation

JAROSŁAW KARWOWSKI AND WOJCIECH SZYNKIEWICZ
Institute of Control and Computation Engineering, Warsaw University of Technology, 00-665 Warsaw, Poland
Corresponding author: Jarosław Karwowski (jaroslaw.karwowski.dokt@pw.edu.pl)

ABSTRACT Social robots have recently gained popularity, and many human-aware navigation approaches have emerged. This work presents a comprehensive benchmark for quantitatively assessing robot navigation methods. As an automated quantitative approach, our Social Robot Planner Benchmark (*SRPB*) produces invariable indicators of the algorithm's performance that can assist the system designer in selecting the best method for the specific application. Our benchmark extends state-of-the-art task performance scores and proposes novel social metrics regarding robot motion naturalness and the perceived safety of humans surrounding the robot. Our social metrics take human tracking reliability into account. Using the *SRPB* integrated with the TIAGo robot, we assessed the robot's behaviour operating with traditional and human-aware trajectory planners in simulated and real-world environments. Our experiments tested whether state-of-the-art human-aware trajectory planners significantly improve human-awareness indicators over traditional approaches yet still maintain reasonable navigation performance. An open-source implementation of our benchmark, compatible with the Robot Operating System, is provided.

INDEX TERMS Benchmark testing, human-aware robot navigation, human-robot interaction, mobile robots, motion planning, social robotics, quantitative evaluation.

I. INTRODUCTION

Navigation is the fundamental skill of mobile robots that is widely integrated into most complex tasks. Since the 1960s, many approaches to robot navigation have been proposed [1]. The main objective of classical navigation algorithms is collision avoidance, considering all objects as generic obstacles. Social robot navigation, instead, relies on principles from social sciences. Based on research from that domain, robot systems designers try to deal with the presence of humans in the environment considering multiple objectives to react in a socially acceptable manner. Recently, due to the growing popularity of social robots, many researchers focused on creating human-aware navigation approaches [2], [3].

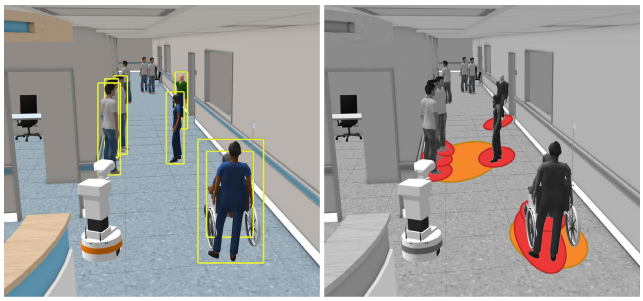
Since various navigation approaches are available, system designers must choose the best algorithm for a specific robotic application. Selecting the optimal method requires conducting comparative experiments that allow confronting investigated methods. Such experiments also benefit developers of new human-aware navigation algorithms, as they can reveal areas for potential improvement.

Robot navigation evaluation is difficult as demonstrating the overall advantage of one method over another is challenging. However, different algorithms can be compared regarding a single aspect, e.g., the undertaken path length or the time required to reach a goal. The evaluation complexity grows with the number of navigation objectives, as in human-aware navigation.

Biswas et al. [4] discussed an ideal method of evaluating social navigation. They state that qualitative methods providing a good approximation of facts are large-scale, costly, and time-consuming. We agree that automated quantitative methods are more appropriate for the iterative evaluation, e.g., during the new algorithm development stage, since they produce invariable indicators of the algorithm's performance.

Quantitative assessment methods are useful for learning-based approaches, where the reward of action must be numeric. Similarly, benchmarking against other methods may benefit planners that employ heuristics or optimise a single criterion.

The associate editor coordinating the review of this manuscript and approving it for publication was Yangming Li.



(a) Online stage (b) Offline stage

FIGURE 1. The two-stage procedure of *SRPB* benchmark for assessing the quality of the robot navigation. (a) Online stage: a navigating robot tracks obstacle locations, humans (marked as bounding boxes in the figure), F-formations and its own state, e.g., a pose and velocity. All the data is recorded and saved to a file. (b) Offline stage: after a finished experiment recordings are used to evaluate the quantitative results of the navigation using multiple metrics. In (b), personal spaces are schematically depicted with red ellipses, whereas F-formations' O-spaces with orange ones.

Designing the appropriate benchmark requires knowledge of the requirements for navigation systems from both classical and human-aware perspectives. Navigation systems exhibiting socially acceptable robot behaviours cannot remarkably sacrifice the general effectiveness of robot motions in favour of social metrics maximisation.

To address the problem of the quantitative assessment of social robot navigation, we developed *SRPB* – the benchmark that evaluates both social and task performance aspects of robot navigation (Fig. 1). Novel metrics proposed in *SRPB* evaluate robot compliance with proxemics rules regarding single humans [5] and F-formations [6], as well as other social norms, e.g., avoidance of heading in the direction of a human [2]. Another original aspect is that our metrics are designed to account for the reduced tracking quality of humans since robot perception systems are imperfect. Our benchmark can be used to test robots operating in simulated and real-world environments. Moreover, metrics were formulated so as to allow benchmark usage with different robot types (either with nonholonomic or holonomic drives). We provide an open-source implementation of our benchmark system[1] that is compatible with the Robot Operating System (ROS) [21].

The rest of the paper is organised as follows: Sec. II reviews state-of-the-art literature regarding robot navigation benchmarks, whereas Sec. III briefly shows robot navigation requirements that *SRPB* metrics are derived from. In the Sec. IV, we present a mathematical formulation of metrics used in our benchmark. The results of our simulated and real-world experiments with the TIAGo robot operating with different short-term trajectory planning algorithms (traditional and human-aware) are presented and discussed in Sec. V. Finally, we summarise our work in Sec. VI.

II. RELATED WORK

Due to a growing set of navigation algorithms available, the importance of quantitative evaluation has increased. Several authors have recently proposed benchmarking frameworks for evaluating robot motion planning algorithms [4], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18].

Heiden et al. [7] have introduced *Bench-MR* – a benchmark concerning sampling-based motion planners for nonholonomic, wheeled mobile robots. *Bench-MR* consists of two main components: motion planning algorithms and evaluation components. These latter indicate diverse navigation scenarios in static environments along with basic performance metrics assessing planning efficiency and path quality.

Another framework for comparing path planning algorithms is *PathBench* proposed by Toma et al. [8]. It provides implementations of classical and learned-based techniques allowing evaluation using typical metrics, e.g., path length, path deviation, success rate, and computational time. *PathBench* is relevant for simulated and real-world applications.

Similarly, Rocha and Vivaldini [9] have proposed *Plannie* framework for developing, testing, and benchmarking various motion planning techniques in real-world 2D and 3D environments. The authors reimplemented classical, meta-heuristics, and machine learning planning algorithms that can be scored with common metrics such as a success rate, path length, time to produce a trajectory, and time to complete the mission.

Tani et al. [10] have introduced a robotics research platform focused on providing reproducibility of experiments. Their framework integrates development and benchmarking, enabling users to create, test, and evaluate various motion planning algorithms in simulation and real robots. They mainly concentrated on autonomous vehicles operating in exemplary urban environments, validating the reproducibility of experiments across different robots using basic spatial metrics.

Mishkin et al. [11] proposed a method of evaluating classical and learning-based approaches to navigation. They tested different navigation algorithms only in simulation environments using basic metrics regarding the success rate, path length, and time required to reach the goal. Perille et al. [12] proposed *BARN* method to examine mobile robot navigation systems in standardised test environments. To evaluate the environment's difficulty, they used *Dynamic Window Approach* (*DWA*) [19] and *Elastic Bands* [20] algorithms scored with simple metrics – traversal time and navigation failures.

Wen et al. [13] proposed *MRPB* framework for evaluating the general performance of mobile robot navigation. Although their approach is suitable for simulated and real-world tests, they did not incorporate any social metrics. Similar features characterise *Arena-Bench* [14], whose authors proposed a complete suite for benchmarking different navigation algorithms but without any human awareness metrics.

MotionBenchMaker [15] is one more open-source tool for benchmarking motion planning datasets. Their approach is intended to ease the evaluation of motion planning algorithms in typical manipulation tasks performed in a simulation. The authors compared different planners using only the average planning time metric. Another mainly performance-focused

[1] https://github.com/rayvburn/srpb

benchmark was proposed by Tafnakaji et al. [16], who assessed the navigation of mobile manipulators. They evaluated, e.g., the robot's accuracy of following the global path or final pose accuracy.

On the other hand, several benchmarks focused on evaluating human awareness in robot navigation were also proposed. For example, *SocNavBench* [4] is intended to regard social aspects in robot navigation, but implements only two basic indicators – distance to the closest pedestrian and time to collision. Moreover, integrating navigation algorithms other than those provided by the authors is considered tricky; therefore, the approach is not yet for practical use.

Another approach, proposed by Tsoi et al. [17], is *SEAN 2.0* – a framework for evaluating robot navigation using different metrics concerning motion efficiency and human awareness. However, despite the variety of tools provided and integration with the most popular robotic framework, ROS [21], their approach is not applicable for evaluating real-world experiments, as the metrics calculation is integrated into the simulator.

Mavrogiannis et al. [18] have also quantified human awareness of robot navigation in the work presenting their *Social Momentum* planning framework. The authors used known metrics – the topological complexity [22] and the path irregularity index [23], to compare their *Social Momentum* with other methods.

We state that the closest to our work is *MRPB* [13]; however, we extended that method concerning metrics diversity, focusing on human awareness indices. Furthermore, our benchmark allows evaluating different methods during on-site tests (simulated or real-world); robot operation in a preprepared environment is not required as in [10] and [12]. Since robot navigation behaviours can be evaluated in target environments, our benchmark allows a more accurate algorithm selection for a specific application. Also, *SRPB* aims not to reimplement state-of-the-art navigation methods (as in [7] and [9]) but relies on ROS-integrated, easily swappable planning algorithms that are under constant development. Furthermore, such an approach does not restrict the usage of the *SRPB* with any specific class of planners. Our benchmark allows comparing path planners [7], [8] and trajectory planners in separation or as combined motion planning methods [9].

III. ROBOT NAVIGATION REQUIREMENTS

Benchmarking navigation algorithms can be named as checking compliance with requirements. Classical navigation mainly accounts for collision-free motions and reaching a goal as fast as possible, whereas social navigation is more complex. We studied relevant review works (referred to below) and established a classification of important requirements in social navigation. Requirements were organised into groups composed as follows:

1) robot task performance maximisation
   - 1.1 avoid collisions with the environment
   - 1.2 plan trajectories that are feasible for the mobile base
   - 1.3 reach the goal as fast as possible
   - 1.4 reach the goal taking the shortest possible path
   - 1.5 avoid rotations along the path [3], [23], [24]
2) human discomfort minimisation
   - 2.1 naturalness of the robot motion [2], [3]
     - 2.1.1 smoothen the velocity profile [2], [3], [25]
     - 2.1.2 avoid backward motions
     - 2.1.3 avoid in-place rotations
     - 2.1.4 avoid oscillating motions
   - 2.2 perceived safety of humans [2], [3]
     - 2.2.1 avoid intrusion into human's personal space [2], [26], [27]
     - 2.2.2 avoid intrusion into F-formation's O-space space [2], [6], [28]
     - 2.2.3 avoid leading straight into a moving human [3]

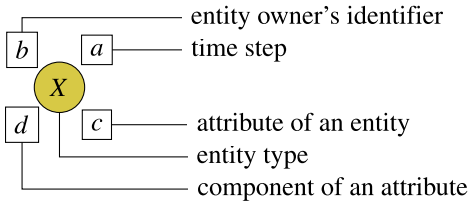


FIGURE 2. A general description of symbol composition method used in the notation.

We extended classical robot motion naturalness requirements with a few indicators that, we argue, are important. Firstly, avoidance of backward motions (Req. 2.1.2) as people in public spaces rarely move backwards. Such movements can also be dangerous collision-wise for mobile bases not equipped with a rear range sensor. Secondly, in-place rotations (Req. 2.1.3) reduce the motion naturalness since they remind of typical robot behaviour from 80s science fiction movies, where machines lacked motion coordination in multiple dimensions. Thirdly, oscillating motions, i.e., slightly moving back and forth (Req. 2.1.4), produce a feeling of robot unreliability.

IV. SOCIAL ROBOT PLANNER BENCHMARK

This section presents metrics calculation methods focusing on social navigation metrics derived from the requirements (Sec. III). Nevertheless, general navigation performance aspects are also discussed briefly.

To describe the metrics for the robot navigation evaluation, we developed a mathematical notation used in equations (Fig. 2). A value of any entity at time t^n is referred to as $(\cdot)^n$. Common symbols are presented in Tab. 1.

The ontology that we propose for social robot navigation is organised as follows: the world configuration at each time t^n consists of the state of a single robot, r^n, and the state of its environment. The latter, recalculated at each time step, aggregates: a set of obstacles, $\mathbb{O}^n$, and a set of humans, $\mathbb{H}^n$, that may be arranged into F-formations, $\mathbb{G}^n$. Therefore, at time t^n, the association of h-th human into g-th F-formation can be expressed as ${}^hH^n \in {}^gG^n$, whereas ${}^gG^n \in \mathbb{G}^n$. The ${}^hH^n$,

TABLE 1. A dictionary of common symbols used for formulating quantitative metrics.

Symbol	Description
r	identifier of a robot
t^n	n-th time stamp
${}^{i}H$	human identified as i
$\mathbb{H}$	set of humans
${}^{i}G$	humans F-formation, i.e., a group, identified as i
$\mathbb{G}$	set of humans F-formations
${}^{i,j}d$	Euclidean distance between i and j
$\mathbf{p}$	pose vector in a form $[x, y, \theta]^T$
$\mathbf{v}$	velocity vector in a form $[v_x, v_y, \omega]^T$
$\mathbf{a}$	acceleration vector in a form $[a_x, a_y, \alpha]^T$
${}^{i,j}\phi$	direction of a vector connecting centers of i and j
${}^{i,j}\delta$	relative location of i regarding the heading of j
$\Re(\mathrm{Z}, \theta)$	rotation matrix around Z axis by the θ angle
var	variance
$\mathbf{\Sigma}$	covariance matrix
$\mathcal{N}$	normal distribution
f	function
m	metric

being prone to collisions with the robot, can also be involved in calculations related to generic obstacles, $\mathbb{O}^n$.

Experiment time stamps, t^n, where $n = \{1, \ldots, N\}$, are shared among the robot, humans and human groups. We commonly refer to the summation of time differences between subsequent time steps to consider that they may not be equal-sampled in non-real-time systems, affecting average values. Conditional summation is represented with the Iverson bracket operator [29].

A. METRICS FOR EVALUATION OF ROBOT NAVIGATION PERFORMANCE

Socially acceptable robot behaviours should not significantly degrade the general performance of the navigation task (Req. 1). The problem of robot performance during navigation was already discussed in multiple works, as shown in Sec. II. However, we propose several metrics that, we argue, are also crucial for a robot behaviour assessment. In this section, metrics from literature are tied to requirements (Sec. III) without presenting relevant equations, but their symbols in our notation are pointed out.

1) OBSTACLE SAFETY

Robot navigation benchmarks usually report the number of collisions along the path to the goal [14], [17]. However, we argue that for robust navigation approaches, assessing the percentage of time the robot has spent in the dangerous area around obstacles (nearer than the configurable distance of ${}^{r,\mathbb{O}}d_{\min}$) is more appropriate. The relevant metric was presented in [13], [14], which we refer to as m_{obs}. We argue that this metric is sufficient to assess compliance with Req. 1.1.

2) MOTION EFFICIENCY

A metric expressing the time required to reach the goal pose (ours m_{mef}) was proposed in [4], [11], [13], [14], [17], [16] and is appropriate for verification of the goal-reaching requirement (Req. 1.3).

3) PATH LENGTH

Classical navigation is often focused on minimising of robot's path length while traversing to the goal (Req. 1.4). The path is determined by a sequence of poses. To evaluate the path length, m_{plin}, the sum of Euclidean displacements of the mobile base during the scenario is computed [4], [14], [16], [17].

4) CUMULATIVE HEADING CHANGE

A metric complementary to the m_{plin} represents robot orientation change along the path (Req. 1.5). For example, the path irregularity metric was discussed in [23], providing a normalised score of unnecessary turning per unit path length. However, since it requires knowing the perfect path to the goal, we argue that it applies only to very small or perfectly known environments. Therefore, in our benchmark, we use the cumulative heading change metric, m_{chc} (1), as in [24], [30], and [31].

$$m_{\text{chc}} = \sum_{n=1}^{N-1} |{}^{r}\theta^{n+1} - {}^{r}\theta^{n}| \tag{1}$$

5) COMPUTATIONAL EFFICIENCY

Trajectory planners for mobile base navigation have different degrees of complexity. Therefore, it is adequate to verify the average computation time the planner takes to accomplish a new velocity command (Req. 1.2). Such a metric was proposed in [13], which we refer to as m_{cef}.

6) COMPUTATIONAL TIME REPEATABILITY

Evaluating how much computation times differ from the mean value, $\bar{c}$, is also important. It shows how likely the planner will violate requested computation times and, thus, whether it can be safely applied in robots operating in highly dynamic environments. Therefore, we proposed the m_{cre} metric, constituting a standard deviation of all computational times (n-th denoted as c^n) during the scenario (2).

$$m_{\text{cre}} = \sqrt{\frac{1}{n}\sum_{n=1}^{N}\left(c^n - \bar{c}\right)^2} \tag{2}$$

B. METRICS FOR EVALUATION OF ROBOT MOTION NATURALNESS

Social metrics are essential for robots operating in dynamic and populated environments. This section discusses metrics related to robot motion naturalness (Req. 2.1).

1) VELOCITY SMOOTHNESS

The velocity smoothness metric, m_{vsm}, defines how much robot's linear velocities, ${}^{r}v_x^n$ and ${}^{r}v_y^n$, differed in subsequent time steps, which indicates a presence of erratic motions (Req. 2.1.1). A similar metric was proposed in [13]; however, their formulation lacks the holonomic drives support. Instead, in our approach, both linear velocity components (along the x and y axes) are taken into consideration (3). Investigated robot velocities are expressed in the mobile base's coordinate system.

$$m_{\text{vsm}} = \frac{1}{N-1}\sum_{n=1}^{N-1}\frac{\sqrt{\sum_{j\in\{x,y\}}\left({}^{r}v_j^{n+1} - {}^{r}v_j^{n}\right)^2}}{t^{n+1}-t^{n}} \tag{3}$$

2) HEADING CHANGE SMOOTHNESS

Another indicator of erratic motions (Req. 1.5), defines an average rate of robot heading changes [30] during the scenario. The m_{hsm} metric is computed by comparing differences of robot angular velocity, ${}^{r}\omega$, in subsequent steps, as in (4).

$$m_{\text{hsm}} = \frac{1}{N-1}\sum_{n=1}^{N-1}\frac{\left|{}^{r}\omega^{n+1} - {}^{r}\omega^{n}\right|}{t^{n+1}-t^{n}} \tag{4}$$

Similar metrics regarding robot motion naturalness (m_{vsm} and m_{hsm}) were also discussed in [4], [14], and [17], yet the authors did not show their calculation methods.

3) OSCILLATIONS

The oscillations metric, m_{osc}, defines the percentage of the total time that the robot has spent oscillating, i.e., has not developed significant linear and angular velocities (Req. 2.1.4). The oscillating behaviour in a given time step occurs when robot velocities, ${}^{r}\mathbf{v}^n$, and oscillation threshold velocities, ${}^{r}\mathbf{v}_{\text{osc}}$, fulfil conditions shown in (5). The linear speed of the robot at time t^n is represented as ${}^{r}v_{\text{lin}}^n = \sqrt{\left({}^{r}v_x^n\right)^2 + \left({}^{r}v_y^n\right)^2}$.

$$m_{\text{osc}} = \frac{100\%}{t^N - t^1}\sum_{n=1}^{N-1}\left(t^{n+1}-t^{n}\right)\begin{bmatrix} {}^{r}v_{\text{lin}}^n < {}^{r}_{\text{lin}}v_{\text{osc}} \\ \wedge \left|{}^{r}v_x^n\right| < {}^{r}_{x}v_{\text{osc}} \\ \wedge \left|{}^{r}v_y^n\right| < {}^{r}_{y}v_{\text{osc}} \\ \wedge \left|{}^{r}\omega^n\right| < {}^{r}\omega_{\text{osc}} \end{bmatrix} \tag{5}$$

4) BACKWARD MOVEMENTS

The backward movements metric, m_{bwd}, defines the percentage of the total execution time that the robot has been advancing in the backward direction (Req. 2.1.2) with a speed of at least ${}^{r}_{x}v_{\text{osc}}$ (6).

$$m_{\text{bwd}} = \frac{100\%}{t^N - t^1}\sum_{n=1}^{N-1}\left(t^{n+1}-t^{n}\right)\left[{}^{r}v_x^n \le -{}^{r}_{x}v_{\text{osc}}\right] \tag{6}$$

5) IN-PLACE ROTATIONS

The in-place rotations metric, m_{iprot}, defines the percentage of the total time that the robot has spent rotating in place (Req. 2.1.3). In-place rotation is an action of the robot when its linear velocities are kept at 0, but the angular velocity is maintained above the threshold value of ${}^{r}\omega_{\text{osc}}$ (7).

$$m_{\text{iprot}} = \frac{100\%}{t^N - t^1}\sum_{n=1}^{N-1}\left(t^{n+1}-t^{n}\right)\begin{bmatrix} {}^{r}v_x^n = 0 \\ \wedge\, {}^{r}v_y^n = 0 \\ \wedge \left|{}^{r}\omega^n\right| \ge {}^{r}\omega_{\text{osc}} \end{bmatrix} \tag{7}$$

It is crucial that m_{osc}, m_{bwd} and m_{iprot} metrics are orthogonal to each other, i.e., in each time step robot's action can be qualified as fulfilling conditions of only one of these metrics.

C. METRICS FOR PERCEIVED SAFETY EVALUATION AMONG HUMANS

In this section, our metrics for the evaluation of the robot's intrusiveness and disturbance to adjacent people are discussed.

1) PERSONAL SPACES INTRUSION

The personal space concept was adopted in social robotics from the proxemics theory [5]. Our personal space intrusion metric, m_{psi}, defines the scale of robot intrusions into any human's personal space [2] throughout the scenario execution (Req. 2.2.1).

Recent studies show that Gaussian functions are legitimate for modelling personal spaces [27], [32]. Therefore, we represent the human's personal space as a multivariate asymmetric Gaussian function, f_{mag}, centred at the h-th human's position, ${}^{h}x^n$ and ${}^{h}y^n$, oriented according to the human's heading ${}^{h}\theta^n$. Variances along the front (${}^{h}\text{var}_{\text{fr}}^n$), side (${}^{h}\text{var}_{\text{sd}}^n$), and rear (${}^{h}\text{var}_{\text{rr}}^n$) directions of the human pose were estimated in [32].

The variance along the heading axis, ${}^{r,h}\text{var}_{\text{hd}}^n$, is selected (${}^{h}\text{var}_{\text{fr}}^n$ or ${}^{h}\text{var}_{\text{rr}}^n$) in a three-step procedure, so the symmetrical variant of the multivariate Gaussian, f_{mg}, can be used in calculations. Firstly, to evolve, where the robot is located compared to the human's heading direction, the angle of the vector connecting the centres of the human and the robot, ${}^{r,h}\phi^n$, is computed (8). Then, the relative location ${}^{r,h}\delta^n$ of the robot r, compared to the human's h heading direction, is calculated as in (9) and presented in Fig. 3a. Finally, using the ${}^{r,h}\delta^n$ indicator, the variance is selected as in (10).

$${}^{r,h}\phi^n = \text{atan2}\left({}^{r}y^n - {}^{h}y^n, {}^{r}x^n - {}^{h}x^n\right) \tag{8}$$

$${}^{r,h}\delta^n = {}^{r,h}\phi^n - {}^{h}\theta^n \tag{9}$$

$${}^{r,h}\text{var}_{\text{hd}}^n = \begin{cases} {}^{h}\text{var}_{\text{fr}}^n, & \text{if } \left|{}^{r,h}\delta^n\right| \le \frac{\pi}{2} \\ {}^{h}\text{var}_{\text{rr}}^n, & \text{otherwise} \end{cases} \tag{10}$$

To compute a value of f_{mg}, the h-th human's personal space covariance matrix needs to be created. Variances defining the personal space are expressed in the human's coordinate system; therefore, the personal space covariance matrix, ${}^{r,h}\mathbf{\Sigma}_{\text{psi}}^n$, must be rotated according to the h-th human's orientation, ${}^{h}\theta^n$, as in (11).

$${}^{r,h}\mathbf{\Sigma}_{\text{psi}}^n = \Re\left(\text{Z}, {}^{h}\theta^n\right)\begin{bmatrix} {}^{r,h}\text{var}_{\text{hd}}^n & 0 \\ 0 & {}^{h}\text{var}_{\text{sd}}^n \end{bmatrix}\Re^T\left(\text{Z}, {}^{h}\theta^n\right) \tag{11}$$

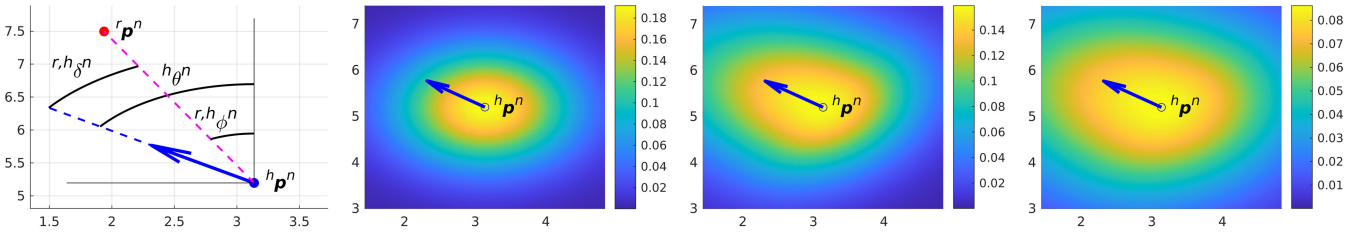


FIGURE 3. Processing of the *h*-th human data. Angles of an example arrangement along with lines reflecting the orientation of the global coordinate system (the x-axis pointing upwards) are presented in (a) . Gaussians of: position uncertainty (b), personal space (c), and resultant distribution (d) are shown with the mean of the estimated *h*-th human pose. The personal space model was created using ${}^{h}\text{var}^{n}_{\text{fr}} = 3.0$, ${}^{h}\text{var}^{n}_{\text{rr}} = 0.75$, ${}^{h}\text{var}^{n}_{\text{sd}} = 1.33$.

In the evaluation process, we also account for human tracking reliability. It aims to prevent excessive penalisation of robot states when, e.g., a tracked human becomes occluded. The covariance matrix of the estimated human position, ${}^{h}\mathbf{\Sigma}^{n}_{p}$, is obtained from the robot perception system. The sum of independent normal random variables is applied to compute the resultant covariance matrix, ${}^{r,h}_{\Delta}\mathbf{\Sigma}^{n}_{\text{psi}}$ (12). It accounts for position estimation uncertainty and the personal space model (Fig. 3).

$$ {}^{r,h}_{\Delta}\mathbf{\Sigma}^{n}_{\text{psi}} = {}^{h}\mathbf{\Sigma}^{n}_{p} + {}^{r,h}\mathbf{\Sigma}^{n}_{\text{psi}} \tag{12} $$

The scale of r robot intrusion into the personal space of h-th human in time t^n is referred to as ${}^{r,h}\text{psi}^n$ (13). It represents a value of the f_{mg} function (modelling the h-th human's personal space) at the robot's pose at that time, ${}^{r}\mathbf{p}^n$. Equation (13) presents arguments that the f_{mg} function takes – a pose and a multivariate normal distribution, the value of which will be computed at the given pose. The multivariate normal distribution, described by a mean of, e.g., $\mathbf{p}$, and covariance matrix of $\mathbf{\Sigma}$, is denoted as $\mathcal{N}\left(\mathbf{p}, \mathbf{\Sigma}\right)$.

The final formulation of the personal space intrusions metric, m_{psi}, is shown in (14). Our method investigates the maximum intrusion in a given time step t^n, provided that some human was detected. The scale of the robot intrusion is normalised to the Gaussian value at the h-th human's centre, ${}^{h}\text{psi}^n$, so the metric value in each time step corresponds to a percentage of the maximum intrusion. If no human was observed during the scenario, $m_{\text{psi}} = 0$.

$$ {}^{r,h}\text{psi}^n = f_{\text{mg}}\left({}^{r}\mathbf{p}^n, \mathcal{N}\left({}^{h}\mathbf{p}^n, {}^{r,h}_{\Delta}\mathbf{\Sigma}^{n}_{\text{psi}}\right)\right) \tag{13} $$

$$ m_{\text{psi}} = \frac{\sum_{n=1}^{N-1}\left(\left(t^{n+1} - t^n\right) \max_{{}^{h}H^n \in \mathbb{H}^n} {}^{r,h}\text{psi}^n\right)}{\sum_{n=1}^{N-1}\left(t^{n+1} - t^n\right)\left[\mathbb{H}^n = \varnothing\right]} \tag{14} $$

2) F-FORMATIONS' O-SPACES INTRUSION

The O-spaces of F-formations were proposed in [6] to reflect the elliptical spaces created by a group of humans involved in a focused interaction [28]. Our m_{fsi} metric aims to penalise a robot for traversing through O-spaces (Req. 2.2.2).

Firstly, to find the pose of the g-th O-space's ellipse, ${}^{g}\mathbf{p}^n$, we employ Taubin's algebraic method of ellipse fitting [33], supplied with mean positions of g-th F-formation members. Then, to assess the cost of robot movement in terms of human groups' presence, we model O-spaces as bivariate Gaussians (Fig. 4). The span of the 2-dimensional O-space's Gaussian model is derived from the lengths of semi-axes (${}^{g}d^n_x$ and ${}^{g}d^n_y$) of the F-formation's ellipse (Fig. 4a). Using the 2σ rule, the variances along the direction of the semi-major and semi-minor axes are derived, ${}^{g}_{x}\text{var}^n_d$ and ${}^{g}_{y}\text{var}^n_d$, accordingly (15). The O-space model's covariance matrix, ${}^{g}\mathbf{\Sigma}^{n}_{\text{fsi}}$, expressed in the global coordinate system, is computed by applying a rotation (by the angle of the F-formation's ellipse orientation, ${}^{g}\theta^n$) to a matrix composed of variances as in (16).

$$ \forall j \in \{x, y\}, {}^{g}_{j}\text{var}^n_d = \left(\frac{{}^{g}d^n_j}{2}\right)^2 \tag{15} $$

$$ {}^{g}\mathbf{\Sigma}^{n}_{\text{fsi}} = \mathfrak{R}\left(\text{Z}, {}^{g}\theta^n\right)\begin{bmatrix} {}^{g}_{x}\text{var}^n_d & 0 \\ 0 & {}^{g}_{y}\text{var}^n_d \end{bmatrix}\mathfrak{R}^T\left(\text{Z}, {}^{g}\theta^n\right) \tag{16} $$

In the spatial model of an F-formation, we also incorporate the uncertainty of the g-th F-formation's position estimation (Fig. 4b), arising from position uncertainties of members, ${}^{g}\mathbb{H}^n$. The uncertainty is represented by the variances: ${}^{g}_{x}\text{var}^n_p$, ${}^{g}_{y}\text{var}^n_p$, ${}^{g}_{\text{xy}}\text{var}^n_p$, and ${}^{g}_{\text{yx}}\text{var}^n_p$, computed as in (17) and (18). The composition of the F-formation's position covariance matrix, ${}^{g}\mathbf{\Sigma}^{n}_{p}$, is shown in (19).

$$ \forall j \in \{x, y\}, {}^{g}_{j}\text{var}^n_p = \max_{{}^{h}H^n \in {}^{g}G^n} {}^{h}_{j}\mathbf{\Sigma}^{n}_{p} \tag{17} $$

$$ {}^{g}_{\text{xy}}\text{var}^n_p = {}^{g}_{\text{yx}}\text{var}^n_p = \max_{{}^{h}H^n \in {}^{g}G^n}\left(\max_{j \in \{\text{xy,yx}\}} {}^{h}_{j}\mathbf{\Sigma}^{n}_{p}\right) \tag{18} $$

$$ {}^{g}\mathbf{\Sigma}^{n}_{p} = \begin{bmatrix} {}^{g}_{x}\text{var}^n_p & {}^{g}_{\text{xy}}\text{var}^n_p \\ {}^{g}_{\text{yx}}\text{var}^n_p & {}^{g}_{y}\text{var}^n_p \end{bmatrix} \tag{19} $$

The covariance matrix that accounts for F-formation's O-space and members' position estimation uncertainties, ${}^{g}_{\Delta}\mathbf{\Sigma}^{n}_{\text{fsi}}$, is formulated as a sum of normally distributed random variables (20). The computation method of the intrusion, ${}^{r,g}\text{fsi}^n$, of r robot into the O-space of g-th F-formation in time t^n, along with arguments that the Gaussian function

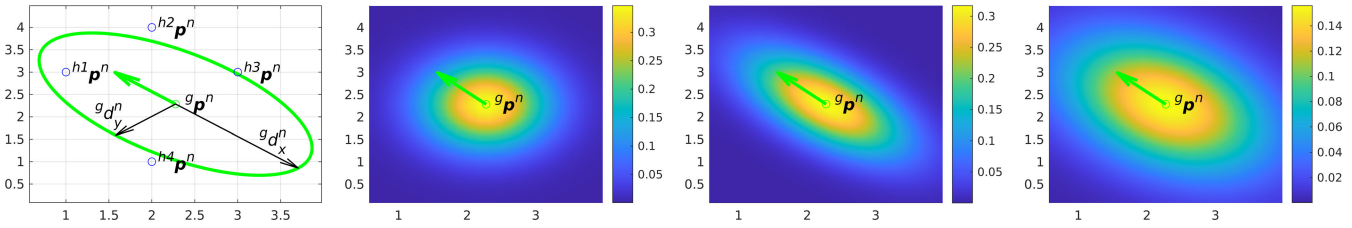


FIGURE 4. Processing of an exemplary F-formation consisting of 4 members. The mean of an estimated pose obtained from ellipse fitting is shown in (a) . The remaining figures present corresponding Gaussians of the: position uncertainty (b), O-space model (c), and resultant distribution (d).

takes, is presented in (21). The final formulation of the O-spaces intrusions metric, m_{fsi}, is shown in (22). The scale of the robot intrusion is normalised to the value of Gaussian at the g-th group's centre, ${}^{g}\text{fsi}^{n}$, so the metric value in each time step corresponds to a percentage of the maximum intrusion. If no F-formation was observed during the scenario, $m_{\text{fsi}} = 0$.

$$ {}^{g}_{\Delta}\boldsymbol{\Sigma}^{n}_{\text{fsi}} = {}^{g}\boldsymbol{\Sigma}^{n}_{p} + {}^{g}\boldsymbol{\Sigma}^{n}_{\text{fsi}} \tag{20} $$

$$ {}^{r,g}\text{fsi}^{n} = f_{\text{mg}}\left({}^{r}\mathbf{p}^{n}, \mathcal{N}\left({}^{g}\mathbf{p}^{n}, {}^{g}_{\Delta}\boldsymbol{\Sigma}^{n}_{\text{fsi}}\right)\right) \tag{21} $$

$$ m_{\text{fsi}} = \frac{\sum_{n=1}^{N-1}\left(\left(t^{n+1} - t^{n}\right) \max_{{}^{g}G^{n} \in \mathbb{G}^{n}} {}^{r,g}\text{fsi}^{n}\right)}{\sum_{n=1}^{N-1}\left(t^{n+1} - t^{n}\right)\left[\mathbb{G}^{n} = \varnothing\right]} \tag{22} $$

3) HEADING STRAIGHT INTO A HUMAN

Reactive approaches to robot navigation usually suffer from late trajectory adjustment in dynamic environments causing the robot to turn just before the imminent collision with, e.g., a human, diminishing their perceived safety (Req. 2.2.3). The problem was initially investigated by Truong and Ngo [26], who tried to assess the robot's approach direction to the humans. However, their approach does not account for human position estimation uncertainty and a robot's dynamics.

Thus, we propose a new metric, m_{dir}, to evaluate the scale of the problem in different algorithms. The metric penalises a robot for undertaking motion directions leading straight into humans, especially when the robot moves with a decent speed near a human.

To compute the metric, we investigate a geometrical arrangement of the human h and the robot r. Namely, we compare the robot's current heading to directions leading into the centre of the human. The span of cross-human robot heading angles arises from the space physically occupied by the human (modelled by the circle with a configurable radius of d_{ocp}) and the human position estimation uncertainty (represented by a covariance matrix, ${}^{h}\boldsymbol{\Sigma}^{n}_{p}$). Both effects are visualised in Fig. 5.

The variance of the bivariate Gaussian representing the circular occupancy model, var_{ocp}, is computed by applying the 2σ rule to the d_{ocp} (23). The resultant covariance matrix, ${}^{h}_{\Delta}\boldsymbol{\Sigma}^{n}_{\text{dir}}$, aggregating the occupancy model and the position uncertainty, is defined as in (24).

$$ \text{var}_{\text{ocp}} = \left(\frac{d_{\text{ocp}}}{2}\right)^{2} \tag{23} $$

$$ {}^{h}_{\Delta}\boldsymbol{\Sigma}^{n}_{\text{dir}} = {}^{h}\boldsymbol{\Sigma}^{n}_{p} + \begin{bmatrix} \text{var}_{\text{ocp}} & 0 \\ 0 & \text{var}_{\text{ocp}} \end{bmatrix} \tag{24} $$

The value of the Gaussian at the ${}^{r,h}\mathbf{p}^{n}_{\text{isc}}$ point, ${}^{r,h}\text{dir}^{n}_{\text{cc}}$, represents how much the robot's direction leads into the centre of the human (26). The ${}^{r,h}\mathbf{p}^{n}_{\text{isc}}$ is an intersection point of the robot's direction axis (ray), $\overrightarrow{{}^{r}\mathbf{p}^{n}}$, and the line, ${}^{r,h}l^{n}_{\text{cc}}$, defined by the crossed point and the direction angle in (25). The geometrical representation of finding the ${}^{r,h}\mathbf{p}^{n}_{\text{isc}}$ is depicted in Fig. 5.

$$ {}^{r,h}l^{n}_{\text{cc}} = \overleftrightarrow{{}^{h}\mathbf{p}^{n}, \measuredangle\left({}^{r,h}\phi^{n} + \frac{\pi}{2}\right)} \tag{25} $$

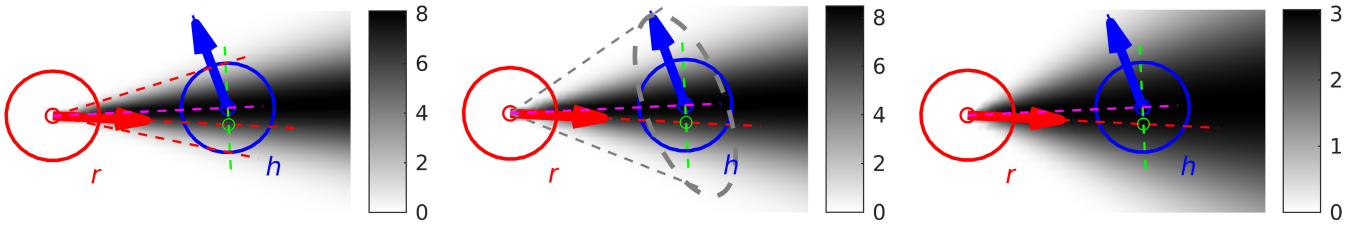
$$ {}^{r,h}\text{dir}^{n}_{\text{cc}} = f_{\text{mg}}\left({}^{r,h}\mathbf{p}^{n}_{\text{isc}}, \mathcal{N}\left({}^{h}\mathbf{p}^{n}, {}^{h}_{\Delta}\boldsymbol{\Sigma}^{n}_{\text{dir}}\right)\right) \tag{26} $$

We also investigate how much the human can notice the robot's movement (potentially disturbing), the scale of which is represented by ${}^{r,h}\text{fov}^{n}$. Applying the 2σ rule to the configurable field of view angle, φ_{fov}, the variance, var_{fov}, is computed. The relative location indicator, ${}^{r,h}\delta^{n}$ (9), determines directly how far the robot is situated from the centre axis of human's sight. Then, the value of the Gaussian appointed in the normalised angle domain, f_{ang}, is computed for the current arrangement of the human and the robot (27).

$$ {}^{r,h}\text{fov}^{n} = f_{\text{ang}}\left({}^{r,h}\delta^{n}, \mathcal{N}\left(0, \text{var}_{\text{fov}}\right)\right) \tag{27} $$

The m_{dir} metric also accounts for the speed of the robot, ${}^{r}v^{n}_{\text{lin}}$, and the distance between the human and the robot, ${}^{r,h}d^{n}$. The final formulation of the robot heading direction penalty, ${}^{r,h}\text{dir}^{n}$, defined for a single human-robot pair, is presented in (28).

The normalisation of the metric value relies on comparing the current arrangement to the worst possible case. To accomplish that, platform-specific features must be determined, namely the circumradius of the mobile base, d_{cr}, and the maximum linear speed of the robot, ${}^{r}_{\max}v_{\text{lin}}$. Moreover, it is assumed that the robot's heading points straight into the human position (${}^{r,h}\mathbf{p}^{n}_{\text{isc}} = {}^{h}\mathbf{p}^{n}$, computed in ${}^{r,h}_{max}\text{dir}^{n}_{\text{cc}}$) and the robot is located along the human's sight axis (${}^{r,h}\delta^{n} = 0$, calculated in ${}^{r,h}\text{fov}^{n}_{max}$). The formula for the normalisation



(a) Human occupancy model (b) Human position uncertainty (c) Resultant distribution

FIGURE 5. An exemplary human-robot arrangement and corresponding: distribution of the human physical space occupancy model (a) , position estimation uncertainty (b), and resultant distribution (c). The robot's and human's heading directions are represented by arrows – red and blue, accordingly. Green dashed lines, constituting ${}^{r,h}I^{n}_{cc}$, are defined to find the intersection point, ${}^{r,h}\mathrm{p}^{n}_{isc}$, represented by green circles. Magenta dashed lines indicate the robot's direction with the maximum likelihood of heading straight into the human. Blue circles with a radius of d_{ocp} represent the human occupancy model, whereas in (b), the grey ellipse represents human position estimation uncertainty (cut-off determined by the 2σ rule).

factor, ${}^{r,h}\mathrm{dir}^{n}_{nrm}$, is shown in (29). The metric for the whole scenario, m_{dir}, is calculated as in (30) and corresponds to the average percentage of heading disturbance generated by the robot. If no human was observed during the scenario, $m_{dir} = 0$.

$$ {}^{r,h}\mathrm{dir}^{n} = \frac{{}^{r,h}\mathrm{dir}^{n}_{cc} \cdot {}^{r,h}\mathrm{fov}^{n} \cdot {}^{r}v^{n}_{lin}}{{}^{r,h}d^{n}} \tag{28} $$

$$ {}^{r,h}\mathrm{dir}^{n}_{nrm} = \frac{{}^{r,h}_{max}\mathrm{dir}^{n}_{cc} \cdot {}^{r,h}\mathrm{fov}^{n}_{max} \cdot {}^{r}_{max}v_{lin}}{d_{cr} + d_{ocp}} \tag{29} $$

$$ m_{dir} = \frac{\sum_{n=1}^{N-1}\left(\left(t^{n+1} - t^{n}\right) \max_{{}^{h}H^{n} \in \mathbb{H}^{n}} \frac{{}^{r,h}\mathrm{dir}^{n}}{{}^{r,h}\mathrm{dir}^{n}_{nrm}}\right)}{\sum_{n=1}^{N-1}\left(t^{n+1} - t^{n}\right)\left[\mathbb{H}^{n} = \varnothing\right]} \tag{30} $$

V. EXPERIMENTS

Our experiments intend to determine whether tested human-aware trajectory planners perform superior to traditional ones regarding human presence in the environment. To accomplish that, we evaluate the perceived safety of humans and robot motion naturalness. Simultaneously, we examine if evaluated social planners perform significantly worse than traditional planners regarding overall robot performance.

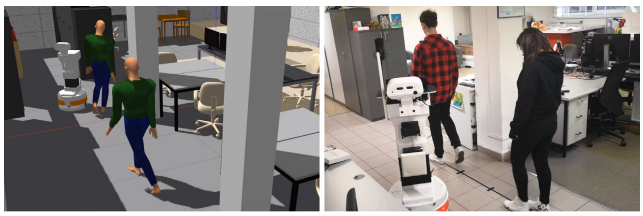
A. EXPERIMENTS DESIGN

Scenarios were designed so that the robot operating with each examined trajectory planner could reach a goal pose, navigating collision-free from the shared start pose. Thus, we started with evaluating the capabilities of the planners while the robot operated in partially unknown static and dynamic environments. Our test environment was a robotics laboratory at Warsaw University of Technology (Fig. 6, 7). We decided to compare real results with ones obtained from the simulation; therefore, we also performed trials in a virtual equivalent of the environment.

We conducted experiments in which humans participate as static or dynamic elements of the robot environment; therefore, tests are identified as *static* in a simulation (Fig. 6a), *static* in the real world (Fig. 6b), *dynamic* in a simulation (Fig. 7a), and *dynamic* in the real world (Fig. 7b). These scenarios are later referred to as 1a, 1b, 2a, and 2b.

(a) Simulation (b) Real world

FIGURE 6. An overview of the *static* scenario.



(a) Simulation (b) Real world

FIGURE 7. An overview of the *dynamic* scenario.

In the *static* scenario (Fig. 6), an F-formation of two humans stays near the robot's goal. Reaching the goal by the robot requires passing the humans, so when approaching the final pose, the robot is foreseen not to distract the humans involved in a focused interaction and take an outside path. Instead, in the *dynamic* scenario (Fig. 7), the robot moving to the goal pose along a narrow corridor encounters a moving human followed by another moving human, both going opposite to the robot. The robot is expected to avoid collisions and maximise the perceived safety of humans.[2]

Each trajectory planner was tested in each scenario's simulated and real-world variants. At least five representative trials

[2]A video presenting test scenarios https://vimeo.com/805337193

TABLE 2. Trajectory planners' parameters that were constant throughout the experiments.

Parameter	${}^{r}_{\min}v_{\text{lin}}$	${}^{r}_{\max}v_{\text{lin}}$	${}^{r}_{\min}\omega$	${}^{r}_{\max}\omega$
Value	-0.1 $\frac{\text{m}}{\text{s}}$	0.5 $\frac{\text{m}}{\text{s}}$	-1.05 $\frac{\text{rad}}{\text{s}}$	1.05 $\frac{\text{rad}}{\text{s}}$
Parameter	${}^{r}_{\min}a_{\text{lin}}$	${}^{r}_{\max}a_{\text{lin}}$	${}^{r}_{\min}\alpha$	${}^{r}_{\max}\alpha$
Value	-1.0 $\frac{\text{m}}{\text{s}^2}$	1.0 $\frac{\text{m}}{\text{s}^2}$	-1.05 $\frac{\text{rad}}{\text{s}^2}$	1.05 $\frac{\text{rad}}{\text{s}^2}$

were benchmarked for each case, and then, the median of each metric was computed to score a trajectory planner (Tab. 4).

For our comparative experiments, we had to replicate the environment configuration in the following trials to test different trajectory planners under the same conditions. The challenging task in the real-world *dynamic* scenario (2b) is to make participating humans move similarly in each trial. To maximise path similarity, the paths of dynamic actors were indicated with a tape glued to the floor. Further, for trajectory similarity, paths were equipped with subsequent pose markers (Fig. 7b) and entrants were asked to finish each step with a tick of a metronome that was programmed to 60 beats per minute.

B. SETUP

Real-world experiments were conducted with PAL's TIAGo Iron robot, and simulation results were obtained with the robot's digital twin provided by PAL. The main sensors of the robot are: a Sick TIM571 LiDAR (0.05 – 25 m scan range, 180° field of view, 0.33° step angle) and an Orbbec Astra RGBD camera (depth stream with a resolution of 640 x 480 pixels and a 0.6 – 8 m depth sensor range).

Since our robot has factory-installed ROS Melodic, we performed simulated trials with the same framework version. We have chosen Gazebo (version 9) as the simulation platform due to its integration with ROS. Simulation experiments were performed on a laptop with an Intel Core i7-4720HQ CPU and 16 GB RAM.

Our test setup uses the ROS navigation system, whose structure consists of a global and a local [2], [34] planners. As the robot's global path planner in all experiments, we have been using wavefront Dijkstra's algorithm.[3] Only trajectory planners were swapped, utilising public ROS implementations of the benchmarked algorithms. Parameters related to kinematic and dynamic constraints of the mobile base, shown in Tab. 2, were common for all planners. In all scenarios, the robot operated with the same preprepared map. Nevertheless, environment obstacles were detected in real-time by the robot sensors and added to the costmaps (of global and local planners), making the robot resistant to the changes not captured in the map. For the global pose estimation, the *AMCL*[4] [35] algorithm was used.

[3] http://wiki.ros.org/navfn
[4] http://wiki.ros.org/amcl

TABLE 3. Configurable parameters of metrics that were used in the experiments.

Parameter	${}^{r,\mathbb{O}}d_{\min}$	${}^{r}_{x}v_{\text{osc}}$	${}^{r}_{y}v_{\text{osc}}$	${}^{r}_{\text{lin}}v_{\text{osc}}$
Value	0.55 m	0.025 $\frac{\text{m}}{\text{s}}$	0.025 $\frac{\text{m}}{\text{s}}$	0.025 $\frac{\text{m}}{\text{s}}$
Parameter	${}^{r}\omega_{\text{osc}}$	φ_{fov}	d_{ocp}	d_{cr}
Value	0.05 $\frac{\text{rad}}{\text{s}}$	3.3 rad	0.28 m	0.275 m

In all experiments, we used SPENCER human perception stack,[5] mainly relying on vision data, as LiDAR-based detections were often false positives. Tracked humans were embedded as bivariate Gaussians into the costmaps representing the robot's environment model used for planning.[6] During the tests, no other types of dynamic objects (obstacles) were present in the environment besides humans.

Moreover, we used the *HuBeRo* framework[7] [36] for controlling humans in the simulation. In all virtual experiments, human agents used the same path planner as the robot and the *Timed Elastic Band* (*TEB*) [37] trajectory planner.

In our experiments, social metrics were computed based on data gathered by the robot's onboard sensors during the run to a goal pose. That approach is appropriate for rapid prototyping and often sufficient to obtain representative results; however, still prone to poor performance of the limited-range robot sensors, e.g., RGBD cameras. Thus, integrating a robot with an external, e.g., vision-based system, can increase the evaluation robustness, decreasing metric deviations between subsequent trials. We argue that external systems for human tracking can be used for benchmarking once the robot control system is integrated with them. Otherwise, planners may be penalised for actions disregarding surrounding humans that the planners are unaware of. Nevertheless, our benchmark can be interfaced with any source of aggregated information about humans surrounding the robot.

C. EVALUATION

We integrated multiple planning approaches with the TIAGo robot and evaluated their operation under the same environmental conditions. In each scenario, we tested traditional trajectory planners for mobile robots, namely: *Elastic Bands* [20], *DWA* [19], *Trajectory Rollout* [38], *TEB* [37], as well as human-aware trajectory planners: *Human-aware TEB* (*HaTEB*) [39] and *Co-operative Human-Aware Navigation* (*CoHAN*) [40]. The *SRPB*'s parameters used for the evaluation are shown in Tab. 3. The results of our experiments, shown in Tab. 4, are discussed in the remainder of this section. Examples of trajectories performed by each planner are shown in Fig. 8.

a: PERFORMANCE METRICS

In terms of keeping a safe distance from obstacles (m_{obs}), *HaTEB* planner was the safest in all scenarios, which is

[5] https://github.com/spencer-project/spencer_people_tracking
[6] http://wiki.ros.org/social_navigation_layers
[7] https://github.com/rayvburn/hubero

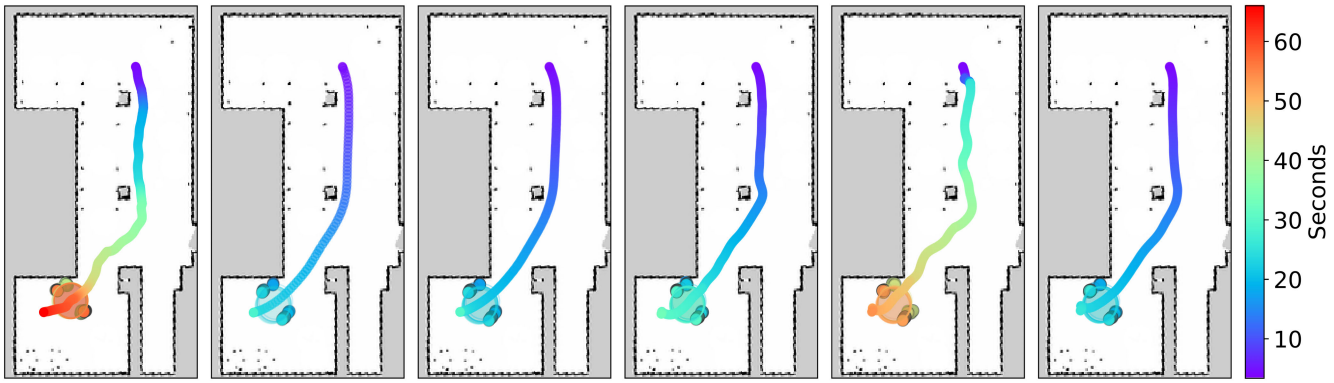


FIGURE 8. Robot trajectories generated by different planners in the *static* scenario. The colour of a symbol represents its occurrence in time (g) . Solid circles with dark edges represent humans, whereas partially transparent circles indicate F-formations. Due to the perception inaccuracy, human positions float over time, especially, after robot rotation at the very end of the scenario.

related to its characteristic of taking wide paths at corners (similar to going along the centre line of the available space). However, it came at the cost of the time required to reach the goal (m_{mef}). By contrast, the robot operating with *DWA* planner traversed closer to obstacles (a higher percentage of time spent by the robot within the ${}^{r,\mathbb{O}}d_{\text{min}}$ distance from obstacles along the path). Still, it reached the goal significantly faster (Req. 1). The reason behind such timing performance is that cost functions of *DWA* and *Trajectory Rollout* planners were tuned with a focus on approaching the goal with the shortest possible path avoiding high-cost areas (humans and obstacles) along the way. The best timing results (m_{mef}) are confirmed by the values of path-related metrics, m_{plin} and m_{chc}, which *DWA* and *Trajectory Rollout* planners have the lowest.

The results of m_{cef} and m_{cre} must be analysed, remembering that the simulated scenarios were performed on a different PC than real-world experiments. Nevertheless, data show that sampling-based methods (*DWA* and *Trajectory Rollout*) exhibit a higher computational burden (m_{cef}) than force-based (*Elastic Bands*) and optimisation-based (*TEB*, *HaTEB*, *CoHAN*) approaches. These latter, mainly *TEB*, have much more stable computation times in different scenarios (m_{cre}).

An interesting observation concerns the values of m_{cef} and m_{cre} metrics between *TEB* and its human-aware variants. Namely, *HaTEB* and *CoHAN* have longer computation times due to the increased number of optimisation objectives regarding humans.

b: NATURALNESS OF ROBOT MOTION

Sampling-based planners provide smoother trajectories (smallest m_{vsm} and m_{hsm}), increasing motion naturalness of the robot (Req. 2.1). In terms of oscillations (m_{osc}) and backward movements (m_{bwd}), the traditional trajectory planners performed the best in most cases, avoiding unnatural motions. However, in the real-world *dynamic* scenario (2b), *DWA* planner often performed recoveries moving backwards. As shown in Tab. 2, we allowed planners to command the mobile base backwards to verify how they will behave against an unexpected human agent.

As for in-place rotations (m_{iprot} metric), *TEB* and derived planners – *HaTEB* and *CoHAN*, generally outperform others due to the feature of this class of planners that adjust the final part of the trajectory to reach the goal position and orientation simultaneously. Instead, *DWA* and *Trajectory Rollout* rotate the mobile base according to the goal orientation after reaching the goal position. Another situation when the robot executes in-place rotations is when it encounters a dynamic obstacle along the path. This issue may be addressed with human trajectory prediction that has been employed in *HaTEB* and *CoHAN*.

c: PERCEIVED SAFETY OF HUMANS

Our human-perceived safety metrics (Req. 2.2) are novelties amid robot planning benchmarks. We evaluated the planners against personal space intrusion (m_{psi}), F-formation space intrusion (specifically, O-spaces, m_{fsi}), and robot heading direction relative to a human position (m_{dir}). Surprising results are related to m_{psi} metric, where generally the human-unaware *TEB* planner outperformed its human-aware specialisations – *HaTEB* and *CoHAN*. Only in the simulated *static* scenario (1a), *HaTEB* performed better than the traditional *DWA* planner only by 1 p.p.

Although none of the planners explicitly accommodate human formations, we expected that in the *static* scenario (1a and 1b), the robot's behaviour will emerge to F-formation avoidance due to regarding personal spaces of single humans. It did not happen with any planner, as the robot has crossed through the O-space of the F-formation in each trial (Fig. 6b). The phenomenon is reflected in m_{fsi} metric, remembering from (22) that the robot escaping from the O-space sooner obtains a smaller metric value. In the *static* scenarios, the human-aware *CoHAN* planner stopped and often oscillated when crossing through the O-space of the F-formation

TABLE 4. Results of simulation and real-world experiments conducted in the WUT laboratory environments.

Metric	Scenario	Elastic Bands	DWA	Trajectory Rollout	TEB	HaTEB	CoHAN
m_{obs} [%]	1a	58.51	33.98	39.15	36.10	**22.83**	44.17
	1b	50.44	42.25	43.39	36.77	**21.88**	41.53
	2a	24.03	35.44	28.13	28.39	**19.16**	53.55
	2b	23.46	52.93	71.65	38.67	**19.81**	46.02
m_{mef} [s]	1a	73.30	**25.50**	26.50	29.75	55.70	27.90
	1b	85.84	**28.63**	29.00	31.15	38.70	39.70
	2a	52.80	29.50	**28.00**	32.90	38.80	55.10
	2b	59.19	**37.70**	47.50	38.70	57.50	40.10
m_{plin} [m]	1a	10.27	9.80	**9.78**	10.18	10.90	10.00
	1b	11.78	**10.59**	10.65	11.14	12.37	12.30
	2a	12.90	11.98	**11.66**	12.03	13.10	15.71
	2b	13.96	13.08	**11.93**	12.78	15.09	12.63
m_{chc} [rad]	1a	32.62	**10.68**	10.72	14.09	28.35	12.09
	1b	50.61	11.55	**8.11**	12.84	25.10	12.37
	2a	38.11	4.55	**3.91**	15.14	30.77	13.14
	2b	66.96	**7.11**	35.48	16.68	55.31	13.54
m_{cef} $[10^{-3} \cdot \text{s}]$	1a	3.33	66.02	33.45	**2.50**	4.42	3.80
	1b	5.12	160.15	78.88	**4.99**	10.30	6.61
	2a	**1.92**	65.23	36.13	2.20	3.31	2.63
	2b	**2.42**	123.13	64.31	4.54	8.80	7.88
m_{cre} $[10^{-3} \cdot \text{s}]$	1a	3.89	34.57	9.99	**2.35**	3.44	3.27
	1b	3.91	81.93	15.95	**2.05**	4.42	4.29
	2a	2.47	28.44	11.56	**2.21**	3.28	3.88
	2b	2.53	75.80	28.46	**1.64**	3.89	7.15
m_{vsm} $\left[\frac{\text{m}}{\text{s}^2}\right]$	1a	0.07	0.05	**0.04**	0.11	0.32	0.14
	1b	0.22	**0.11**	0.12	0.14	0.39	0.22
	2a	0.15	0.09	**0.05**	0.12	0.34	0.16
	2b	0.28	0.18	**0.17**	0.19	0.35	0.25
m_{hsm} $\left[\frac{\text{rad}}{\text{s}^2}\right]$	1a	1.44	**0.18**	0.21	0.93	0.89	0.85
	1b	1.54	**0.22**	0.37	0.86	1.16	0.78
	2a	1.44	**0.18**	0.19	1.10	0.92	0.56
	2b	1.68	**0.33**	0.57	0.96	1.01	0.78
m_{osc} [%]	1a	3.79	1.95	**0.95**	1.18	3.67	1.69
	1b	9.70	3.89	2.62	**1.30**	3.27	2.12
	2a	**0.73**	1.78	0.90	0.91	2.11	7.05
	2b	6.01	**2.15**	2.15	4.57	5.68	2.50
m_{bwd} [%]	1a	0.00	0.00	0.00	1.50	3.44	6.04
	1b	0.00	0.00	0.00	1.58	5.35	7.79
	2a	0.00	0.00	0.00	0.00	1.74	0.00
	2b	**0.00**	10.82	0.26	0.26	2.92	2.50
m_{iprot} [%]	1a	5.18	13.72	13.69	**0.00**	1.63	0.40
	1b	5.74	9.79	10.19	0.94	1.48	**0.12**
	2a	6.41	3.51	3.63	0.30	0.31	**0.00**
	2b	4.07	0.40	24.06	0.28	3.24	**0.00**
m_{psi} [%]	1a	22.35	15.09	16.47	19.32	**14.30**	19.78
	1b	**15.88**	20.94	31.37	17.88	24.29	24.88
	2a	**20.24**	31.53	38.07	40.70	35.35	40.44
	2b	**19.88**	42.60	30.00	25.40	34.48	39.74
m_{fsi} [%]	1a	38.29	42.07	53.25	**32.11**	36.60	45.32
	1b	**3.33**	31.85	41.05	13.30	8.25	35.73
	2a	0.00	0.00	0.00	0.00	0.00	0.00
	2b	0.00	0.00	0.00	0.00	0.00	0.00
m_{dir} [%]	1a	**0.29**	0.76	1.04	1.31	0.77	0.87
	1b	**0.23**	2.38	0.69	0.46	1.09	0.62
	2a	**0.22**	1.32	0.39	0.56	1.34	0.55
	2b	**0.31**	2.13	0.74	0.62	0.97	1.46

(therefore increased m_{fsi} values). In the *dynamic* scenarios (2a and 2b), the robot's perception did not qualify moving humans as an F-formation (as expected); thus values of m_{fsi} are 0.0.

The metric representing human disturbance induced by a robot heading direction (m_{dir}) is useful for evaluating whether a planner is capable of adjusting the trajectory heading towards a human as soon as it detects such an agent. Again, human-aware trajectory planners performed similarly to traditional ones across all scenarios. Here, *Elastic Bands* accomplished the best metric scores, which are caused by its frequent heading changes (m_{chc}).

In our tests, examined state-of-the-art human-aware trajectory planners do not improve robot navigation regarding social metrics over the traditional approaches that treat humans as generic obstacles.

d: ROBUSTNESS

The results of our experiments are based on successful trials of the robot navigating from the initial pose to the goal pose. However, we performed multiple test runs beyond the benchmarked trials to find the start and goal poses accessible for all planners. During these tests, we could observe the robustness of each planner. *TEB* planner outperforms others with 0% of failed runs, whereas *HaTEB* commonly aborted further navigation being stuck, e.g., before an F-formation in the *static* scenario. By contrast, although the *Elastic Bands* planner appeared to be as robust as *TEB* and performed well in terms of the perceived safety of humans, it is impractical to use due to generating multiple erratic motions (reflected by m_{chc} and m_{hsm} metrics). These, in turn, cause much longer times needed to reach the goal (m_{mef} metric).

e: COMPARISON STRATEGY

We have found that comparing different planners in a simulation generally allows finding the one that will also perform best in a similar real-world scenario regarding a particular metric. We observed the robot operating with different planners during our simulation and real-world experiments, and some distinctive behaviours of certain planners are visible in both cases. These include, e.g., wide turns of *HaTEB*, mostly straight trajectories of *DWA*, and smooth stopping of *Trajectory Rollout*.

VI. SUMMARY

In this work, we presented *SRPB*, the social robot navigation benchmark that evaluates both the performance and the human-awareness aspects. It was designed to verify the fulfilment of the robot navigation requirements and assist system designers in selecting the best method for the application. Our approach allows comparing different navigation algorithms rapidly in both simulated and real-world environments. It also ensures easy integration with popular ROS-driven robots.

We focused on implementing quantitative metrics to evaluate common robot behaviour patterns. Most of the metrics in our benchmark allow confronting navigation algorithms, provided that the initial and final conditions of the evaluated scenario are the same in each trial. Therefore, path and trajectory similarities must be guaranteed in subsequent tests for a given scenario. Although physical sensors' inaccuracy impacts bigger metrics dispersion in real-world scenarios, our results show that, in general, comparing planners in a simulation environment is a legitimate and effective way to find the best planner for an equivalent real-world application.

In this work, we compared local (short-term) trajectory planners, but an analogical assessment of global path planners is also reasonable.

Our method investigates only non-focused interactions, so only the movement behaviours of humans and the robot in a shared space are evaluated. Extending our benchmark for evaluating focused human-robot interactions would be another significant contribution to social robotics. Initial research on this topic has already started and relates to, e.g., the approach pose of a robot that initiates an interaction with a human [26].

The results obtained with our benchmark show that state-of-the-art human-aware trajectory planners do not significantly improve the social aspects of robot navigation. Therefore, we argue that properly including social behaviour patterns in novel trajectory planners is still an open problem. We hope that our benchmark gives a good foundation for the further development of human-aware navigation algorithms. We also plan to add ROS2 compatibility to *SRPB*.

REFERENCES

[1] R. Raj and A. Kos, "A comprehensive study of mobile robot: History, developments, applications, and future research perspectives," *Appl. Sci.*, vol. 12, no. 14, p. 6951, Jul. 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/14/6951

[2] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robot. Auto. Syst.*, vol. 61, no. 12, pp. 1726–1743, Dec. 2013.

[3] Y. Gao and C.-M. Huang, "Evaluation of socially-aware robot navigation," *Frontiers Robot. AI*, vol. 8, Jan. 2022.

[4] A. Biswas, A. Wang, G. Silvera, A. Steinfeld, and H. Admoni, "SocNavBench: A grounded simulation testing framework for evaluating social navigation," *ACM Trans. Hum.-Robot Interact.*, vol. 11, no. 3, pp. 1–24, Jul. 2022.

[5] E. T. Hall, *The Hidden Dimension* (A Doubleday Anchor book). Garden City, NY, USA: Anchor Books, 1969.

[6] A. Kendon, *Spacing and Orientation in Co-Present Interaction*. Berlin, Germany: Springer, 2010, pp. 1–15.

[7] E. Heiden, L. Palmieri, L. Bruns, K. O. Arras, G. S. Sukhatme, and S. Koenig, "Bench-MR: A motion planning benchmark for wheeled mobile robots," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4536–4543, Jul. 2021.

[8] A.-I. Toma, H.-Y. Hsueh, H. A. Jaafar, R. Murai, P. H. J. Kelly, and S. Saeedi, "PathBench: A benchmarking platform for classical and learned path planning algorithms," in *Proc. 18th Conf. Robots Vis. (CRV)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2021, pp. 79–86, doi: 10.1109/CRV52889.2021.00019.

[9] L. Rocha and K. Vivaldini, "Plannie: A benchmark framework for autonomous robots path planning algorithms integrated to simulated and real environments," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2022, pp. 402–411.

[10] J. Tani, A. F. Daniele, G. Bernasconi, A. Camus, A. Petrov, A. Courchesne, B. Mehta, R. Suri, T. Zaluska, M. R. Walter, E. Frazzoli, L. Paull, and A. Censi, "Integrated benchmarking and design for reproducible and accessible evaluation of robotic agents," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 6229–6236.

[11] D. Mishkin, A. Dosovitskiy, and V. Koltun, "Benchmarking classic and learned navigation in complex 3D environments," 2019, *arXiv:1901.10915*.

[12] D. Perille, A. Truong, X. Xiao, and P. Stone, "Benchmarking metric ground navigation," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Nov. 2020, pp. 116–121.

[13] J. Wen, X. Zhang, Q. Bi, Z. Pan, Y. Feng, J. Yuan, and Y. Fang, "MRPB 1.0: A unified benchmark for the evaluation of mobile robot local planning approaches," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 8238–8244.

[14] L. Kästner, T. Bhuiyan, T. A. Le, E. Treis, J. Cox, B. Meinardus, J. Kmiecik, R. Carstens, D. Pichel, B. Fatloun, N. Khorsandi, and J. Lambrecht, "Arena-bench: A benchmarking suite for obstacle avoidance approaches in highly dynamic environments," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 9477–9484, Oct. 2022.

[15] C. Chamzas, C. Quintero-Peña, Z. Kingston, A. Orthey, D. Rakita, M. Gleicher, M. Toussaint, and L. E. Kavraki, "MotionBenchMaker: A tool to generate and benchmark motion planning datasets," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 882–889, Apr. 2022.

[16] S. Tafnakaji, H. Hajieghrary, Q. Teixeira, and Y. Bekiroglu, "Benchmarking local motion planners for navigation of mobile manipulators," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Jan. 2023, pp. 1–6.

[17] N. Tsoi, A. Xiang, P. Yu, S. S. Sohn, G. Schwartz, S. Ramesh, M. Hussein, A. W. Gupta, M. Kapadia, and M. Vázquez, "SEAN 2.0: Formalizing and generating social situations for robot navigation," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 11047–11054, Oct. 2022.

[18] C. Mavrogiannis, P. Alves-Oliveira, W. Thomason, and R. A. Knepper, "Social momentum: Design and evaluation of a framework for socially competent robot navigation," *ACM Trans. Hum.-Robot Interact.*, vol. 11, no. 2, pp. 1–37, Feb. 2022, doi: 10.1145/3495244.

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

[20] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, pp. 802–807.

[21] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, vol. 3, 2009, pp. 1–6.

[22] I. Dynnikov and B. Wiest, "On the complexity of braids," *J. Eur. Math. Soc.*, vol. 9, no. 4, pp. 801–840, 2007. [Online]. Available: https://hal.science/hal-00001267

[23] J. Guzzi, A. Giusti, L. M. Gambardella, G. Theraulaz, and G. A. Di Caro, "Human-friendly robot navigation in dynamic environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 423–430.

[24] S. B. Banisetty, "Multi-context socially-aware navigation using non-linear optimization," Ph.D. dissertation, Dept. Comput. Sci. Eng., Univ. Nevada, Reno, NV, USA, May 2020.

[25] M. Daza, D. Barrios-Aranibar, J. Diaz-Amado, Y. Cardinale, and J. Vilasboas, "An approach of social navigation based on proxemics for crowded environments of humans and robots," *Micromachines*, vol. 12, no. 2, p. 193, Feb. 2021. [Online]. Available: https://www.mdpi.com/2072-666X/12/2/193

[26] X.-T. Truong and T.-D. Ngo, "'To approach humans?': A unified framework for approaching pose prediction and socially aware robot navigation," *IEEE Trans. Cognit. Develop. Syst.*, vol. 10, no. 3, pp. 557–572, Sep. 2018.

[27] M. Neggers, R. Cuijpers, P. Ruijten, and W. Ijsselsteijn, "Determining shape and size of personal space of a human when passed by a robot," *Int. J. Social Robot.*, vol. 14, pp. 1–12, Mar. 2022.

[28] J. Rios-Martinez, A. Spalanzani, and C. Laugier, "From proxemics theory to socially-aware navigation: A survey," *Int. J. Social Robot.*, vol. 7, no. 2, pp. 137–153, Apr. 2015.

[29] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics: A Foundation for Computer Science*. Reading, MA, USA: Addison-Wesley, 1989.

[30] B. Okal and K. O. Arras, "Learning socially normative robot navigation behaviors with Bayesian inverse reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2889–2895.

[31] A. V. Magro, R. Cintas, L. Manso, P. Bustos, and P. Núñez, *Socially-Accepted Path Planning for Robot Navigation Based on Social Interaction Spaces*. Cham, Switzerland: Springer, Jan. 2020, pp. 644–655.

[32] R. Kirby, "Social robot navigation," Ph.D. dissertation, Robotics Inst., Carnegie Mellon Univ. Pittsburgh, Pennsylvania, PA, USA, May 2010.

[33] G. Taubin, "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 11, pp. 1115–1138, Nov. 1991.

[34] D. V. Lu, D. Hershberger, and W. D. Smart, "Layered costmaps for context-sensitive navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 709–715.

[35] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo localization for mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1999, pp. 1322–1328.

[36] J. Karwowski, W. Dudek, M. Węgierek, and T. Winiarski, "HuBeRo—A framework to simulate human behaviour in robot research," *J. Autom., Mobile Robot. Intell. Syst.*, vol. 15, no. 1, pp. 31–38, Jul. 2021. [Online]. Available: https://en.wikipedia.org/wiki/%C4%98
[37] C. Rösmann, F. Hoffmann, and T. Bertram, "Integrated online trajectory planning and optimization in distinctive topologies," *Robot. Auto. Syst.*, vol. 88, pp. 142–153, Feb. 2017.
[38] B. Gerkey and K. Konolige, "Planning and control in unstructured terrain," in *Proc. IEEE Int. Conf. Robot. Automat.*, Pasadena, CA, USA, May 2008.
[39] H. Khambhaita and R. Alami, "A human–robot cooperative navigation planner," in *Proc. Companion ACM/IEEE Int. Conf. Human-Robot Interact.* New York, NY, USA: Association for Computing Machinery, Mar. 2017, pp. 161–162.
[40] P. Teja Singamaneni, A. Favier, and R. Alami, "Human-aware navigation planner for diverse human-robot interaction contexts," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 5817–5824.

JAROSŁAW KARWOWSKI received the M.Sc.Eng. degree in automation and robotics from the Faculty of Electrical Engineering, Warsaw University of Technology (WUT), in 2018, where he is currently pursuing the Ph.D. degree with the Faculty of Electronics and Information Technology, Institute of Control and Computation Engineering.

His research interests include robot control systems modeling, social robot motion planning, autonomous vehicle control systems integration, and embedded programming of robotic actuators. He contributes to the open-source robotic community, sharing with the public his ROS packages concerning human behavior simulation, robot perception, and planning.

WOJCIECH SZYNKIEWICZ received the Ph.D. and D.Sc. (habilitation) degrees in control and robotics from the Warsaw University of Technology (WUT).

He is currently a Professor with the Institute of Control and Computation Engineering, WUT. His research interests include motion and task planning, autonomous mobile robots, robot controller structures, and real-time and distributed systems. He works on sensor-based motion planning and control algorithms for robot systems, including service and personal mobile robots. He is the author/coauthor of more than 100 publications (in proceedings of conferences, journals, and books) concerned with the above-mentioned research subjects.