POLITECNICO DI TORINO

Master's Degree in Computer Engineering

MASTER THESIS

Human-Aware Robot Navigation around Groups in Narrow Spaces

at KTH Royal Institute of Technology

Supervisors

Prof. Marina Indri

Prof. Iolanda Leite (KTH)

Prof. Patric Jensfelt (KTH)

Candidate

Angelo Briamonte

Academic Year 2020-2021

Alla mia famiglia,
per avermi sempre incoraggiato e supportato

Abstract

The technological advancement in the field of Robotics and related areas is leading to the increasing deployment of artificial agents not only in factories but also in offices, hospitals, airports and, more recently, also in homes. One of the problems that robots face when deployed in environments with people is how to perform the so-called human-aware navigation, i.e., how to move around the environment complying with people's social conventions so that the humans around them feel safe and comfortable.

In this regard, the social conventions typically followed by humans include keeping a certain distance from each other and not invading the space shared by the people gathered in a group. This thesis aims to develop and evaluate a robotic navigation framework that generates paths satisfying these two constraints. To achieve this goal, we developed a social model to represent both single humans and groups. In particular regarding group formations, we designed a geometric model able to detect, classify, and represent them adequately in the space. We integrated this social model with a state-of-the-art global path planner, obtaining an overall human and group aware navigation framework.

We evaluated the system through a test set reproducing the typical structures of human groups, and then compared our resulting framework with the used state-of-the-art planner in terms of both performance and sociability. The results of the simulations confirm that, with the developed geometric model, it is possible to identify the formations of people in the environment, and consequently generating group-aware trajectories which result in higher levels of sociability.

Sammanfattning

Den tekniska utvecklingen inom området robotik och relaterade områden leder till ett ökat användande av artificiella agenter, inte bara i fabriker utan också på kontor, sjukhus, flygplatser och, nyligen, också i hem. Ett av problemen som robotar möter när de används i miljöer med människor är hur de kan röra sig runt miljön i enlighet med människors sociala konventioner så att människorna omkring dem känner sig säkra och bekväma.

De sociala konventionerna som normalt följs av människor inkluderar att hålla ett visst avstånd från varandra och inte röra sig in i utrymmet som delas av människorna samlade i en grupp. Detta examensarbete syftar till att utveckla och utvärdera en robotnavigeringsmetod som genererar vägar som uppfyller dessa två begränsningar. För att uppnå detta mål utvecklade vi en social modell som representerar både enskilda människor och grupper. I synnerhet när det gäller gruppformationer, designade vi en geometrisk modell som kan upptäcka, klassificera och representera dem i rummet. Vi integrerade den sociala modellen med en modern global vägplanerare och kunde på så sätt realisera ett navigationssystem kapabelt att hantera både enskilda människor och grupper av människor.

Vi utvärderade systemet genom en simulering som reproducerade de typiska strukturerna för mänskliga grupper och jämförde vårt med de bästa existerande metoderna när det gäller både prestanda och socialt hänsynstagande. Resultaten av simuleringarna bekräftar att det med den utvecklade geometriska modellen är möjligt att identifiera människor i miljön och därmed generera gruppmedvetna rörelsemönster som resulterar i högre nivåer av socialt hänsynstagande.

Acknowledgments

I would first like to thank Prof. Marina Indri (Politecnico di Torino) for supervising this thesis from Italy, for her useful feedback, and for her availability.

I would like to express my most sincere gratitude to Prof. Iolanda Leite (KTH) for the continuous support that I received from her. I really appreciate all the insightful feedback and suggestions that Prof. Leite has given me on every step of the process and for giving me the opportunity and privilege to work in her group.

I would also like to acknowledge Prof. Patric Jensfelt (KTH) for accepting being the examiner of my thesis at KTH and for his suggestions that allowed me to develop the best version of this thesis.

Finally, there are no words that can express how grateful I am to my family and friends.

First of all, I would infinitely like to thank my parents. Thank you for giving me the opportunity to continue my studies, for always supporting me (despite my constant worries and anxieties), for encouraging and motivating me, and for always believing in me ... I will always be grateful to you.

I would like to thank my brother Peppe, for his advice and his help whenever I needed it, but above all for always being an example to follow for me.

Special thanks go to my grandparents who taught me and passed on the values of family, love and life, making me the person I am today.

I would like to thank my uncle Rocco, for always being there during these five years and for believing in me.

I would like to thank all my friends: from those, I hold in my heart from childhood to those I met on my journey and who have decided to walk by my side and accompany me here. To Francesca and Marialucia, thank you for always being there and for teaching me the value of friendship. To Gigi, for the example of strength, tenacity, and life that he gave me.

I also thank my family of Collegio Einaudi: Pastore, Rik Rik, Napoli, il Maestro, Ritina and Simo. Thank you for sharing with me over the years a journey as tiring as it is unforgettable, full of worries and exams but also joys and gratifications.

Finally, I would like to thank Andrea, known during the Erasmus project in Stockholm, for sharing this wonderful experience with me but, above all, for the beers drunk together.

Contents

Chapter 1

Introduction

1.1 Problem description

In recent years, continuous research in the field of robotics and automation has led to the introduction of assistance robots not only in factories but also in domestic environments, such as offices and hospitals [1]. Despite a large number of works in this regard, there are still many open problems, including the human-safe navigation problem [2], that is, the property of a robot moving autonomously in the environment taking the presence of humans into account.

The use of robots in everyday situations, indeed, requires that these agents should be able to move in the environment satisfying a multitude of constraints that go beyond the performance and the efficiency of motion. Robots should respect the so-called *human comfort* and navigate as much as possible respecting the social norms used by people in navigation [3] (see Chapter 3 for further discussion). Consequently, to achieve these goals, the agent should navigate taking into account the current positions of people, their possible future positions, as well as the social conventions existing between them. In other words, it should behave in a *human-appropriate* fashion. This represents a significant aspect for the future deployment of robots in daily life because only if people are safe and feel safe, they could accept to share their space with these new entities.

As mentioned above, robots should explicitly consider social conventions that govern human navigation. This means respecting several constraints, such as not getting too close to humans, not crossing groups of people (Fig. 1.1) and moving to the right (or left, depending on the social conventions of the specific Country) side of a hallway when a person is moving in the opposite direction, just to mention some of them (Section 3.3 will expand this topic,

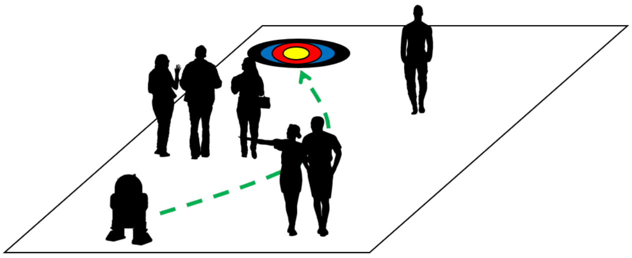


Figure 1.1: Human-aware navigation taking human groups into account.

explaining some of the existing works that address these aspects).

This work investigates the ability of an agent to perform autonomous human-aware and, specifically, group-aware navigation in narrow spaces. In other words, it analyzes the ability of an agent to move around the environment respecting the personal space of individuals but also the space shared by people when they interact with each other.

However, to satisfy these requirements (together with all the other social conventions), it is not enough to simply employ the navigation techniques traditionally used in dynamic environments, i.e., obstacle avoidance methods [4]. The movement resulting from the application of a static path planner with an obstacle avoidance module would be unnatural and difficult to predict by humans. Furthermore, it would be quite difficult to incorporate high-level social conventions into local modules. As for the other option of using a standard path planner, therefore leaving humans the task of avoiding the robot by changing their own poses/paths, it does not seem to be a sufficiently suitable choice for human safety. For these reasons, the approach typically used in social-aware navigation consists of integrating the social conventions in the high-level planner of the robot navigation framework, considering humans as special entities and not simply as moving obstacles.

Based on these considerations, this thesis investigates the ability of an agent to perform autonomous human-aware and group-aware navigation from a global path planning perspective.

1.2 Purpose and goals

In order to answer the research question stated above, this work proposes an A-star based path planner that enables robots to navigate in an efficient and socially acceptable way in narrow spaces, which are very recurrent in daily life, for example hallways of offices or hospitals. In particular, the social conventions that will be modeled and integrated into the navigation framework concern the distance a robot needs to keep from people (the theoretical model will be presented in Section 2.1) and the respect for human groups (the models will be described in Section 2.2), in the static and dynamic cases.

1.3 Contribution

Human-aware navigation frameworks that consider the presence of humans in the environment have been widely proposed in the literature, both in the static and in the dynamic case [2][3]. However, to the best of our knowledge, only a limited number of works take the presence of human groups into account, and none of them explicitly consider narrow environments, such as hallways.

In this regard, this work introduces a new geometric model to represent human formations. Consequently, by combining this model with an existing A*-based global planner, this work proposes an overall human/group-aware path planner easily tunable for the application in different narrow environments (which are typically populated by a low number of people).

1.4 Ethics, Sustainability and Societal Aspects

This work belongs to the field of human-aware robot navigation, which aims to study and improve the interaction between people and robots while moving around the environment. The final goal of this branch of robotics is to allow the future employment of these new autonomous agents in everyday environments.

This leads to ethical issues relating, in particular, to the safety in the use of robots in the society of the future. In other words, the deployment of a robot in the public spaces of hospitals, offices or homes raises questions about the reliability of autonomous navigation solutions, or, stated differently, about the safety of people who share the same environment of the robot. What happens if a robot collides with a person, and the person is injured? Who is responsible for this? Who should pay the injured person? These are just some of the questions to be answered to allow the future deployment of robots in daily life.

From an economic sustainability point of view, a robot equipped with several skills, including the ability to move around the environment in a human-aware way, could be used in a multitude of working and non-working scenarios, with positive effects on economic growth. Once these service robots have been purchased, indeed, the buyers not only do not have to face additional costs but can also use them for different applications. In the case of an office service robot, for example, it can be used to perform various trivial tasks, allowing employees to focus on higher value-added activities.

If, on the one hand, the use of these autonomous agents allows the economic growth of companies and, in general, higher savings for users, on the other hand, their use could reduce the number of job opportunities. In this regard, however, it should be highlighted that the employment of these artificial agents in society has the purpose of supporting human activity rather than replacing it.

1.5 Outline

The rest of the thesis is structured as follows. Chapter 2 provides the background theory, introducing the *Proxemics* and the human-formation social models. It also briefly introduces the motion planning problem. Chapter 3 provides an overview of the work done before in the area, dividing the existing works into three macro-requirements for social-aware navigation. Chapter 4 describes the A*-based planner that we will use as the basic framework. In addition, starting from the theoretical social models introduced in Chapter 2, the mathematical models are designed and, subsequently, integrated into the planner. In Chapter 5, the obtained group-aware planner is tested in simulation and the results are analyzed. In particular, the resulting planner is first evaluated in a set of use cases and then compared with the basic framework. Finally, Chapter 6 concludes this work discussing the main results before presenting possible future work.

Chapter 2

Background

In this chapter, some basic concepts that will be used throughout the thesis are presented. Specifically, Section 2.1 introduces the concept of human personal space and Section 2.2 focuses on human arrangements in space. These two social concepts will be implemented in mathematical models in Chapter 4. Finally, Section 2.3 presents the path planning problem, providing a basic definition limited to the context of this work.

2.1 Human personal space

One of the necessary factors to consider in social navigation is respect for the personal space of the human. It is easy to understand, indeed, how people tend to self-distance each other according to the particular context.

The first scholar to formalize this concept in a mathematical model was the anthropologist Edward T. Hall, who proposed the *Proxemics* framework in 1966 [5]. Based on this model, the space around a person can be divided into four concentric regions, corresponding to four distance values:

- *intimate* distance: this ranges from a few centimeters up to 45 cm;
- *personal* distance: this ranges from 46 cm to 1.2 m;
- *social* distance: this ranges from 1.2 m to 3.6 m;
- *public* distance: this ranges from 3.6 m to 7.6 m or more.

The corresponding regions of space, in increasing order, are (Fig. 2.1):

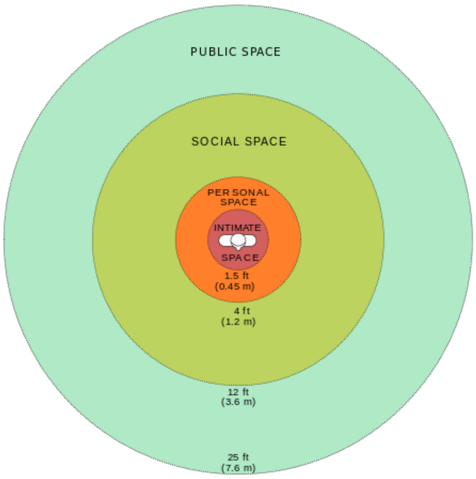


Figure 2.1: *Proxemics* framework proposed by Hall [6].

- *intimate space*: the interactions within this space indicate a close relationship between participants and can include physical contacts, as in the case of hugs or whispers;
- *personal space*: in this space, the interactions between family members or close friends occur. Typically, the smaller the distance the closer the relationship between the participants;
- *social space*: this space is dedicated to interactions with acquaintances, colleagues or strangers;
- *public space*: the interactions in this space are typically one-way interactions, such as lectures, speeches, theatrical performances and, in general, one-to-many interactions.

It is important to note that these subdivisions are not fixed, but can be influenced by several factors such as ethnicity, culture, gender, age, the method of approach, the human state (standing or sitting) as well as the dimensions of the environment itself [7][8]. Furthermore, in the case of human-robot interaction, parameters such as the physical dimensions of the robots, e.g. the height, and the familiarity of operating with them, are also relevant aspects in determining the dimensions of these subspaces [9]. Finally, in the case of moving people, the speed and direction of motion can also contribute to determining the shape and size of the human personal space.

2.2 Arrangement of people in space when they form a group

The way people orient and space themselves while interacting with each other is a topic studied since the 1960s. It, indeed, involves numerous disciplines, such as psychology, sociology, design, electronics, robotics, etc. In the specific case of robotics, understanding how people place themselves in groups is crucial, for example, both to avoid passing through these formations and to approach them in the most natural and social possible way.

The following sections will expose two of the main existing frameworks which model human formations in space, in the static and in the dynamic case.

2.2.1 Static formation: the F-formation model

When people interact with each other, very often they enter into a distinctive spatial-orientational arrangement, which is jointly maintained. Therefore, if a person changes his or her pose, the others adjust their position and orientation accordingly. Adam Kendon [10][11], who called these arrangements F-formations, found that their structure typically has a circle shape. In particular, he distinguished three concentric regions of space (Fig. 2.2):

- the *o-space*: this is the shared space and is dedicated to the main activity of the group;
- the *p-space*: this is the area where the bodies of the participants are located. To be considered part of the group, a person should be in this region;
- the *r-space*: this is the external space and can be considered as a buffer between F-formation itself and the external world. Before entering an existing formation, typically, a human is located in this region of space.

It is worth underlining that in an F-formation, the orientation of each participant is directed towards the central region, i.e., towards the *o-space*.

Moreover, in groups with just two people, a further distinction is possible. Depending on the intent or the topic of the communication itself, three configurations can occur:

- *L-shape*: in this configuration, the directions of the two participants are perpendicular to each other (Fig. 2.3-a). This situation typically occurs

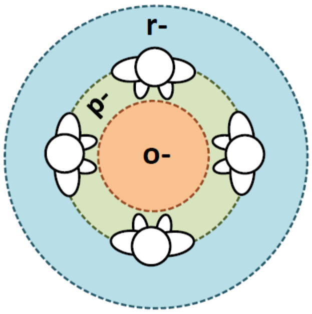


Figure 2.2: Structure of the three spaces of an F-formation: starting from the centre, O-space, P-space and R-space [12].

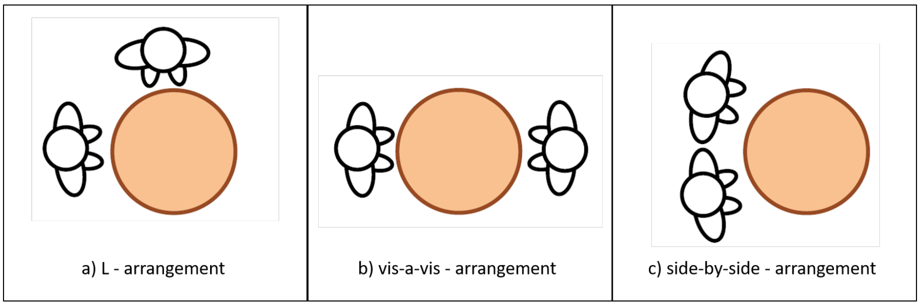


Figure 2.3: The three standard configurations of an F-formation with two people [12].

in cooperative interactions, for example when one participant helps the other, or when the discussion topic is disembodied;

- *vis-a-vis* (*vav*): in this case, the two people are facing each other (Fig. 2.3-b). Typically, this situation occurs in competitive interactions, for example when one participant disputes the idea of the other, or when the topic of the discussion is their relationship, for example during greetings;

- *side-by-side* (*sbs*): in this configuration, the two people are close and face the same direction (Fig. 2.3-c). This scenario is typical of situations in which both participants are interested in something that is in the immediate environment.

Configurations, which are combinations of these three basic categories, are also possible.

Figure 2.4: The three standard human arrangement patterns in a dynamic group.

2.2.2 Moving human groups patterns

Unlike static formations, for which, as just described, a well-defined and globally recognized model is available, in the case of dynamic formations such a model is not yet available. Nevertheless, in recent years, several authors [13][14], analyzing crowd behaviors, have tried to extract recurring patterns also for moving groups. Three typical patterns have been identified according to human density in the environment (Fig. 2.4): *side-by-side*, *V-like* (or *U-like*, depending on whether there are three or four members in the group) and *river-like*, i.e. with people walking in line.

Furthermore, as regards the number of participants, this model reveals that, in the dynamic case, the groups are composed of a maximum of 3-4 people. Formations with more than 4 members inevitably split into smaller subgroups during their navigation.

2.3 The Path planning problem

To be able to move in the environment taking into account static and dynamic obstacles, a robot typically includes a path planning module in its navigation framework. In this regard, this section will briefly introduce the general problem of path planning, also known as motion planning or piano mover's problem, and will present some of the algorithms existing in the literature to solve it.

2.3.1 Problem definition

In order to define the problem, some basic definitions are given below:

- The Workspace *W* is the space where the moving agent exists;
- A Configuration q is a set containing all the parameters required for defining the positional state of the agent;
- The Configuration space, known as *C*-space, is the space of all possible configurations. This space is defined as a *topological manifold*;
- Given $O \in$ *W*, i.e. the set of obstacles in the workspace, and *A(q)*, i.e. the agent in configuration $q \in C$, the *Free-space* is defined as $C_{free} = \{q \in C | A(q) \cap O = \emptyset\}$.
- The *Obstacle-space* is, therefore, defined as $C_{obs} = C \setminus C_{free}$.

The **motion planning** problem can therefore be defined as the problem of finding a path that moves an agent from a starting pose to a goal pose, avoiding obstacles. Formally, it is the problem of finding a continuous path τ, such that

$$\tau : [0,1] \to C_{free}, \qquad \tau(0) = q_I, \quad \tau(1) = q_G$$

where q_I is the starting configuration and q_G is the goal configuration (Fig. 2.5).

It is important to emphasize that, in reality, the motion planning problem is broader and more complex than the problem described by the statement above. The provided definition, indeed, has the purpose of limiting the topic to the specific case of this work.

2.3.2 Path planning algorithms

In order to perform motion planning, the *C*-space needs to be discretized. Based on how discretization is performed, two categories of algorithms can be identified: *combinatorial planning* and *sampling-based planning*.

Combinatorial planning

The combinatorial-planning techniques characterize the C_{free} subspace by explicitly capturing its connectivity in a graph, called *roadmap* RM. Each vertex of this graph is a configuration in C_{free} and each edge is a collision-free path through C_{free}. In particular, the following three properties hold for the

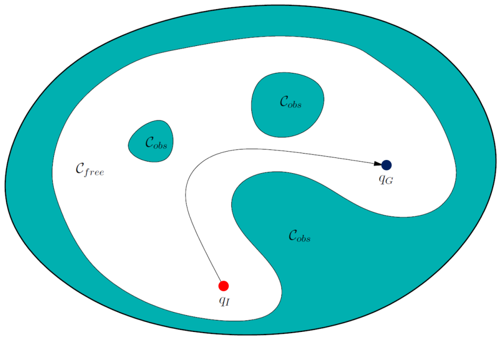


Figure 2.5: The path planning problem illustrated with the *C*-space concept. The goal is to find a path from q_I to q_G in C_{free} [15].

roadmap: there is a path between q_I and some $q_A \in RM$, there is a path between q_G and some $q_Z \in RM$ and, finally, there is a path in the roadmap between q_A and q_Z. To build a roadmap from the *C*-space, several techniques have been proposed, such as the Voronoi diagrams, the visibility graphs, and the exact and approximate cell decompositions [15].

Given the roadmap, then the path between the initial and the goal configurations can be easily obtained using a graph-based search. Without discussing the details of each algorithm, some of the search techniques commonly used in the field of path planning are the Greedy search, the Dijkstra algorithm [16], A* [17], Theta* [18], D* [19] and D* Lite [20] (see [21] for a more exhaustive list). The main advantage of these techniques is the *completeness* property, i.e. they find a solution if it exists and report a failure otherwise. On the other hand, they become quickly intractable when *C*-space dimensionality increases (*Combinatorial explosion*).

Sampling-based planning

Unlike previous approaches, sampling-based algorithms do not explicitly characterize the *C*-space, identifying C_{free} and C_{obs}, but they search for collision-free path only by sampling points in the *C*-space itself.

In general, the idea behind all these approaches consists of incrementally build a graph $G(V, E)$ (i.e. a *roadmap*), where the new vertices are extracted from the *C*-space. In more detail, these algorithms, starting from a graph containing only the initial q_I and goal q_G configurations, proceed to the expansion of the graph by sampling the *C*-space one point at a time. To ensure the admissibility of this new sample/configuration, a *Local Planner Method*, which attempts to build a collision-free path between the graph and the new point, is

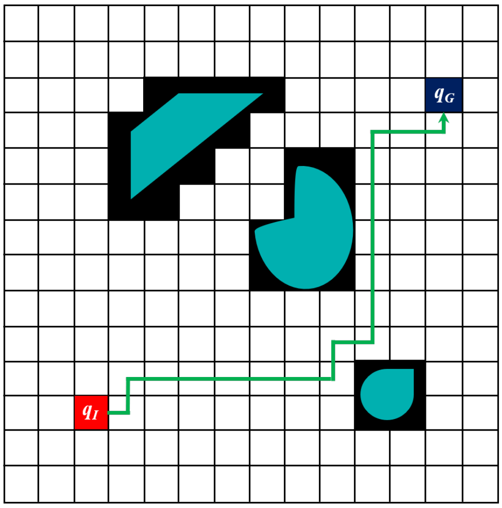


Figure 2.6: Example of combinatorial path planning. The C_{free} space is divided into smaller cells obtaining a grid map. This one can be considered as a graph and, therefore, the path from q_I to q_G can be determined by applying a graph search algorithm.

used. This process continues until a solution is reached, i.e. G contains a path connecting q_I with q_G, or some termination condition is satisfied [15] (Fig. 2.7).

Based on the specific customizations made to this scheme, then, different algorithms have been proposed over the years, such as PRM [22], OBPRM [23], RRT [24], RRT* [25] and RRT^X [26] just to mention some of them (see [27] for a more exhaustive list). These approaches are typically more efficient than the *combinatorial* ones, especially for high-dimensional *C*-spaces. However, these algorithms are not complete but are only *probabilistically complete*, i.e. the probability of finding a solution tends to 1 as time tends to infinity, and their performance typically degrades in problems with narrow passages.

It is important to emphasize that the two approaches described above do not include all algorithms existing in the literature; many of these, indeed, are based on specific ideas or concepts derived from other disciplines. Examples are the Potential Field Methods [1] [28] , which use the concept of electric charges derived from physics.

[1] Given the local nature of these methods, however, they are primarily used as obstacle avoidance techniques [4], i.e. as local planning techniques.

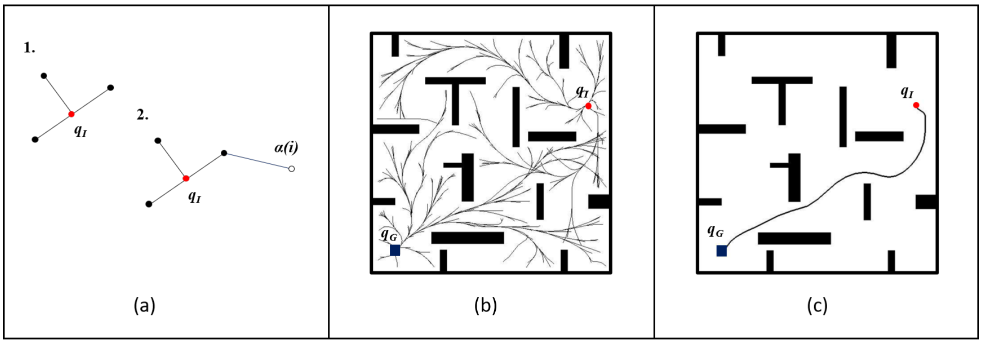


Figure 2.7: Example of sampling-based motion planning. a) Expansion step of the roadmap ($\alpha(i)$ is the sampled point). b) Roadmap resulting from several executions of the sampling process. c) Final path from q_I to q_G.

To conclude, given that combinatorial-planning techniques are more suitable for incorporating further navigation constraints (e.g. social conventions), a framework that adopts such approaches will be proposed in this work. In particular, as will be detailed in Chapter 4, an A* based solution will be used because, in addition to being complete, this algorithm is also optimal, i.e., it determines the solution with the lowest possible cost to the search problem considered.

Chapter 3

Related Work

The continuous increase and improvement of robot skills have made human-aware navigation a very active research field so that robots could shortly enter everyday environments. Furthermore, given the multiple aspects that characterize this problem, the number of approaches solving it, present in the literature, is very high. In this chapter, a short review of these techniques is presented.

In human-aware robot navigation, the robot should behave by satisfying additional requirements compared to navigating in static and unpopulated environments. These requirements could be grouped into three categories such as *comfort*, i.e. the robot should not cause fear or feelings of danger in humans, *naturalness*, that is, it should move as much as possible in a similar way to humans, and *sociability*, i.e. it should behave by obeying high-level cultural rules [3].

To perform navigation obeying these constraints, in addition to the standard navigation modules, e.g. the planning, obstacle avoidance and SLAM modules, specific components are required: modules dedicated to the prediction of the motion of dynamic obstacles or humans, which allow not only to plan a faster trajectory but also to give a greater feeling of safety to people in the same environment [29][30], modules for the selection of behavior, which attempt to increase the acceptability by reducing differences between robot and human motions [31], and modules for the selection of the pose to adopt when interacting explicitly with people [32].

The following sections will focus on the three requirements mentioned above, i.e. comfort, naturalness and sociability, presenting some of the techniques in the literature that attempt to satisfy them. Particular emphasis will be given to *sociability* as the work presented in this thesis is closer to this requirement, although the proposed categorization does not represent a clear distinction at all.

3.1 Comfort

The concept of human comfort refers to the requirement that, during robot navigation, the humans present in the environment should not only be safe but should also feel safe.

In the literature, the comfort requirement is mostly correlated to a distance a robot needs to keep from people. This is endorsed by the *Proxemics* model proposed by Hall [5], presented in more detail in section 2.1. In general terms, this work divided the area around a human into different zones of interaction, each for a specific social relation. Failure to respect these limits could disturb the emotional state of the human. The *Proxemics* model is the basis of numerous works on human-aware navigation, which, typically, use the combination of two or more two-dimensional Gaussian functions to represent the personal space of the people in the environment. The resulting function is therefore used to assign a crossing cost to the corresponding regions of the space as in [33][34][35], or to completely prevent passage to these areas as in [36].

In addition to keeping a certain distance from humans, Pacchierotti et al. [37][38] suggested that both the speed of motion of the robot during its navigation and the signaling of the perception of the person by the robot, and, therefore, the distance at which this signal is made, affect the feeling of safety perceived by people.

Another factor that affects a person's discomfort is the so-called *surprise effect*, i.e., the sudden appearance of the robot in the human's field of view. In this regard, Sisbot et al. [39], in addition to the concept of "safety criterion", also introduced the concepts of "visibility criterion" and "hidden zone criterion", proposing a solution based on a grid map where higher costs are assigned to places in the environment outside the person's field of view or hidden by obstacles. As a side effect, however, the paths generated by this planner may be very unnatural due to their attempts to stay visible to people. Similarly, Pandey et al. [36] modified the planned trajectory in the proximity of the corners, to travel a longer but more visible path.

3.2 Naturalness

The concept of naturalness refers to the property of an agent to navigate in the environment in a similar way to how a human behaves.

Pandey et al. [36] proposed a technique that is based on the dynamic selection of social rules to be applied according to the environment in which the robot is located. The application of these social norms leads to the relocation of the milestones that make up the planned path allowing the robot to both perform socially acceptable navigation but also to follow a smooth path.

In [35] and [40], assigning a cost not only to the position corresponding to the static obstacles but also to the surrounding places, causes the robot to avoid passing very close to the obstacles, thus emulating the behavior of a human.

Another aspect that can be considered in the naturalness topic is the ability to navigate in crowds. This, which represents a very active field in current research, is a problem typically addressed by resorting to techniques derived from machine learning. Examples are [41], which uses a Monte Carlo Search Tree planner with a Recurrent Neural Network to predict the movement of the crowd, and [42] where an approach based on deep reinforcement learning is employed.

3.3 Sociability

Beyond minimizing the feeling of discomfort that its movement could cause in humans and to navigate in a *natural* way, a robot should also navigate respecting the high-level rules that govern social relations between humans.

Some of these rules include keeping a certain distance from the people in the environment, a factor already mentioned in section 3.1, overtaking a person from his/her left (or right, depending on the social conventions of the specific Country), moving to the right (left) side of a hallway when a person is moving in the opposite direction, approaching a human from the front, approaching a group of people without invading the central space that they share, etc. As can be seen, the conventions listed above are tendencies rather than strict rules, which are applied in a more or less flexible way depending on the considered context.

Kirby et al. [43], in the proposed COMPANION framework, codified some of the social conventions mentioned above, specifically the pass on the right rule and not to invade people's *Personal* space rule, through mathematical cost functions, which are then used by an A* path planner. The proposed model constructs a cost function which is a linear combination of different social and non-social constraints, then used by a heuristic A* planner. This cost-based approach easily enables the extension of the framework to consider new constraints. However, the work only considered a single human scenario, thus limiting the range of application of the solution. Based on the same idea, Kollmitz et al. [44] proposed a time-depended planner which takes humans and their movement over time into account, using a layered dynamic social cost map (we will describe this framework in detail in Section 4.1).

Pandey et al. [36], on the other hand, proposed an approach where a decision tree is used; it dynamically selects the social rules to be applied based on the current state in which the robot is located, that is, based on the relative distance from humans present in the environment and the structure of the environment itself. The resulting planner, for example, penalizes the paths that pass through narrow passages to prevent the robot from getting stuck there if a human is also present. The idea of dynamically selecting the social rules to be used for path planning was consequently exploited by Bellarbi et al. [45], which added new conventions to the set of social rules proposed in [36], also considering the person's body orientation.

The techniques presented above, in different ways, try to codify the social conventions of navigation in high-level models to be integrated into the planning system. In recent years, however, alongside these types of approaches, numerous techniques derived from machine learning have been applied to the problem of social navigation. In fact, unlike *model-based* techniques, which may require adjustments of the parameters when moving from one social environment to another, these techniques, also called *learning-based* methods, have the advantage of automatically adapting to the environment thanks to the experience acquired through the trial and error learning process. In this regard, Tung et al. [46] proposed a social-aware navigation system based on deep reinforcement learning, which takes both the presence of humans and social relationships between them into account. Chung et al. [47][48], instead, proposed a socially acceptable planner by introducing the "Spacial Behaviour Cognition Model". This system integrates the human motion model (represented by a Markov Decision Process) with the "spatial effects" present in the environment, i.e. with the characteristics of the environment such as doors, hallways, humans, etc. (the weight of each spatial effect is obtained through Inverse Reinforcement Learning). On the other hand, it is important to realize that these methods require large amounts of data not always available, and the resulting models have low interpretability.

To our knowledge, none of the solutions we presented so far consider the presence of human groups in the environment or, more precisely, none of them adopt a model that explicitly represents these formations. In this regard, as explained in detail in section 2.2, the F-formation model [10] aims to explicitly represent the arrangements of humans in space and constitutes the theoretical foundations of different works. As an example, several authors [33][34][49], using the F-formation and *o-space* concepts, proposed a Risk-RRT based path planner able to detect and then avoid static formations in the environment. A further important contribution of these works is the formulation of a method to determine the center of these human arrangements. Nevertheless, the proposed navigation system can detect and avoid only groups of two people.

Truong's works [50][51], based on the concept of "Dynamic Social Zone" (DSZ), take a step further in this direction by proposing a planner that, indeed, can take into account groups of more than two members. Moreover, this approach is not limited to avoiding groups of people but also enables a robot to join them in a socially appropriate manner, even in the dynamic case. The weak point of these solutions, however, is the use of Dijkstra's algorithm (in [50]) and D* algorithm (in [51]) which, as it is known, perform well in sparse and quasi-dynamic environments, but are not sufficient in crowded and dynamic environments.

For this reason, first Gómez et al. [52], which considered only static groups, and then Yang et al. [35] (which considered dynamic groups too) adopted a fast marching path planning solution. This type of approach builds a speed map of the environment, i.e. assigns to each point in the environment a speed value, which is modulated based on the possible presence of obstacles, humans and social relations between them. The use of this technique makes the planner proposed by Yung et al. [35] able to generate smooth and time-optimal paths that take into account the groups of people (static and dynamic) present in the environment. It, therefore, constitutes a considerable contribution to the field of group-aware navigation. However, this framework, which we will refer to again in the rest of this thesis, was explicitly designed for serious games with virtual characters and not for robotics applications [1]. In addition, if, on the one hand, the usage of the Graph-Cuts for F-formation (GCFF) method [12] makes this framework capable of dealing with complex populated environments, on the other hand, it is difficult to adapt the framework to different scenarios.

On the contrary, in this thesis, we will propose a group-aware navigation framework able to deal with less populated environments (such as hallways and narrow spaces in general) but that can be easily adapted to different human

[1] This implies, for example, not considering some of the typical aspects of robot navigation, such as the motion constraints.

environments.

Summary Plenty of research has been conducted on socially-aware navigation, from works that consider single-human scenarios to works that consider high-dynamic crowds. However, only a limited number of these have proposed solutions that explicitly deal with the presence of human groups, and even fewer works have addressed this problem in narrow environments. For this reason, based on the work of Kollmitz et al. [44], which proposes a robot path planner able to optimize among multiple customizable constraints, this thesis will propose a navigation framework that allows an autonomous agent to move in the environment optimally taking the presence of humans and groups of humans into account. It is worth underlining that, in this context, optimality regards a multitude of factors, some related to performance (such as the distance traveled and the time spent by the robot to reach the goal pose) and some to sociability (such as the compliance with the human personal space and group-shared space constraints).

Chapter 4

Methods

In this chapter, we will present a human-aware and group-aware global planner. In particular, in Section 4.1, we will present the state-of-the-art framework that served as the basis for our group detection and handling models, and which we will refer to as Baseline, in the rest of the thesis. In Section 4.2, we will detail the model used to represent individual humans during the planning phase, i.e. the Personal Space model. Finally, in Section 4.3, we will describe the models designed to identify and take into account the groups of people in the environment surrounding the autonomous agent.

4.1 Baseline Planner

In order to implement a global planner capable of taking into account both individual and groups of humans, we have chosen the *social* planner designed by Kollmitz et al. [44] as the basic framework. This choice is motivated by the characteristic of the framework to present a clear separation between the planning procedure and the definition of planning constraints. This property allows us to develop, independently of the core planner, a social model which, encoded in cost functions, can be used as input to the planning procedure.

In general terms, this baseline framework consists of an A*-based path planner, which uses a cost function that is a weighted linear combination of different factors, some related to performance and others to sociability. The resulting trajectory is, therefore, the one that jointly optimizes all these elements.

Regarding performance, the factors taken into consideration include the navigation time and the total traveled distance between the initial and the goal positions.

As for sociability, on the other hand, the framework defines a so-called **Social Cost Model**, which encodes the social relations to take into account while moving around. Specifically, the baseline framework models the personal space of humans, thus avoiding that the robot generates paths that collide (or pass too close) to people. This social cost model is exactly the innovation element introduced by this thesis, which, in addition to defining a more accurate personal space representation (Section 4.2), implements new models that allow the identification and representation of static and dynamic human groups present in the environment (Section 4.3).

Back to the baseline framework, it also presents another important property, which plays a significant role in producing human-aware trajectories. In fact, in addition to its social planner functions, it is also a time-dependent planner, i.e., where time is considered a variable of the search space, or, stated differently, time is one of the variables that define the planning configuration (see Section 2.3.1 for the meaning of configuration space). This aspect allows the planner to take into account not only the current positions of the people, but also their most likely future poses within a specified time range. Thanks to this ability to predict and, therefore, anticipate the movements of humans, the trajectories generated by the robot are smooth, easy to predict for people, and, therefore, more comfortable.

4.1.1 The Dynamic Cost Map

To combine the two properties described above, i.e., the use of a social cost model and the time dependence, the framework uses a layered dynamic cost map.

In detail, this dynamic cost map consists of a static layer, which represents the static objects present in the environment such as walls and furniture, and a dynamic layer (one for each discrete-time instant in the future) which encodes the Social Cost Model as Gaussian cost functions.

Therefore, depending on the discrete-time instant considered (recall that time is a state variable), the corresponding dynamic layer is selected. This dynamic layer is built based on the predicted positions of the people (and groups of people, in the approach we will propose), who, depending on their starting positions and the prediction of their motions, will occupy a different region of space depending on the considered time instant. In this regard, Figure 4.1 shows a possible example scenario and the corresponding dynamic cost maps.

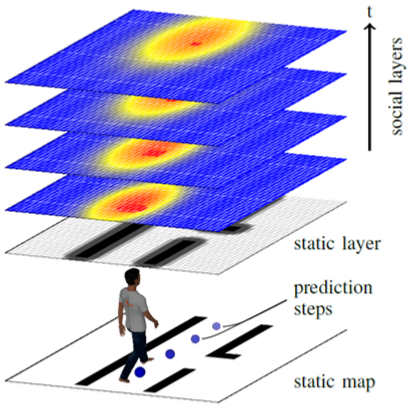


Figure 4.1: The dynamic cost map is composed of a static layer and a dynamic one for each considered time step [44].

Once the cost map has been created, it is used to create a complete cost function, which is a weighted linear combination of the different factors that characterize each planning state, such as the distance from the initial configuration, the time spent to reach the current state, the social cost of crossing intermediate states, etc. This function is, subsequently, employed by the deterministic planning algorithm A*, which generates the minimum cost path to the target state [1] (the A* algorithm can be found in Appendix A).

We will now describe how we designed the personal space model and the identification models of the different types of groups that people tend to form when they interact with each other. We will also present the cost functions used to represent these models within the dynamic layer of the cost map described above.

4.2 Human personal space model

As explained in Chapter 2, to take human comfort into account, it is not possible to treat people as simple obstacles, but it is necessary to consider them as special entities from which the robot should keep a certain distance. In this regard, Section 2.1 detailed the *Proxemics* model introduced by Hall [5], in which specific regions of space around the human are identified. In the current section, we will use this theoretical model to implement a Gaussian cost function that represents the person in the space.

[1] To speed up the execution of the algorithm, the framework uses an admissible and consistent heuristic based on the assumption that no people are present in the environment.

First of all, it is worth underlining that in this section the term **Personal Space** is more general and refers to the union of the *intimate space* and the *personal space* (using Hall's terminology). Moreover, since people are usually more sensitive to events that occur within their field of view, it is reasonable to consider a frontal space larger than the back one. For these reasons, to model the Personal Space, we use an approach derived from Laga et al. [53], which adopts a linear combination of two 2D Gaussian functions, one for the front area and the other for the back area of the person.

In more detail, given a human H located at position $h(x, y)$ we define a local coordinate system with origin in h, Y-axis along the shoulder direction and X-axis in the face direction (Fig. 4.2-a). The Personal Space is therefore represented by the following function (Fig. 4.3):

$$\Phi_{h,\Sigma_{front},\Sigma_{back}}(q) = \delta(x_q)\Phi_{h,\Sigma_{front}}(q) + (1 - \delta(x_q))\Phi_{h,\Sigma_{back}}(q) \tag{4.1}$$

where $q = (x_q, y_q)$ are the coordinates of a point in the map coordinate system, $\Phi_{h,\Sigma_{front}}$ is the Gaussian function that models the frontal area of the person, $\Phi_{h,\Sigma_{back}}$ is the Gaussian function that models the back space, and $\delta(x)$ is such that

$$\delta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

i.e., it selects the correct Gaussian function depending on whether the considered point is in the front or back space of the person.

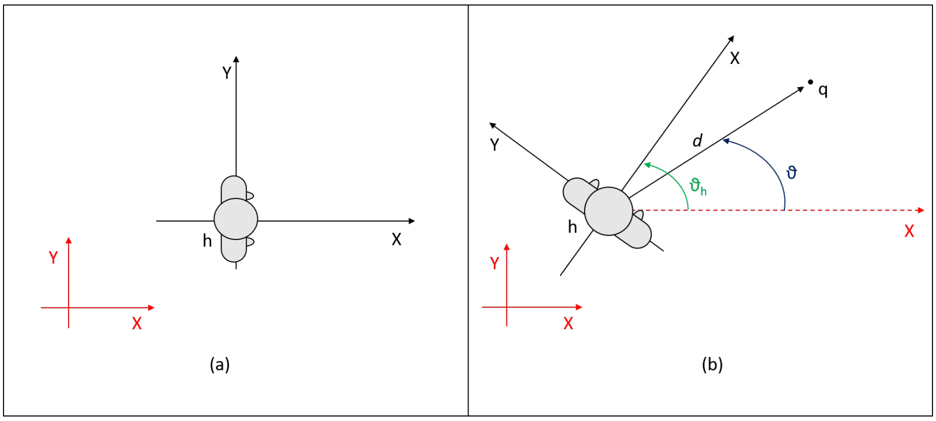


Figure 4.2: Local coordinate system centered in the position of the human.

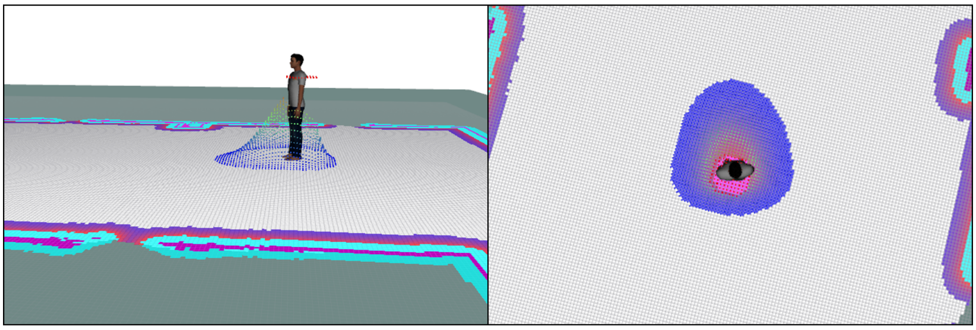


Figure 4.3: Gaussian function that models the personal space of a standing human.

As for the individual Gaussian functions, we adopt the following function proposed by Yang et al. [35]:

$$\Phi_{h,\Sigma}(q) = A\, exp(-\frac{(d\cos(\theta-\theta_h))^2}{2\sigma_x^2} - \frac{(d\sin(\theta-\theta_h))^2}{2\sigma_y^2}) \tag{4.2}$$

where d and θ are the Euclidean distance and angle from $q = (x_q, y_q)$ to $h = (x, y)$, θ_h is the angle between the person's face direction and the X-axis of the map coordinate system (Fig. 4.2-b), $A = 255$ is an amplitude parameter and σ_x^2, σ_y^2 are the diagonal entries of the Σ covariance matrix, which determines the shape of the personal space. In particular, the two covariance matrices in 4.1 are defined in the following way:

$$\Sigma_{front} = \begin{pmatrix} \sigma^2_{x\_front} & 0 \\ 0 & \sigma^2_{y\_front} \end{pmatrix};\ \Sigma_{back} = \begin{pmatrix} \sigma^2_{x\_back} & 0 \\ 0 & \sigma^2_{y\_back} \end{pmatrix} \tag{4.3}$$

By choosing different values for σ_x and σ_y it is therefore possible to model different shapes for the front and back personal spaces.

To also take the person's status (i.e. if the human is stationary or moving) into account, we introduce the parameter $p_{mot} = \frac{vel}{max\_speed}$, where vel is the current human speed and max_speed is the maximum speed supposed for the movement of the person. Based on the logical reasoning, the four parameters of 4.3 are then redefined as follows:

$$\begin{cases} \sigma^2_{x\_front} &= (1+p_{mot})\sigma^2_{x\_front} \\ \sigma^2_{y\_front} &= (1-\frac{p_{mot}}{2})\sigma^2_{y\_front} \\ \sigma^2_{x\_back} &= (1-p_{mot})\sigma^2_{x\_back} \\ \sigma^2_{y\_back} &= (1-\frac{p_{mot}}{4})\sigma^2_{y\_back} \end{cases}$$

obtaining an overall lengthening of the Gaussian cost function in the direction of human motion and a narrowing of the lateral area. This choice is motivated by different studies on people's motion [54], which reveal that in her/his movement the person tends to minimize energy consumption, thus avoiding accelerations.

Finally, considering that a human does not occupy only one point on a plane but covers a certain area of space, using the same approach proposed by Kollmitz et al. [44], we define a region of circular space centered in the position of the person to whom the maximum cost is assigned [2]. This choice implies that, during the planning phase, the autonomous agent does not plan paths passing too close to the human, which would cause discomfort and, in the worst cases, clashes.

Distinction between front and back spaces To fully understand function 4.1, a deeper understanding of how the delta function is implemented is necessary. This aspect will also be addressed in Section 4.3.1 when we will talk about the conditions necessary for a static group to be identified.

Let us consider again a human H located at position $h(x, y)$, whose face orientation forms an angle θ_h with the positive X-semiaxis of the map coordinate system, as shown in Figure 4.2-b. To determine if a point $q(x, y)$ is inside her/his frontal space, it is possible to apply the mathematical concept of **inner product** between vectors. In our case, therefore, indicating with $\overrightarrow{f}$ the vector corresponding to the face direction of the person and with $\overrightarrow{q}$ the vector corresponding to the considered point (in the local coordinate system of the human), the required condition is implemented by the following inequality:

$$\overrightarrow{f} \cdot \overrightarrow{q} \geq 0 \Longrightarrow \cos|\theta_h - \theta| \geq 0 \tag{4.4}$$

Figure 4.4 shows the two types of possible cases. In the first scenario (Fig. 4.4-a), where the q point is in the human frontal space, the projection of the vector $\overrightarrow{q}$ along the person's face direction is parallel to the vector $\overrightarrow{f}$ thus generating a positive inner product. In the second case (Fig. 4.4-b), with the q point in the human back space, instead, the projection of $\overrightarrow{q}$ has opposite direction to $\overrightarrow{f}$ producing a negative inner product.

[2] This region of space models part of the *intimate* space (in Kendon's terminology).

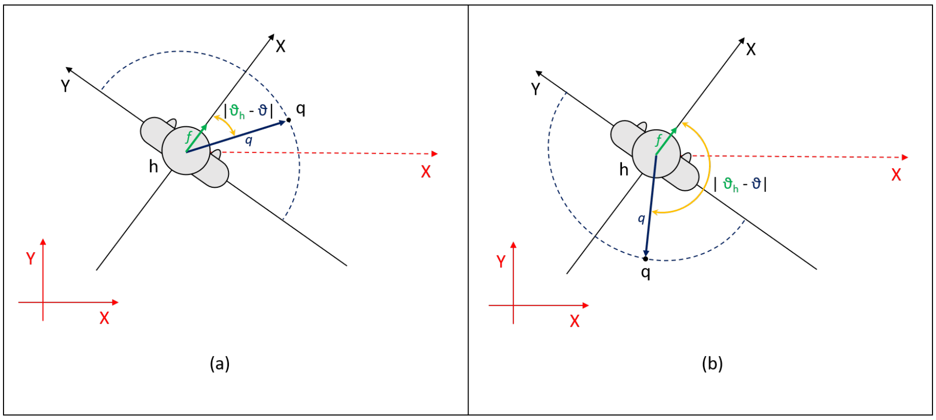


Figure 4.4: a) The point q is within the human frontal space. b) q is within the back area.

4.3 Human group models

To take the presence of human formations in the environment into account during the planning phase, it is necessary to determine their structure and correct positioning in the space. In this regard, as explained in Section 2.2, people, during interactions (particularly in a static scenario), tend to jointly assume and maintain typically circular space-orientational arrangements, called F-formations. Consequently, using Kendon's terminology [11], the problem becomes that of identifying the type of formation and subsequently determining its center, thus the center of the so-called *o-space*. In this section, therefore, we will present the mathematical models designed to identify and, therefore, represent static and dynamic human formations, which represent the main contribution of this thesis. We will assume to know at every moment the pose, i.e. position and orientation, and velocity (both in the map coordinate system) of each person present in the environment surrounding the robot.

4.3.1 Static formation models

Given a set of standing people, the first problem to be addressed is to check if their poses determine a group, and if not, to state if there are subsets within it that, instead, can be considered groups.

To solve this problem, we introduce a first constraint, which is called **Distance Constraint**. Specifically, this condition requires that the relative Euclidean distance between at least two members of a formation be less than

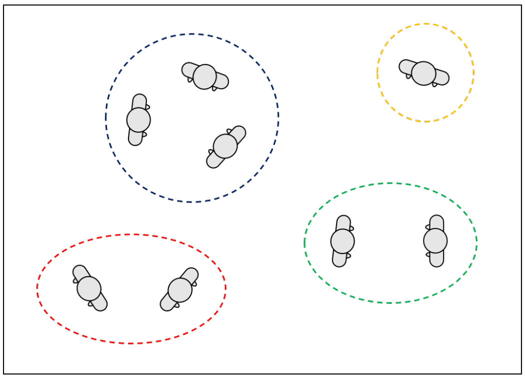


Figure 4.5: Applying the distance constraint to the set of people has the effect of partitioning them into subsets, which are potential formations. In this particular scenario, the blue and green subsets are strictly formations, while the red and orange subsets not.

a specified threshold *st_group_dist_thr*. This condition, it should be emphasized, does not require that the relative distance between each possible pair of group members be less than the threshold, but that at least one pair respects this constraint. The result of applying this requirement is the partitioning of people into several subsets, which *potentially* represent different formations, as shown in Figure 4.5. The algorithm used to enforce this constraint, or, stated differently, to perform the clustering of humans can be found in Appendix B.

The set of identified partitions is then analyzed by treating the different subsets based on the number of members within it. It is worth underlining that subsets containing a single member are not explicitly handled as groups and are therefore discarded.

Groups with two members

Particularly interesting is the case of subsets with only two members, where, depending on the poses of the participants, three different structures are possible, as seen in Section 2.2.1. First of all, before analyzing in detail the models relating to these three structures, it is necessary to focus attention on one of the aspects common to all these types of formations regarding the orientation of the participants. As shown in Figure 4.6, indeed, in all three types of two-member formations, each person must be located within the frontal space of the other. For this reason, we introduce a second constraint, called **Orientation Constraint**, which must be satisfied for a subset of two members to be considered a real formation (based on Kendon's theory). Using the same

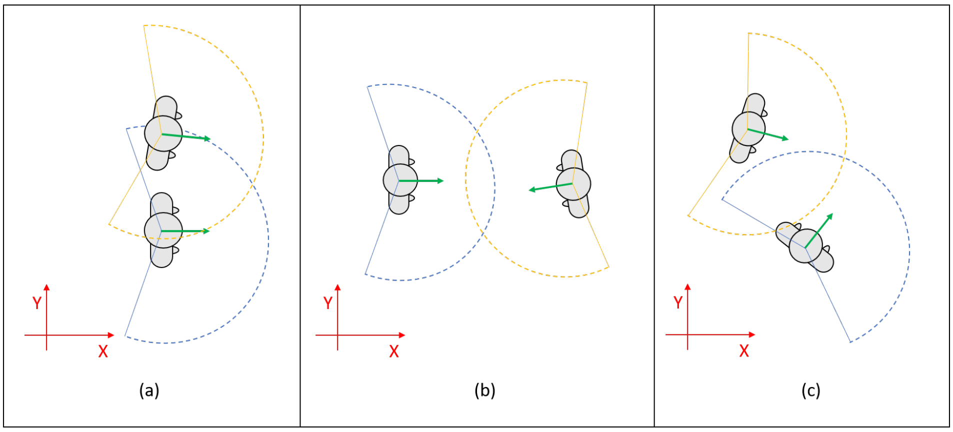


Figure 4.6: In all three types of 2-member formations identified by Kendon, each of the participants is located in the frontal area of the other member.

approach described in Section 4.2, we define the following condition, which must be met for **both** participants (Fig. 4.7):

$$\vec{f} \cdot \vec{q} \geq angle\_thr \implies \cos|\theta_h - \theta| \geq angle\_thr \tag{4.5}$$

where *angle_thr* is a parameter to be set according to the desired width for the human frontal space.

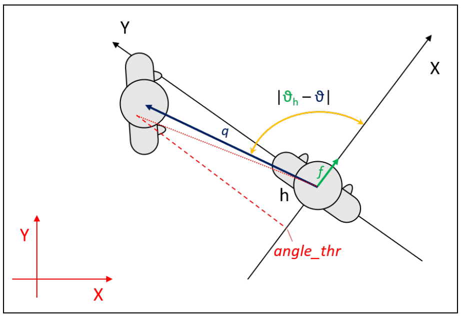


Figure 4.7: Orientation Constraint illustrated for one of the two group participants.

We will now describe what are the conditions that allow distinguishing the different two-member formations (assuming the fulfillment of the two constraints of *distance* and *orientation* described above) and the criteria used to determine the center of the corresponding o-space.

Side-by-side To be in a *side-by-side* configuration, the orientations of the two participants of the formation must satisfy the following system of inequal-

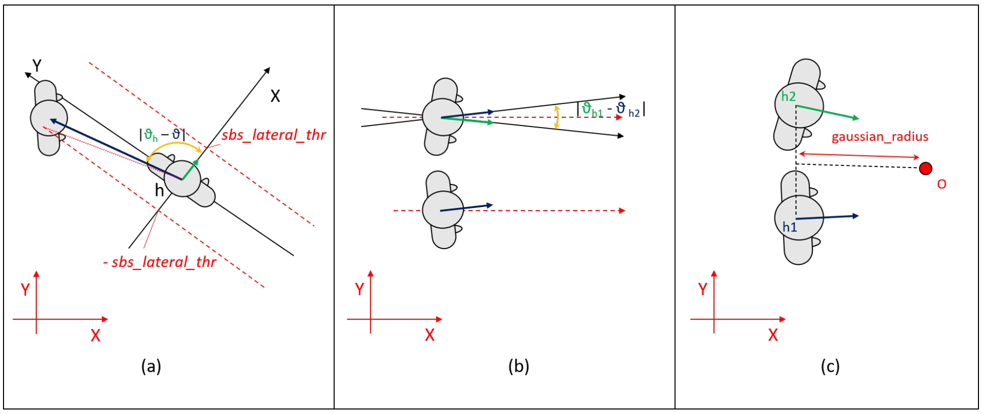


Figure 4.8: a) - b) Orientation constraints that identify a side-by-side formation and c) center location of the corresponding o-space region.

ities:

$$\begin{cases} |\cos(|\theta_h - \theta|)| & \leq & sbs\_lateral\_thr \\ \cos(|\theta_{h1} - \theta_{h2})|) & \geq & sbs\_front\_thr \end{cases} \tag{4.6}$$

It is important to realize that the angles that appear in these two formulas do not refer to the same variables. In the first inequality, which must be satisfied for at least **one** of the two members and whose graphic representation is shown in Figure 4.8-a, θ_h is the angle between the face orientation of the considered person H and the positive X-semiaxis of the map coordinate system, and θ is the angle (in polar coordinates) from the position of the second participant to the position of H (as we also illustrated in section 4.2). In the second inequality, instead, both θ_{h1} and θ_{h2} represent the angle between the positive X-semiaxis of the map coordinate system and the face direction of the two participants respectively (Fig. 4.8-b).

In general terms, the first condition of 4.6 requires that fixing the position of one of the two participants, let's call it H, the other member must be located within the plane strip having a direction perpendicular to the face direction of H and width equal to $2 * |\overrightarrow{q}| * sbs\_lateral\_thr$. The second inequality, on the other hand, requires that the two members must face approximately the same direction, with a tolerance of $\theta = \pm \cos^{-1}(sbs\_front\_thr)$ in the difference of orientation angle.

Once the side-by-side formation has been identified, it is necessary to represent it in the cost map through a cost function, so that the global planner can take it into account when creating the path to the goal position. To do this, we use a 2D Gaussian function, which we will describe in detail in Section 4.3.1. At this point, however, we have left to describe how, for this type of formation, we select the origin of the cost function, that is where the center of the o-space is located. In this regard, based on Kendon's model (Section 2.2.1), the group center O is located in the midpoint between the positions of the two members and shifted in their frontal region by a distance equal to the *gaussian_radius* parameter and an angle equal to the average value of the orientation angles of the two participants (Fig. 4.8-c). In mathematical terms, it is defined by the following formulas:

$$\begin{cases} x_O = x_m + gaussian\_radius * \cos\alpha \quad , \quad x_m = \frac{x_{h1}+x_{h2}}{2} \\ y_O = y_m + gaussian\_radius * \sin\alpha \quad , \quad y_m = \frac{y_{h1}+y_{h2}}{2} \end{cases} \tag{4.7}$$

where $\alpha = \frac{\theta_{h1}+\theta_{h2}}{2}$.

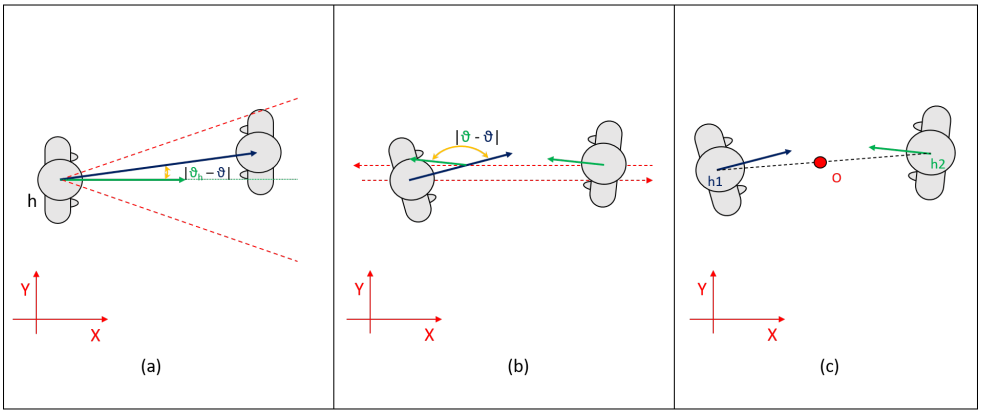


Figure 4.9: a) - b) Orientation constraints that identify a vis-a-vis formation and c) center location of the corresponding o-space region.

Vis-a-vis A *vis-a-vis* configuration occurs when the poses of the two people satisfy the following system of inequalities:

$$\begin{cases} \cos(|\theta_h - \theta|) \geq vav\_lateral\_thr \\ |\sin(|\theta_{h1} - \theta_{h2}|)| \leq vav\_front\_thr \end{cases} \tag{4.8}$$

where the angles θ_h, θ, θ_{h1} and θ_{h2} have exactly the same meaning described in the previous paragraph. In general terms, the first condition, which must be met by **both** members, requires that fixing the position of one of the two participants, let's call it H, the other member must be located inside the plane angular section with origin in the position of H and angular amplitude equal to $2 * \cos^{-1}(vav\_lateral\_thr)$ (Fig. 4.9-a). In this regard, it is important to underline that *vav_lateral_thr* should be set so that $vav\_lateral\_thr > sbs\_lateral\_thr$, otherwise, conditions 4.6 and 4.8 would not be disjoint, in particular, the set of configurations (between the two members of the group) allowed by inequalities 4.8 would include the set identified by 4.6 [3]. The second condition, on the other hand, requires that each person must be oriented approximately towards the other, consequently generating a difference orientation angle of about π radians, with a tolerance of $\theta = \pm \sin^{-1}(vav\_front\_thr)$ (Fig. 4.9-b).

Regarding the position of the center of the Gaussian function for this type of structure, it is merely located at the midpoint between the positions of the two members, as shown in Figure 4.9-c.

V-shape The last type of formations introduced in Section 2.2.1 is the one called by Kendon *L-shape*, however, in this Chapter, we consider more generally the so-called *V-shape* formations, in which the face directions of the two people are not necessarily perpendicular with each other, as occurs in the L-shape case. This choice allows us to identify and consider a larger number of configurations, thus making the model more flexible and suitable for the multitude of real applications. This category, indeed, includes almost all the remaining configurations, or at least all those that satisfy the following property.

To be in a *V-shape* configuration, the face directions of the two people, let's call them H1 and H2, must intersect in a point that belongs to both the frontal region of H1 and the frontal region of H2, as shown in Figure 4.10-a. In order to check the fulfillment of this convergence constraint, we use the procedure shown below in Algorithm 1:

[3] This situation, although not rigorously correct, is not, however, a real problem, since it is sufficient to first check the most stringent condition (i.e. 4.6), in the process of identifying the type of group, to obtain a correct classification.

Algorithm 1 Convergence Check

$h1 \leftarrow$ position of H1
$h2 \leftarrow$ position of H2
$curr\_dist \leftarrow distance(h1, h2)$
$i \leftarrow 0$
while $i <$ *max_cycles* $\&$ *curr_dist* $>$ *v_dist_thr* **do**
    $next\_h1 \leftarrow h1 +$ *incr* ▷ (along H1 face direction)
    $next\_h2 \leftarrow h2 +$ *incr* ▷ (along H2 face direction)
    $next\_dist \leftarrow distance(next\_h1, next\_h2)$
    if $next\_dist > curr\_dist$ **then**
        return *no-formation*
    end if
    $curr\_dist \leftarrow next\_dist$
    $i \leftarrow i + 1$
end while
return *formation-detected*

In general terms, after computing the current distance between the two people positions, this algorithm determines the "virtual" positions that they would reach by moving in the direction they are oriented. If the distance between these new points increases, then the face directions of the two participants are divergent and, therefore, they do not determine a formation (Fig. 4.10-b). On the contrary, if the distance continuously decreases until reaching a given threshold or a maximum number of iterations is exceeded [4], then the two people have convergent orientations determining a real V-shape formation (Fig. 4.10-a).

In the latter scenario, to determine the center of the o-space, we use a technique proposed by Martinez et al. [33] whose graphic representation is shown in Figure 4.10-c. Specifically, given the positions of the two participants ($h1$ and $h2$ in the figure), we call M the midpoint between $h1$ and $h2$, and V the intersection point of their face directions [5]. The center of the o-space O is then determined as the midpoint between M and V.

[4] This additional condition allows to limit the execution time of the procedure in case a rather small *incr* step is set.

[5] In reality, we do not determine the exact intersection point of the two people's directions, but an estimate of it, obtained through Algorithm 1 as the midpoint between the two "virtual" points *next_h*1 and *next_h*2 once the convergence condition is confirmed.

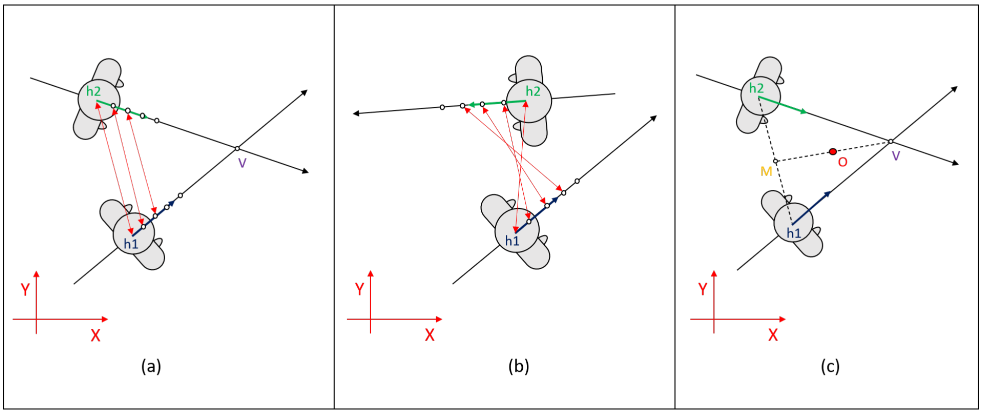


Figure 4.10: a) V-shape formation: the convergence constraint of the face directions of the two people is satisfied. b) No formation: the face directions of the two people are divergent. Both in a) and in b) the white circles represent the "virtual" points used in Algorithm 1, to check the convergence of the two human directions. c) Estimation of the center of the o-space region in the V-shape formation scenario.

Groups with more than two members

The class of formations with three, four, or more members presents a weaker classification than that of groups with two members analyzed above. Based on Kendon's theory, indeed, in this type of formations, people typically tend to assume and maintain a circular-shape space-orientational arrangement, regardless of the number of participants.

For this reason, to establish the actual presence of this type of group, we only introduce and check the satisfaction of the following constraint regarding the orientation of the participants in the subset, called **Circle Orientation Constraint** [6]. This constraint derives from the observation that when people arrange themselves in a circle, each person is located within the frontal space of the other, or, stated differently, each person's frontal space includes all the other members of the group (Fig. 4.11-a). However, to design a more flexible model that considers a larger number of scenarios, the Circle Orientation Constraint requires the condition, that a group member must include all the other participants in his/her frontal space, does not necessarily have to be met by all the participants, but at least by more than half of them (Fig. 4.11-{a,b}).

If the formation is detected, the center O of the corresponding o-space is directly positioned at the midpoint of the locations of all members, considering

[6] The Distance Constraint is assumed to be already satisfied.

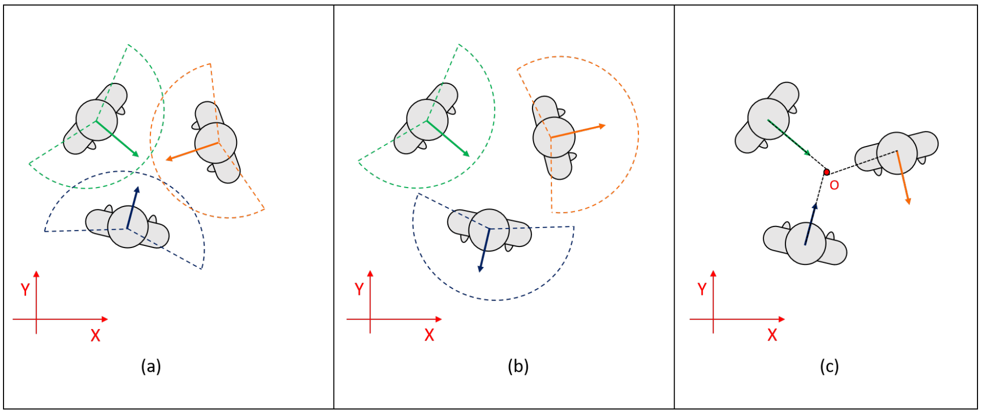


Figure 4.11: The figure analyzes the case of a subset with three humans. a) Each participant is inside the front space of the others, thus satisfying the Circle Orientation Constraint. Therefore, they determine a formation. b) Since only one member (green) out of three includes all the other participants in its front space, the orientation constraint is not satisfied and, therefore, the subset is not identified as a formation. c) Estimation of the center of the o-space region. The subset is identified as a group because more than half of the members (green and blue) satisfy the orientation constraint.

both those that satisfy the orientation constraint and those that do not (Fig. 4.11-c).

Group cost function

As previously mentioned, we use a 2D Gaussian cost function to represent the formations in the cost map, regardless of the number of members (obviously greater than or equal to two). In particular, similarly to what we described in Section 4.2, the Gaussian function is defined by the following equation:

$$\Phi_{O,\Sigma_{SG}}(q) = A\, exp(-\frac{(d\cos\theta)^2}{2\sigma^2_{x\_SG}} - \frac{(d\sin\theta)^2}{2\sigma^2_{y\_SG}}) \tag{4.9}$$

where $O(x, y)$ is the center of the o-space, $q(x_q, y_q)$ is the considered point (both the coordinates of O and q are in the map coordinate system), d and θ are the Euclidean distance and angle from q to O, $A = 255$ is an amplitude parameter and $\sigma^2_{x\_SG}$, $\sigma^2_{y\_SG}$ are the diagonal entries of the Σ_{SG} covariance matrix, which determines the shape and size of the formation space.

Finally, using the same approach adopted in Section 4.2 for the Personal Space, we define a circular region of space centered in the origin of the o-space to whom the maximum cost is assigned. In this way, the autonomous

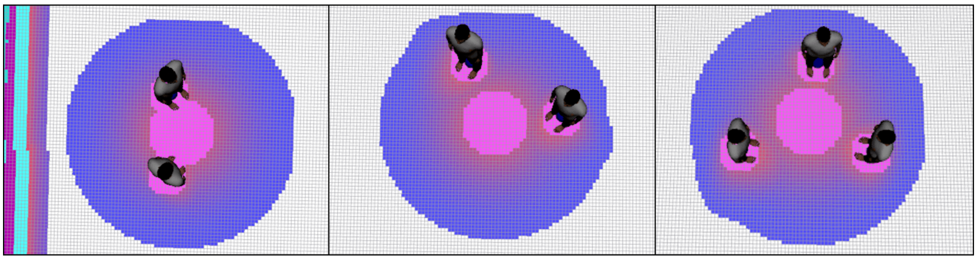


Figure 4.12: Gaussian functions that model the static formations V-shape, side-by-side, and 3-members from left to right.

agent will surely avoid taking paths that cross the central area of the group, which would cause discomfort to the participants (Fig. 4.12).

4.3.2 Dynamic group models

Similarly to static formations, also for the dynamic ones, we define a set of constraints on the poses and velocities of humans to identify and represent them adequately during the planning phase. Here too, indeed, we introduce a **Distance Constraint**, which requires that the relative Euclidean distance between at least two moving members of a formation be less than a specified threshold *dy_group_dist_thr*. The application of this condition to the set of moving people perceived by the autonomous agent implies the partitioning of it into subsets that *potentially* represent dynamic formations (Fig. 4.13-a). The algorithm used to enforce this constraint can be found in Appendix B.

The set of identified partitions [7] is then analyzed, checking for each subset the fulfillment of two further constraints, called **Dynamic Orientation Constraint** and **Speed Constraint**, which have been defined based on the theoretical model of dynamic groups discussed in Section 2.2.2.

As the name suggests, the Dynamic Orientation Constraint is a condition on the orientation of the members of the formation and, given two members H1 and H2, can be represented by the following formula:

$$\cos\left(|\theta_{h1} - \theta_{h2}|\right) \geq dy\_front\_thr \tag{4.10}$$

where θ_{h1} and θ_{h2} represent the angle between the positive X-semiaxis of the map coordinate system and the face direction of the two participants respectively. In general terms, this constraint, which must be satisfied by **every** pos-

[7] Even in the dynamic case, subsets consisting of a single member are not explicitly handled and are therefore discarded.

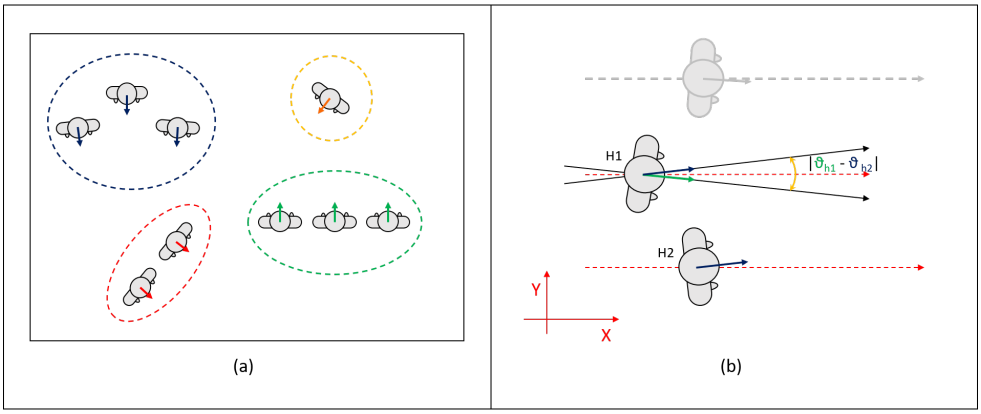


Figure 4.13: a) Applying the distance constraint to the set of moving people has the effect of partitioning them into subsets, which are potential formations. b) Orientation constraint illustrated for one couple of the three-members dynamic group.

sible pair of members of the subset, requires that all the participants have approximately the same direction of motion, as shown in Figure 4.13-b.

Finally, the Speed Constraint requires that the difference in speed between each possible pair of people in the subset be less than the *dy_group_vel_thr* threshold.

Once the dynamic group has been identified, it is represented in the cost map using the same approach employed for static groups, that is, through a 2D Gaussian cost function. Before defining this function, however, it is necessary to describe how the center of the Gaussian is determined, that is where the center of the formation is placed [8]. In this regard, we set it in the midpoint of the locations of all the members shifted, in the motion direction of the group, by a quantity equal to *offset_x*, which in turn is defined by the following system of equations:

$$\begin{cases} \mathit{offset\_x} = \mathit{offset\_x\_DGroup} * (1 + p_{mot}) \\ p_{mot} = \frac{vel}{max\_speed\_DG} \end{cases} \tag{4.11}$$

where *offset_x_DGroup* is a constant parameter, *vel* represents the average speed of all members and max_speed_DG is the maximum speed supposed

[8] The patterns of dynamic formations described in Section 2.2.2 do not explicitly model an o-space, as instead happens for the Kendon model relating to static formations (Section 2.2.1), however, we use this concept in this scenario too, as it is a suitable tool for modeling the group shared space also for moving formations.

for the movement of the group [9].

The Gaussian function is then defined by the following equation:

$$\Phi_{O,\Sigma_{DG}}(q) = A\,exp(-\frac{(d\cos(\theta-\theta_{DG}))^2}{2\sigma^2_{x\_DG}} - \frac{(d\sin(\theta-\theta_{DG}))^2}{2\sigma^2_{y\_DG}}) \quad (4.12)$$

where $O(x, y)$ is the center of the formation, $q(x_q, y_q)$ is the considered point (both the coordinates of O and q are in the map coordinate system), d and θ are the Euclidean distance and angle from q to O, θ_{DG} is the average value of the members' orientations, $A = 255$ is an amplitude parameter and $\sigma^2_{x\_DG}$, $\sigma^2_{y\_DG}$ are the diagonal entries of the Σ_{DG} covariance matrix, which determines the shape and size of the formation space.

Furthermore, using the same approach adopted in Section 4.3.1 for static groups, we define a circular region of space centered in the origin of the group to whom the maximum cost is assigned.

Finally, similarly to what we defined for the Personal Space (4.2), we set the entries of the covariance matrix as functions of the current speed of the formation. In mathematical terms:

$$\begin{cases} \sigma^2_{x\_DG} & = & (1+p_{mot})\sigma^2_{x\_DG} \\ \sigma^2_{y\_DG} & = & (1-\frac{p_{mot}}{4})\sigma^2_{y\_DG} \end{cases}$$

The overall result of this modification is a lengthening of the Gaussian function proportional to the group velocity (Fig. 4.14). In particular, from the planning stage perspective, this implies that the global planner should avoid selecting paths that cross the regions of space immediately ahead of the moving groups.

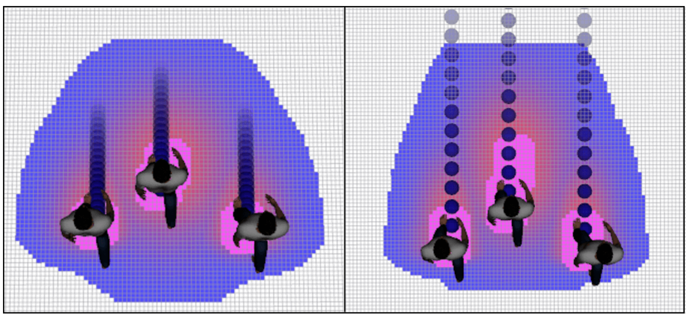


Figure 4.14: Gaussian functions that model a dynamic formation with group speed of 0.1 m/s (left) and 0.5 m/s (right).

[9] The p_{mot} parameter is similar to that introduced in Section 4.2, for Personal Space.

To conclude, it is important to underline that the choice not to distinguish the different dynamic formations (as done in the static case) by handling them all in the same way, is dictated by the consideration that the computation of the group center would be very similar regardless of the type of structure under investigation, making the classification meaningless.

Chapter 5

Simulations and results

This chapter details the experiments performed to test the framework and the obtained results. In particular, two categories of simulations are performed: Evaluation Tests and Comparison Tests.

In the Evaluation Tests (5.1), the proposed framework is evaluated through a set of use cases corresponding to different configurations of humans in space, verifying the correct identification and handling of human groups (when they are present). In the Comparison Tests (5.2), on the other hand, the proposed approach is compared, in terms of performance and sociability, against the Baseline in multiple scenarios, one for each type of formation(s).

Simulations are performed using the robotics simulator Gazebo [55], together with Robot Operating System (ROS) [56] and Rviz, a 3D visualization tool for ROS. The static testing environment consists of a 6 m wide and 14 m long hallway built with Gazebo, while the humans, who populate it, are simulated with ROS (emulating a people detector tool). The tests are performed using a simulated TurtleBot Burger robot.

The threshold-parameters of the model used to perform the set of simulations are provided in Table 5.1, which specifies the macro-model (where the constant has been defined), the specific parameter, and its corresponding numerical value.

To have a global view of the performed simulations, we provide a summary table (Table 5.2) where, for each type of test and condition/behavior analyzed, the corresponding simulations are specified.

Mod	Parameter	Value
PS	σ_{x_front}	0.3
	σ_{y_front}	0.12
	σ_{x_back}	0.05
	σ_{y_back}	0.12
	max_speed	1.0 (m/s)
SG	$st\_group\_dist\_thr$	2.0 (m)
	$angle\_thr$	-0.5
	$sbs\_lateral\_thr$	0.5
	sbs_front_thr	0.8660
	$gaussian\_radius$	0.6 (m)
	$vav\_lateral\_thr$	0.8660
	vav_front_thr	0.34

Mod	Parameter	Value
SG	v_dist_thr	0.5 (m)
	$incr$	0.05 (m)
	max_cycles	100
	σ_{x_SG}	0.4
	σ_{y_SG}	0.4
DG	$dy\_group\_dist\_thr$	2.0 (m)
	dy_front_thr	0.9397
	$dy\_group\_vel\_thr$	0.5 (m/s)
	$offset\_x\_DGroup$	0.5 (m)
	max_speed_DG	0.8 (m/s)
	σ_{x_DG}	0.3
	σ_{y_DG}	0.3

Table 5.1: Numerical values of the model parameters used to perform the simulations.

Test type	Simulation condition	Use case
Evaluation Tests	Handling of static formations	1.{a,b} 2.{a,b} 3.{a,b} 4.{a,b} 5
	Handling of dynamic groups	6.{a,b} 7
Comparison Tests	Hallway populated with static formations	1 2 3 4 5.{a,b}
	Hallway populated with dynamic groups	6
	Mixed scenario: the overtaking maneuver	7

Table 5.2: Summary table of the performed simulations.

5.1 Evaluation Tests

This test set is used to determine whether the system behaves as expected in different situations. For each use case, the initial positions and orientations of the humans and the robot are set. Subsequently, a goal pose is provided to the navigation stack of the robot [1], which updates the cost map according to the people and formations detected within its surrounding space. This map is then used in the robot's global planner to determine a path to the required location.

5.1.1 Handling of static formations

In this section, we will analyze environments populated only with static groups.

Use case 1.a: 2-people side-by-side In this use case, the poses of the two people (described in Table 5.3) determine a side-by-side formation. Consequently, the TurtleBot, after identifying it and updating the cost map accordingly, plans a path towards the goal pose (which is on the opposite side of the human group), that circles the group, without passing through it (Fig. 5.1).

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-0.7	-1.8	270
Human 2 (right)	0.8	-2.0	261

Table 5.3: Human poses (positions and orientations) in use case 1.a. All values are relative to the map coordinate system.

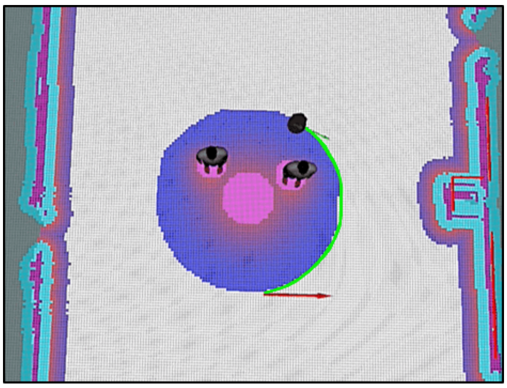


Figure 5.1: Use case 1.a. The robot detects the side-by-side formation and consequently avoids passing through it.

[1] The planner also takes the orientation of the robot in the goal state into account to determine the trajectory. However, for the purpose of this thesis, this final orientation is not particularly relevant.

Use case 1.b: 2-people subset without side-by-side structure This use case represents a variant of the previous one in which the orientation of one of the two people of the set has been changed (Table 5.4). This modification, however, violates one of the two constraints required by the side-by-side configuration related to orientation, specifically the one about the concordant orientation of the members (second equation of 4.6), making the subset not compliant with the side-by-side structure. Consequently, by providing the robot navigation stack with the same target position as in the previous case, the resulting path now passes through the set of humans (Fig. 5.2).

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-0.7	-1.8	270
Human 2 (right)	0.8	-2.0	81

Table 5.4: Human poses in use case 1.b. All values are relative to the map coordinate system.

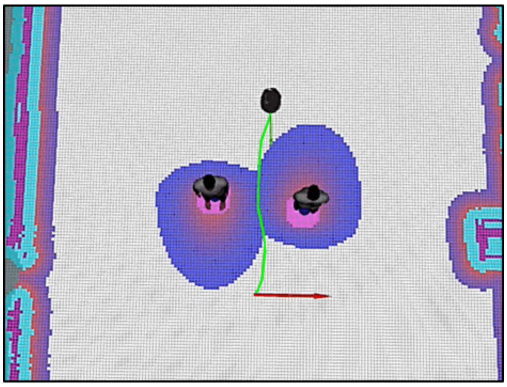


Figure 5.2: Use case 1.b. Although the two people are side by side (from the position point of view), the robot now passes through them because their orientations do not satisfy the conditions necessary to identify a side-by-side formation (in Kendon's terminology).

Use case 2.a: 2-people vis-a-vis In this use case, the two humans are positioned in space defining a vis-a-vis formation (Table 5.5). As a result, the TurtleBot robot plans a trajectory that avoids the group to reach the goal pose indicated with the red arrow in Figure 5.3.

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-0.7	-1.5	0
Human 2 (right)	0.6	-1.6	180

Table 5.5: Human poses in use case 2.a. All values are relative to the map coordinate system.

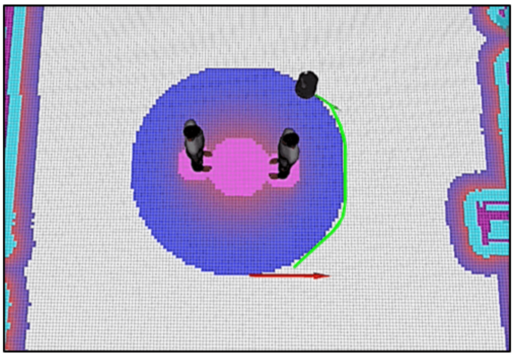


Figure 5.3: Use case 2.a. The robot detects the vis-a-vis formation and consequently plans a path that does not cross it.

Use case 2.b: 2-people subset without vis-a-vis structure This use case, as opposed to the previous one, shows the situation in which the poses of the two humans do not satisfy the condition of a vis-a-vis formation. In particular, in this scenario, the condition of correct frontal positioning (first equation of 4.8) is violated, in which the two members of the formation should face each other.

An important observation about this scenario is that, as shown in Figure 5.4, the path generated by the global planner does not pass through the group of people even if the formation is not detected. This behavior (also observed in some runs relating to the other use cases of type b) can be justified by the observation that if the two people are still very close to each other, due to the related Personal Spaces, the planner still chooses the path that does not cross the human set.

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-0.8	-1.0	0
Human 2 (right)	0.6	-2.0	180

Table 5.6: Human poses in use case 2.b. All values are relative to the map coordinate system.

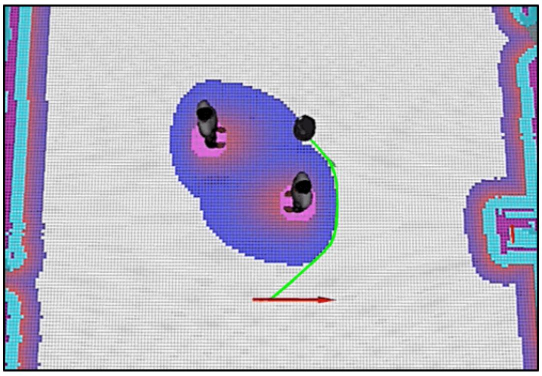


Figure 5.4: Use case 2.b. Two humans not in vis-a-vis formation.

Use case 3.a: 2-people in V-shape arrangement In this use case and the following, we test the correct recognition of the last type of two-members formations.

In this first situation, since the two humans are placed in such a way that their face directions converge (Fig. 5.5), they constitute a V-shape group (from the theoretical point of view) and, consequently, the robot plans a trajectory that circumnavigates the space shared by the formation, i.e., the *p-space*.

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-1.0	-1.5	0
Human 2 (right)	0.0	-3.0	60

Table 5.7: Human poses in use case 3.a. All values are relative to the map coordinate system.

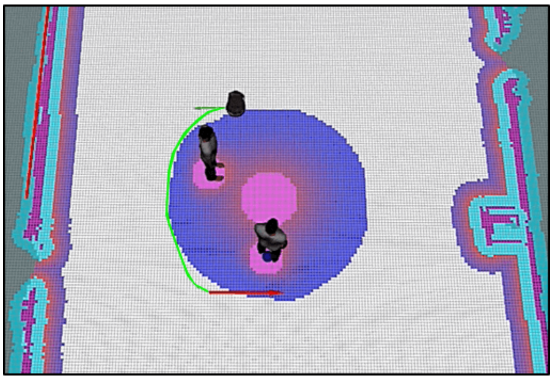


Figure 5.5: Use case 3.a. Human group with V-shape structure identified by the robot.

Use case 3.b: 2-people subset without V-shape structure This use case aims to test the correct recognition of V-shape formations but, unlike the previous one, from the opposite point of view, i.e. verifying the correct non-identification of the aforementioned formation (in the same way as described in cases 1.b and 2.b for *sbs* and *vav* formations respectively).

In this situation, the orientations of the two people are divergent, not respecting the convergence constraint required by the V-shape formation, and then not generating any group. As a result, the global planner produces a path that directly crosses the set of humans, as shown in Figure 5.6.

Human	x (m)	y (m)	θ (°)
Human 1 (left)	-0.5	-2.2	0
Human 2 (right)	1.5	-1.5	150

Table 5.8: Human poses in use case 3.b. All values are relative to the map coordinate system.

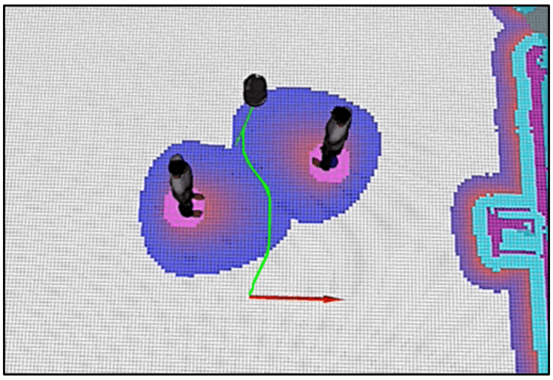


Figure 5.6: Use case 3.b. Human group without V structure correctly not identified by the robot.

Use case 4.a: 5-people formation This use case and the following, on the heels of the previous ones, test the correct recognition of formations with more than two members.

Specifically, in this first scenario, we consider a set of five people [2], where each person is located inside the frontal space of the others. Consequently, the identification conditions of the formation are satisfied and the group is

[2] In this use case, as well as in all the others, unless explicitly stated, the Distance Constraint is considered satisfied (see Section 4.3.1).

correctly taken into account by the TurtleBot robot during the navigation, as shown in Figure 5.7.

Human (clockwise from top-left)	x (m)	y (m)	θ (°)
Human 1	-1.05	-1.0	300
Human 2	1.05	-1.0	230
Human 3	1.20	-2.6	140
Human 4	0.0	-3.5	90
Human 5	-1.20	-2.5	30

Table 5.9: Human poses in use case 4.a. All values are relative to the map coordinate system.

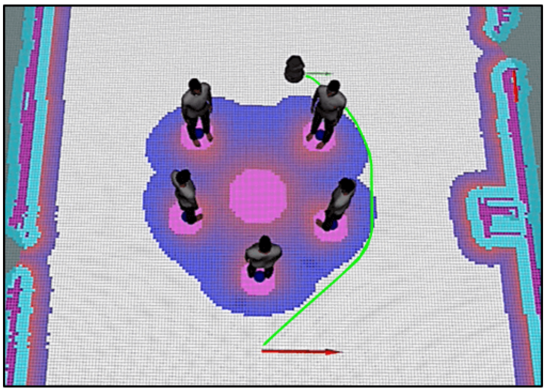


Figure 5.7: Use case 4.a. The 5-members human group represented in the cost map and taken into account by the robot during its navigation to the goal pose.

Use case 4.b: 5-people subset without formation structure As mentioned above, this use case aims to test the correct recognition of groups with more than two members using, however, an opposite approach, compared to the previous one.

In detail, since only two out of five members satisfy the orientation constraint (see Section 4.3.1), the robot does not recognize the set as a formation and consequently selects a path that crosses it (Fig. 5.8). It is important to realize that the Distance Constraint is satisfied, but this is not sufficient to consider the human set as a group.

Human (clockwise from top-left)	x (m)	y (m)	θ (°)
Human 1	-1.05	-1.0	300
Human 2	1.05	-1.0	230
Human 3	1.20	-2.6	320
Human 4	0.0	-3.5	270
Human 5	-1.20	-2.5	210

Table 5.10: Human poses in use case 4.b. All values are relative to the map coordinate system.

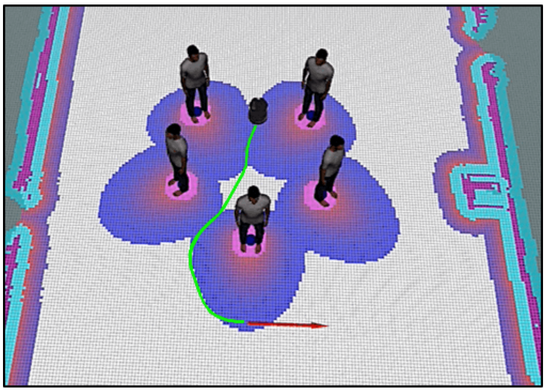


Figure 5.8: Use case 4.b. The path planned by the robot passes correctly through the group of people.

Use case 5: Multiple static formations This last use case about static formations is a combination of the previous scenarios since it includes simultaneously multiple subsets of people, organized into different types of groups. The main goal of this class of simulations is to test the correct identification and partitioning of the different groups in the environment, or, stated differently, the exact clustering of humans in the respective groups.

For this reason, in this scenario, we set up an environment with three subsets of people, who constitute a V-shape, a side-by-side, and a 3-members formation respectively (Fig. 5.9).

During navigation, the TurtleBot robot is able to correctly partition the set of people (provided by the people detection module simulated via ROS) and, updating the corresponding positions of the cost map, determines a socially acceptable path that does not pass through any of the identified groups (Fig. 5.10).

Human (clockwise from top-left)	x (m)	y (m)	θ (°)
Human 1 (V-shape)	-1.6	3.5	320
Human 2 (V-shape)	-1.6	2.3	50
Human 3 (3-members)	-0.2	-0.2	270
Human 4 (3-members)	0.6	-1.5	165
Human 5 (3-members)	-1.3	-1.5	350
Human 6 (sbs)	-0.75	-4.50	270
Human 7 (sbs)	0.5	-4.55	260

Table 5.11: Human poses in use case 5. All values are relative to the map coordinate system.

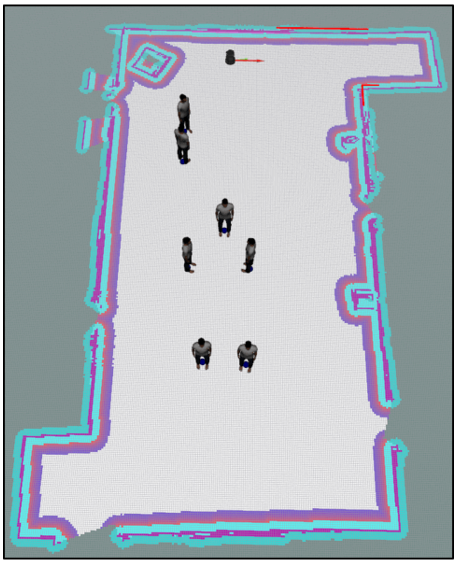


Figure 5.9: Use case 5. Top view of the hallway.

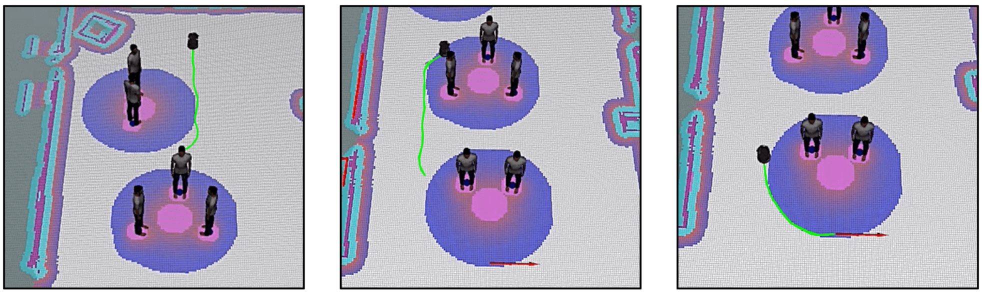


Figure 5.10: Use case 5. Path planned by the robot at different time instants during a simulation run.

5.1.2 Handling of dynamic groups

We will now analyze environments populated only with dynamic groups.

Use case 6.a: 3-people in V-like structure This use case is intended to analyze the correct recognition of the V-like dynamic formation structure.

In details, the environment is populated with a set of moving humans, whose distance, orientation, and relative speed fulfill the constraints imposed by the model defined in Section 4.3.2, and whose arrangements in space determine a V-like structure. The goal is to verify that the TurtleBot robot is capable of identifying the group and, consequently, generating a path that does not hinder it.

The simulation result (Fig. 5.11) shows that providing the robot with a goal position localized on the direction of movement of the group (behind it), results in a path that does not pass through the dynamic formation, as expected.

Human (from left to right)	x (m)	y (m)	θ (°)	vel (m/s)
Human 1	-1.0	-7.2	90	0.4
Human 2	-0.25	-6.75	90	0.4
Human 3	0.75	-7.25	90	0.4

Table 5.12: Human starting poses and velocities in use case 6.a. All values are relative to the map coordinate system.

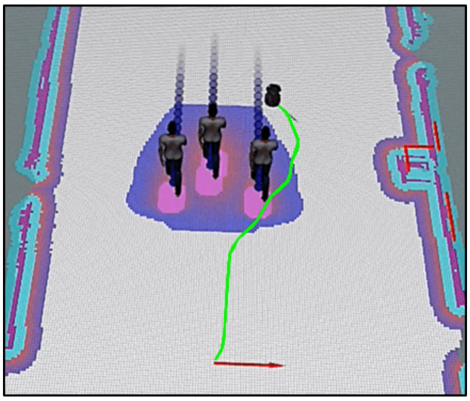


Figure 5.11: Use case 6.a. Path planned by the robot when its goal position is located on the motion direction of the human group.

Use case 6.b: 3-people subset without V-like structure In this use case, which is the dual compared to the previous one, we configure the environment with a group of moving people who do not generate a formation. In particular, the set of three people does not satisfy the Dynamic Orientation Constraint, required by the model (Section 4.3.2), as the directions of motion of the three humans are not parallel (Table 5.13). As a result, the robot, not identifying the presence of any formation, generates a "direct" path towards the target position (chosen in the same way as that of the previous use case) (Fig. 5.12).

Human (from left to right)	x (m)	y (m)	θ (°)	vel (m/s)
Human 1	-1.0	-7.2	120	0.4
Human 2	-0.25	-6.75	90	0.4
Human 3	0.75	-7.25	60	0.4

Table 5.13: Human starting poses in use case 6.b. All values are relative to the map coordinate system.

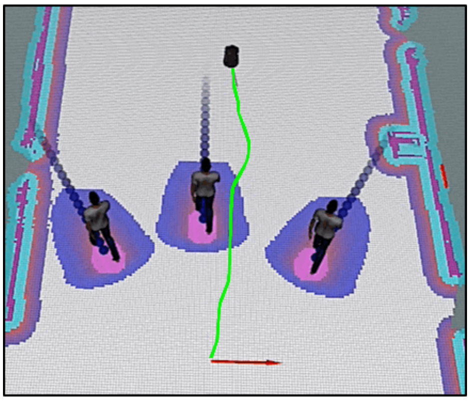


Figure 5.12: Use case 6.b. The path planned by the robot is direct to the target pose because the moving humans are treated as single entities.

Use case 7: Multiple dynamic formations This last use case examines a more complex dynamic scenario, consisting of two moving groups having a V-shape and a side-by-side structure respectively. This type of simulations has two principal goals, i.e., analyze how the model handles side-by-side dynamic formations and verify the proper partitioning of multiple human groups in the dynamic case.

The simulation result (Fig. 5.13) shows that the robot is able both to suitably cluster the humans identified by the people detector, and to adequately represent these formations in space in order to generate a socially acceptable path towards the target position (located on the opposite side of the hallway from its starting position), as desired.

Human (from left to right)	x (m)	y (m)	θ (°)	vel (m/s)
Human 1 (V-shape)	-2.0	-4.7	90	0.3
Human 2 (V-shape)	-1.0	-5.1	90	0.3
Human 3 (V-shape)	0.0	-4.7	90	0.3
Human 4 (sbs)	0.25	-7.7	90	0.2
Human 5 (sbs)	1.5	-7.75	90	0.2

Table 5.14: Human starting poses and velocities in use case 7. All values are relative to the map coordinate system.

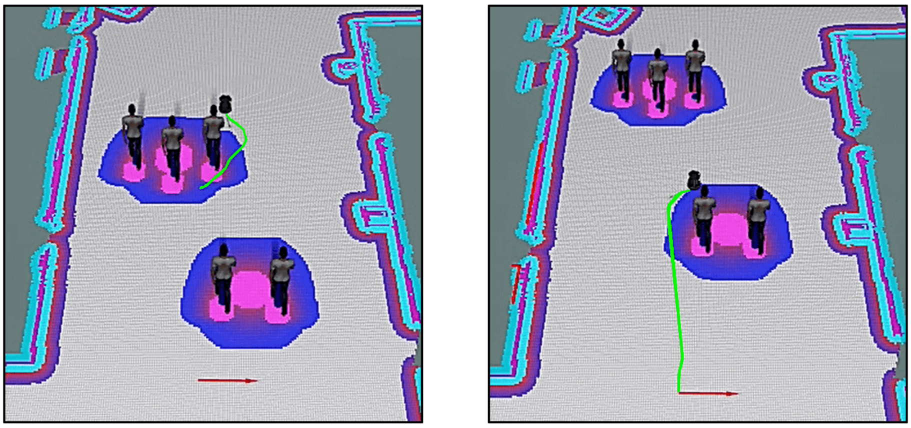


Figure 5.13: Use case 7. Path planned by the robot at different time instants during a simulation run, when multiple dynamic groups populate the environment.

5.2 Comparison Tests

We now present a test set aiming at comparing the baseline model with our solution, in terms of both performance and sociability. We expected that our proposed framework would result in a decline in performance (i.e., an increase of the average time the TurtleBot takes to reach the target pose) but an increase in the level of sociability of the robot.

Here too, the test set covers all the structures analyzed in the model (Section 4.3), i.e., single static formations with two or more members, multiple static formations, and single and multiple dynamic groups.

Before analyzing the different use cases, however, it is necessary to define two metrics that allow us to compare the two solutions. In this regard, as mentioned above, to measure the performance of the two approaches, we use the **Navigation Time** the robot spends to reach the goal pose from the starting position. As for the sociability of the robot's behavior, we adopt an index called **Social Group Index** (**SGI**), proposed by Truong et al. [51], whose mathematical formulation is described by the following equation:

$$\text{SGI (r)} = \max_{k=1:K} \exp\left(-\frac{(x_r - x_k^O)^2}{2\sigma^2_{x\_k}} - \frac{(y_r - y_k^O)^2}{2\sigma^2_{y\_k}}\right) \tag{5.1}$$

where K is the number of human groups in the environment, $r = (x_r, y_r)$ is the current position of the robot, $O_k = (x_k^O, y_k^O)$ is the center of the k-th formation (both the coordinates of r and O are in the map coordinate system), and $\sigma^2_{x\_k}$, $\sigma^2_{y\_k}$ are the diagonal entries of the Σ_k covariance matrix, which determines the shape of the k-th formation.

In general terms, this index, which ranges from 0 to 1, describes the sense of discomfort that the movement of the robot causes in humans (members of groups), populating the surrounding environment. Therefore, the higher the index value, then the lower is the level of sociability of the autonomous agent. The SGI index evolves based on the position of the robot in the environment during its navigation towards the final pose. Therefore, it is a function of both position and time.

To acquire more reliable results, we carried out 10 executions for each described use case. Consequently, regarding the Navigation Time index, we use a box plot to represent the overall values obtained in the different executions; as for the SGI index, instead, we illustrate the time series corresponding to one of the ten executions.

5.2.1 Hallway populated with static formations

In this section, we will compare the two approaches considering environments populated only with static groups.

Use case 1: side-by-side formation In this use case, we examine the behavior of the baseline framework and our approach, when the environment is populated with a side-by-side formation. The poses of humans and the starting and goal locations of the TurtleBot robot are shown in Table 5.15.

Human	x (m)	y (m)	θ (°)
H. 1 (left)	-1.0	-1.5	240
H. 2 (right)	0.4	-2.3	220

Position	x (m)	y (m)
Start	0.0	0.0
Goal	-1.0	-3.7

Table 5.15: Use case 1. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

As illustrated in Figure 5.14, by setting the same conditions, the two solutions provide different results. While the baseline generates a trajectory that crosses the group of people, our approach, once identified the formation, generates an alternative path that gets around the group.

As a result, the average time that the robot takes to reach the final position increases (43.3 s instead of 35.3 s, Fig. 5.15-a), however, the level of discomfort caused to the humans by its movement decreases considerably, as shown in graph 5.15-b.

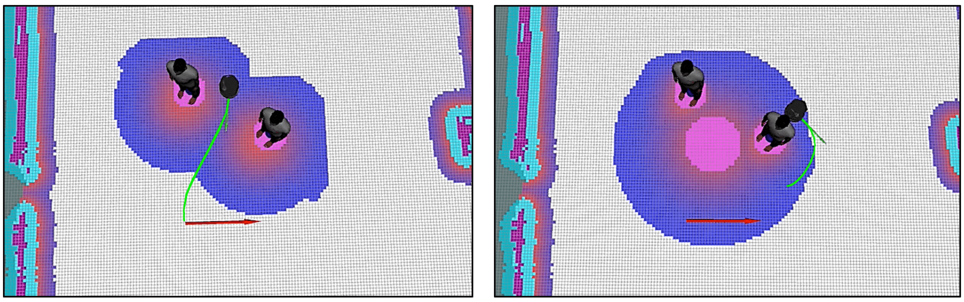
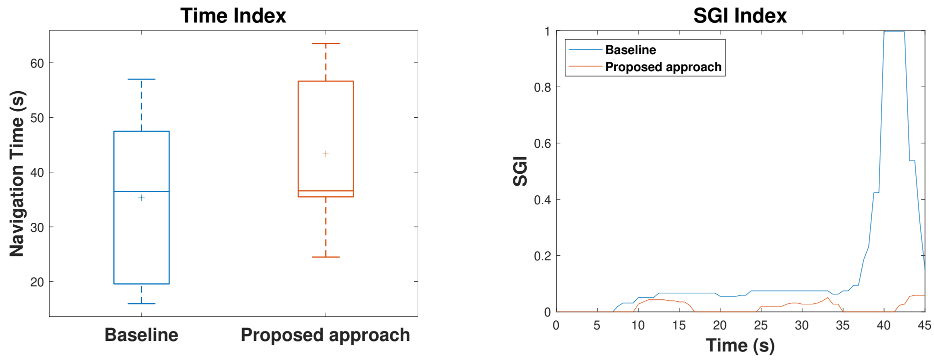


Figure 5.14: Paths generated by the baseline (left) and by our solution (right), in use case 1.



(a) (b)

Figure 5.15: Use case 1. Evaluation indices comparing the Baseline and the Proposed approach.

Use case 2: V-shape formation In this use case, we compare the two solutions in the presence of a formation with a V-shape structure. The poses of humans and the starting and goal locations of the TurtleBot robot are shown in Table 5.16.

Human	x (m)	y (m)	θ (°)
H. 1 (left)	-1.5	-1.5	0
H. 2 (right)	0.0	-2.5	105

Position	x (m)	y (m)
Start	0.0	0.0
Goal	-1.4	-3.0

Table 5.16: Use case 2. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

As illustrated in Figure 5.16, also in this case, the two approaches generate different results. Furthermore, the two trajectories have properties comparable to those of the previous case. In particular, our method generates a path that takes a longer average time to be traveled (30.3 s instead of 21.5 s), but which, on the other hand, is socially more acceptable to humans (Fig. 5.17).

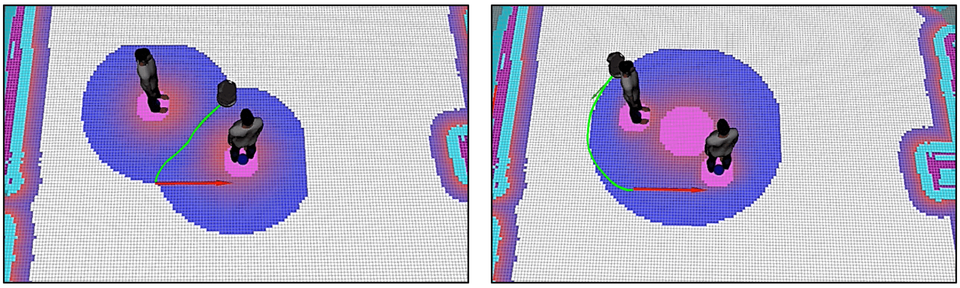
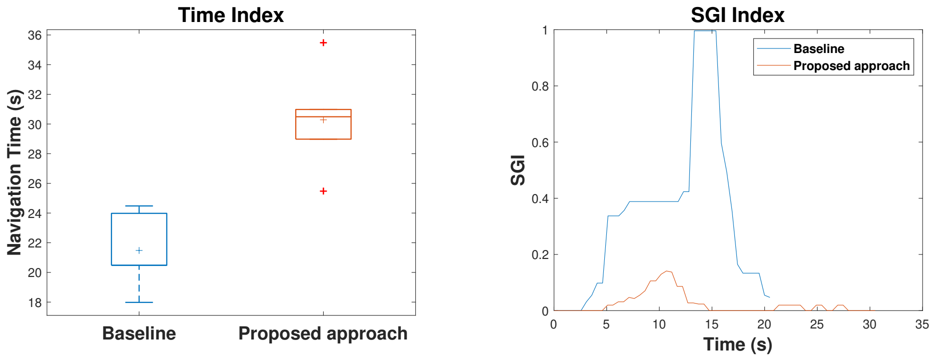


Figure 5.16: Paths generated by the baseline (left) and by our solution (right), in use case 2.



(a) (b)

Figure 5.17: Use case 2. Evaluation indices comparing the Baseline and the Proposed approach.

Use case 3: vis-a-vis formation In this use case, we compare the two approaches in a scenario populated with a vis-a-vis formation. The poses of humans and the starting and goal locations of the robot are provided in Table 5.17.

Human	x (m)	y (m)	θ (°)
H. 1 (left)	-1.2	-1.0	345
H. 2 (right)	0.5	-2.0	150

Position	x (m)	y (m)
Start	0.0	0.0
Goal	-1.1	-2.7

Table 5.17: Use case 3. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

The simulation results show a behavior similar to that found in previous use cases, characterized by an increase in the average navigation time (28.3 s instead of 16.3 s) and a decrease in the level of discomfort that the movement of the autonomous agent causes in the members of the formation (Fig. 5.19).

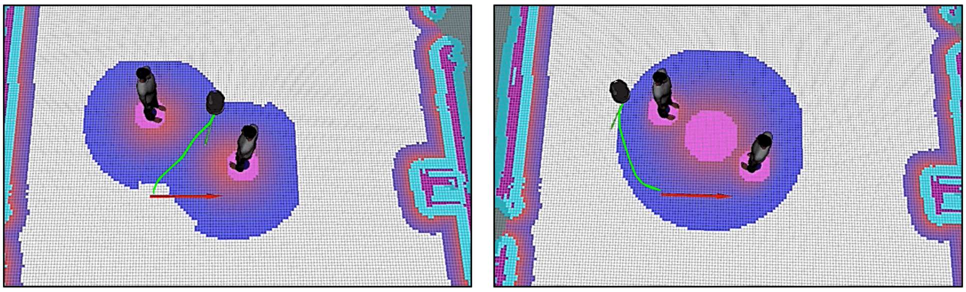


Figure 5.18: Paths generated by the baseline (left) and by our solution (right), in use case 3.

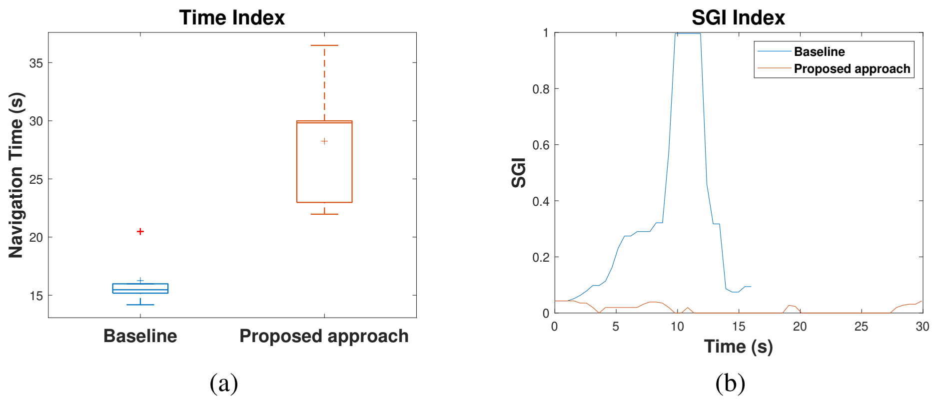


Figure 5.19: Use case 3. Evaluation indices comparing the Baseline and the Proposed approach.

Use case 4: 3-people formation In this use case, we analyze the two approaches in a scenario populated with a 3-members formation. The poses of humans and the starting and goal positions of the robot are provided in Table 5.18.

Human (left to right)	x (m)	y (m)	θ (°)
H. 1	-1.0	-1.5	290
H. 2	0.0	-4.0	90
H. 3	0.75	-1.5	240

Position	x (m)	y (m)
Start	0.0	0.0
Goal	0.1	-6.0

Table 5.18: Use case 4. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

Here too, the baseline plans a path that crosses the group of people while our approach determines a trajectory that circumnavigates it (Fig. 5.20). However, unlike previous cases, in this scenario, the average time that the robot takes to reach the target position is higher for the baseline than for our solution (49.8 s instead of 77.6 s). The proposed approach, therefore, generates a path that is not only more comfortable for humans but also faster (Fig. 5.21).

The explanation for this result is that, due to the re-planning mechanism, the baseline framework does not always generate an exact trajectory to the destination, but, in some cases, it only produces the most promising path. Consequently, in situations such as the current one, in which the number of members of the formation grows (the same phenomenon will, indeed, be found also in the following simulations), this mechanism can lead to a temporary "deadlock" of the robot, where it frequently changes its trajectory. This inevitably determines a longer navigation time. This event does not occur, instead, in our approach since by making the space occupied by the group (almost) impassable, the planner determines, a priori, an alternative trajectory.

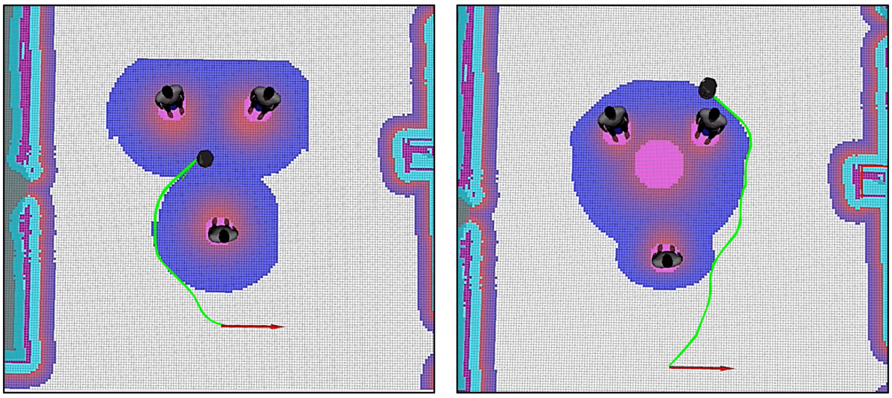
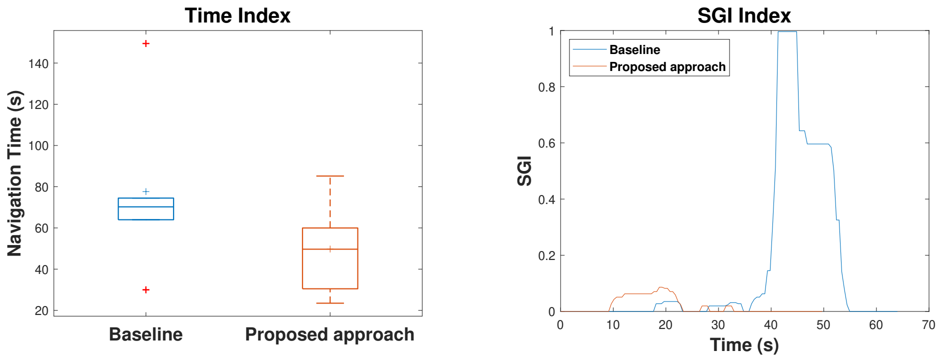


Figure 5.20: Paths generated by the baseline (left) and by our solution (right), in use case 4.



(a) (b)

Figure 5.21: Use case 4. Evaluation indices comparing the Baseline and the Proposed approach.

Use case 5.a: multiple formations (3 groups) In this simulation, we examine the baseline and our approach behaviors in a more complex environment, populated with multiple static groups. This test aims at comparing the two solutions in a scenario that represents as much as possible a real situation. The human poses (which determine a 3-members, a V-shape, and a vis-a-vis formations) and the starting and goal locations of the TurtleBot robot are provided in Table 5.19. Figure 5.22 provides a top view of the hallway.

Human (→↓)	x (m)	y (m)	θ (°)
H. 1 (3-m)	-1.0	4.5	320
H. 2 (3-m)	0.25	3.3	110
H. 3 (3-m)	1.0	4.5	210
H. 4 (V)	-1.1	0.5	320
H. 5 (V)	-1.1	-0.7	50
H. 6 (vav)	-1.2	-4.5	350
H. 7 (vav)	0.5	-4.5	165

Position	x (m)	y (m)
Start	0.0	6.5
Goal	-0.3	-6.7

Table 5.19: Use case 5.a. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

As illustrated in Figures 5.23 and 5.24, the robot behaves in the same way in the regions of free space, while it follows different trajectories in correspondence of the human groups, crossing them in the baseline approach while circumventing them in our solution. Consequently, the proposed technique has a generally longer navigation time (97.6 s instead of 93.9 s, Fig. 5.25-a), however, its level of sociability is decidedly higher, as shown in Figure 5.25-b. The two peaks illustrated in this last figure correspond precisely to the crossing of two of the three groups of people, performed by the robot in the baseline approach.

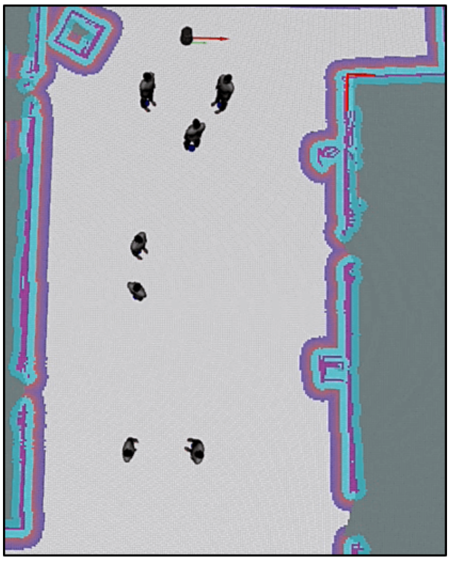


Figure 5.22: Top view of the hallway in use case 5.a.

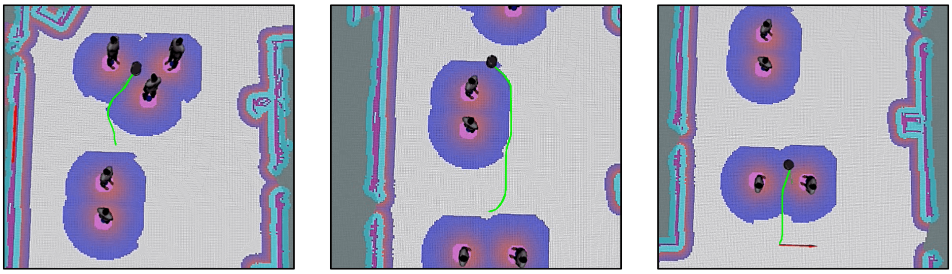


Figure 5.23: Use case 5.a. Path generated by the baseline.

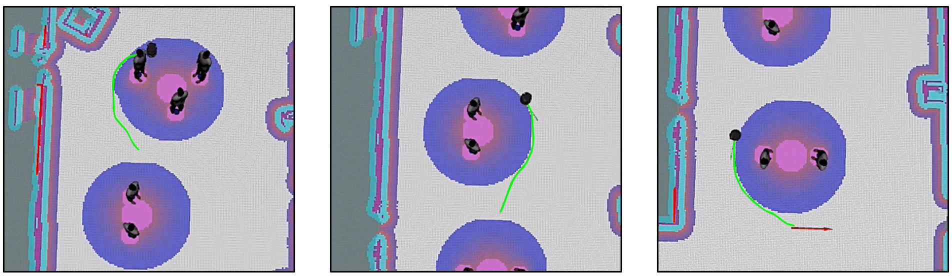
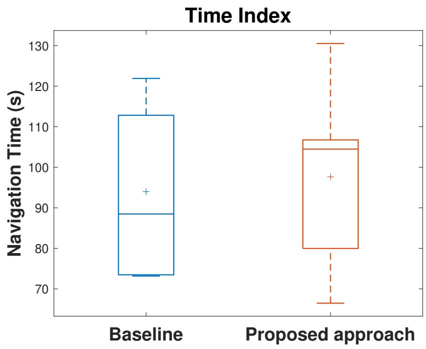
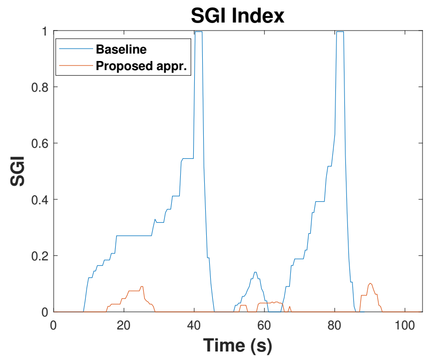


Figure 5.24: Use case 5.a. Path generated by our approach.



(a)



(b)

Figure 5.25: Use case 5.a. Evaluation indices comparing the Baseline and the Proposed approach.

Use case 5.b: multiple formations (4 groups) Like the previous situation, in this use case, we compare the two approaches in an environment populated with several groups. In particular, we set up a scenario with a 4-members, a V-shape, a side-by-side, and a 3-members formations (Fig. 5.26). The data of the poses of each human and the starting-target locations of the robot are in the Table 5.20.

Human (→↓)	x (m)	y (m)	θ (°)
H. 1 (4-m)	-1.2	4.5	320
H. 2 (4-m)	-0.7	2.7	85
H. 3 (4-m)	0.7	2.8	110
H. 4 (4-m)	1.2	4.5	210
H. 5 (V)	-1.6	0.5	320
H. 6 (V)	-1.6	-0.7	50
H. 7 (sbs)	0.3	-2.0	240
H. 8 (sbs)	1.7	-2.8	220
H. 9 (3-m)	-1.3	-5.5	350
H. 10 (3-m)	-0.2	-4.2	270
H. 11 (3-m)	0.6	-5.5	165

Position	x (m)	y (m)
Start	-0.2	6.6
Goal	0.0	-7.0

Table 5.20: Use case 5.b. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

Here too, the generated trajectories exhibit a higher level of sociability of our approach (Fig. 5.28) compared to the baseline (Fig. 5.27), since the path produced by our solution does not cross any of the groups occupying the hallway. It is also important to underline that, as in use case 4, also here, the average navigation time of the robot is lower in our solution than in the baseline (86.6 s instead of 110.9 s) (Fig. 5.29). The motivation is the same as that provided in that simulation.

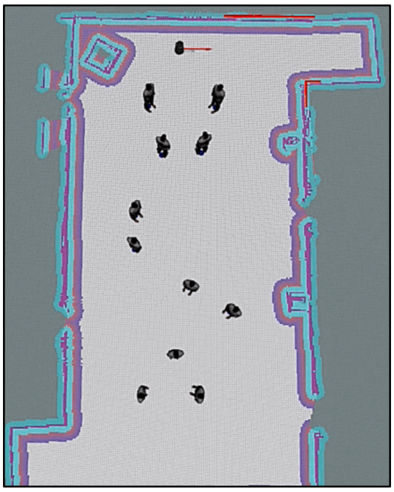


Figure 5.26: Top view of the hallway in use case 5.b.

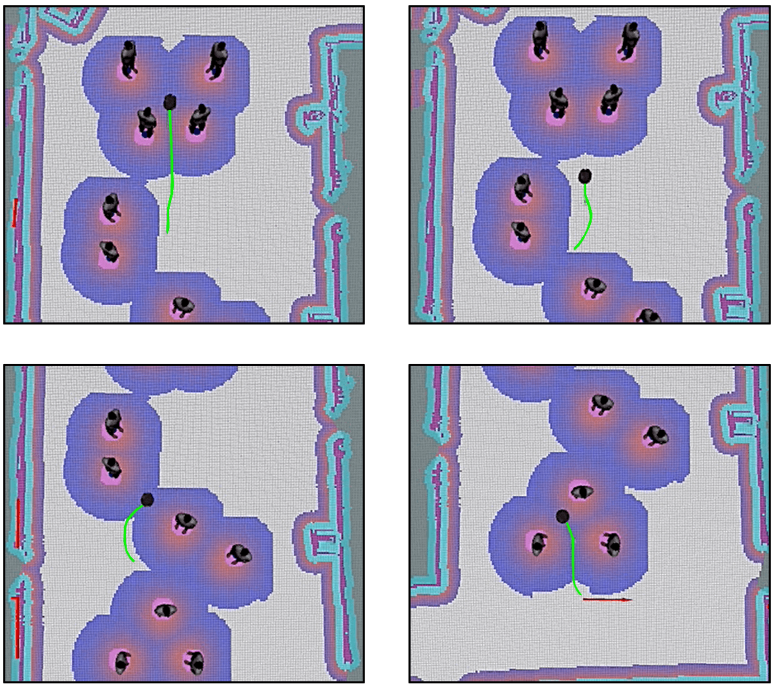


Figure 5.27: Use case 5.b. Path generated by the baseline.

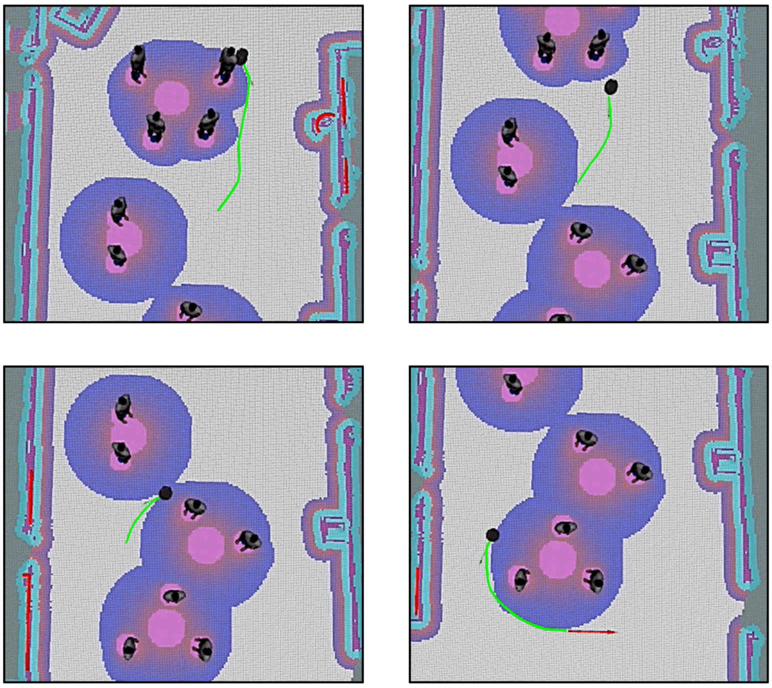


Figure 5.28: Use case 5.b. Path generated by our approach.

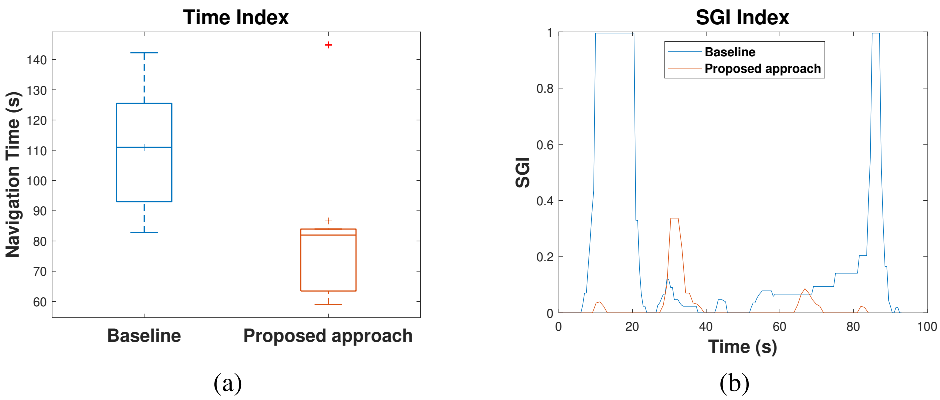


Figure 5.29: Use case 5.b. Evaluation indices comparing the Baseline and the Proposed approach.

5.2.2 Hallway populated with dynamic groups

We will now analyze a use case where the environment is populated only with moving humans and groups.

Use case 6: dynamic group with side-by-side pattern This use case compares the trajectories generated by the baseline framework and by our approach in the presence of people walking down the hallway in the opposite direction to the robot movement. In particular, we set the environment with three moving people (Table 5.21), two of which form a group with a side-by-side structure (membership to the formation is only determined, in this case, by the relative distance between the different humans). As for the robot's initial and target poses, they are positioned on the direction of the group's motion, to verify the presence or absence of a deviation of the chosen trajectory by the optimal-direct path that would occur if people are not present.

Human (→)	x (m)	y (m)	θ (°)	vel (m/s)
H. 1	-2.0	-5.1	90	0.3
H. 2	-0.5	-5.1	90	0.3
H. 3	1.75	-5.1	90	0.3

Pos.	x (m)	y (m)
Start	-2.3	4.0
Goal	-1.3	-6.0

Table 5.21: Use case 6. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

As shown in Figure 5.30, while the path determined by the baseline approach is exactly the "direct" one, i.e., the one that passes through the group of humans, the route generated by our technique presents a deviation that allows the agent to circumvent the formation. The consequence of this behavior is, obviously, a longer navigation time of the robot (the average time is 35.8 s instead of 31.2 s) but also a notable lowering of the sense of discomfort caused to people, as illustrated in Figure 5.31.

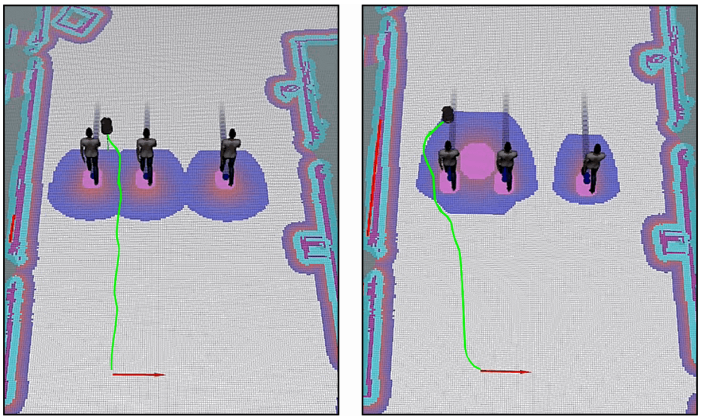
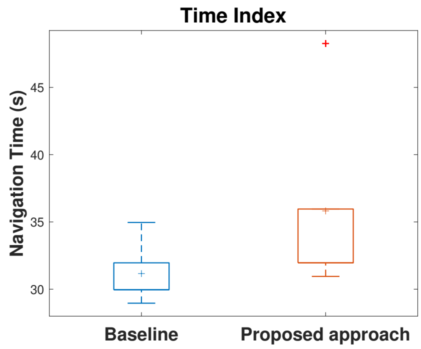
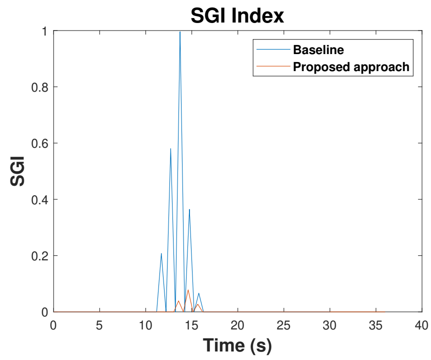


Figure 5.30: Paths generated by the baseline (left) and by our solution (right), in use case 6.



(a)



(b)

Figure 5.31: Use case 6. Evaluation indices comparing the Baseline and the Proposed approach.

5.2.3 Mixed scenario: the overtaking maneuver

In this last section, we examine a use case that includes both static and dynamic groups.

Use case 7: the overtaking maneuver In this use case, we analyze a typical situation that can occur inside a hallway, that is the overtaking maneuver of a group of standing humans. We also included in the environment a group of dynamic people who crosses the corridor in the opposite direction, to further limit the robot's movement options. Data about the starting positions and speeds of humans and the starting and target locations of the robot are provided in Table 5.22.

Human (→)	x (m)	y (m)	θ (°)	vel (m/s)
H. 1 (st.)	-2.25	0.0	280	0.0
H. 2 (st.)	-0.65	0.0	260	0.0
H. 3 (dyn.)	1.5	-5.1	90	0.2
H. 4 (dyn.)	2.8	-5.1	90	0.2

Pos.	x (m)	y (m)
Start	-1.6	4.0
Goal	-1.4	-6.0

Table 5.22: Use case 7. (Left) Human poses - (Right) Robot start and target positions. All values are relative to the map coordinate system.

As shown in Figure 5.32, in this case also, the two approaches generate different paths. While the baseline produces a trajectory that crosses the group of standing people, our solution produces a trajectory that overtakes it. As a result, the proposed approach shows a higher level of sociability but a lower performance (Fig. 5.33). The clear difference in navigation time (63.1 s instead of 43.9 s) is mainly due to the presence of the dynamic group that occupies the opposite lane of the corridor; in fact, before starting the overtaking maneuver, the robot allows people in the opposite lane to pass and move away from it, as can be seen in Figure 5.32.

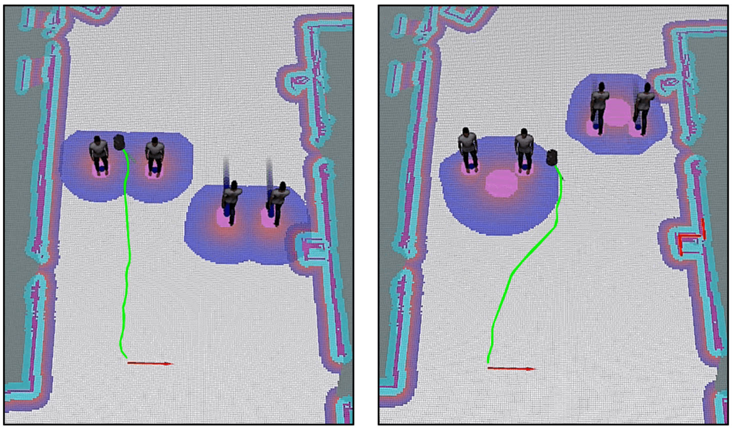
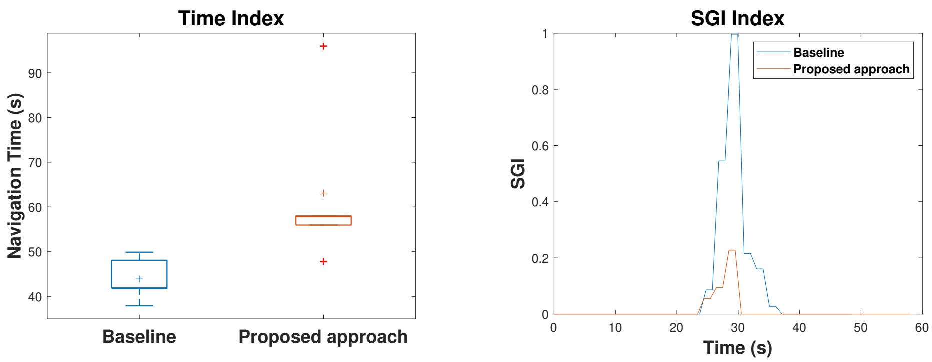


Figure 5.32: Paths generated by the baseline (left) and by our solution (right), in use case 7.



(a) (b)

Figure 5.33: Use case 7. Evaluation indices comparing the Baseline and the Proposed approach.

Summary of results

The purpose of the Evaluation Tests was mainly to showcase and evaluate the proposed group-aware navigation framework. In particular, the aim was to test the correct recognition of human formations when present in the environment by the robot, and the consequent planning of group-aware paths. Based on the obtained results, we can conclude that the framework behaves as expected, given that it generates paths that circumnavigate the human formations while passing through simple subsets of individual people (i.e. non-groups).

On the other hand, the purpose of the Comparison Tests was to compare the baseline framework to our proposed work, both in terms of performance and sociability. As expected, the simulation results show that the navigation time of the robot is higher in our approach than in the baseline (even if there is only a slight difference and, in some cases, our solution takes less time). However, our solution generates paths with a clearly higher level of sociability, or, in other words, with a much lower level of discomfort for humans. The trade-offs between time and sociability need to be considered case by case but, in many real-world social environments, it would be desirable that a robot avoids disrupting a group interaction even if that comes at the cost of taking a bit longer to achieve its goal.

It is worth highlighting, however, that the index used to measure the level of sociability of the two approaches, although reasonable, is based only on the theoretical definition of human formations. In other words, given the simulated environment, it does not consider the actual level of comfort experienced by group members when the robot moves around the environment. However, it provides at least an estimate proportional to the real sociability levels of the two approaches [3].

To conclude, the proper functioning of the framework, concerning the identification and handling of formations, also depends on the correct setting of the model parameters. In our simulations, given that the testing environment (the hallway) was not too restricted in dimensions, the parameters were set strictly following the theoretical models, without taking the available space into account. However, in more narrow environments (such as corridors), the parameters should be adjusted accordingly – for example, reducing the distance thresholds between group members.

[3] The measure of the level of sociability of a robot's behavior is far from being an objective measure, as it depends on the people who evaluate it; it depends, indeed, on the level of familiarity of the people involved in the test with the robots, on their emotional state, on their culture, etc.

Despite some initial parameterization tuning for a specific environment, we are confident that the presented framework can positively contribute to human-aware path planning.

Chapter 6

Conclusions

The purpose of this thesis was to develop a human and group aware navigation framework that would allow a robot to move in narrow spaces (such as hallways of offices and hospitals) in a socially acceptable way. Specifically, the robot should have been able to not only keep a comfortable distance from people, but also to identify group formations and keep a certain distance from them too (i.e. avoiding to cross the group).

To do this, we employed a state-of-the-art A*-based global path planner that, using a linear combination of different parameters as a cost function, can determine the minimum cost path to the desired location. Among these parameters, we included the requirement to maintain a minimum distance to humans, based on the Personal Space model proposed by Hall [5]. In particular, we used a combination of two 2D Gaussian cost functions to represent individual people in the environment.

To reason around the spatial arrangements of people in the environment, we designed a geometric model based on the F-formations theory proposed by Kendon [10][11], and on recurring dynamic group patterns [13][14] of moving groups. In particular, given the poses (positions and orientations) and velocities of the people detected in the environment surrounding the autonomous agent, this model could determine the presence of groups, both static and dynamic. Subsequently, the formations were converted into 2D Gaussian cost functions which, based on the group structure, were centered at different points in space.

The resulting framework was then tested in a hallway scenario through two test sets. The first test set had the purpose of evaluating the proper functioning of the solution in terms of the identification and handling of human groups. The second test set compared the baseline framework (i.e., the state-of-the-art A*-based global path planner without our Personal space and Group space models) with the one we proposed, in terms of both performance and sociability.

The results of the simulations showed that our framework is able to handle the presence of human formations and that, although with a slightly lower performance than the baseline (in terms of total distance and time to reach the goal), it clearly shows a higher level of human and group awareness. In none of the simulations, indeed, the path generated by the global planner passed through the group(s) of people.

These results are compatible with our expectations; it is important to underline, however, that as mentioned in Chapter 1, in human-aware navigation it is necessary to take into account further constraints (e.g., keep a certain distance to humans) than those strictly related to the performance and the efficiency of motion, such as the distance traveled and the elapsed time to reach the goal.

From all these considerations, we can conclude that the question investigated in this thesis has successfully been answered.

6.1 Future work

This thesis represents the starting point of many possible future works in different directions.

In Chapter 5, the framework was evaluated and compared with the baseline. However, this evaluation was carried out only in simulation. The implementation of the proposed solution in a real robot would entail two notable improvements: on the one hand, we would evaluate and test the integration of the model with a real people detector (as explained above, we used a simulated people detector); on the other hand, there would be a more reliable assessment of the sociability level of the robot's navigation. This last aspect was evaluated in Chapter 5 but based only on the theoretical concepts of the formations. Testing the framework in a real environment with real people would allow a more trustworthy evaluation of the robot's social behavior, thanks to the feedback and judgments of the participants in the experiments.

On the other hand, this improvement would involve several challenges regarding the acquisition, merging, and processing of data on people's poses and velocities. In particular, regarding the detection of people in the environment and the estimation of their speed of movement, an onboard sensing system (e.g., a laser rangefinder or an RGB-camera) would be sufficient. However, the acquisition of data about people's orientations would probably require an additional offboard sensing system – for example, a camera mounted on the ceiling of the room, which tracks people from above. A deeper study about the state-of-the-art people detection and tracking systems would, therefore, be needed to implement the proposed solution in a real autonomous agent.

Moving to the topic of static group identification and handling model, at least two improvements are possible. This model was implemented based mainly on the theory proposed by Kendon [10][11]. The continuous research in fields ranging from sociology to robotics, however, has improved this theoretical model continuously. To mention an example, the recent work by Hedayati et al. [57], starting from Kendon's models, observed how static formations have larger or smaller dimensions based on whether the group members are interacting with each other using only their voices or also a shared electronic device. These aspects could, therefore, be included in the implemented solution obtaining a model that is more conforming to real situations.

Another improvement regarding the group detection model concerns the prediction of the motion of dynamic groups. The baseline framework, as detailed in Chapter 4, implements a time-dependent planner, thus allowing us to take into account the prediction of the movement of people during the path planning process. This aspect could be extended to the case of dynamic formations, developing a method that, analyzing the trajectories of the moving people, could determine the potential formation of groups in the incoming future and, in that case, their poses in space.

Bibliography

[1] *Pratande robotar sköter logistiken*. Accessed: 2020-06-02. URL: https://www.sll.se/verksamhet/halsa-och-vard/Aktuella-projekt-Halsa-och-vard/Nya-Karolinska-Solna/pratande-robotar-skoter-logistiken/.

[2] Konstantinos Charalampous, Ioannis Kostavelis, and Antonios Gasteratos. «Recent trends in social aware robot navigation: A survey». In: *Robotics and Autonomous Systems* 93 (Apr. 2017). DOI: 10.1016/j.robot.2017.03.002.

[3] Thibault Kruse et al. «Human-Aware Robot Navigation: A Survey». In: *Robotics and Autonomous Systems* 61 (Dec. 2013), pp. 1726–1743. DOI: 10.1016/j.robot.2013.05.007.

[4] S. U. Kamat and K. Rasane. «A Survey on Autonomous Navigation Techniques». In: *2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC)*. 2018, pp. 1–6.

[5] Edward T. Hall. *The Hidden Dimension*. Anchor Books, 1966.

[6] WebHamster. *A chart depicting Edward T. Hall's interpersonal distances of man, showing radius in feet and meters*. File: Personal Space.svg. 2009. URL: https://en.wikipedia.org/wiki/Proxemics#/media/File:Personal_Space.svg.

[7] Martin Remland, Tricia Jones, and Heidi Brinkman. «Proxemic and haptic behavior in three European countries». In: *Journal of Nonverbal Behavior* 15 (Dec. 1991), pp. 215–232. DOI: 10.1007/BF00986923.

[8] Mohammad Aliakbari, Elham Faraji, and Parnaz Pourshakibaee. «Investigation of the proxemic behavior of Iranian professors and university students: Effects of gender and status». In: *Journal of Pragmatics* 43.5 (2011). Multilingual structures and agencies, pp. 1392–1402. ISSN:

0378-2166. DOI: https://doi.org/10.1016/j.pragma.2010.10.021.

[9] Vineeth Rajamohan et al. «Factors Influencing The Human Preferred Interaction Distance». In: Oct. 2019, pp. 1–7. DOI: 10.1109/RO-MAN46459.2019.8956404.

[10] Adam Kendon. *Conducting Interaction: Patterns of Behavior in Focused Encounters*. Cambridge University Press, 1990.

[11] Adam Kendon. «Spacing and Orientation in Co-present Interaction». In: Jan. 2009, pp. 1–15. DOI: 10.1007/978-3-642-12397-9_1.

[12] Francesco Setti et al. «F-Formation Detection: Individuating Free-Standing Conversational Groups in Images». In: *PLOS ONE* 10 (Sept. 2014). DOI: 10.1371/journal.pone.0123783.

[13] Mehdi Moussaïd et al. «The Walking Behaviour of Pedestrian Social Groups and Its Impact on Crowd Dynamics». In: *PloS one* 5 (Apr. 2010), e10047. DOI: 10.1371/journal.pone.0010047.

[14] Mizar Federici et al. «Data Collection for Modeling and Simulation: Case Study at the University of Milan-Bicocca». In: Jan. 2012, pp. 699–708. DOI: 10.13140/2.1.2424.4164.

[15] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, 2006. DOI: 10.1017/CBO9780511546877.

[16] E. W. Dijkstra. «A Note on Two Problems in Connexion with Graphs». In: *Numer. Math.* 1.1 (Dec. 1959), pp. 269–271. ISSN: 0029-599X. DOI: 10.1007/BF01386390.

[17] P. E. Hart, N. J. Nilsson, and B. Raphael. «A Formal Basis for the Heuristic Determination of Minimum Cost Paths». In: *IEEE Transactions on Systems Science and Cybernetics* 4.2 (1968), pp. 100–107.

[18] Kenny Daniel et al. «Theta*: Any-Angle Path Planning on Grids». In: *J. Artif. Intell. Res. (JAIR)* 39 (Jan. 2014). DOI: 10.1613/jair.2994.

[19] Anthony Stentz. «Optimal and Efficient Path Planning for Partially-Known Environments». In: *1994 International Conference on Robotics and Automation* 4 (Feb. 2000). DOI: 10.1007/978-1-4615-6325-9_11.

[20] Sven Koenig and Maxim Likhachev. «Fast replanning for navigation in unknown terrain». In: *Robotics, IEEE Transactions on* 21 (July 2005), pp. 354–363. DOI: 10.1109/TRO.2004.838026.

[21] Mohammadreza Radmanesh et al. «Overview of Path Planning and Obstacle Avoidance Algorithms for UAVs: A Comparative Study». In: *Unmanned Systems* 6 (Apr. 2018), pp. 1–24. DOI: 10.1142/S2301385018400022.

[22] Lydia Kavraki et al. «Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces». In: *Robotics and Automation, IEEE Transactions on* 12 (Sept. 1996), pp. 566–580. DOI: 10.1109/70.508439.

[23] Nancy M. Amato et al. *OBPRM: An Obstacle-Based PRM for 3D Workspaces*. 1998.

[24] Steven LaValle and James Kuffner. «Randomized Kinodynamic Planning». In: *International Journal of Robotic Research - IJRR* 20 (Jan. 1999). DOI: 10.1177/02783640122067453.

[25] Sertac Karaman and Emilio Frazzoli. «Incremental Sampling-based Algorithms for Optimal Motion Planning». In: (May 2010).

[26] Michael Otte and Emilio Frazzoli. «RRTX: Asymptotically optimal single-query sampling-based motion planning with quick replanning». In: *The International Journal of Robotics Research* 35 (Sept. 2015). DOI: 10.1177/0278364915594679.

[27] Mohamed Elbanhawi and Milan Simic. «Sampling-Based Robot Motion Planning: A Review». In: *IEEE Access* 2 (Feb. 2014), pp. 56–77. DOI: 10.1109/ACCESS.2014.2302442.

[28] Sabudin elia nadira, Rosli Omar, and Che Ku Nor Hailma. «Potential field methods and their inherent approaches for path planning». In: 11 (Jan. 2016), pp. 10801–10805.

[29] Consuelo Granata and Philippe Bidaud. «A framework for the design of person following behaviors for social mobile robots». In: Oct. 2012, pp. 4652–4659. ISBN: 978-1-4673-1737-5. DOI: 10.1109/IROS.2012.6385976.

[30] Andrey Rudenko et al. *Human Motion Trajectory Prediction: A Survey*. May 2019.

[31] Henny Admoni and Brian Scassellati. «Social Eye Gaze in Human-Robot Interaction: A Review». In: *J. Hum.-Robot Interact.* 6.1 (May 2017), pp. 25–63. DOI: 10.5898/JHRI.6.1.Admoni.

[32] Gonzalo Ferrer et al. «Robot social-aware navigation framework to accompany people walking side-by-side». In: *Autonomous Robots* 41 (July 2016). DOI: 10.1007/s10514-016-9584-y.

[33] Jorge Rios-Martinez, Anne Spalanzani, and Christian Laugier. «Probabilistic Autonomous Navigation Using Rist-RRT approach and Models of Human Interaction». In: Jan. 2011.

[34] Jorge Rios-Martinez, Anne Spalanzani, and Christian Laugier. «Understanding human interaction for probabilistic autonomous navigation using Risk-RRT approach». In: Sept. 2011, pp. 2014–2019. DOI: 10.1109/IROS.2011.6094496.

[35] F. Yang and C. Peters. «Social-aware navigation in crowds with static and dynamic groups». In: *2019 11th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*. 2019, pp. 1–4.

[36] A. K. Pandey and R. Alami. «A framework towards a socially aware Mobile Robot motion in Human-Centered dynamic environment». In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2010, pp. 5855–5860.

[37] Elena Pacchierotti, Henrik Christensen, and P. Jensfelt. «Human-robot embodied interaction in hallway settings: A pilot user study». In: vol. 2005. Sept. 2005, pp. 164–171. ISBN: 0-7803-9274-4. DOI: 10.1109/ROMAN.2005.1513774.

[38] Elena Pacchierotti, Henrik Christensen, and Patric Jensfelt. «Evaluation of Passing Distance for Social Robots». In: Oct. 2006, pp. 315–320. DOI: 10.1109/ROMAN.2006.314436.

[39] Emrah Sisbot et al. «A Human Aware Mobile Robot Motion Planner». In: *Robotics, IEEE Transactions on* 23 (Nov. 2007), pp. 874–883. DOI: 10.1109/TRO.2007.904911.

[40] André Mateus et al. «Human-Aware Navigation using External Omnidirectional Cameras». In: Nov. 2015.

[41] Stuart Eiffert et al. *Path Planning in Dynamic Environments using Generative RNNs and Monte Carlo Tree Search*. Jan. 2020.

[42] Changan Chen et al. *Crowd-Robot Interaction: Crowd-aware Robot Navigation with Attention-based Deep Reinforcement Learning*. Sept. 2018.

[43] Rachel Kirby, Reid Simmons, and Jodi Forlizzi. «COMPANION: A constraint-optimizing method for person-acceptable navigation». In: Nov. 2009, pp. 607–612. DOI: 10.1109/ROMAN.2009.5326271.

[44] Marina Kollmitz et al. «Time dependent planning on a layered social cost map for human-aware robot navigation». In: Sept. 2015, pp. 1–6. DOI: 10.1109/ECMR.2015.7324184.

[45] A. Bellarbi et al. «A social planning and navigation for tour-guide robot in human environment». In: *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*. 2016, pp. 622–627.

[46] Truong Tung and Trung Dung Ngo. «Socially Aware Robot Navigation Using Deep Reinforcement Learning». In: May 2018, pp. 1–5. DOI: 10.1109/CCECE.2018.8447854.

[47] Shu Chung and h.c Huang. «A mobile robot that understands pedestrian spatial behaviors». In: Oct. 2010, pp. 5861–5866. DOI: 10.1109/IROS.2010.5649718.

[48] Shu-Yun Chung and h.c Huang. «Incremental learning of human social behaviors with feature-based spatial effects». In: Oct. 2012, pp. 2417–2422. ISBN: 978-1-4673-1737-5. DOI: 10.1109/IROS.2012.6385852.

[49] Dizan Vasquez et al. «Human Aware Navigation for Assistive Robotics». In: vol. 88. June 2012. DOI: 10.1007/978-3-319-00065-7_31.

[50] Xuan-Tung Truong and Trung Dung Ngo. «Dynamic Social Zone based Mobile Robot Navigation for Human Comfortable Safety in Social Environments». In: *International Journal of Social Robotics* 8 (2016), pp. 663–684.

[51] Xuan-Tung Truong and Trung Dung Ngo. «"To Approach Humans?": A Unified Framework for Approaching Pose Prediction and Socially Aware Robot Navigation». In: *IEEE Transactions on Cognitive and Developmental Systems* PP (Sept. 2017), pp. 1–1. DOI: 10.1109/TCDS.2017.2751963.

[52] Javier Gómez, Nikolaos Mavridis, and Santiago Garrido. «Fast Marching Solution for the Social Path Planning Problem». In: June 2014. DOI: 10.1109/ICRA.2014.6907169.

[53] Hamid Laga and Toshitaka Amaoka. «Modeling the spatial behavior of virtual agents in groups for non-verbal communication in virtual worlds». In: Jan. 2009, pp. 154–159. DOI: 10.1145/1667780.1667811.

[54] Tung-Wu Lu and Chu-fen Chang. «Biomechanics of human movement and its clinical applications.» In: *The Kaohsiung journal of medical sciences* 28 2 Suppl (2012), S13–25.

[55] *Gazebo*. Accessed: 2020-05-25. URL: http://gazebosim.org/.

[56] *Ros*. Accessed: 2020-05-25. URL: https://www.ros.org/.

[57] Hooman Hedayati, James Kennedy, and Daniel Szafir. «Comparing F-Formations Between Humans and On-Screen Agents». In: Apr. 2020. DOI: 10.1145/3334480.3383015.

Appendix A

The A* planning algorithm

The pseudo-code of the deterministic planning algorithm **A*** [17] is presented below. The algorithm is able to determine the minimum cost path between the *start* and *goal* states. It belongs to the class of informed search algorithms thanks to the use of heuristics, which indicate the most promising node to expand during the search. The **trajectory_rollout** function rebuilds the complete path starting from the goal configuration once the optimal sequence of states is found.

Algorithm 2 A* Planning Algorithm (Part-1)

```
function A*(start, goal)
    closedset ← ∅                          ▷ Set of nodes already evaluated
    openset ← start                        ▷ Set of tentative nodes to be evaluated
    g[start] := 0                          ▷ Distance from start along optimal path
    h[start] := HEURISTIC_ESTIMATE(start, goal)
    f[start] := h[start]                   ▷ Estimated total distance from start to goal
    while openset is not empty do
        x := node in openset with the lowest f value
        if x == goal then
            return TRAJECTORY_ROLLOUT(came_from, goal)
        end if
        delete x from openset
        add x to closedset
        for each y in NEIGHBOURS(x) do
            if y in closedset then
                continue
            end if
```

Algorithm 3 A* Planning Algorithm (Part-2)

```
            tentative_g := g[x] + COST_BETWEEN(x, y)
            if y not in openset then
                add y to openset
                tentative_is_better := true
            else if tentative_g < g[y] then
                tentative_is_better := true
            else
                tentative_is_better := false
            end if
            if tentative_is_better == true then
                came_from[y] := x
                g[y] := tentative_g
                h[y] := HEURISTIC_ESTIMATE(y, goal)
                f[y] := g[y] + h[y]
            end if
        end for
    end while
    return failure
end function

function TRAJECTORY_ROLLOUT(came_from, current)
    if came_from[current] is set then
        n = TRAJECTORY_ROLLOUT(came_from,came_from[current])
        return n + current
    else
        return empty path
    end if
end function
```

Appendix B

People clustering algorithm

We provide below the algorithm used to enforce the *Distance Constraint* between members of the formations (both in static and dynamic cases). The inputs provided to the algorithm are a set of people and the distance threshold to impose. As a result, it determines the partitioning of people into their respective groups, returning the set of identified groups as output. In each of them, for each member, there is at least one other member such that their relative distance is less than the specified threshold.

Algorithm 4 People Clustering Algorithm (Part-1)

```
function GROUPING(people, threshold)
    groupset ← ∅                                    ▷ Set of detected groups
    for each h in people do
        if h already belongs to a group then
            continue
        end if
        add h to a new group G_i
        for each p in G_i do
            DISCOVER_NEIGHBOURS(p, G_i, people, threshold)
        end for
        add G_i to groupset
    end for
    return groupset
end function
```

Algorithm 5 People Clustering Algorithm (Part-2)

```
function DISCOVER_NEIGHBOURS(person, group, people, threshold)
    for each h in people do
        if h already belongs to a group then
            continue
        end if
        if DISTANCE(person, h) < threshold then
            add h to group
        end if
    end for
end function
```