Human-friendly Robot Navigation in Dynamic Environments

Jérôme Guzzi[1] Alessandro Giusti[1] Luca M. Gambardella[1] Guy Theraulaz[2] Gianni A. Di Caro[1]

***Abstract*— The vision-based mechanisms that pedestrians in social groups use to navigate in dynamic environments, avoiding obstacles and each others, have been subject to a large amount of research in social anthropology and biological sciences. We build on recent results in these fields to develop a novel fully-distributed algorithm for robot local navigation, which implements the same heuristics for mutual avoidance adopted by humans. The resulting trajectories are *human-friendly*, because they can intuitively be predicted and interpreted by humans, making the algorithm suitable for the use on robots sharing navigation spaces with humans. The algorithm is computationally light and simple to implement. We study its efficiency and safety in presence of sensing uncertainty, and demonstrate its implementation on real robots. Through extensive quantitative simulations we explore various parameters of the system and demonstrate its good properties in scenarios of different complexity. When the algorithm is implemented on robot swarms, we could observe emergent collective behaviors similar to those observed in human crowds.**

I. INTRODUCTION

We present a novel, human-inspired approach for ground robot navigation and obstacle avoidance in dynamic environments. The scenarios that we address feature the simultaneous presence of *multiple robots* (hereafter generically referred to as *robot swarms*) as well as of *human pedestrians*. The problem of robot swarms sharing common navigation spaces with humans arises in many practical contexts, including the use of mobile autonomous robots for assistive tasks in domestic environments, for patrolling and surveillance, for service delivery in health care institutions, and so on. In more general terms, given the expected pervasive deployment of mobile robot technologies in societal and industrial settings, it is customary to endow mobile robots with navigation capabilities that meet both engineering and societal objectives. From an *engineering* point of view, robot navigation needs to be effective in terms of followed trajectories, robust to inherent sensing errors, scalable to differently crowded environments, and distributed, meaning that robots to not need to rely on any external infrastructure to effectively move around. From a *societal* point of view, robot navigation behaviors need to be safe, as well as be *human-friendly*.

With human-friendly we mean that robot movements must exhibit behaviors that are *acceptable* by humans. In terms of local navigation, this can be ensured by following trajectories that are similar to those that a person in a similar setting would follow. Such trajectories have the property to be *predictable* and *legible* by humans, meaning that a person observing robot motion can intuitively understand the spatial target the robot is heading to [1]. Other than limiting emotional stress, this ensures that both humans and robots can navigate the space efficiently. In fact, if navigation algorithms generate unpredictable trajectories, humans have to frequently change their local plans to move around the robots, ultimately resulting in less efficient navigation for both groups.

In this work, we address both the engineering and societal aspects of the navigation problem in shared human-robot spaces, with a particular emphasis on the societal aspects. We propose a robot navigation algorithm which only relies on *local sensing* (i.e., it is fully distributed and does not need external infrastructures), is *reactive*, which allows it to operate in real-time in dynamic environments, and also includes a *proactive* planning component, based on a local estimation of the mobility of the agents in its field of view. To address the societal issues, the algorithm is based on a recently published heuristic model, proposed by Moussaid et al. [2] for explaining pedestrian behavior. The heuristic has been validated in large-scale comparative experiments and has shown to closely model observed human trajectories in controlled conditions [3].

To the best of our knowledge, this is the first porting of such a heuristic into the robotics domain. In our implementation, we provide a number of extensions to the basic model, taking into account the core differences between robots and humans. In particular, we included changes aimed at: (i) enforcing safety (lateral contacts and pushing are an integral part of human crowd motion but are unacceptable in robotics); (ii) respecting human personal spaces whenever possible; (iii) preventing local crowding situations, which could naturally lead to reduced efficiency; (iv) dealing with deadlock conditions that can arise among robotic agents. It must be noted that the resulting robot navigation and obstacle avoidance algorithm, while designed having in mind mixed, human-robot crowds, works equally well when only robots are in the environment. In this respect, it can play the role of a simpler, and more flexible alternative to existing methods [4] (e.g., the ones based on velocity obstacle, which are described in the related work, Section II).

The main contribution of the paper is, therefore, a bio-inspired, local navigation algorithm with both reactive and proactive components, which can generate human-like and human-friendly trajectories. At the same time, the algorithm is conceptually simple, computationally light, independent on the specific sensing technique, inherently able to handle

[1] J. Guzzi, A. Giusti, L. M. Gambardella, and G. A. Di Caro are with the Dalle Molle Institute for Artificial Intelligence (IDSIA), Manno-Lugano, Switzerland {jerome, alessandrog, luca, gianni}@idsia.ch

[2] G. Theraulaz is with the Université Paul Sabatier, Toulouse, France guy.theraulaz@univ-tlse3.fr

heterogeneous agents, and able to provide effective, robust, and scalable navigation behaviors. The algorithm is described in Section III. We discuss and demonstrate its implementation on both *real* and *simulated robots* (in Section IV), validating the simulation results with real experiments, and we also provide quantitative results obtained from large-scale simulations (Section V). In particular, we test the scalability of the approach to large robot swarms, the robustness to sensing inaccuracies, and the effect of various parameters affecting safety and human friendliness. We additionally demonstrate that robot swarms implementing the proposed algorithm exhibit macroscopic behavioral patterns (i.e., the emergence of lanes of opposite flows in corridors) matching those observed in human crowds [3].

II. RELATED WORK

The problem of dynamic obstacle avoidance is widely researched both in the robotics and in the sociology literature. Our works builds upon results in both fields.

In robotics, the most common approach is based on the concept of *velocity obstacle* [5], also known as *collision cone* or *forbidden velocity map* – i.e. the sets of velocities that will lead a robot to collision: choosing a velocity outside such set ensures that no collision will occur. Different variants have been presented to improve the prediction of the other agents' trajectories [6], [7], [8], to add recursion and account for sensing errors [9] in a probabilistic framework, and to ensure smooth trajectories by sharing the responsibility to avoid a collision with other agents (*reciprocal velocity obstacle* [4]). Applications to very crowded scenarios also introduce asymmetries in the obstacle velocity construction [10] so to enforce conventions allowing smooth, deterministic interactions between agents. The velocity-obstacle model was successfully applied to explain certain characteristics of pedestrian behavior [11] and validated by comparing real and simulated trajectories of two crossing humans.

All such works build basically on a mechanicistic and artificial approach to navigation, which is designed to ensure safety, and adapt it to produce smooth trajectories. On the contrary, our work stems from a heuristic [2] modeling human behavior – which produces paths with good efficiency, smoothness, and legibility – to which we add some modifications to ensure safety. This paper represents the first implementation of this heuristic to robotics. Implementation-wise, such heuristic allows us to decouple the computation of the desired heading and speed. This leads to a simpler implementation than velocity-obstacle approaches, which requires a search over the two-dimensional velocity space.

Humans mutual avoidance and sharing of space has been extensively studied in sociology, among other for the prediction of the behavior of crowds. The original models are based on the study of *proxemics* [12], which formalizes the concept of *personal* and *social* space; pedestrian behavior based on *social forces* [13] enforces people to keep a minimum distance from neighbors whenever possible. Such model was successfully used for crowd simulation and also inspired several human tracking and avoidance models in robotics [14], [15]. Simple rules on pedestrian navigation (passing left and shared collision avoidance responsibility) were incorporated in a sampling-based [16] planner for collision avoidance among robots. Moussaid et al. [2], [3] recently proposed a fundamentally different model based on a simple heuristic; in this paper, we consider such heuristic and extend it for implementation in robots.

By adopting such heuristic, we ensure that robot will exhibit a human-like behavior. In turn, this ensures that humans sharing space with the robot will be able to predict its intentions thus improving efficiency and social acceptance. Our work achieves this goal in the context of obstacle avoidance, or local path planning. In the significantly different context of *global* path planning – in which complete information about the environment is available – different approaches have been proposed with the same goals; discomfort to humans is minimized through the enforcement of specific constraints [17], additional cost terms [18], [19], [20] or rules implementing specific social conventions [21]. In the same framework, extensions were proposed for addressing more complex interaction scenarios like a joint working space [22], for ensuring that robots do not obstruct human's view of the environment [23] and for navigating among moving humans [24], [25]. A related line of research is concerned with learning-based prediction of human behavior [26], [27], for avoiding unwanted interference with human activity [28], or navigate shared spaces [29].

The interplay between local and global navigation algorithms has been shown to be of large importance for the legibility and acceptance of the resulting trajectories: in particular, human-aware global path planners only resulted in acceptable trajectories when coupled with human-like obstacle avoidance algorithms [1], which further highlights the importance of human-friendly local navigation algorithms.

III. METHOD

A. Behavioral model for human locomotion

In this section we summarize the behavioral model of vision-based human locomotion in social groups that we used to design our robot navigation algorithm [2], [3]. To account for the targeted robotic context, we generically speak in terms of an 'agent', rather than a human, and introduce some minor modifications to the original model.

Given a 2D reference frame F depicted in Figure 1, a moving agent directed to a target point $\vec{O}$ is characterized by an optimal (open space) moving speed v_{opt} and a horizontal field of view 2ϕ (in radians). In the reference frame F, $\vec{x}(t)$, $\vec{v}(t)$ are agent's position and velocity vectors at time t, and $\alpha(t)$ is agent's heading, (i.e., the direction it is facing with respect to F's horizontal axis). We assume that agent's ground occupancy is approximated by a circle of known radius r.

To direct its movements, the agent makes use of a cognitive function $f(\alpha),\ \alpha \in [\alpha(t) - \phi,\ \alpha(t) + \phi]$, based on visual information, that maps each heading α within the field of view to the distance that the agent can travel in $\alpha's$ direction before colliding with any visible obstacle, when

moving at speed v_{opt}. The distance is bounded by a maximum horizon H. With $s(\alpha)$ we denote the 2D segment connecting $\vec{x}$ with the point at distance $f(\alpha)$ along direction α (i.e., the point of first collision for heading α). When computing $f(\alpha)$, all obstacles are assumed to keep their current heading and speed, thus moving according to a uniform linear motion [1].

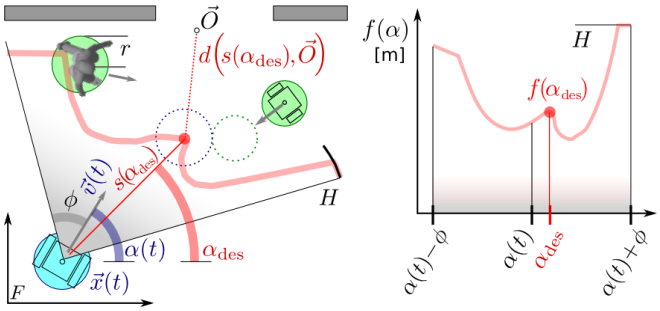


Fig. 1: Illustration of variables and functions defined in text. The red curve is the estimated free distance that the blue agent can travel in direction α up to the first collision, that in direction α_{des} will happen at the red point where the blue (agent) and green (robot neighbor) dotted circles touch.

Given the above notations, the agent walking behavior can be explained by the following simple heuristic rules.

First, the agent determines its desired heading $\alpha_{\text{des}}(t)$ as the direction allowing the most direct path to destination point $\vec{O}$, taking into account the presence of obstacles:

$$\alpha_{\text{des}}(t) = \text{argmin}_{\alpha}\, d\left(s\left(\alpha\right), \vec{O}\right) , \tag{1}$$

where d is the minimal distance between segment and point. In practice, if the robot moves towards direction α_{des} at a constant speed v_{opt}, it will reach a point closer to the target than any point it would reach when moving along any other direction $\alpha \neq \alpha_{\text{des}}$.

The desired velocity vector $\vec{v}_{\text{des}}(t)$ is then determined. It has direction defined by the heading $\alpha_{\text{des}}(t)$ and modulus $v_{\text{des}}(t)$, set to allow stopping in a fixed time τ_1 within the free distance $D(\alpha_{\text{des}}) \in [0, H]$, *currently* seen in direction α_{des}:

$$v_{\text{des}}(t) = \min\left(v_{\text{opt}}, \frac{D\left(\alpha_{\text{des}}\right)}{\tau_1}\right) . \tag{2}$$

The actual velocity vector $\vec{v}(t)$ is continuously adjusted depending on $\vec{v}_{\text{des}}(t)$:

$$\frac{d\vec{v}}{dt} = \frac{\vec{v}_{\text{des}}(t) - \vec{v}(t)}{\tau_2} , \tag{3}$$

where the fixed parameter τ_2 represents the *time constant* characterizing the exponential speed profile, which in practice modulates the smoothness of motion. Controlled laboratory experiments measured $\tau_1 = \tau_2 = 0.5$ s for pedestrians in normal walking conditions [31].

Note that, since computing $f(\alpha)$ values involves a rough prediction of agent's and obstacles' future trajectories, the resulting behavior is *proactive* in that it attempts to avoid potential collisions well before they are expected to occur.

[1]Humans feature a dedicated neural mechanisms to detect object motion [30] and predict the time-to-collision with obstacles.

B. Application to robotic navigation

The model described above results in smooth paths, which have shown to closely match the characteristics of pedestrian motion in large-scale controlled experiments, both for single trajectories and macroscopic crowd motion patterns [3]. Robots following the same rules would therefore exhibit a behavior which is predictable, legible, and acceptable by humans sharing the same environment with robots.

Neverthless, the immediate application of the model to robotics is hindered by several shortcomings, with the main one related to the fact that trajectories are *not (entirely) safe*: in fact, *collisions* among humans happen routinely and, especially in crowded situations, contribute to define the motion of tightly-packed groups through reciprocal pushing forces. Even with sparse agents, collisions may happen when agents with limited field of view are unable to perceive each other when traveling side by side, or in presence of sudden direction changes, which are only partially accounted for by the heuristic model.

Clearly, in our context, collisions (both among robots and between robots and humans) should be avoided as much as possible. Therefore, we extend the heuristic with the concept of *safety margin*, which is common to many obstacle avoidance approaches. In particular, when computing $f(\alpha)$, we account for an increased radius $r' = r + m_s$ for each agent, with m_s being a fixed safety margin parameter.

Under the unrealistic assumption of perfectly-accurate and 360° sensing, choosing a sufficiently large value for m_s ensures that no collisions can occur. However, with realistic sensing inaccuracies and limited field of view, a completely safe behavior cannot be guaranteed. On the other hand, large safety margins also lead to inefficient and unnatural trajectories (e.g., see [32] for a recent study on planning in a probabilistic framework with imperfect sensing). Therefore, given the characteristics of the sensing subsystem, the safety margin m_s sets the tradeoff between efficiency and safety of the trajectories, which is investigated in Section V-B. Unlike [2], in our approach, if an obstacle is inside the safety margin, we set $f(\alpha) = D(\alpha) = 0$ for all angles α which would bring the robot closer to the obstacle.

One can observe that, when crossing in opposite directions, robots tend to pass each other (and humans) as close as allowed by the safety margin, regardless of how much space is available. While being appropriate for modeling human behavior, this behavior is not always suited to robots, for two reasons: (i) a robot passing a human should, *if possible*, keep a distance larger than the minimum safe distance, to avoid invading its personal space and causing discomfort; (ii) groups of robots passing close to each other can induce a temporary situation of *local crowding*, which occasionally results in temporary *deadlocks*. Increasing m_s is not an appropriate solution for either problems, because agents should be allowed to come close to each other *when needed* (e.g., in order to negotiate tight spaces).

Therefore, we introduce the concept of *social margin*, $m_t \geq m_s$, and redefine $r' = r + m(d)$, with $m(d)$ being a piecewise linear function of the distance d between the agent and its closest neighbor. $m(d)$ is bounded between m_s, for small values of d, and m_t, for large d values (see Figure 2).

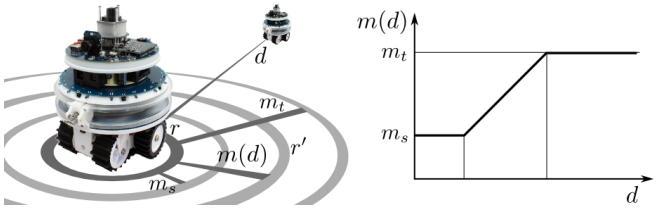


Fig. 2: Illustration of safety margin m_s and social margin m_t. As a function of the distance d of the closest neighbor, a margin $m_s \leq m(d) \leq m_t$ is added to each obstacle's physical radius r. The figure also illustrates the foot-bot robot, on which we implemented the navigation system.

As a result, *when enough space is available*, d is large and the robots tend to keep a distance larger than strictly necessary. On the one hand, this increases social acceptance by humans, on the other hand, this reduces the likelihood of forming local high-density clusters of robots (further enhancing safety as side-effect) which may result in deadlocks. Still, deadlocks cannot be completely ruled out and may occasionally occur, especially with large sensing errors and/or large numbers of robots packed in tight spaces.

General solutions for avoiding traffic jams and escaping from deadlocks require higher-level planning and/or explicit coordination among agents, both of which do not fit in our current, low-level approach. Partial solutions include the use of implicit coordination (e.g., in the form of social conventions), or of *evasive behaviors*. We implemented the latter approach, with a clause loosely inspired by human behavior: after being stuck for a short time, an agent recognizes that it has been likely involved in a deadlock. Then, after a random time interval, it becomes ochlophobic (i.e., exhibits a fear of crowds) and spends a random amount of time moving towards a random, obstacle-free direction (regardless of its target), before switching back to the normal behavior. Experiments in Section V-C show that this simple heuristic leads to a virtuous emerging collective behavior, quickly unraveling complex deadlocks involving tens of agents (see the video attachment).

IV. REAL AND SIMULATED ROBOT MODELS

The navigation algorithm described in Section III has been implemented on real robots (the *foot-bots*) as well as in simulations, based on simulated foot-bot models.

A. The foot-bot platform and algorithm implementation

The *foot-bot* robot (Figure 2) is a small mobile platform, directly derived from the *marXbot* [33], specifically designed for swarm robotics [34]. The robot is 30 cm wide and 20 cm tall, and is based on an on-board ARM-11 processor programmed in a Linux-based operating environment. Differential-driven motorized tracks allow mobility at speeds up to 0.3 m/s.

In the context of this work, foot-bots use two distinct sensing modalities. (i) An IR-based *range-and-bearing* (RAB) sensor and communication system, which allows a robot to detect its *line-of-sight* robot neighbors within a 4-meter range and estimate their relative distance and bearing; each robot also advertises its current speed and relative bearing to neighbors through the same system. (ii) A forward-facing camera with a $2\phi = 50°$ field of view and a down-sampled resolution of 128×92 px, which is used for localizing humans, targets, and walls at 25 frames-per-second.

Because our main focus is on navigation algorithms and not on sensing, we use straightforward techniques for processing camera images: entities of interest, (e.g., landmarks used to identify a destination point, or humans) are marked with differently colored bands at a known height from the floor. Robots convert each frame to the HSV color space, and segment pixels corresponding to each object. After performing connected component analysis, this results in a set of binary blobs. From the image coordinates of each blob's centroid, the robot computes distance and bearing of the corresponding entity by means of an homography transform, which can be estimated in advance given that the camera parameters and height of each entity are known. The velocity of humans is estimated by finite differencing, after smoothing position readings with a moving average filter defined over a period of 0.5 s. Note that the position of targets (i.e., destination points) is sensed online through vision, and not given by an external observer or a higher-level path planning algorithm.

Robot controllers operate on a 0.1 s timestep and are not synchronized with each other. At each timestep, rules in Eq. (1) and (2) yield the desired values for heading (α_{des}) and speed (v_{des}), respectively. Both in simulation and in the implementation on the foot-bots, we use a mobility model similar to Eq. (3) that takes into account the robots constraints and controls the speed of left (w^l) and right (w^r) differential driven track wheels as follows. First, we define the desired rotation speed in terms of the desired differential $\chi_{\text{des}}(t)$ between the left and right wheels' tangential velocities; $\chi_{\text{des}}(t)$ is found as a function of $\alpha(t)$ and $\alpha_{\text{des}}(t)$, bounded by a maximum allowed value $\chi_{\max} = 0.1\text{m/s}$:

$$\chi'_{\text{des}}(t) = \frac{\alpha_{\text{des}}(t) - \alpha(t)}{\tau_r} \frac{L}{2} \tag{4}$$

$$\chi_{\text{des}}(t) = \begin{cases} -\chi_{\max} & \text{if } \chi'_{\text{des}} < -\chi_{\max} \\ +\chi_{\max} & \text{if } \chi'_{\text{des}} > +\chi_{\max} \\ \chi'_{\text{des}}(t) & \text{otherwise}, \end{cases} \tag{5}$$

where L is the wheel axis length and τ_r determines the time constant for robot rotation smoothing. We fix $\tau_1 = \tau_2 = \tau_r = 0.5s$ to mirror human motion characteristics, but such parameters can be decreased for more reactive (but less predictable) behavior as well to increase its caution when $\tau_1 > \tau_2$. The desired tangential speed of the left (w^l_{des}) and right (w^r_{des}) wheel is then defined as $w^l_{\text{des}} = v_{\text{des}}(t) - \chi_{\text{des}}(t)$,

$w^r_{\text{des}} = v_{\text{des}}(t) + \chi_{\text{des}}(t)$. For each of w^l and w^r, the actual rotation speed is modulated as $\dot{w} = \frac{w - w_{\text{des}}}{\tau_2}$, and clipped in such a way that $|w|$ does not exceed a maximum speed w_{max}, which for foot-bot robots corresponds to 0.3 m/s.

B. Robot simulation

We developed a custom simulator for performing large-scale experiments with both foot-bot and human models.

Simulated vision sensor readings approximate the statistical properties of localization errors from monocular, catadioptric, or stereo cameras. That is, precise and uniform bearing resolution but large uncertainty in depth estimation, which increases for objects farther away. More specifically, given an obstacle whose ground truth relative position is expressed in robot-centered polar coordinates as (ρ, θ). The observed position (ρ', θ') is given by $\theta' = \theta + \phi e$; $\rho' = \rho + k\rho\phi e$, where: $e \sim \mathcal{N}(0, \sigma)$ models the localization error in the normalized image space, ϕ denotes the camera field of view, and k is a constant depending on the characteristics of the depth estimation approach. In the following, we set $\sigma = 1/128$ (i.e., 1 pixel on a 128×96 sensor) and $k = 10$, which well fits the errors observed in real robots. We can evaluate the impact of sensing errors by tuning the σ parameter. As in the real robot implementation, velocity vectors are estimated by finite differencing.

Simulated range and bearing sensors, which well model other sensing modalities like laser, ultrasound, time-of-flight or structured illumination (e.g. Kinect), are instead characterized by constant angular resolution for bearing and distance-independent uncertainty for range (within maximum limits) and a constant probability (set as 80%) for the message to be received.

V. EXPERIMENTS

We report the results of several simulation experiments exploring the characteristics of the proposed navigation approach in three different settings. In the setting *cross* (Sections V-A, V-B), we consider four target destinations at the vertices of a square with an edge of 4 m. Robots are divided in two equally-sized groups: robots of each group travels back and forth between two opposite vertices, thus creating a busy crossroad in the middle. In setting *circle* (Section V-C), robots are initialized at regular intervals along a 5 m radius circumference, and need to reach once a target opposite to their starting position, which creates challenging conditions close to the center of the circumference. In setting *periodic corridor* (Section V-D), groups of robots and humans travel in opposite directions in an infinite straight corridor bounded by walls, which is a setup often considered in crowd analysis literature.

For each experiment, we test system's performance by varying a specific parameter. For each value of the parameter we perform $R = 50$ simulation runs (replicas), each lasting T seconds and initialized randomly. The following performance measures have been considered.

- *Relative throughput:* defined as the number of targets that a robot has reached during the simulation. It is expressed as a fraction of the number of targets that the robot could reach if traveling along straight paths and ignoring collisions (which is an ideal upper bound for throughput). Throughput is directly related to the minimum time the robot would take to reach the same targets under the same conditions above. We also measure time, discounting, for each agent, the time it takes to reach the first target in order to reduce the impact of initial conditions. The resulting quantity is adimensional, bounded between 0 (worst) and 1 (optimal), and is averaged over all the robots in the simulation.
- *Path irregularity:* defined as the amount of *unnecessary turning* per unit path length performed by a robot, where *unnecessary turning* corresponds to the total amount of robot rotation minus the minimum amount of rotation which would be needed to reach the same targets with the most direct path. Path irregularity is measured in rad/m, and is averaged over all the robots in the simulation.
- *Total number of collisions:* measuring the collisions occurring during simulation, which is reported as collisions per robot per minute.

A. Scalability and validation with real robots

Within scenario *cross*, we initially verify the scalability of the algorithm versus an increasing number of robots, and validate simulation results by comparison with the performance measured on foot-bot robots. Results are reported in Figure 3. We can observe that the results obtained with real robots in the same conditions closely match simulations. When large swarms are considered, the relative throughput decreases and path irregularity increases, because robots must follow longer and more curvy trajectories in order to avoid collisions. Even in very crowded scenarios, paths remain well smooth and predictable (well shown in the video attachment).

In the real robot implementation, despite the severe hardware limitations, the navigation controller requires invariably less than 20 ms of computation time per timestep. In simulation, we also tested robustness to timesteps longer than 0.1 s, and found that in all considered scenarios, performance begins to degrade only when the timestep exceeds 0.4 s.

B. Safety of trajectories

In the same *cross* scenario we study the effect of different parameters on system performance in terms of relative throughput and safety, which is measured as number of collisions per robot per minute. We consider the challenging case in which all obstacles are perceived through vision, thus dropping the range-and-bearing sensor and ruling out any explicit communication mechanism among robots. Figure 4 shows the effects of different visual sensing errors σ, of the camera field of view ϕ, and of the safety margin m_s.

As expected, unreliable sensing (larger σ) leads to more collisions (Figure 4a). At the same time, trajectories are less efficient, because robots need to change plans often in order to make it up for sensing errors. Also a narrow field of view ϕ

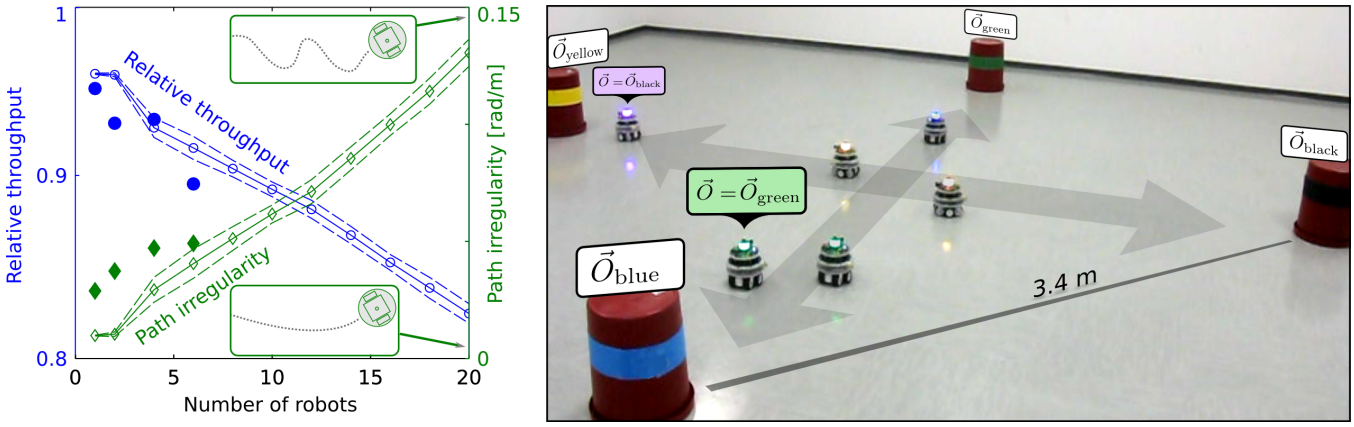
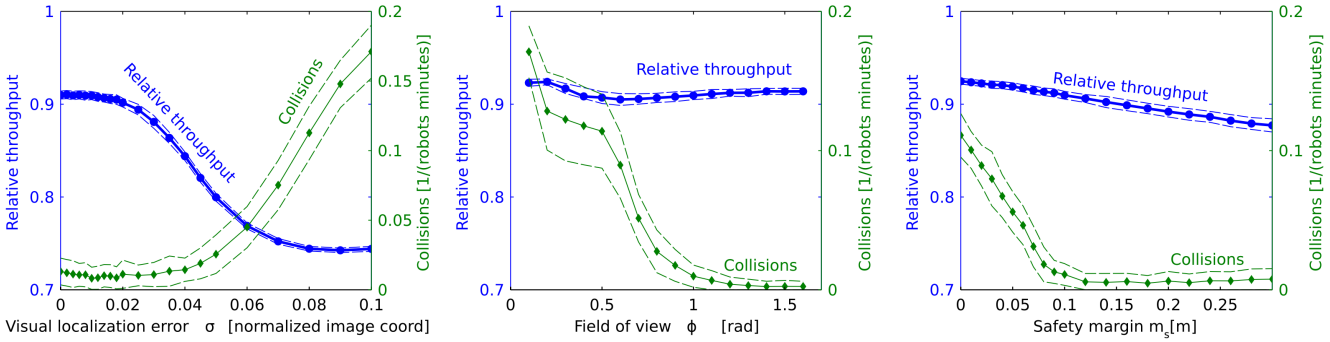


Fig. 3: Experimental results (left) for scalability in the *cross* scenario (pictured on the right). Large filled markers correspond to results measured on real robots. Blue thick line with round markers denotes relative throughput. Green thin line with diamond markers shows path irregularity. Dashed lines delimit $\pm$ standard deviation over $R = 50$ replicas. In this experiment, both real and simulated robots use $360°$ range-and-bearing sensing, $H = 3$ m, $m_s = m_t = 10$ cm, $T = 360$ s.



(a) $m_s = m_t = 10$ cm, $\phi = 1$ rad (b) $m_s = m_t = 10$ cm, $\sigma = 0.008$ (c) $\phi = 1$ rad, $\sigma = 0.008$

Fig. 4: Experimental results for safety in the *cross* scenario. $N = 10$ robots, $T = 900$ s. Visual localization error $\sigma = 0.008$ corresponds to 1 pixel in a 128×96 image and matches errors observed on real robots.

leads to collisions (Figure 4b), but in this case trajectories are slightly more efficient because robots tend to follow (unsafe) direct paths since they cannot perceive (and thus account for) obstacles at their sides.

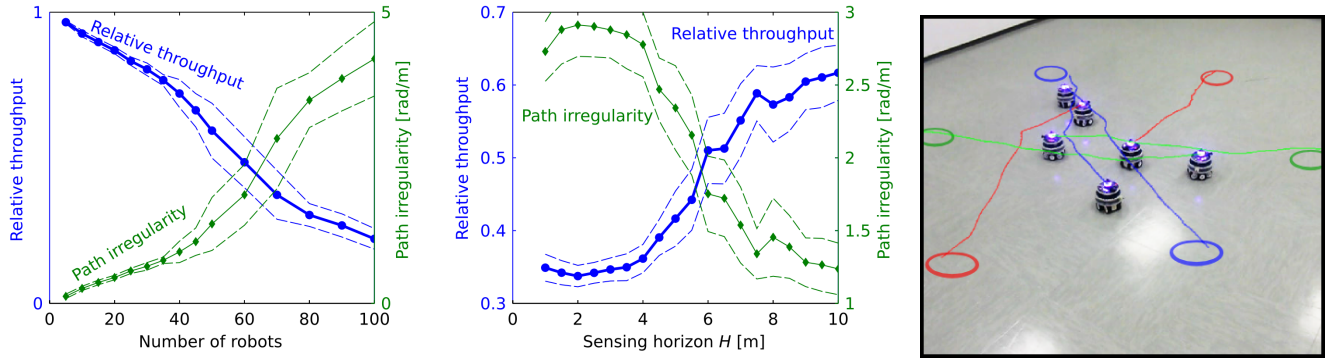
The rightmost plot in Figure 4 shows the impact of the parameter m_s (safety margin) under realistic operating conditions ($\phi = 1$rad, $\sigma = 0.008$). A safety margin larger than 10 cm effectively prevents collisions among robots. However, large m_s values lead to less efficient trajectories because robots deviate significantly from the optimal path in order to keep large distances from neighbors.

C. *Scalability and planning horizon in the* circle *topology*

The scenario *circle* is quite a challenging one. It was also studied in previous works [10]. We investigate the system performance for different values of swarm size N and horizon H. Larger swarms obviously lead to less efficient and regular paths, mostly due to crowding at the center of the circle (Figure 5a). Moreover, in this context the impact of H is apparent. A relatively large horizon allows the robots to earlier react to the upcoming crowding in the center of the circle, causing the emergence of an interesting collective behavior: all robots pass to the right of the center (or, in an equal number of simulations, to the left), despite having no explicit communication nor implementing social conventions dictating so. The resulting trajectories (see Figure 6) are extremely efficient. When a shorter horizon or social radius do not grant the time to develop such coordination, robots jam in the center of the circle, but can efficiently escape the resulting deadlocks using the heuristic for evasive behaviors described in Section III. This is well illustrated in the video attachment.

D. *Macroscopic crowd behaviors in the* periodic corridor

Figure 7 reports the results obtained in the *periodic corridor* scenario (i.e., a straight line corridor in which the top and the bottom ends are connected, like on the lateral surface of a cylinder). Agents traveling in opposite directions exhibit collective behaviors matching those observed in studies of human crowds: they tend to form ordered *flow lines* (despite being initialized randomly and implementing no explicit rules promoting such behavior). Figure 7a explores this behavior versus time when 60 robots travel along the corridor, 30 robots for each direction. After 40 seconds,



(a) $H = 10$ m (b) $N = 50$ robots (see video attachment) (c) Real robot simulation with $N = 6$

Fig. 5: Experimental results for the *circle* scenario, $\phi = 1$ rad, $\sigma = 0.008$.

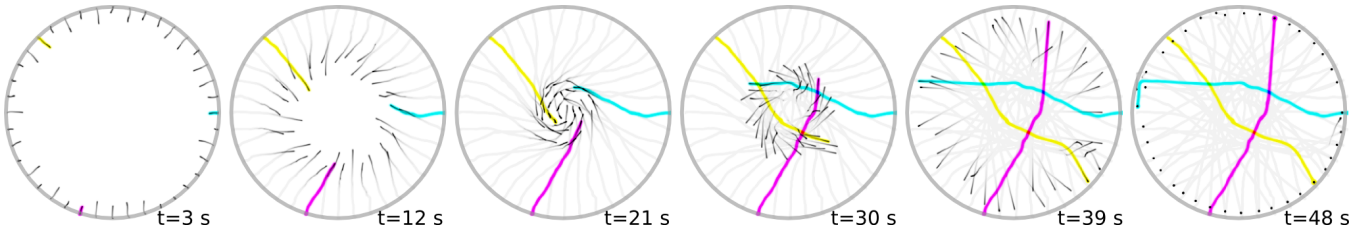
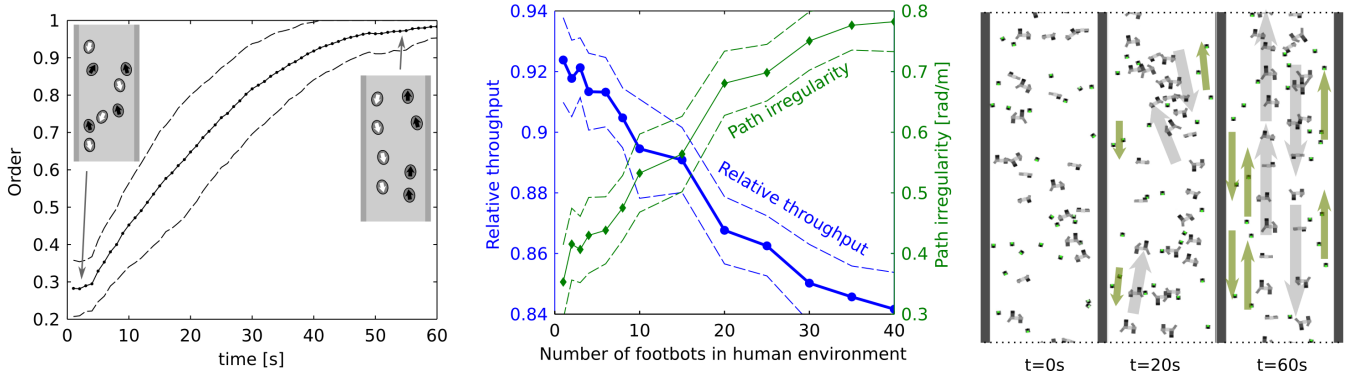


Fig. 6: Visualization of trajectories for $N = 40$ robots in *circle* scenario: $H = 10$ m, $\phi = 1$ rad, $\sigma = 0.008$, radius=5 m.



(a) $N = 60$ robots, $\phi = 1$ rad, $\sigma = 0.008$, corridor width is 2 m (b) 30 humans + N robots. $\phi = 1$ rad, $\sigma = 0.008$, corridor width 4 m (c) 30 humans + $N = 30$ robots. $\phi = 1$ rad, $\sigma = 0.008$, corridor width 4 m

Fig. 7: Experimental results for the *periodic corridor* scenario. The Left plot shows the order of agents' configuration versus time when only robots, traveling in two opposite directions, are included. Order is 0 when agents are scattered in random positions, whereas it approaches 1 when they form clear vertical lines of flow (see [3] for a formal definition of order). The Middle plot shows the performance when different numbers of robots share the corridor with 30 humans. The Right plot illustrates how humans and robots (which travel at significantly different speeds) tend to auto-organize themselves in vertical flow fields characterized by homogeneous agents traveling in the same direction (also shown in the video attachment).

order approaches 1, meaning that agents are organized in vertical flow lines. Figures 7b and Figure 7c consider a corridor shared among 30 humans and N robots. Each group is divided in equal-sized subgroups traveling in opposite directions. As expected, as N increases, throughput (which is reported for robot agents only) decreases, while path irregularity increases (Figure 7b). The setting with $N = 30$ is shown Figure 7c at three different times, showing the random initial setting, an intermediate configuration, and the ordered organization reached after 60s.

VI. CONCLUSIONS

We introduced a novel local navigation algorithm for ground robots, based on a simple obstacle avoidance heuristic which well models pedestrian behavior [3]. We adapted the heuristic to a multi-robot context and designed a few extensions of it in order to ensure effective, safe, and smooth behavior in challenging settings.

The algorithm is demonstrated on real robots and on large-scale simulations considering multiple scenarios with different characteristics. We explored the effects of differ-

ent parameters on the system's efficiency, smoothness, and safety. The simulated experiments also demonstrate good behavior when sharing spaces with humans.

Current research is aimed at validating the acceptance of the resulting robots' trajectories in real settings with humans. Moreover, additional experiments are planned for validating the ability of the algorithm to handle heterogeneous agents, such as robots with dramatically different speeds or motion behaviors.

REFERENCES

[1] C. Lichtenthaler, T. Lorenz, M. Karg, and A. Kirsch, "Increasing perceived value between human and robots – Measuring legibility in human aware navigation," in *IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, 2012, pp. 89–94.

[2] M. Moussaïd, D. Helbing, and G. Theraulaz, "How simple rules determine pedestrian behavior and crowd disasters." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 108, no. 17, pp. 6884–6888, 2011.

[3] M. Moussaïd, E. G. Guillot, M. Moreau, J. Fehrenbach, O. Chabiron, S. Lemercier, J. Pettré, C. Appert-Rolland, P. Degond, and G. Theraulaz, "Traffic instabilities in self-organized pedestrian crowds." *PLoS Computational Biology*, vol. 8, no. 3, p. e1002442, 2012.

[4] J. van den Berg, D. Manocha, and M. Lin, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2008, pp. 1928–1935.

[5] P. Fiorini and Z. Shiller, "Motion pplanning in fynamic environments using velocity obstacles," *The International Journal of Robotics Research*, no. 7, pp. 760–772.

[6] A. B. E. Yasuaki, M. Yoshiki, and Y. Abe, "Collision avoidance method for multiple autonomous mobile agents by implicit cooperation," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, 2001, pp. 1207–1212.

[7] D. Wilkie, J. van den Berg, and D. Manocha, "Generalized velocity obstacles," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 5573–5578.

[8] J. van den Berg, J. Snape, S. J. Guy, and D. Manocha, "Reciprocal collision avoidance with acceleration-velocity obstacles," *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3475–3482, 2011.

[9] B. Kluge and E. Prassler, "Recursive agent modeling with probabilistic velocity obstacles for mobile robot navigation among humans," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, no. October, 2003, pp. 376–380.

[10] J. Snape, J. van den Berg, S. J. Guy, and D. Manocha, "The hybrid reciprocal velocity obstacle," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 696–706, 2011.

[11] S. J. Guy, M. C. Lin, and D. Manocha, "Modeling collision avoidance behavior for virtual humans," in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, 2010, pp. 575–582.

[12] E. T. Hall, "A system for the notation of proxemic behavior," *American Anthropologist*, vol. 65, no. 5, pp. 1003–1026, 1963.

[13] D. Helbing, I. Farkas, and T. Vicsek, "Simulating dynamical features of escape panic." *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.

[14] Y. Tamura, T. Fukuzawa, and H. Asama, "Smooth collision avoidance in human-robot coexisting environment," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 3887–3892.

[15] M. Luber, J. A. Stork, G. D. Tipaldi, and K. O. Arras, "People tracking with human motion predictions from social forces," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2010, pp. 464–469.

[16] R. Knepper and D. Rus, "Pedestrian-inspired sampling-based multi-robot collision avoidance," in *Proceedings of the International Symposium on Robot and Human Interactive Communication*, 2012.

[17] E. G. Collins Jr, B. Goldiez, A. Donate, and D. Dunlap, "Human-aware robot motion planning with velocity constraints," in *Proceedings of the International Symposium on Collaborative Technologies and Systems*, 2008, pp. 490–497.

[18] E. E. Sisbot, R. Alami, T. Simeon, K. Dautenhahn, M. Walters, and S. Woods, "Navigation in the presence of humans," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, 2005, pp. 181–188.

[19] E. A. Sisbot, L. F. Marin, R. Alami, T. Simeon, and C. Roche, "A mobile robot that performs human acceptable motions," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 1811–1816.

[20] T. Kruse, P. Basili, S. Glasauer, and A. Kirsch, "Legible robot navigation in the proximity of moving humans," in *Proceedings of IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, 2012, pp. 83–88.

[21] R. Kirby, R. Simmons, and J. Forlizzi, "COMPANION: A constraint-optimizing method for person-acceptable navigation," in *Proceedings of the 18th IEEE International Symposium on Robot and Human Interactive Communication*, 2009, pp. 607–612.

[22] C.-P. Lam, C.-T. Chou, C.-F. Chang, and L.-C. Fu, "Human-centered robot navigation – Toward a harmoniously coexisting multi-human and multi-robot environment," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 1813–1818.

[23] J. Rios-Martinez, A. Renzaglia, A. Spalanzani, A. Martinelli, and C. Laugier, "Navigating between people: a stochastic optimization approach," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 231855, 2012, pp. 2880–2885.

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

[25] L. Scandolo and T. Fraichard, "An anthropomorphic navigation scheme for dynamic scenarios," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011, pp. 809–814.

[26] P. Trautman and A. Krause, "Unfreezing the robot: navigation in dense, interacting crowds," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 797–803.

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

[28] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 3931–3936.

[29] C. Fulgenzi, A. Spalanzani, and C. Laugier, "Probabilistic motion planning among moving obstacles following typical motion patterns," in *Proceedings of 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 4027–4033.

[30] P. R. Schrater, D. C. Knill, and E. P. Simoncelli, "Mechanisms of visual motion detection." *Nature neuroscience*, vol. 3, no. 1, pp. 64–68, 2000.

[31] M. Moussaïd, D. Helbing, S. Garnier, A. Johansson, M. Combe, and G. Theraulaz, "Experimental study of the behavioural mechanisms underlying self-organization in human crowds." in *Proceedings of the Royal Society B: Biological sciences*, vol. 276, no. 1668, 2009, pp. 2755–2762.

[32] N. E. D. Toit, J. W. Burdick, and N. E. Du Toit, "Robot Motion Planning in Dynamic, Uncertain Environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, 2012.

[33] M. Bonani, V. Longchamp, S. Magnenat, P. Rétornaz, D. Burnier, G. Roulet, F. Vaussard, H. Bleuler, and F. Mondada, "The marXbot, a Miniature Mobile Robot Opening new Perspectives for the Collective-robotic Research," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 4187–4193.

[34] M. Dorigo, D. Floreano, L. M. Gambardella, F. Mondada, S. Nolfi, T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, D. Burnier, A. Campo, A. L. Christensen, A. Decugnière, G. A. Di Caro, F. Ducatelle, E. Ferrante, A. Förster, J. M. Gonzales, J. Guzzi, V. Longchamp, S. Magnenat, N. Mathews, M. M. de Oca, R. O'Grady, C. Pinciroli, G. Pini, P. Rétornaz, J. Roberts, V. Sperati, T. Stirling, A. Stranieri, T. Stützle, V. Trianni, E. Tuci, A. E. Turgut, and F. Vaussard, "Swarmanoid: a novel concept for the study of heterogeneous robotic swarms," *IEEE Robotics & Automation Magazine*, 2012 (to appear).