Lloyd-based Approach for Robots Navigation in Human-shared environments

Manuel Boldrer, Luigi Palopoli and Daniele Fontanelli

***Abstract*—We present a Lloyd-based navigation solution for robots that are required to move in a dynamic environment, where static obstacles (e.g, furnitures, parked cars) and unpredicted moving obstacles (e.g., humans, other robots) have to be detected and avoided on the fly. The algorithm can be computed in real-time and falls in the category of the reactive methods. The simplicity, the small amount of information required for the control inputs synthesis, and the low number of parameters to be tuned are the highlights of this method. Moreover, the multiagent flocking is considered in the same framework and opens to several interesting applications. particular, we show how a team of robots can flocks The goodness of the method is proved through extensive simulations and, for the single agent navigation in human-shared environment, also with experiments on a unicycle-like robot.**

***Index Terms*—Reactive navigation, Multi-agent systems, Distributed control.**

I. Introduction

Securing a safe and efficient navigation for mobile robots in unstructured environments is one of the most urgent problems in modern robotics. Two are the main approaches that traditionally compete for prominence in this area. The first, called *deliberative*, requires a complete *a priori* knowledge of the environment and relies on optimisation solutions that produce a motion plan maximising the elected cost function. The second, which is frequently called *reactive*, emphasises the importance of reacting quickly to unexpected conditions detected by the on-board sensing system. The reactive approaches are quick and efficient and they integrate the sensor data in real-time. Clearly, reactive algorithms cannot rely on global information on the environment and on the mission and therefore they do not usually seek optimality. In the class of applications considered in this paper, we have three requirements: 1. each robot has a global mission, we could live with sub-optimal solutions but we need some guarantee that the robot will eventually fulfil the mission goals, 2. the environment is extremely dynamic, we have got a "nominal" map but the presence of temporary obstacles and of humans changes the landscape on a per-minute basis, 3. some of the missions are assigned to groups of robots, which travel together but are not stiffly constrained to a formation. Two meaningful paradigms in the many that could describe our class of applications are the delivery of food and medicines inside an hospital or of small packets within the pedestrian area of an historical town.

M. Boldrer and D. Fontanelli are with the Department of Industrial Engineering (DII), University of Trento, Via Sommarive 5, Trento, Italy {manuel.boldrer, daniele.fontanelli}@unitn.it.
L. Palopoli is with the Department of Information Engineering and Computer Science (DISI), University of Trento, Via Sommarive 5, Trento, Italy luigi.palopoli@unitn.it

Digital Object Identifier (DOI): see top of this page.

Related Work. Robotic navigation in the presence of obstacles has been a constant topic of high interest for robotic researchers throughout the globe, with a massive number of results that can be used directly or as a source of inspiration. Some of the first proposal revolved around the idea of potential fields [1], which reduce to the minimum the computation time but which under certain condition generate local minima preventing the progress of the robot to the goal. Other authors propose a Deformable Virtual Zone [2], i.e., the definition of a safe area around the robot in which the presence of unforeseen obstacles triggers the robot's reactions. Borenstein et al. [3] propose the vector field histogram for densely cluttered environment navigation in order to manage the uncertain presence of obstacles from the sensor readings. In the class of deliberative approaches that operate knowing up-front the position of the obstacles, we find sampling based approaches such as RRT [4], RRT* [5] and combinatorial approaches based on cell-decomposition [6]. A particular way to generate a cell decomposition is by using computational geometry and Voronoi diagrams [7]. A potential limitation of these approaches is that they cannot be directly applied to manage unexpected or moving obstacles. The computational efficiency of Voronoi diagrams makes this technique a good fit also for local re-planning [8].

Dynamic obstacles are the core concern of a whole lot of research works. Fox et al. [9] propose the dynamic window approach in order to identify point-wise the set of velocity securing an obstacle free progress. A similar idea is the so called velocity obstacle approach [10], in which the set of velocities producing a collision with an object moving at a known speed is computed at each sampling time. Many different combinations of this idea can be found in the literature. In [11] motion models of the surrounding humans and social constraints [12] are considered to reactively re-plan the robot route.

Extending single agent navigation algorithms to the multi-agent case is in general far from trivial. Reynolds [13] is one of the first authors who introduced a (flock) model of polarised, non-colliding aggregate motion e.g. flocks, herds and schools. Using potential function for distributed flocking is at the heart of the work of Olfati [14]. Extensions of the velocity obstacle algorithm to the multi-agent case are studied in [15], [16]. Once again the computational efficiency of Voronoi diagrams can be particularly appealing, Zhou et al. [17] present a distributed Voronoi-based algorithm for

collision avoidance between agents in a group, while Sud et al. [18] exploit first and second order Voronoi diagram constructions for real-time multi-agent path planning. Our approach is based on the Llyod algorithm [19] which can be applied to deploy a team of robot in order to achieve a good coverage of an area of interest both in convex [20] and in non-convex maps [21]. The use of Voronoi tessellation for navigation has been proposed by Lindhé et al. [22], who consider a particle model for a formation of robots. This approach was an important inspiration for this work, but it suffers from two important limitations that prevent its direct applicability to our context. At the level of the single agent, we explicitly consider the presence of human being in the environment, showing experimental evidences on an unicycle-like robot. At the level of the group of agents, we consider a group motion in which robots compose a loose group, just as much as human pedestrians do [23], [24]. In contrast, Lindhé et al. consider the group of agents as a stiff formation, whose motion can be difficult in adverse environments and can seem strange and possibly intimidating to human by-standers.

Paper Contribution. Our contribution is twofold: 1. we propose a Lloyd-based algorithm able to move a robot safely and to make it reach its goal position in a complex environment with static and dynamic obstacles, and human beings; 2. we show how the proposed algorithm is naturally extended to the multiagent case introducing a simple method to make the agents flock together in a safe manner by adapting the overall shape to the surrounding environment and to the surrounding agents. Contrary to the previous literature, our solution produces a loose formation in which the robots remain confined within a compact set but are allowed to change their relative positions, in a way that resembles a group of human pedestrians [23], [24]. Moreover, we introduce what we called the Convex Weighted Voronoi Diagram (CWVD) and we provide, in addition to the theoretical analysis, simulative and experimental evidences.

The paper is organised as follows. Section II introduces the background material and formalises the problem at hand. The technical developments of the Lloyd-based navigation approach is presented in Section III, while the effectiveness and major improvements with respect to the state-of-the-art solutions and the experiments on a wheeled robots are reported in Section IV. The multi-agent extension, with simulation results, is instead offered in Section V. Finally, Section VI presents the concluding remarks of the paper and defines the future research directions.

II. Background Material and Problem Description

In this paper we consider the robot as a unicycle, whose model is

$$[\dot{x}, \dot{y}, \dot{\theta}]^T = [v\cos\theta, v\sin\theta, \omega]^T \tag{1}$$

where $p_r = [x, y]^T$ is the position of the mid point on the rear axle and θ is the yaw angle. The control input vector $u = [v, \omega]^T$ comprises the forward velocity v and the angular velocity ω. To ensure a correct (i.e. collision-free) execution of the robot navigation in a known environment from a starting position s to an end position e, the mobile robot firstly plans a safe path that links s to the goal e. The planner adopted to this purpose considers only the obstacles that belong to the map. It is worthwhile to note that this is a necessary step to avoid local minima that otherwise cannot be ruled out when a reactive method is considered. The chosen global planner is arbitrary (e.g. RRT* [5]) and it is in general not required to satisfy any optimal criterium, i.e. the synthesised path can be coarse and needs only to satisfy the path safety constraint. The desired path is then discretised in way-points $\mathcal{WP} = \{wp^1, wp^2, \ldots, wp^n\}$, where $wp^1 \equiv s$ and $wp^n \equiv e$.

A. Lloyd algorithm

Let the mission space $\mathcal{Q} \in \mathbb{R}^2$ be a convex polytope and let us introduce the distance $||q - p_r||$, where q is a generic point in the $\mathcal{Q}$ space and p_r is the robot position. Given the (generic robot) positions $P = \{p_1, \ldots, p_n\}$, the i–th Voronoi cell is

$$V_i = \{q \in \mathcal{Q} | \, ||q - p_i|| \leq ||q - p_j||, \forall j \neq i\}. \tag{2}$$

Hence, let us consider the following cost function

$$\mathcal{H}(P) = \frac{1}{2}\sum_{i=1}^{n}\int_{V_i} ||q - p_i||^2 \phi(q) dq, \tag{3}$$

where n is the number of agents in the scene, and the probability density function $\phi : \mathcal{Q} \to \mathbb{R}_+$ is an indicator of the probability to have an interesting location or a desired goal location in the $\mathcal{Q}$ space. By assuming a single integrator dynamics, and following a gradient descent logic, we have $\dot{p}_i = -\frac{\partial \mathcal{H}(P)}{\partial p_i} = u_i$. By taking into account the mass of the i–th Voronoi cell

$$M_{V_i} = \int_{V_i} \phi(q) dq, \tag{4}$$

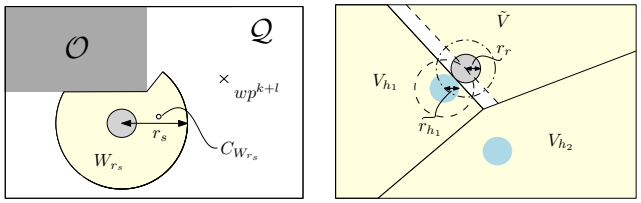
and the centroid position for the i–th Voronoi cell

$$C_{V_i} = \frac{1}{M_{V_i}}\int_{V_i} q\phi(q) dq, \tag{5}$$

we compute the derivative of (3) and obtain $\dot{p}_i = -M_{V_i}(p_i - C_{V_i})$. We can assert that by applying the control law $u_i = -k_{\text{prop}}(p_i - C_{V_i})$, each agent will converge asymptotically to its Voronoi centroid position, which is an optimal deployment for the coverage problem [19], [20].

B. Problem formulation

The problem we are facing is the navigation of a mobile platform (or a group thereof) from an initial position to a desired final position belonging to an environment comprising static obstacles (e.g. indoor corridors, alleys, doors, furnitures). We explicitly consider the presence of dynamic obstacles as well, such as other mobile robots or human beings. The controller takes inspiration from the Lloyd algorithm summarised in Section II-A, but it has to explicitly consider the nonholonomic constraints of the adopted unicycle-like robot (of course it can be applied to holonomic vehicles as well). Moreover, we assume that the robot has to preferably (i.e. most of the time) move at a desired forward velocity v^D. This problem takes the inspiration by the robotic platform



(a) (b)

Figure 1. The small faded circles with the boundary represents the robot. (a) Modified Voronoi tessellation W_{r_s}, used to avoid static obstacles, with the set $\mathcal{O}$ of the obstacle points and the set $\mathcal{Q}$ representing the environment points. (b) Modified Voronoi tessellation $\tilde{V}$, used to avoid collisions with dynamic obstacles. The circles without the thick boundary represent the human beings.

FriWalk [25], [26], which is an assistive cart [27] developed in the European project ACANTO [28] to help elderly in their navigation activities.

III. Single Agent Control

As stated in Section II, the path is assumed discretised in way-points $\mathcal{WP} = \{wp^1, wp^2, \dots, wp^n\}$ from starting s to ending e positions. By considering the robot position p and assuming that wp^k is the closest way-point, we select as next way-point to reach wp^{k+l}, which is the point ahead of a preview length l with respect to wp^k. Notice that the preview length is a function of the visibility set of the robot, i.e. if the straight line between p and wp^{k+l} intersects an obstacle of the map (i.e. static obstacle) we reduce the preview l until the line of sight in between is obstacle free (condition that is guaranteed to hold since the planned path is safe). An hybrid between the reactive and the deliberative approaches guides the robot toward the goal position. Notice that after the path is planned, the proposed solution is purely reactive.

A. Reactive control for static obstacles

To avoid collisions with static obstacles, we start by considering the single Voronoi partition V associated with the robot position p_r, i.e. using (2) we get $V \equiv \mathcal{Q}$. Then we constrained V on the visibility set of the robot itself. In particular, we start by defining the sensing area

$$S = \{q \in \mathcal{Q} | \, p_r + a(q - p_r) \notin \mathcal{O}\}, \;\; \forall a \in [0, 1],$$

where $\mathcal{O}$ is the set of points in $\mathcal{Q}$ belonging to the static obstacles in the map. This leads to the Voronoi-constrained partition $W = S \cap V$. By adding the constraints on the limited sensing range r_s, the Voronoi-Visible partition turns to

$$W_{r_s} = \{q \in W | \, ||q - p_r|| \leq r_s\}. \tag{6}$$

With this construction, the centroid position $C_{W_{r_s}}$ of the Voronoi-Visible partition, computed as in (5) with $\phi(q)$ uniform, is in general in a safe location, i.e. $C_{W_{r_s}} \in W_{r_s}$ and hence the segment connecting $C_{W_{r_s}}$ and p_r do not intersect $\mathcal{O}$ (see Figure 1-a). However, since W_{r_s} may not be convex, pathological cases in which $C_{W_{r_s}} \in \mathcal{O}$ could occur. When it happens, we modify the centroid position as the closest safe point to the actual centroid such that $C_{W_{r_s}} \in W_{r_s}$, as in [29].

B. Reactive control for dynamic obstacles

To deal with dynamic obstacles (e.g. robot-to-human interactions), we act again on the Voronoi cell. To this end, let us consider the Voronoi tessellation as generated by both the robot and the human beings in the scene. More precisely, assuming $P = \{p_r, p_{h_1}, \dots, p_{h_w}\}$, i.e. w human beings in the scene, we have the new partition V_i defined by (2) where $i, j \in \{r, h_1, \dots, h_w\}$. We further assume a circular occupancy region with radius r_r for the mobile robot, and with radius r_h for each human being. As it will be evident in the next Section, to avoid collision with a moving human being, i.e. a moving obstacle, a safety margin is needed, which is a function of $R_{rh} = r_r + r_h$, i.e., the sum of the occupancy regions. Indeed, the Voronoi cell associated to the robot is modified as follows:

$$\tilde{V} = \begin{cases} \{q \in \mathcal{Q} | \, ||q - p|| \leq ||q - p_h||\}, & \text{if } R_{rh} \leq \frac{||p - p_h||}{2} \\ \{q \in \mathcal{Q} | \, ||q - p|| \leq ||q - \tilde{p}_h||\}, & \text{otherwise,} \end{cases} \tag{7}$$

where $\tilde{p}_h = p_h + 2\left(R_{rh} - \frac{||p - p_h||}{2}\right) \frac{p - p_h}{||p - p_h||}$. The safety margin is thus computed when the circle centred in p_r and with radius R_{rh} exceeds the limit of the robot cell. An example of this construction is depicted in Figure 1-b.

C. Control synthesis

For the robot navigation, we control the yaw angle derivative and the forward velocity independently. To synthesise both control laws, we first compute the desired heading h^D using a Lloyd inspired approach (see Section II-A): we define a Laplacian distribution density function $\phi(q, p_r)$ with active way-point wp^{k+l} (the way-point to be reached) as mean value, i.e.

$$\phi(q, p_r) = \alpha \mathrm{e}^{\left(-\frac{||\mathrm{q} - \mathrm{wp}^{\mathrm{k+l}}||}{\rho}\right)}, \tag{8}$$

where ρ is a tuning parameter that controls the spread of the pdf (i.e. related to the density variance) and α is the normalisation constant computed as the inverse of the mass $M_{\tilde{V}}$ in (4). To implement the collision avoidance, we consider a new Voronoi cell that retains both the characteristics of the static (6) and dynamic obstacles (7), that is $\mathcal{F} = W_{r_s} \cap \tilde{V}$. The centroid $C_{\mathcal{F}}$ of this new cell, given by (5) and weighted by $\phi(q, p_r)$, defines the desired heading as

$$h^D = \frac{C_{\mathcal{F}} - p_r}{||C_{\mathcal{F}} - p_r||}, \tag{9}$$

where $h^D = [h_x^D, h_y^D]^T$. The angular velocity control law is given by

$$\omega = -\kappa(1 - \langle h^D, h \rangle)^\gamma \mathrm{sign}(h_x^D h_y - h_y^D h_x), \tag{10}$$

where $h = [h_x, h_y]^T = [\cos\theta, \sin\theta]^T$ is the current robot heading, $\gamma \in (0, \frac{1}{2})$ is a tuning parameter and $\kappa > 0$ is related to the vehicle steering capabilities (the proof of the stability of the vehicle given (10) is here omitted for space limits but reported in [30]). For what concerns the forward velocity, we consider two cases: if the vehicle is well oriented, that is if $\langle h^D, h \rangle \geq \cos\psi$, with $\psi \in (-\pi/2, \pi/2)$, the vehicle follows

the desired velocity v^D; if the orientation error is too high, the vehicle will brake to reduce the turning radius. Hence:

$$\dot{v} = \begin{cases} k_a(v^D - v), & \text{if} \quad \langle h^D, h \rangle \geq \cos\psi \\ -k_b v, & \text{otherwise,} \end{cases} \tag{11}$$

where k_a and k_b are two parameters that are representative of the accelerating and braking capabilities of the vehicle.

The effect of ρ: In (8), we have introduced the tuning parameter ρ, which regulates the spread of the pdf $\phi(q, p_r)$. If $\rho \to +\infty$, all the points assume the same weight and, hence, the robot is not attracted by the next way-point but tries to maximise the Voronoi partition coverage. On the other hand, if $\rho \to 0$, the mass is concentrated on the way-point, hence the agent exhibits a greedy behaviour, i.e. the robot will execute trajectories closer to the obstacles. Therefore, ρ can be regarded as a tuning parameter for the agent behaviour. Since different ρ choices could radically change the behaviour of the system (sometimes even leading to deadlock for large values), we propose an adaptive control that follows a switching logic ruled by the distance $d = ||C_{\mathcal{F}} - p_r||$, i.e.

$$\dot{\rho} = \begin{cases} -\rho, & \text{if } d < d_{\min} \\ (\rho - \rho^D), & \text{otherwise,} \end{cases} \tag{12}$$

where ρ^D is the desired spread factor, and $d_{\min}$ is a threshold value for the distance between the centroid and the robot. In practice, since smaller values of ρ move the centroid $C_{\mathcal{F}}$ towards the way-point wp^{k+l}, the point here is to reduce the weight of the cell geometry in favour of the target position, hence avoiding deadlock configurations.

Stability and convergence: We will first show that the control laws (10) and (11) ensures $p_r \to e$ for static obstacles, i.e. $\mathcal{F} = W_{r_s}$, and $wp^{k+l} = e$ when $\rho = 0$ in (8).

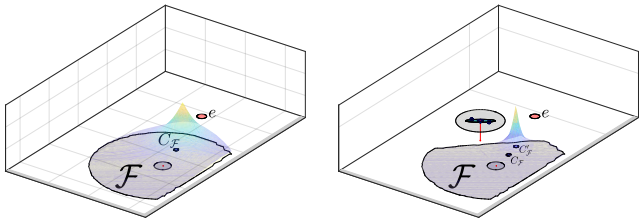
Theorem 1. *If $\mathcal{F} = W_{r_s}$, $wp^{k+l} = e$ and $\rho = 0$, the control laws* (10) *and* (11) *ensures $p_r \to e$.*

Proof. If $wp^{k+l} = e$ and $\rho = 0$ the centroid $C_{\mathcal{F}}$ does not change due to the robot motion. Moreover, if $\langle h^D, h \rangle < \cos\psi$, the robot is not pointing towards $C_{\mathcal{F}}$ but, by means of (10), the robot rotates towards the desired direction h^D. Notice that (11) ensures a smaller turning radius in this condition (eventually the robot would stop and turn on the spot). Once $\langle h^D, h \rangle \geq \cos\psi$, the robot moves along directions that ensure $\dot{d} < 0$, or, in other words, $p_r \to C_{\mathcal{F}}$. Using the fact that $\rho = 0$, the pdf in (8) turns to be a Dirac delta function centred in $wp^{k+l} = e$ and therefore we have immediately $e = C_{\mathcal{F}}$, which concludes the proof. □

Corollary 1. *If $\mathcal{F} = W_{r_s}$, $wp^{k+l} = e$ and ρ follows* (12)*, the control laws* (10) *and* (11) *ensures $p_r \to e$.*

Proof. Irrespective of the value of ρ, the control laws (10) and (11) ensures $p_r \to C_{\mathcal{F}}$, i.e. $d \to 0$. Therefore, there exists a time instant such that $d < d_{\min}$, hence, by (12), ρ decreases and $C_{\mathcal{F}} \to wp^{k+l}$. Hence, the proof. □

Figure 2-a depicts the situation where the robot moves towards the centroid $C_{\mathcal{F}}$ that is located between the current position and the desired ending position. Instead, Figure 2-b shows the modified cell in the presence of a person and the effect of the ρ spread factor. Next, we will prove what happens along the sequence of way-points, that again is reported in Figure 2-a.



(a) (b)

Figure 2. Modified Voronoi cells: (a) the robot with its Voronoi-Visibile set $\mathcal{F}$, the function $\phi(q, p_r)$ in (8) computed over $\mathcal{F}$, the centroid position $C_{\mathcal{F}}$ (small blue circle) and the goal position e (big red circle); (b) the presence of a human being modifies the cell and, moreover, the effect of the spread factor ρ on the centroid position ($C_{\mathcal{F}}$ and $C'_{\mathcal{F}}$ when ρ decreases).

Theorem 2. *If $\mathcal{F} = W_{r_s}$ and ρ follows* (12) *, the control laws* (10) *and* (11) *ensures $p_r \to e$.*

Proof. By Corollary 1, the distance $||p_r - wp^k|| \to 0$. By construction, when $p_r \approx wp^k$, the next way-point wp^{k+l} is selected, with $l \in \mathbb{N}$. Notice that the existence of a new reachable way-point is ensured by discretisation of the planned path. The proof is then derived recalling that $wp^n \equiv e$. □

We are now in a position to prove one of the main theoretical results of this paper.

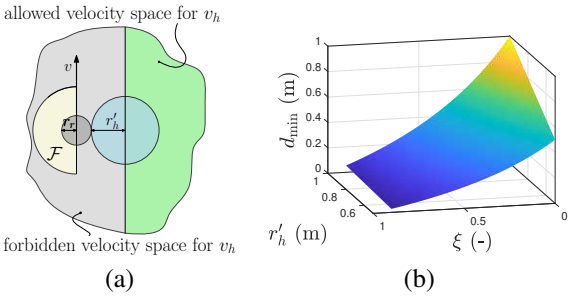
Theorem 3. *If $\mathcal{F} = W_{r_s} \cap \tilde{V}$, ρ follows* (12) *and the dynamic obstacles in the scene are finite in number and do not play against the robot (i.e. does not follow a pursuit policy [31]), the control laws* (10) *and* (11) *ensures $p_r \to e$.*

Proof. When a (group of) dynamic obstacle(s) modifies the Voronoi cell, it may happens that the vehicle is locally pushed away from the way-point. In the worst case, this negative phenomenon can occur infinitely often in a finite time (e.g. human adopts a pursuit policy). With the sentence "the dynamic obstacles in the scene do not play against the robot" we denote the situation in which this is not the case, that is the interactions with the dynamic obstacles happens within a finite interval (i.e. a dwell time). Since the human beings are finite in number, by means of the previous Theorems the proof is readily derived. □

It has to be noted that at least one way-point should remain in the visible set of the robot when it is pushed away: if it is not the case a replanning of the way-points is needed.

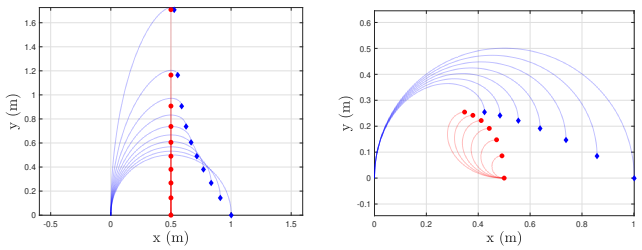
D. Safety guarantees

The safety provision is considered for a static and a dynamic (e.g. a pedestrian) separately. For static obstacles, safety is guaranteed by design (i.e., the robot is constrained to follow a point belonging to its visibility set) providing that the control parameters (κ, k_b, k_a and ψ) ensure a responsive behaviour (i.e., they depend on v^D and ρ). For dynamic obstacles (humans, in our application), since their velocities are not controllable by the robot, safety guarantees cannot be given in general. However, if the dynamic obstacle is not in pursuit of the robot when $\|p_r - p_h\| = R_{rh}$, i.e. if its velocity v_h lies



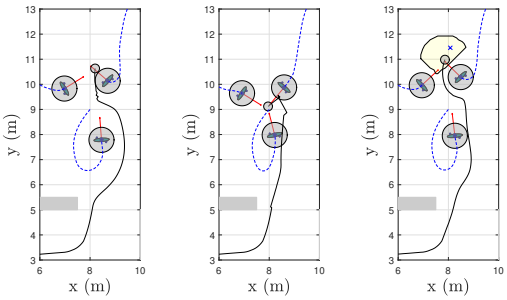
(a) (b)

Figure 3. (a) Construction of the allowed human velocity v_h space ensuring a sufficient condition for human-agent collision avoidance. (b) $d_{\min}$ as a function of the human velocity v_h and occupancy radius r'_h for the case of human constant dynamics.



(a) (b)

Figure 4. Trajectories of human (red circle) and robot (blue diamond) for safety margin $r'_h = 0.5$ m and increasing human velocities v_h: (a) constant human dynamics, (b) human in pursuit dynamics. According to Figure 3-b, when $\xi \to 1$, the collision cannot be avoided in the worst case scenario.



(a) (b) (c)

Figure 5. Trajectory (black solid) traveled by a single agent (grey circle) in a human shared environment. Red arrows represent the noisy human being sensed velocities. Dashed blue lines are the past human beings trajectories (human beings follows a pursuing behaviour). (a) APF, (b) VO and (c) proposed approach.

on the right half plane of Figure 3-a, then safety is guaranteed because the agent's velocity is directed towards the cell's centroid that belongs to the cell $\mathcal{F}$ (left semicircle in Figure 3-a). Notice that this is actually a sufficient condition for safety, thus if the human violates this condition it does not imply collision. In practice, we are assuming that the human shares the responsibility for the collision avoidance. Nevertheless, if the velocity of the robots is greater than that of human (v_h), the robot takes all the responsibility for the avoidance manoeuvre (e.g., this is always the case for the simulations in Section IV-A and V-A).

The safety condition on the relative velocity between the robot and the human thus proposed can be relaxed if we assume that the safety margin R_{rh} in (7) is computed on $r'_h > r_h$, i.e., if we inflate the strict occupancy of the pedestrian with a larger radius, while if $||p - p_h|| < R_{rh}$ the safety margin is computed on $||p - p_h||$. In this case, with reference to Figure 3-a, we observe that the worst velocity direction (in terms of safety) for the robot belongs to the straight edge of the cell $\mathcal{F}$. Given the system dynamics (1), this corresponds to $\theta = \alpha - \pi/2$, where $\alpha = \arctan(\Delta_y/\Delta_x)$, $\Delta_x = x - x_h$ and $\Delta_y = y - y_h$. For convenience, the relative position between the vehicle and the pedestrian can be rewritten in polar coordinates, with the origin being located on the human, i.e. $d = \sqrt{\Delta_x^2 + \Delta_y^2}$ and α, coming up with the following:

$$[\dot{d}, \dot{\alpha}]^T = \left[-v_h \cos(\alpha - \beta), -\frac{v + v_h \cos\alpha}{d}\right]^T, \qquad (13)$$

where β is the human orientation angle. We further notice that the robot has to reach its way-point wp^{k+l}, thus the worst case velocity selection (i.e., belonging to the straight edge of the cell $\mathcal{F}$, see Figure 3-a) is maintained for at most a semicircle. Indeed, at most along one complete semicircle, the robot will finally depart towards the way-point changing its direction (hence the velocity no more will belong to the straight edge of the cell $\mathcal{F}$). The minimum distance $d_{\min}$ between the robot and the pedestrian can then be computed integrating (13), which is a function of r'_h and $\xi = \frac{v_h}{v}$ and is shown in Figure 3-b. Moreover, we report the trajectories of the robot (along a semicircle) and of the pedestrian with two different behaviours: constant dynamics (Figure 4-a) and a pursuing (dynamics 4-b). In Section V we will see that in the multi-agent case the safety condition is always satisfied between agents, while in Section IV-B, we validate these safety features through experiments on the field.

IV. Single Agent Results

In this Section, we first propose a comparative analysis with respect to some solutions available in the literature and then an actual application of the method is discussed.

A. Comparison with the state-of-the-art

A comparative analysis with respect to two well known reactive solutions, which may also have a multiagent extension, is firstly presented. In Figure 5, the qualitative comparison is reported when three human beings have a pursuing behaviour. The artificial potential functions (APF) method (Figure 5-a, [32]), the velocity obstacle (VO) approach (Figure 5-b, [10]), and the proposed solution (Figure 5-c) are reported in the same scenario. For the sake of simplicity, we consider a single agent having a holonomic kinematic model. We assume here, to mimic an actual situation, that the velocities of human beings are estimated with a white noise distributed as a $\mathcal{N}(0, \sigma_v^2)$, where σ_v^2 is 10% the actual human velocity (the red arrows in Figure 5 represent the velocities affected by the measurement noise). Let all the pedestrians move with the same velocity v_h and v be the robot velocity, the desired robot velocity v^D in (11) is set to $3v_h$. It can be noticed how the VO solution does not work properly (i.e. trajectory is jerky and possibly unsafe)

Table I
QUANTITATIVE COMPARISON FOR THE TRAJECTORIES GENERATED BY APF, VO AND THE PROPOSED ALGORITHM, IN 4 DIFFERENT SCENARIOS. WE REPORT THE PATH LENGTH l, THE TIME TO REACH THE GOAL POSITION t, THE CURVATURE OF THE PATH AND ITS DERIVATIVE κ AND $\dot{\kappa}$.

-	APF				VO				Lloyd-based			
scen. #	1	2	3	4	1	2	3	4	1	2	3	4
l (m)	12.7	12.5	13.8	12.4	11.9	11.6	**11.0**	15.5	**11.8**	**11.4**	11.2	**11.6**
κ $(\text{m})^{-1}$	2.83	0.16	>100	0.18	6.58	>100	0.45	>100	**0.18**	**0.06**	**0.05**	**0.15**
$\dot{\kappa}$ $(\text{ms})^{-1}$	0.67	0.53	>100	0.47	1.11	25.18	0.50	10.64	**0.56**	**0.45**	**0.39**	**0.39**
t (s)	4.75	4.69	5.11	4.65	4.78	5.64	4.92	6.83	**4.45**	**4.32**	**4.26**	**4.39**

because of the noise in the v_h variable (Figure 5-b). The APF solution appears instead to be better (Figure 5-a), however it shows oscillatory behaviours, not to mention the relatively high number of parameters to be tuned in the algorithm and the high sensitivity of the solution to parameter variations. The proposed solution, instead, generates good trajectories in terms of smoothness, path length and safety (Figure 5-c). In order to validate quantitatively the effectiveness of our approach we report in Table I an analysis using as metrics the path length l, the time to reach the goal position t, the average path curvature κ and the average normalised curvature derivative $\dot{\kappa}$.
We report the results for four different scenarios with random robot initial and goal position, human beings positions and velocities. The bolded values in Table I indicates the highest performance. It is evident how the proposed approach performs better than the other two with respect to the chosen metrics.

B. Experimental Results

The experiments with the single agent navigation algorithm in two different scenarios are here discussed to prove the effectiveness of the solution in an actual scenario also with nonholonomic constraints. The experiments have been carried out in the DISI Department at the University of Trento. The robot localises itself through its encoders and through a RPLIDAR-A3, with the latter being also used to detect static and dynamic obstacles. In the first scenario, depicted in Figure 6-a, the avoidance manoeuvre in a corridor where the robot and a human being are moving in opposite directions (frontal case). In this scenario, the human being facilitates the avoidance manoeuvre by swerving to one side; as a result the robot can easily follow the desired heading without major braking actions (Figure 6-b). The second scenario, illustrated in Figure 6-c, considers the case of a human being walking alongside the robot in a corridor. In this experiment the human has a clearly adverse behaviour, thus the robot must brake several times to avoid collisions, as clearly visible from Figure 6-d. In Figure 7, 8 snapshots of an experiment with multiple humans in the scene are reported. The video material accompanying this paper provides additional meaningful experiments. As a final remark, the algorithm, implemented on a Nvidia Jetson TX2 hardware mounted on the robot, computes the control inputs with an average computation time of 20 ms. The computation time increases with the selected agent's cell dimension r_s, which in this case is set to be equal to 1 m: by increasing the range up to 1.5 m, the computation time reaches 30 ms. Higher ranges would increase the computation time as well, even though the desired reactive behaviour has by definition a limited area of interest [33].

V. MULTIAGENT EXTENSION

In this Section, we propose a natural extension of the navigation strategy presented in Section III for the single agent. As the solution built upon (9), (10) and (11) deals with dynamic obstacles, a quite straightforward extension is just to apply those rules to all the n agents in the team. However, the idea is pushed even further to derive a completely distributed approach addressing the *group cohesion*. A first important step, is to introduce a limited communication range, which is set to $r_{c,i} = cr_{s,i}$, where $c > 2$ and $i = 1, \ldots, n$. Requiring the communication range to be twice as much as the sensing range is, in our evaluation, a realistic assumption.
An important point is that our notion of group cohesion bears a close resemblance with the cohesion of a group of humans rather than with that of a classic "lattice" formation of robots. Therefore, moving cohesively means to establish and maintain a social-like link among the team agents, which we believe is another important contribution of this work. Considering n robots, we assume that all the starting $\mathcal{S} = \{s_1, s_2, ..., s_n\}$ and ending $\mathcal{E} = \{e_1, e_2, ..., e_n\}$ positions allow a connected communication graph between the agents. The first step to confer cohesion to the group is to compute its average position (i.e., the centroid of the group). This can be made by using distributed techniques such as the average consensus algorithm based on the Metropolis-Hastings approach [34]. In this scheme, all the agents have an estimate of $p_c = \sum_{i=1}^{n} p_{r,i}/n$, denoted as $\hat{p}_{c,i}$. Along the trajectory, the robots are constrained in a region defined by a circular area (but any convex shape can be considered) with radius R_{coh}, computed for the i–th agent as $\mathcal{U}_i = \{q \in \mathcal{Q} | \, ||q - \hat{p}_{c,i}|| < R_{\text{coh}}\}$. Notice that R_{coh} can be regarded as a parameter governing the group compactness. In order to secure the cohesion of the group, it is sufficient to use the set $\mathcal{F}_i = \mathcal{U}_i \cap W_{r_{s,i}} \cap \tilde{V}_i$, where $\tilde{V}_i$ is the Voronoi partition of the i–th agent computed as in (7). Figure 8 depicts such a partition for the i-th agent.
The socially-aware flocking not only drives the agents in a non-constrained but cohesive configuration (as pedestrians do) while avoiding obstacles and human beings, but also adapts agents behaviour according to neighbours and the environment. To this end, we introduce the convex weigthed Voronoi diagram (CWVD) by taking inspiration from the power weighted Voronoi diagram (PWVD) and the multiplicatively weighted Voronoi diagram (MWVD) (see [35], [36]). The main novelties here are: 1. the cells preserve their convexity, which is not true for PWVD and 2. the agent moves inside the cell, which is not valid for MWVD. We define the CWVD as

$$\mathcal{H}_i = \left\{ q \in Q | w_x \cos\alpha_{ij} + w_y \sin\alpha_{ij} < \frac{r_{s,i}}{r_{s,i} + r_{s,j}} \left\| p_i - p_j \right\| \right\} \tag{14}$$

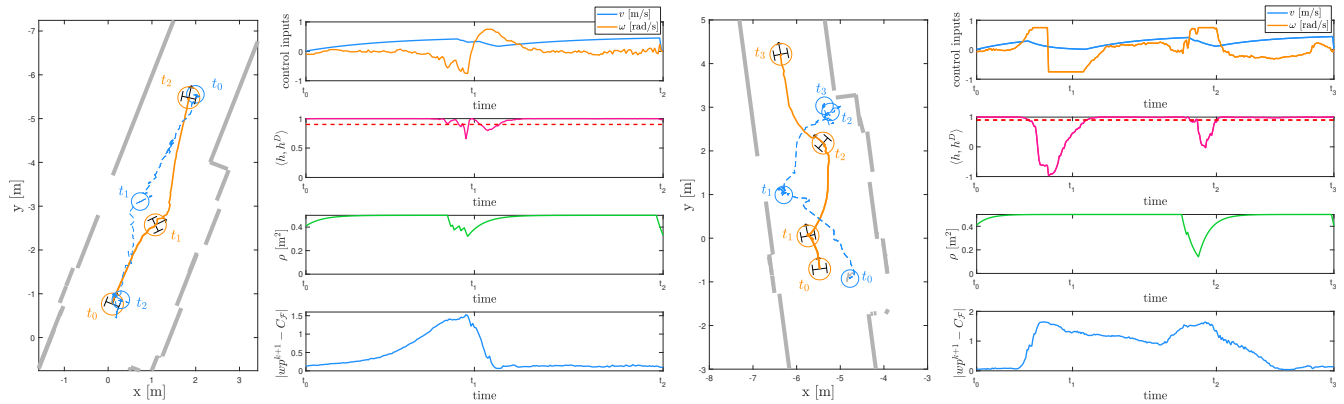


Figure 6. In (a) and (c) we show respectively three and four time instants (denoted with t_i) of the single agent navigation in a cluttered environment. The grey thickest solid lines are the static obstacles detected by the LIDAR sensor, the solid orange curves are the robot paths and the dashed blue curves are the human trajectories. In (b) and (d) the control inputs in (1), the heading error in (11) with the dashed threshold value $\cos\psi$, the spread factor (12) and the way-point distance $||wp^{k+l} - C_{\mathcal{F}}||$ are reported.

Figure 7. Example of an experiment in the corridor with multiple human beings with 8 snapshots in chronological time sequence from t_0^1 (upper left) to t_7^2 (lower right), where t_i^j indicates the photo taken by camera j at the i–th instant of time.

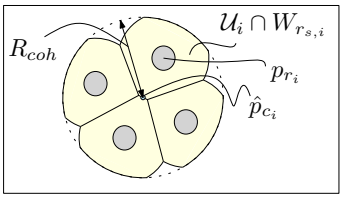


Figure 8. Modified Voronoi tessellation $\mathcal{U}_i \cap W_{r_{s,i}}$ for the cohesion purpose of a group with four agents. The small faded circles are the agents of the group in positions $p_{r,i}$, $\forall i = 1 \dots 4$.

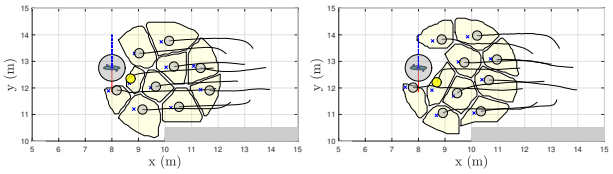
where $w = q - p_i = [w_x, w_y]^{\mathrm{T}}$, and $\alpha_{ij} = \text{atan}\left(\frac{y_j - y_i}{x_j - x_i}\right)$. With the following adaptive law

$$\dot{r}_{s,i} = \begin{cases} -\left(r_{s,i} - \underline{r}_{s,i}\right) & \text{if } \dim\left(\mathcal{Y}_i\right) \geq \dim\left(\frac{1}{n_{\mathcal{V}_i}} \sum_{j \in \mathcal{V}_i} \mathcal{Y}_j\right) \\ -\left(r_{s,i} - \overline{r}_{s,i}\right) & \text{if } \dim\left(\mathcal{Y}_i\right) < \dim\left(\frac{1}{n_{\mathcal{V}_i}} \sum_{j \in \mathcal{V}_i} \mathcal{Y}_j\right) \end{cases}, \quad (15)$$

where $\underline{r}_{s,i} > 0$ and $\overline{r}_{s,i} < R_{s,i}$ are the lower and upper bounds for $r_{s,i}$, the set $\mathcal{V}_i$ indicates the neighbours of the i–th agent and $n_{\mathcal{V}_i}$ its number, and the set $\mathcal{Y}_i = \mathcal{U}_i \cap \mathcal{H}_i \cap W_{r_{s,i}}$, we can easily change the CWVD $\mathcal{H}_i$ for the i-th robot. In particular, the role of (15) is to balance the cell dimensions inside the cohesive flocking region, thus giving an equal share to each agent. This way we enforce a "social" behaviour that tends to ensure an equal mobility to all the agents. This is of major relevance, since more natural and comfortable trajectories can be generated. We also adapt the cohesion shape to the environmental conditions: when $\|p_{r,i} - \hat{p}_{c,i}\|$ exceeds $R_{\text{coh},i} - \epsilon$, the cohesion radius is adapted to $R_{\text{coh},i} = \|\hat{p}_{c,i} - p_i\| + \epsilon$; it allows the most obstructed agents to continue to live inside their cells, and then to reconnect the group.

A. Simulation Results

Due to the lack of an adequate number of mobile agents, the proposed approach has been extensively tested only in simulations. We mainly compare quantitatively the multi-agent flocking algorithm with constant and adaptive ranges $r_{s,i}$. We select as performance $\mathcal{I} = \sum_i^n \left(\dim(\mathcal{Y}_i) - \dim(\bar{\mathcal{Y}})\right)^2$, which is indicative of how much the space inside the cohesion hull is equally divided among the agents, on both adaptive $\mathcal{I}_{\text{a}}$ (when (15) is adopted) and non-adaptive $\mathcal{I}_{\text{na}}$ cases. The simulations in complex environments with static and dynamic obstacles always result in $\mathcal{I}_{\text{na}} \gg \mathcal{I}_{\text{a}}$. In Figure 9 we provide a qualitative comparison for the two approaches; in the non-adaptive case (Figure 9-a), when the team gets close to a human being, the robot that is closer has very little mobility, as testified by the small available cell. In this case, instead, it should have a large cell since it is impaired by the presence of the pedestrian: adopting the adaptive solution, the same robot



(a) (b)

Figure 9. Two snapshots, at the same time instant, with the non-adaptive (a) and adaptive (b) adaptation rule (15). The solid lines are the agents trajectories followed from the starting positions, the dashed lines are the human past trajectories and the blue crosses are the centroids of the cells.

has a larger region, while the other team members reduce their mobility freedom accordingly (Figure 9-b), thus implementing the social cooperative behaviour. Finally, the comparison with APF and VO reported in Section IV-A is not carried out in the multiagent case due to space limits, however the advantages of the proposed solution are similar also in this case, mainly due to the built in social compliance.

VI. Conclusions

We have proposed the application of Lloyd algorithm as a means to navigate a ground robot from a starting position to a goal. Our navigation solution is offered in two different flavours. For the single agent case, the robot is guaranteed to safely reach its final destination avoiding both fixed and moving obstacles, unless the latter behave adversely and impede on purpose the progress of the robot. For the multiagent case, not only can our solution guarantee progress toward the goals and collision avoidance for the entire group, but it also secures cohesion to the group and a cooperative behaviour between the agents in the team. Our notion of group cohesion is quite different form the standard definition of robot formation, since our robots are not constrained to a stiff formation lattice but they can move inside a cohesion area. The cooperation between agents is the result of local interaction which tends to distribute agents' mobility equally among the group. An extension of the work, planned for the near future, will be implement the multiagent algorithm on real robotic platforms.

References

[1] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous robot vehicles*. Springer, 1986, pp. 396–404.

[2] L. Lapierre, R. Zapata, and P. Lepinay, "Combined path-following and obstacle avoidance control of a wheeled robot," *The International Journal of Robotics Research*, vol. 26, no. 4, pp. 361–375, 2007.

[3] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE transactions on robotics and automation*, vol. 7, no. 3, pp. 278–288, 1991.

[4] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 2. IEEE, 2000, pp. 995–1001.

[5] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime motion planning using the rrt," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1478–1483.

[6] F. Lingelbach, "Path planning using probabilistic cell decomposition," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 1. IEEE, 2004, pp. 467–472.

[7] O. Takahashi and R. J. Schilling, "Motion planning in a plane using generalized voronoi diagrams," *IEEE Transactions on robotics and automation*, vol. 5, no. 2, pp. 143–150, 1989.

[8] S. Garrido, L. Moreno, M. Abderrahim, and F. Martin, "Path planning for mobile robot navigation using voronoi diagram and fast marching," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 2376–2381.

[9] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[10] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[11] P. Bevilacqua, M. Frego, D. Fontanelli, and L. Palopoli, "Reactive Planning for Assistive Robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1276–1283, April 2018.

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

[13] C. W. Reynolds, *Flocks, herds and schools: A distributed behavioral model*. ACM, 1987, vol. 21, no. 4.

[14] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on automatic control*, vol. 51, no. 3, pp. 401–420, 2006.

[15] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Distributed Autonomous Robotic Systems*. Springer, 2013, pp. 203–216.

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

[17] D. Zhou, Z. Wang, S. Bandyopadhyay, and M. Schwager, "Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1047–1054, 2017.

[18] A. Sud, E. Andersen, S. Curtis, M. C. Lin, and D. Manocha, "Real-time path planning in dynamic virtual environments using multiagent navigation graphs," *IEEE transactions on visualization and computer graphics*, vol. 14, no. 3, pp. 526–538, 2008.

[19] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[20] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[21] M. Boldrer, D. Fontanelli, and L. Palopoli, "Coverage Control and Distributed Consensus-Based Estimation for Mobile Sensing Networks in Complex Environments," in *Proc. IEEE Int. Conf. on Decision and Control (CDC)*. Nice, France: IEEE, Dec. 2019, accepted.

[22] M. Lindhé, P. Ogren, and K. H. Johansson, "Flocking with obstacle avoidance: A new distributed coordination algorithm based on voronoi partitions," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 1785–1790.

[23] I. Karamouzas and M. Overmars, "Simulating and evaluating the local behavior of small pedestrian groups," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 3, pp. 394–406, 2011.

[24] F. Farina, D. Fontanelli, A. Garulli, A. Giannitrapani, and D. Prattichizzo, "Walking Ahead: The Headed Social Force Model," *PLOS ONE*, vol. 12, no. 1, pp. 1–23, 1 2017.

[25] L. Palopoli *et al.*, "Navigation Assistance and Guidance of Older Adults across Complex Public Spaces: the DALi Approach," *Intelligent Service Robotics*, vol. 8, no. 2, pp. 77–92, 2015.

[26] M. Andreetto, S. Divan, F. Ferrari, D. Fontanelli, L. Palopoli, and F. Zenatti, "Simulating passivity for Robotic Walkers via Authority-Sharing," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1306–1313, April 2018.

[27] F. Ferrari, S. Divan, C. Guerrero, F. Zenatti, R. Guidolin, L. Palopoli, and D. Fontanelli, "Human–Robot Interaction Analysis for a Smart Walker for Elderly: The ACANTO Interactive Guidance System," *International Journal of Social Robotics*, June 2019.

[28] "ACANTO: A CyberphysicAl social NeTwOrk using robot friends," http://www.ict-acanto.eu/acanto, February 2015, EU Project.

[29] S. Bhattacharya, N. Michael, and V. Kumar, "Distributed coverage and exploration in unknown non-convex environments," in *Distributed autonomous robotic systems*. Springer, 2013, pp. 61–75.

[30] M. Boldrer, M. Andreetto, S. Divan, L. Palopoli, and D. Fontanelli, "Socially-aware Reactive Obstacle Avoidance Strategy based on Limit Cycle," *IEEE Robotics and Automation Letters*, 2020.

[31] J. P. Hespanha, H. J. Kim, and S. Sastry, "Multiple-agent probabilistic pursuit-evasion games," in *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304)*, vol. 3. IEEE, 1999, pp. 2432–2437.

[32] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[33] G. Ferrer, A. G. Zulueta, F. H. Cotarelo, and A. Sanfeliu, "Robot social-aware navigation framework to accompany people walking side-by-side," *Autonomous robots*, vol. 41, no. 4, pp. 775–793, 2017.

[34] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.* IEEE, 2005, pp. 63–70.

[35] F. Aurenhammer, "Power diagrams: properties, algorithms and applications," *SIAM Journal on Computing*, vol. 16, no. 1, pp. 78–96, 1987.

[36] A. Kwok and S. Martinez, "Deployment algorithms for a power-constrained mobile sensor network," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 20, no. 7, pp. 745–763, 2010.