An experiment on squad navigation of human and robots

L. Nomdedeu[1], J. Sales[1], E. Cervera[1], J. Alemany[1],
R. Sebastia[2], J. Penders[3], V. Gazi[4]

Abstract

Mobile robots are nowadays capable of navigating in a semi-structured environment, either alone or in formations. Dealing with mobile obstacles, e.g. persons, in everyday environments poses additional challenges which are being addressed by the research community. This paper presents an experiment with a set of robots -acting as a swarm- and one person. The aim of such robot-human squad is to explore an unmapped area indoors. The swarm is programmed such that obstacle avoidance, robot avoidance and human avoidance is combined with staying in the vicinity of the human and gradually proceeding towards the end point. The robots are provided with enough sensors to perform avoidance, and determine/estimate the distance to the human. Communication is one-way only, from robots to human. The aim of the experiment is to provide insights about the real problems of the multirobot system when dealing with a non-expert person in a cooperative exploration task.

1 Introduction

Mobile robots are nowadays capable of navigating in a semi-structured environment, either alone or in formations. Dealing with mobile obstacles, e.g. persons, in everyday environments poses additional challenges which are being addressed by the research community . If the robots need to cooperate with humans in the navigation squad, the difficulty is increased, since aspects of human-robot interaction arise. In such cases, simulation becomes useless, since physical interaction between humans and robots cannot be considered in the simulator. Thus, real experiments are needed, to test and validate the control approaches of the multirobot system, and the interaction with the cooperating persons. This paper presents an experiment with a set of robots -acting as a swarm- and one person. The aim of such robot-human squad is to explore an indoors area (see figure 1). The swarm is programmed such that obstacle avoidance, robot avoidance and human avoidance is combined with staying in the vicinity of the human and gradually proceeding towards the end point. The robots are provided with enough sensors to perform avoidance, and determine/estimate

[1]Robotic Intelligence Lab, Jaume-I University, Castelló (SPAIN)
[2]Robotnik Automation, SL, Valencia (SPAIN)
[3]Sheffield Hallam University (UK)
[4]TOBB University of Economics and Technology, Ankara, (TURKEY)

Figure 1: Experiment layout scheme. 3D recreation.

the distance to the human. Communication is one-way only, from robots to human. The human being is provided with a first experimental (one way) Robot to Human communication device. The aim of the experiment is to provide insights about the real problems of the multirobot system when dealing with a non-expert person in a cooperative exploration task.

1.1 Motivation

While in a fire, fire-fighters have virtually no sensing capabilities. They have poor visibility, very rude tactile perception, no way of hearing clearly due to the usual noise, etc. In addition they have no map of the building, and so they have to manage to get in, secure the place, take of any victim, and get out of the building. Under this conditions one of the most important problems they face is how to get out. It's quite easy to lose orientation and to get trapped inside the building. Many fire-fighters have died trapped inside buildings while doing his job.

1.2 Statement of the problem

Our system must be able to integrate exploration and obstacle avoidance, while following the human and localizing itself in order to be able to find the way out. With human and robot mobile agents operating in the same workspace we need to know location and orientation (aka pose) in order to guarantee safety in terms of a robot hurting a human. [1]. We address this pose problem tracking the human with the escorts lasers and providing him with an accelerometer-gyroscope-magnetometer (IMU) that lets us know a bit more about the real pose of the fire-fighter.

This demo will guide an artificially blinded human being through a building with only the information shown in the screen of his laptop.

1.3 Related Work

Using laser range finders to find and track humans has been recently studied, but the special conditions of our environment don't let us use any available approach. Detecting multiple moving objects, multiple humans, by means of the laser range finder has been addressed [2] [3]. A number of proprietary devices (light-emitting [4] [5], radio transmitter-receiver [6], ...) has been developed. By using video data the task could be more easily addressed, but one of the biggest constraints fire-fighters have is the lack of visibility, and so, the robots will see nothing with this kind of sensors.

In order to escort the fire-fighter, a mobile robot needs to know the position of the person and must be able to determine its own path in order to accompany his target [7]. Usually finding the human relies in the shape typical legs have [8] [9], but once more, we could not assume a perfect convex pair of legs due to the special trousers fire-fighter wear and the special way they move, almost without any separation between his legs.

When talking about localization, obstacle avoidance, or path-planing, we are talking about very well known problems in mobile robotics.

The Adaptive Monte Carlo Localization Plus algorithm [10] provides us a solution for self localization. It uses the environment map and the readings from a range sensor to maintain a particle-based probabilistic density representation of possible places to filter. Its good enough for our purpose and is quite easy to find an implementation.

In the field of obstacle avoidance there are tons of algorithms. The more relevant two algorithms may be VFH (Vector Field Histogram [11]) and ND (Nearness Diagram [12] [13]). Each one uses its own technique to avoid colliding with the surrounding objects, by processing range data from sensors. Both of them are able to perform a local navigation taking the robot starting point as the origin and moving towards the goal using odometry (if there is no better information) to know when to stop.

For the purpose of or demonstration the classical wavefront-propagation path-planning algorithm [14] is quite a good choice. It provides us a good enough set of waypoints at almost no cost, in terms of effort and computation, as there are a lot of optimized implementations ready to use.

1.4 Overview

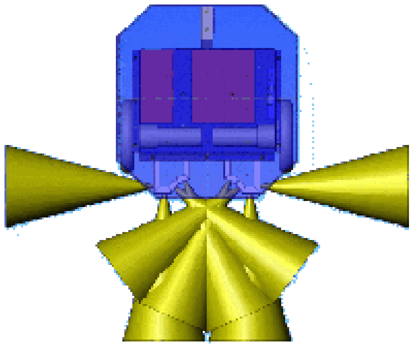
The paper is organized as follows: In Section II we explain the system configuration, robots, sensors, and capabilities. Section III deals with the software architecture we have developed while Section IV presents the experiments and results. Finally we draw some conclusions and future work in Section V.

2 System configuration

2.1 Hardware setup

For this demo we have used the mobile commercial platform Erratic from Videredesign (Fig. 2). Each Erratic mobile platform is equipped with the sensors shown in figure 2.

(a) Mobile platform Erratic: H8S controller, Servo and analog/digital I/O, 2x 75W motors, 500 CPR encoders, IR floor sensors, 3x 7A 12V batteries. Onboard PC: 1.4 GHz Celeron, 802.11b/g, USB 2.0, IEEE 1394b, 40 GB, 512 MB



(b) Sonar ring: 8 MaxSonar EZ1 sonars, Maximum range 6m, Typical usable range 2m - 3m, Total firing rate 20 Hz

(c) XSens MTi IMU device: accurate full 360 degrees 3D orientation output (Attitude and Heading), highly dynamic response combined with long-term stability (no drift), 3D acceleration, 3D rate of turn and 3D earth-magnetic field data, high update rate, temperature, 3D misalignment and sensor cross-sensitivity compensated

(d) Hokuyo URG Laser Rangefinder: Range 4m, Accuracy +- 10mm, Field of view 240 degrees, Angular resolution 0.36 degrees, Max scan rate 10 Hz, Power 2.5W, Weight 0.16 Kg

Figure 2: Mobile platform and sensors used

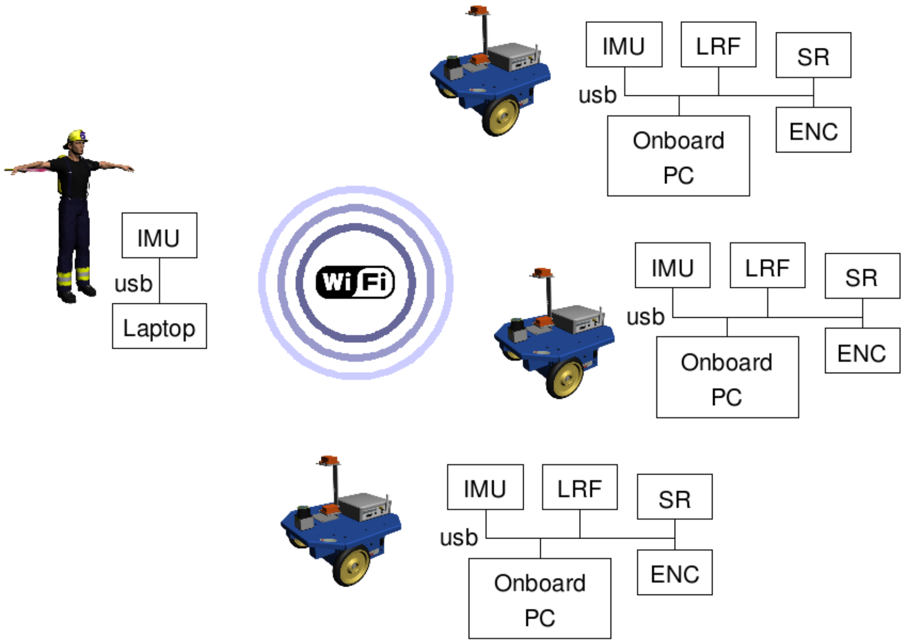


Figure 3: Experiment participants scheme[1].

This platform provides us lasers readings, encoder and sonar data, as well as all the imu data. All this data is locally processed for localizing and obstacle avoiding proposes, as well as for human detection.

The fire-fighter will carry a laptop along with a imu sensor (Fig. 3).. The readings from this imu will be used for posterior filtering and accurate detection of the fire-fighter pose itself.

The communication between different robots and the fire-fighter laptop will be provided by a wireless infrastructure network, with several access points distributed all around the building. The roaming between this APs will be also taken into account.

2.2 Software setup

Once the mobile platforms (specially the escorts) have done the self-localization and the human tracking, they will communicate via the wireless network with other mobile platforms if needed and with the fire-fighter laptop. Using all escorts data, and its own imu data the laptop will be able to provide the fire-fighter with an estimation of his own pose, and using the leader data, will be able to provide the fire-fighter a new orientation and velocity to take.

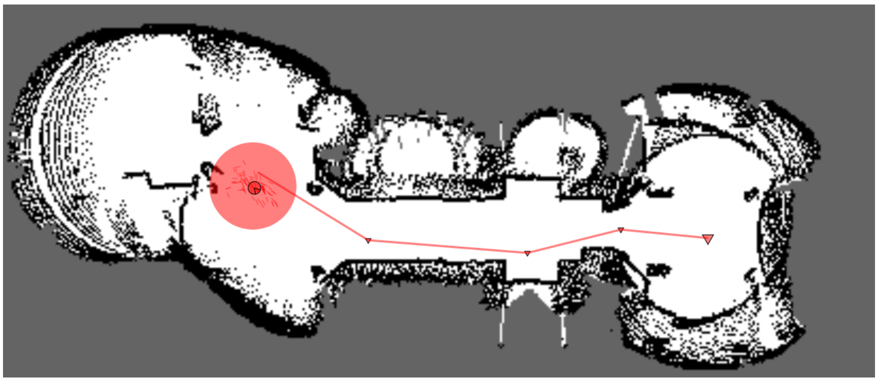
We obviously assume that the fire-fighter is cooperative and does not deliberately run, hide, jump, or takes any other action that will lead to the lose of

[1]
IMU: Accelerometer sensor.
LRF: Laser rangefinder sensor.
SR: Sonar ring.
ENC: Encoders and motors.

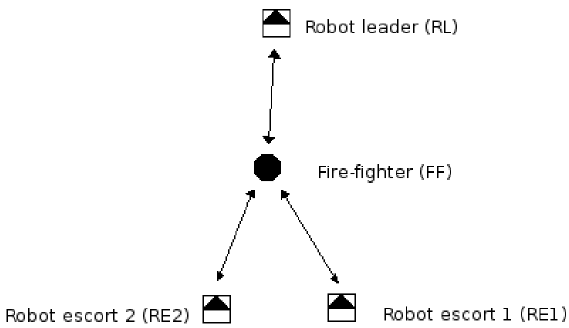
tracking.

Under this circumstances the robots are able to predict the next position and speed of the human, based on the history of his position and the robots own position and velocity, and so, they are able to effectively follow the fire-fighter in addition to localize him.

As we could see in diagram 4(b) there are three roles in our human-robot system. First we have the leader robot, extending the fire-fighter capabilities and providing him with the path (and probably in the near future the whole map), the position, and, among other possible data, any guidance information available. The second role is the one played by the fire-fighter. He will follow the instructions in his user interface, and act, when needed, as a fire-fighter will. And finally, the escort robots, moving along with the fire-fighter, localizing him, detecting any possible dangers, and showing the way out if needed.



(a) Playernav tool (from Player project) screenshot showing trajectory and localization



(b) Formation and communication diagram

Figure 4: Demo diagrams

In figure 5 we could see the logical scheme of the system. The modules are interconnected, receiving data from more low level modules, processing this data, and providing some more high level data to other modules that wish to

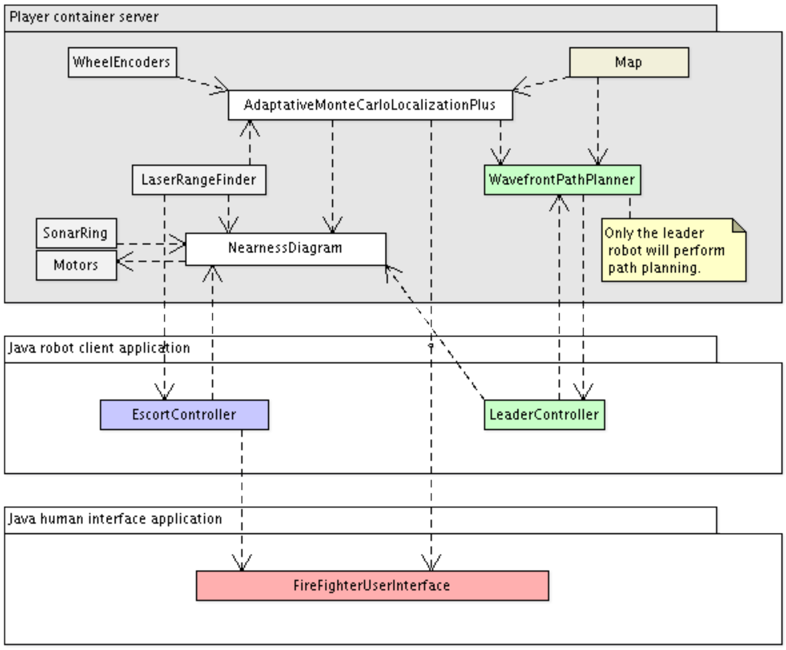


Figure 5: Logical system scheme

obtain this data. In figure 4(a) we could see an screenshot of the playernav tool, showing the path the robot leader is going to follow, and the uncertainty in its self localization (the red big semi-transparent circle around the robot circle).

The robot escort application layer reads data from several modules, computes the human tracking algorithm, and sends commands to the appropriate local navigation module. The relative human estimated localization calculated will be provided to any other layer interested.

The robot leader application layer reads the waypoint list and manages it by commanding the local navigation module to the next waypoint and discarding waypoints when certain conditions (mainly proximity) are reached. Although we could make the wavefront module send the waypoints directly to the ND local navigation module, we prefer to control the way waypoints are sent and when they are discarded. This lets us perform future waypoint modifications if we detect any problem in the waypoint list provided.

Firefighter application layer receives the global estimated localization of each robot, plus his own relative estimated localization from each escort robot, and computes his own global estimated localization. With his own global estimated localization and the robot leader ones the application provides a primitive user interface indicating in a graphical fashion the angular displacement of the robot leader and the linear gap between the robot leader and the fire-fighter. With this information the fire-fighter will be able to follow the robot leader without environment vision, only seen his laptop's screen.

3 Implementation

3.1 Software architecture

In the context of the GUARDIANS project we will be using several different and heterogeneous types of mobile platforms. This fact reinforces the current tendency of layered software development, in special regarding the Hardware abstraction layer (HAL). Based on this and other several arguments the Player / Stage [2] platform [15] has been chosen for simulation and HAL for robot development in Guardians. Player is a control layer, a middleware, that provides a common interface to heterogeneous robot platforms and sensors. It is an open source project supported by many community developers interested in robotics.

With the correct configuration and the correct set of device proxies, we are able to access any device, and any important data we could need from the mobile platform and its sensors. The actuators (the motors) could be easily accessed too.

In addition we could use several state of the art algorithms for solving several specific problems. They, from the Player / Stage point of view, are called abstract drivers. This is the case of, for example, the Adaptive Monte Carlo Localization Plus algorithm [10], encapsulated in the AMCL driver, or the local navigation and obstacle avoidance algorithms VFH (Vector Field Histogram [11]) and ND (Nearness Diagram [12] [13]), as well as the path planning algorithm Wavefront-propagation path-planner.

To be able to access all the mobile platform devices, Player provides a socket based communication system. Data through this sockets is encoded using the platform independent data representation standard called XDR (eXternal Data Representation). This allows any program, written in any programing language, to access all the Player provided features, with the only requirement of having a socket based communications stack and an XDR standard encoding-decoding library. The Java Player Client [3] project provides such a library, and more, in fact the it provides a complete set of device access proxies on the top of this communication layer. Our specific tracking algorithm execution will then be carried by the Application layer (Fig. 6).

3.2 Tracking

With all this setup the main task remains, the human being localization through only the laser rangefinder data. We assume that only one fire-fighter will be in the field of view of the escort robots (Fig. 7). Using this precondition we search for an object of a certain preset width and in an specific range of view and distance from the robot. This object will be the fire-fighter as no other object will be in this initial ranges.

To know the object's width we perform a linear search at both sides of the object centroid (Fig. 8). We assume that a point belongs to the object if it is inside certain threshold of distance (δ) of the object's centroid from the robot (the laser reading value) and not far from this centroid in terms of the angle of the reading (ϵ). Those two parameters provides the dimensions of the blue

[2] `http://playerstage.sourceforge.net`
[3] `http://java-player.sourceforge.net`

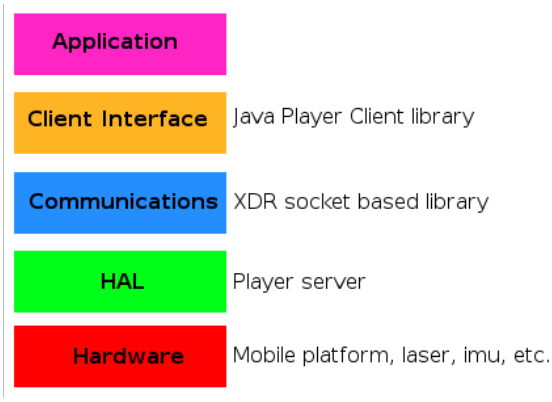
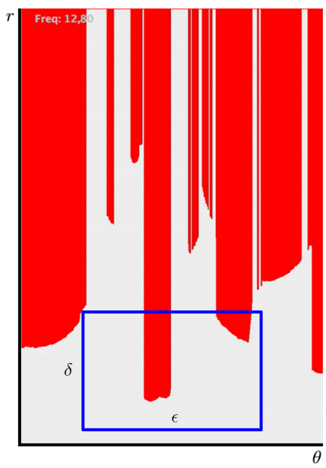
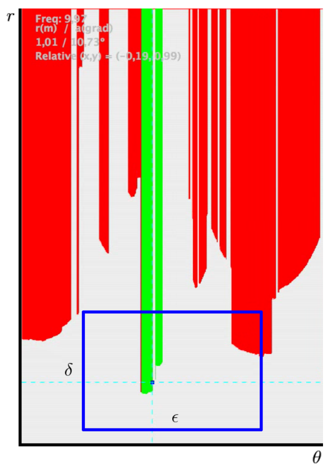


Figure 6: Software architecture



(a) Starting laser reads. We have to search in the sensible area.



(b) Fire-fighter has been located inside the sensible starting region.

Figure 7: First step: finding the fire-fighter.

rectangle centered in the centroid that we could see in each screenshot of the figure 8.

The first centroid is selected arbitrary from within the legs readings, and will be centered in the next loop of the control loop by growing the object at both sides of the centroid and recalculating it. This time the sensible area is bigger than while tracking. We could appreciate the bigger size of the blue rectangle in figure 7.

At the beginning steps we could appreciate how the distance between fire-fighter and the robot increases, the robot is moving deliberately slow to reach the desired formation. Once the fire-fighter is in the desired relative position, the robot starts moving faster to keep this formation the more stable it can.

Due to the small velocity and the high scan rate we assume that the fire-fighter will be in the near neighborhood of the previous instant detected centroid. So we use the last instant centroid to calculate the object width and the new centroid in the actual instant, used for the next calculations.

The tracking is then done. A control loop is employed (see fig. 9), which calculates the distance from the person and sets the maximum velocity and turn rate based on the calculated distance. Then the destination pose of the robot is set according to the fire-fighter estimated relative position and the formation parameters (swarm behavior). An obstacle avoidance routine that uses sonar and laser information is allways used to ensure the safety of the fire-fighter and of the robot itself.

Let (θ, r) be the relative polar coordinates of the fire-fighter centroid with respect to the laser range finder at instant t. In the next instant we will search, beginning at angle θ each reading z_i within the radius range $[r-\delta, r+\delta]$ being i in $[\theta-\epsilon, \theta+\epsilon]$[4]. If we find one reading out of the radius range, we skip this reading and continue, but if we find more than λ successive readings out of the radius range we stop searching at this side. At the end we obtain a set of readings z_i of the form (θ_i, r_i). We then calculate the current instant centroid of the followed fire-fighter needed at next step as an arithmetic mean of the selected j reads, as seen in equation (1). In general, δ and ϵ are parameters that will depend on the shape and size of the object we want to track.

$$x_c = \frac{\sum_{j=0}^{n-1} x_j}{n} \quad y_c = \frac{\sum_{j=0}^{n-1} y_j}{n} \tag{1}$$

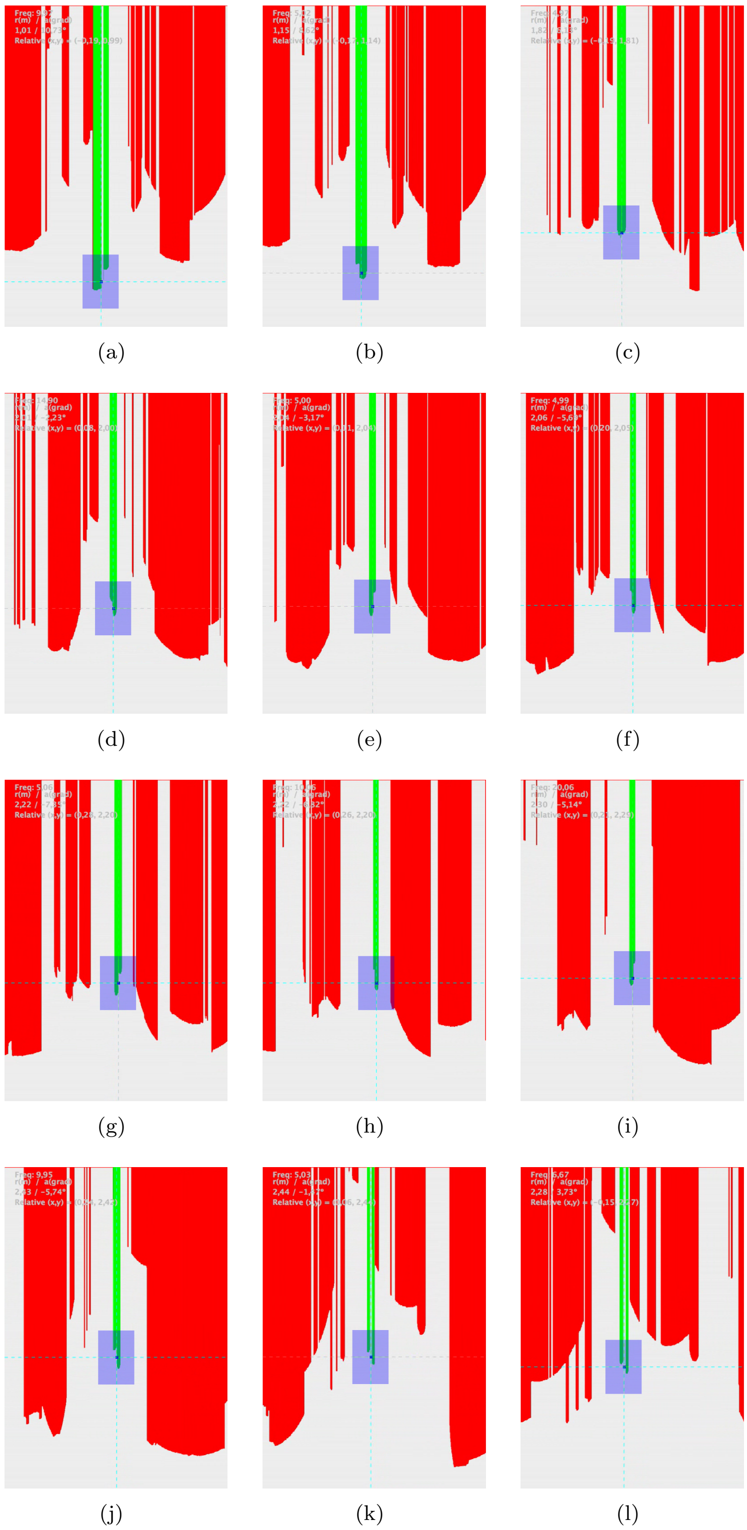
For navigation purposes we then transform polar in Cartesian coordinates by the well known system equation (2).

$$\begin{aligned} x &= r * sin(\theta) \\ y &= r * cos(\theta) \end{aligned} \tag{2}$$

The relative velocities in each Cartesian coordinate are then approximated as $(\Delta x/t, \Delta y/t)$ using the relative positions history and the timestamp readings.

Once the relative fire-fighter position and velocity are obtained, the robot calculates the estimated global position and velocity of the fire-fighter using its

[4] δ is half the height of the blue box in the images from the sequence of figure 8, while ϵ is half the width.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Figure 8: Second step: tracking the fire-fighter.

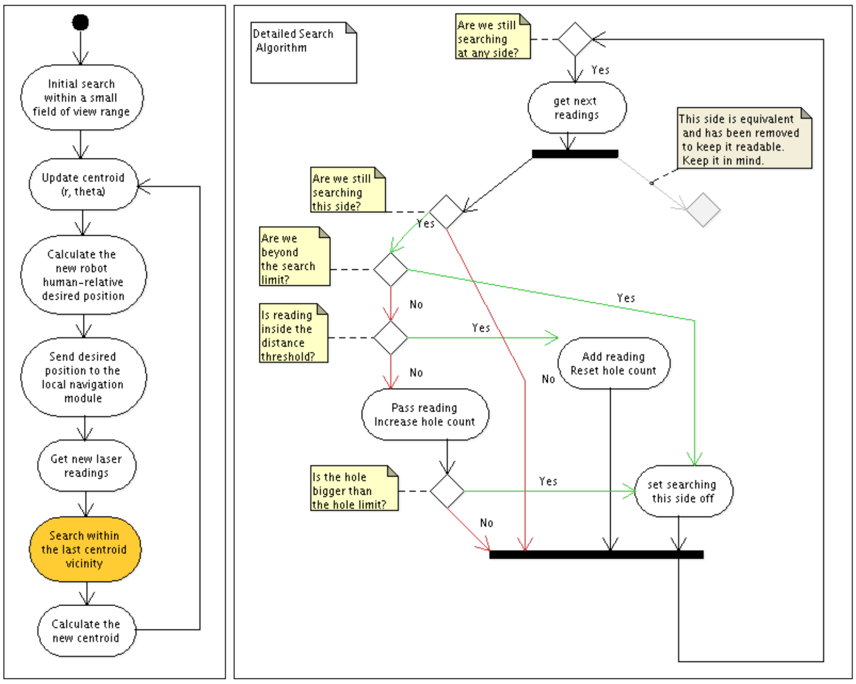


Figure 9: Tracking algorithm

own estimated position and velocity. Then the desired relative position of the robot with respect to the fire-fighter is calculated according to the formation rules. For the demo the formation will be a simple isosceles triangle of approximately 1.5 meters high and 40 centimeters wide as seen in figure 10. This position is then transformed to a global position and passed to the local navigation module (ND) which will move the robot to this temporary goal while doing obstacle avoidance with laser and sonar data. The fact that the fire-fighter is in front of the robot does not conform a problem for the obstacle avoidance routine as the firefighter is far from the obstacle avoidance distance threshold.

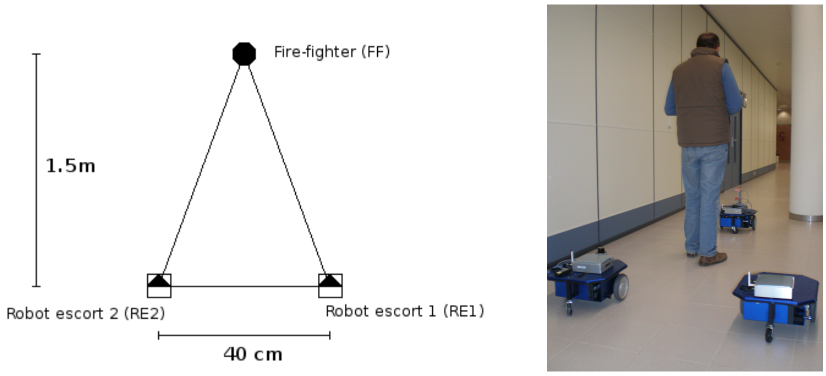
4 The experiment

First, the experiment presented has been partially tested in simulation. The figure 11 shows the scheme of what we have perform.

After the simulation in the real environment the robots has guided the human through our department, from a lab to the lift and stairs. The human has no other environment information that the one shown in his laptop. With only the laptop information we achieve a collision free path for the human, reaching his destination without major problems.

The tracking algorithm parameters used were:

- $\delta = 0.3$ meters
- $\epsilon = 50$ laser readings ($\approx 18.3°$)



(a) Formation scheme

(b) Demo picture

Figure 10: Robot relative formation

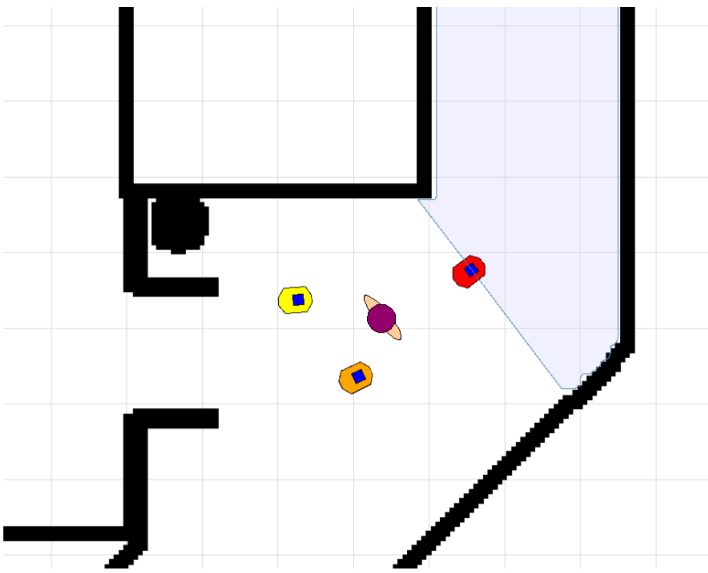


Figure 11: Experiment layout scheme.

- $\lambda = 15$ laser readings ($\approx 5.5°$)

The escort team has not loosed his target along the way. This is remarkable since there have been some occlusions along the path, but the robots were able to regain his target by only following the last formation pose provided before the occlusion.

Self localization and obstacle avoidance worked well. The waypoint provided by the Wavefront module where not the best ones from a human point of view but were good enough for the task.

To conclude, the whole experiment seemed to validate our approach, and spear a ray of light to the complex problem of aiding very well prepared fire-fighter in hazardous and risky search and rescue task.

5 Conclusion and future work

In this article we have presented an experiment for validating the use of mobile platforms in hazardous environments as a support for human beings. Although the system has proven to work, we have in mind some changes for making it more robust, as it will be used in very unstructured environments and human lives will depend somehow in the system reliability.

Implementation of Kalman filtering to improve pose fusion and prediction is planned, along with an online SLAM approach carried out by the mobile platforms, mainly the leader robot. We hope that Kalman filter to address occlusions and sensor noise problems.

We think that this demonstrations is a good step, but there is a long way before reaching a state of the art when robots could effectively be working side-by-side with human beings in search and rescue tasks.

Simultaneous mapping capabilities (SLAM) have been investigated in the last decade, and an armamentarium of algorithms and implementations are at disposal of researchers worldwide for experimentation and testing. to explore an unmapped area indoors

What we aim to achieve is a full Human Augmented Mapping (HAM) system, capable of guiding fire-fighter in, and more important, out of a building being devastated by flames.

References

[1] J. Graham and K. Shillcutt, "Robot tracking of human subjects in field environments," in *iSAIRAS 2003, May 2003.*, 2005.

[2] A. Almeida, J. Almeida, and R. Araujo, "Real-time tracking of moving objects using particle filters," in *Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on*, vol. 4, June 2005, pp. 1327 – 1332.

[3] A. Fod, A. Howard, and M. Mataric, "Laser-based people tracking," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2002.

[4] Y. Nagumo and A. Ohya, "Human following behavior of an autonomous mobile robot usinglight-emitting device," in *Robot and Human Interactive Communication, 2001. Proceedings. 10th IEEE International Workshop on*, 2001, pp. 225 – 230.

[5] O. Gigliotta, M. Caretti, S. Shokur, and S. Nolfi, "Toward a person-follower robot," in *Proceedings of the Second Robocare Workshop*, 2003.

[6] R. Bianco, M. Caretti, and S. Nolfi, "Developing a robot able to follow a human target in a domestic environment," in *Proceedings of the First Robocare Workshop*, 2002.

[7] A. Ohya and T. Munekata, "Intelligent escort robot moving together with human -interaction in accompanying behavior-," in *2002 FIRA Robot Congress*, 2002.

[8] A. Milella, C. Dimiccoli, G. Cicirelli, and A. Distante, "Laser-based people-following for human-augmented mapping of indoor environments," in *AIAP'07: Proceedings of the 25th conference on Proceedings of the 25th IASTED International Multi-Conference.* Anaheim, CA, USA: ACTA Press, 2007, pp. 151–155.

[9] E. A. Topp and H. I. Christensen, "Tracking for following and passing persons," in *Intelligent Robots and Systems, 2005. (IROS 2005)*, 2005.

[10] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, 1999, pp. 1322–1328.

[11] I. Ulrich and J. Borenstein, "Vfh+: Reliable obstacle avoidance for fast mobile robots," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, 1998, pp. 1572–1577.

[12] J. Minguez, J. Osuna, and L. Montano, "Nearness diagram navigation (nd): Collision avoidance in troublesome scenarios." *IEEE Transactions on Robotics and Automation*, p. 154, 2004.

[13] ——, "A divide and conquer strategy based on situations to achieve reactive collision avoidance in troublesome scenarios," in *IEEE International Conference on Robotics and Automation*, 2004.

[14] J. Barraquand, B. Langlois, and J. Latombe, "Numerical potential field techniques for robot path planning," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 22, no. 2, pp. 224–241, 1992.

[15] R. T. Vaughan and B. P. Gerkey, "Really reusable robot code and the player / stage project," in *Software Engineering for Experimental Robotics*, ser. Springer Tracts on Advanced Robotics, D. Brugali, Ed. Springer, 2007.