

# A New Efficiency-Weighted Strategy for Continuous Human/Robot Cooperation in Navigation

Alberto Poncela, Cristina Urdiales, Eduardo J. Pérez, and Francisco Sandoval, *Member, IEEE*

**Abstract**—Autonomous robots are capable of navigating on their own. Shared control approaches, however, allow humans to make some navigation decisions. This is typically executed either by overriding the human or the robot control at some specific situations. In this paper, we propose a method to allow cooperation between humans and robots at each point of any given trajectory so that both have some weight in the emergent behavior of the mobile robot. This is achieved by evaluating their efficiencies at each time instant and combining their commands into a single order. In order to achieve a seamless combination, this procedure is integrated into a bottom-up architecture via a reactive layer. We have tested the proposed method using a real robot and several volunteers, and results have been satisfactory both from a quantitative and qualitative point of view.

**Index Terms**—Autonomous navigation, behaviors, robot, shared control.

## I. INTRODUCTION

**A**UTONOMY in an agent can be defined as the ability to make independent choices. A mobile robot is considered to be autonomous when it can perform a given task in an unstructured environment without continuous human guidance. Many devices are autonomous to some degree but require some human supervision to perform a task. People with disabilities may as well require some assistance, either from a machine or from other persons, to achieve some tasks. Situations where machines and persons cooperate to achieve a common goal fall within the field of collaborative control.

There are many studies on the level of autonomy a robot might have when interacting with a human and vice versa [1]–[4]. Depending on how much autonomy the machine has, all approaches between human control and machine autonomous operation can be categorized into the following: 1) teleoperation; 2) safeguarded operation; 3) shared control; and 4) autonomous control [1]. Among these approaches, safeguarded operation and shared control fall between full teleoperation and full robotic autonomy. In the first case, mobile robots can be totally controlled by humans; however, in some cases, the robot makes some decisions to avoid imminent danger when

communication interruptions and delays are frequent [5], [6] or when human control is not adequate [7]–[9]. The robot may be able to exert more control in mobility assistance for people with disabilities, as the user may choose to give control to the machine to overcome difficult situations such as door crossing (DC), U-turn, etc. [1], [3]. In the extreme case, the human operator might only point the target, and the machine would be in charge of motion planning and path tracking on its own.

Although there are many approaches to shared control, they usually rely on giving control either to the human user or to the machine according to, more or less, complex algorithms. In this paper, we propose a new approach consisting of combining both the machine and the human orders at every moment in order to achieve a better result in situations where one or the other performs better. Our approach relies on evaluating the performance of the human driver and the machine for each given situation. To achieve this in a simple intuitive way, we rely on a hierarchical architecture proposed by the authors in [10]. As all reactive algorithms, the reactive layer of this architecture provides a simple combination of both command sources and integrates them with more complex layers in a bottom-up way. This architecture is described briefly in Section II. Depending on the effectiveness of each of them, the response of the system is a weighted linear combination of both the user and the robot, as explained in Section III. The proposed system has been tested in different situations for different drivers, and performance has been evaluated from an analytical and subjective point of view. Experiments are described in Section IV. Finally, conclusions and future work are presented in Section V.

## II. AUTONOMOUS NAVIGATION CONTROL

In order to navigate in an autonomous way, a robot needs to be equipped with sensors to perceive its environment. In most cases, these sensors return range measurement, indicating how far the system is from nearby obstacles. Navigation basically consists of reaching an established goal in a safe way, and control architectures are used to turn robot available data (position with respect to goal and obstacles) into appropriate motor commands. The more the system knows about the environment, the more efficient it can be. However, it may also take more time to process that knowledge and take an action.

Early control architectures relied on the so-called sense-plan-act scheme (SPA) (e.g., [11], [12]), where a model of the environment is used to calculate a safe path to the goal. Such models can be provided beforehand or calculated online; nevertheless, SPA relies on dealing with fairly static and predictable environments. Even in these situations, the system

Manuscript received March 28, 2007; revised December 18, 2007. First published February 18, 2009; current version published April 17, 2009. This work was supported in part by the Spanish Ministerio de Educación y Ciencia (MEC) and in part by FEDER funds under Project TEC2006-11689-C01/TCM and Project STREP 045088. This paper was recommended by Associate Editor A. A. Maciejewski.

The authors are with the Departamento de Tecnología Electrónica, Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad de Málaga, 29071 Málaga, Spain.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMCA.2009.2013191

performance degrades with odometry errors that must be corrected by techniques such as Kalman filters [13] or Monte Carlo localization [14]. SPA control presents some basic drawbacks: 1) All modules had to be fully functional to perform a basic test; 2) a single failure provokes a collapse in global functioning; 3) it has poor flexibility; and 4) it cannot act rapidly because information is linearly processed by every module.

In order to solve the SPA drawbacks, the subsumption architecture [15] is based on a bottom-up behavioral approach: 1) Simple behaviors are the result of coupling sensor readings and actions, and 2) the combination of several basic behaviors running concurrently produces a complex one. Reactive behaviors are fast and quite robust against sensor errors and noise and can easily adapt to changes in hardware or tasks. Unfortunately, emergent behaviors are unpredictable, not necessarily efficient, and prone to fall into local traps. However, in this paper, it is even more interesting that emergent behaviors deal with several sensors and goals in a simple way; hence, they can also combine the human and robotic commands and goals. The human influence on a reactive layer may be of help to gain global efficiency and to avoid local traps.

Hybrid schemes solve the aforementioned problems by combining both reactive and deliberative paradigms to achieve the best possible performance [16]. The hybrid style facilitates the design of efficient low-level control connected to high-level reasoning. These schemes typically include reactive, control, and deliberative systems for high-level planning. There are many approaches to implement hybrid schemes such as *Laboratoire d'Analyse et d'Architecture des Systèmes* [17] and *Task Control Architecture* [18]. Most of them are based on the well-known *3T* scheme [19], and they usually offer a top-down or bottom-up hierarchical control strategy. In this paper, we rely on a new control architecture called distributed and layered architecture (DLA).<sup>1</sup> The main advantage of this architecture with respect to other ones is that it has been conceived keeping transparency in mind through simplicity and portability in its use. Hence, DLA may not be as efficient as other solutions for specific systems; however, it tends to be more general for application to different problems. DLA was presented by the authors in [10]. It implements a distributed shared memory scheme that allows a combination of cooperative algorithms through the asynchronous interaction of freely distributed processes. Some of its features are the following: 1) *simplicity*—the user only has to use the DLALibrary to exchange data among modules; 2) *extendibility*—adding new modules and/or hardware with new functions and behaviors to the system is easily performed without further changes in the rest of the modules; 3) *portability*—DLALibrary has been developed in *MATLAB*, *JAVA*, and *C* for *Linux* and *Windows*; hence, any module using those languages or platforms can be integrated with the other modules, and new robotic platforms can be used without modifications to existing modules; and 4) *efficiency*—DLA has additional debugging tools and several mechanisms to improve the system performance.

The basic DLA architecture scheme for hierarchical navigation is shown in Fig. 1. Its advantages and drawbacks are

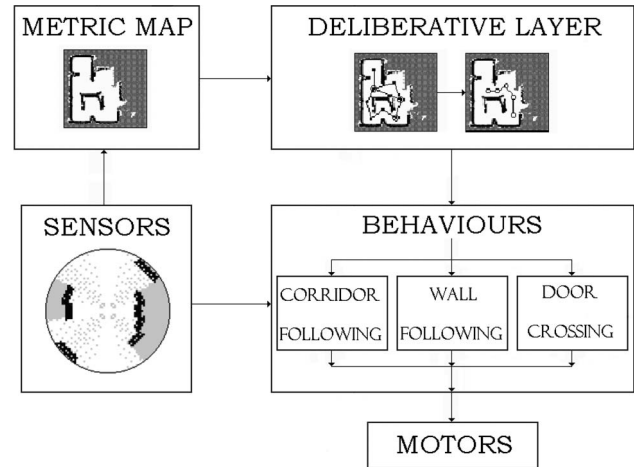


Fig. 1. DLA architecture scheme.

discussed in [20]. Odometrical data and sonar readings are captured at the robotic platform. Using these data, a geometrical model of the environment is built using occupancy grids. This geometrical map, however, is not suitable for fast planning because, in medium or large environments, it presents a too-high data volume. It has been reported that symbolic representations are better for high-level spatial reasoning [21], [22]. Hence, the geometric map is used to generate a grounded graph in a hierarchical way, as proposed by the authors in [20]. This process allows fast interaction among deliberative and reactive layers in the system. The resulting graph is the input instance for the planning stage, which returns a high-level plan for the robot to reach the desired goal. This proposed plan is translated into a set of easy-to-reach partial goals using the geometrical map for low-level planning. Finally, these partial goals are sequentially reached in a purely reactive way, to efficiently face unexpected situations and mobile obstacles.

### III. SHARED CONTROL

As mentioned earlier, shared control can be defined as a situation in which both a human and a machine have an effect on how that machine achieves a certain goal, in our case, navigation. In most cases, research on shared control focuses either on interfaces to support different disabilities or in control architectures mostly based on previous robotic research. In this last case, shared control mostly focuses on switching the control from human to machine in an automatic way.

A first group of approaches leaves control mostly to the person, and automatic navigation is only triggered when a given situation is detected, such as imminent collision. Under these circumstances, a reactive algorithm, most frequently a dynamic-window-approach-based algorithm [14], is used to avoid obstacles. These approaches typically rely on a remote-control interface to teleoperate a robot manipulator or a mobile robot [1], [2]. The user may have full manual control of the mobile robot, allowing the machine to take the initiative to prevent collisions [1]. Otherwise, a robot manipulator may be visually controlled by a human operator, where the system helps the human driver in object placement on a target [2].

<sup>1</sup>General Public License available at <http://campusvirtual.uma.es/eperez/>.

A second group of approaches [23]–[26] relies on a basic set of primitives, such as AvoidObstacle, FollowWall, and PassDoorway, which can be used to assist the person in difficult maneuvers, either by manual selection or automatic triggering. Hence, the operator may guide the robot directly or switch among various autonomous behaviors to deal with complex situations. These approaches are mostly based on variations of the subsumption architecture, and the main difference among them is how behaviors are implemented.

Some other systems work like a conventional autonomous robot; the user simply provides a destination, and the mobile robot is totally in charge of getting there [27]–[30]. At any point, the user may override automatic control and take over. SENARIO [31], for example, uses a local navigation layer controlled by a high-level planner based on a topological representation of the environment. This planner takes into account several risk assessments and even the user condition. Véhicule Autonome pour Handicapés Moteurs [30] relies on a subsumptionlike architecture, where free space is represented by a set of rectangles and obstacles are defined by their boundaries. Using this model, an A\* algorithm is applied to obtain a trajectory free of obstacles to the goal. Wheellesley [32] also relies on the subsumption architecture, where the deliberative level is controlled by the user via EagleEyes and the reactive one is a simple obstacle-avoidance module.

It can be noted, however, that control is not really shared by human and machine at a given time instant; it is held by one or the other. In order to avoid brusque control changes that might cause discomfort to the user and slippage to the mobile robot, it would be better to actually distribute control between robot and user at all times, increasing or decreasing the weight of each one depending on the user's driving efficiency. If the combination of commands is properly performed, it may give a sense of continuity to both sources, and desired working parameters may be preserved. However, it is necessary to decide how to combine the robot-proposed trajectory and the human commands at every point of any given trajectory.

As commented, one of the main advantages of reactive approaches is how easily they can combine different sensors and goals. Since the DLA includes a reactive layer based on the potential fields approach (PFA), this layer can be used to include the user control interface as a goal function. In our case, we use a joystick to provide the direction and velocity that the user wants to impose to the mobile robot at each time instant. At the same time, the robot is willing to perform its own motion command, based on the desired reactive behavior. It needs to be noted that pure reactive behaviors tend to fall in local traps, may present oscillations, and do not deal well with subtleties, whereas humans are less precise and do not preserve curvature so well. In this paper, we propose a PFA-based method to merge behaviors at each time instant. Basically, the joystick direction is included as another vector in the potential field at each position (Fig. 2). If the weight of that vector is large, motion mostly obeys the driver, whereas if it is small, the robot tends to move on its own. However, they both have influence on motion at the same time. Weights are calculated in terms of efficiency, where this efficiency is evaluated in terms of local factors that depend on the desired behavior but always include

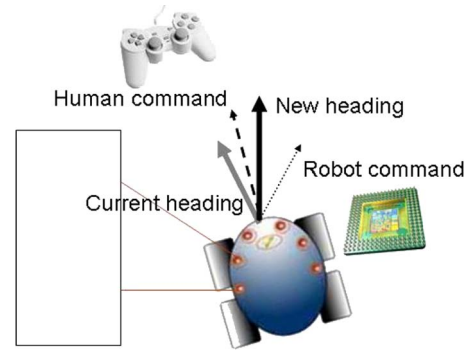


Fig. 2. Combination of robot and human commands.

safety and curvature. Thus, the main advantage of this approach is that it preserves curvature and grants safety, as most PFA-based algorithms. In the next section, test behaviors used in this approach are described, and efficiency in each case is specified.

#### A. Behaviors

Depending on the experiment, robot motion commands are provided by one of the three implemented behaviors: wall following (WF), corridor following (CF), and DC. Since these behaviors are based on a PFA, their performance is affected by oscillations and refusal to move close to obstacles. Thus, the robot might oscillate while navigating through a narrow corridor. Furthermore, it might not be able to cross a narrow door due to the presence of close obstacles. It must be noted that current implementations of the PFA cover the aforementioned problems via hybridization, by using local models of the environment instead of sensor readings in a straight way. However, to check the potential of human/robot collaboration, we wanted to use a PFA that can actually be helped by humans. Thus, we chose to use a purely reactive PFA, where disadvantages can be clearly perceived.

In order to achieve shared control, the emergent behavior will include robot motion commands and the joystick input. These commands will be added as weighted vectors. Consequently, it is important to determine the weight of these vectors at each time instant. Implemented behaviors obtain, from sonar readings, the rotational ( $v_{rR}$ ) and translational ( $v_{tR}$ ) velocities the robot supplies as its own motion command. Our model is based on an addition of forces (Fig. 3).

WF behavior tries to move the robot around the environment following the contour of its right wall at a predefined distance. Three forces are added. The first ( $f_{WFparallel}$ ) maintains the robot parallel to the wall. The second ( $f_{WFdist}$ ) tries to maintain the robot at a predefined distance to the right wall, and the third ( $f_{WFavoid}$ ) avoids obstacles that appear while the robot is navigating. Let  $k_{WFparallel}$ ,  $k_{WFdist}$ , and  $k_{WFavoid}$  be three constants that weigh  $f_{WFparallel}$ ,  $f_{WFdist}$ , and  $f_{WFavoid}$ , respectively. Assuming that  $\alpha_W$  is the angle to the nearest right wall,  $d_W$  is the current distance to the right wall, and  $d_F$  is the distance to obstacles in front of the robot,  $v_{rR}$  is calculated as

$$\begin{aligned} v_{rR} &= f_{WFparallel} + f_{WFdist} + f_{WFavoid} \\ &= k_{WFparallel} \cdot \alpha_W + k_{WFdist} \cdot (d_R - d_W) \\ &\quad + k_{WFavoid} \cdot (d_{WFSEC} - d_F) \end{aligned} \quad (1)$$

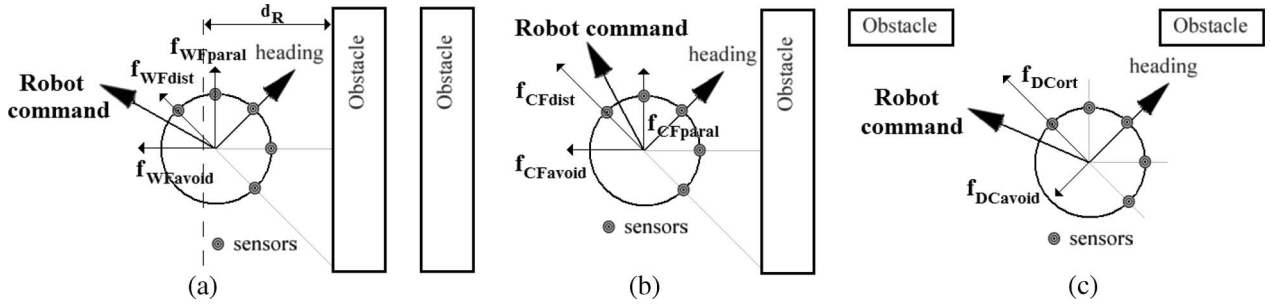


Fig. 3. Forces. (a) WF behavior. (b) CF behavior. (c) DC behavior.

where  $d_R$  is the distance the robot must keep to the right wall and  $d_{WFSEC}$  is a threshold distance to consider that an obstacle is near the robot.  $d_W$  and  $d_F$  are determined from sensor readings at run time, while constants  $d_R$  and  $d_{WFSEC}$  have been empirically established to 700 and 1000 mm, respectively. The three forces involved are shown in Fig. 3(a), where it is supposed that  $d_W$  is lower than  $d_R$ .

CF behavior allows the robot to go across a corridor by its center. As in the WF behavior, three forces are involved to perform the task.  $f_{CFpar}$  tries to move the robot parallel to a wall. However,  $f_{CFdist}$  does not try to maintain the robot at a distance of the wall as in the WF behavior, since it should move by the center of the corridor. Thus,  $f_{CFdist}$  takes into account its wideness. Finally,  $f_{CFAvoid}$  avoids obstacles that may appear in the way of the robot. Let  $k_{CFpar}$ ,  $k_{CFdist}$ , and  $k_{CFAvoid}$  be three factors that weigh  $f_{WFpar}$ ,  $f_{WFdist}$ , and  $f_{WFAvoid}$ , respectively. In this case, these factors may change as they depend on the corridor width. As stated in the WF behavior,  $\alpha_W$  is the angle to the nearest right wall,  $d_W$  is the current distance to the right wall, and  $d_F$  is the distance to obstacles in front of the robot. Then, if  $d_C$  is half of the wideness of the corridor,  $v_{tR}$  is calculated as

$$\begin{aligned} v_{tR} &= f_{CFpar} + f_{CFdist} + f_{CFAvoid} \\ &= k_{CFpar} \cdot \alpha_W + k_{CFdist} \cdot (d_C - d_W) \\ &\quad + k_{CFAvoid} \cdot (d_{CFSEC} - d_F) \end{aligned} \quad (2)$$

where  $d_{CFSEC}$  is a threshold distance to consider that an obstacle is near the robot.  $d_C$ ,  $d_W$ , and  $d_F$  are determined from sensor readings at run time, while constant  $d_{CFSEC}$  has been empirically established to 400 mm.

DC behavior allows the robot to cross a door by its middle to come in or come out of a room. This behavior is concerned with two forces.  $f_{DCCort}$  allows the robot to stay orthogonal to the doorframe, and  $f_{DCAvoid}$  avoids the doorframe, preventing the robot from colliding with it. Let  $k_{DCCort}$  be a constant that weighs  $f_{DCCort}$  and  $k_{DCAvoid}$  be another constant that weighs  $f_{DCAvoid}$ . Let  $d_L$  and  $d_R$  also be the distances to the left and right parts of the door, respectively. If, as commented in the WF and CF behaviors,  $d_F$  is the distance to obstacles in front of the robot,  $v_{tR}$  will be

$$\begin{aligned} v_{tR} &= f_{DCCort} + f_{DCAvoid} \\ &= +k_{DCCort} \cdot (d_L - d_R) \\ &\quad + k_{DCAvoid} \cdot (d_{DCSEC} - d_F) \end{aligned} \quad (3)$$

where  $d_{DCSEC}$  is a threshold distance to consider that an obstacle is close enough to the robot to be of interest.  $d_L$ ,  $d_R$ , and  $d_F$  are determined from sensor readings at run time, while the constant  $d_{DCSEC}$  has been experimentally fixed to 400 mm.

Translational ( $v_{tR}$ ) velocity is based on  $v_{tR}$ . For each of the behaviors,  $v_{tR}$  is

$$v_{tR} = v_{tMAX} \cdot \left[ 1 - \frac{v_{rR}}{v_{rMAX}} \right] \quad (4)$$

where  $v_{tMAX}$  and  $v_{rMAX}$  are the maximum translational and rotational velocities, respectively. Thus, the higher the rotational velocity is, the lower the translational one will be. In this paper, our experiments were carried out with  $v_{rMAX} = 15^\circ/s$  and  $v_{tMAX} = 0.2$  m/s. This simply means that the robot reduces its speed to turn safely.

### B. Shared Control Algorithm

In order to combine the robot response and human orders, shared control is achieved through a weighted linear combination of robot and joystick commands. The driver applies the rotational  $v_{tH}$  and translational  $v_{tH}$  velocities as human commands via joystick. Hence, a combination of both robot and human commands supplies velocities of the shared control, which are actually applied to the robot motors. Shared motion commands (rotational velocity  $v_{tS}$  and translational velocity  $v_{tS}$ ) are defined by

$$v_{tS} = 0.5 \cdot \eta_R \cdot v_{tR} + 0.5 \cdot \eta_H \cdot v_{tH} \quad (5)$$

$$v_{tS} = 0.5 \cdot \eta_R \cdot v_{tR} + 0.5 \cdot \eta_H \cdot v_{tH} \quad (6)$$

where  $\eta_R$  is the efficiency of robot motion commands and  $\eta_H$  is the efficiency of human motion commands. The shared motion command efficiency is defined as  $\eta_S$ . Efficiencies range from zero to one, with one being the maximum efficiency. It must be noted that  $\eta_S$  is not equal to  $\eta_R$  nor to  $\eta_H$ . Since shared commands linearly combine both robot and human ones,  $\eta_S$  will tend to average  $\eta_R$  and  $\eta_H$ . It can be observed that vectors are also weighted by a 0.5 factor. This factor will be used in the future to reflect the user and environment conditions and other external and/or global factors. The state of the user and the diagnostic achieved by medical staff should be taken into account. For example, if the driver is not very efficient due to personal factors, his/her influence on shared control may be reduced. Since it has been considered that both the user

and the environment are under normal conditions, we have not given more importance to any of them; however, the system is prepared to consider a different weighting. In this sense, we have already performed some experiments with different factors [33]. Results show that when the user's factor increases, he/she exerts more control over the system. In the same way, when the system increases its factor, the autonomous control increases.

Efficiencies (generalizing,  $\eta$ ) are evaluated in terms of local factors having an immediate effect on them, as they must be evaluated from a purely reactive point of view. Three factors are involved in efficiencies: softness ( $\eta_{sf}$ ), trajectory length ( $\eta_{tl}$ ), and security ( $\eta_{sc}$ ), each of them ranging from zero to one. Usually, robots will be more precise, whereas humans will be more versatile. If both perform equally well, the global command will be the average of them both.

Softness was evaluated as the angle between the current direction of the robot and the command vector. It is meant to take into account that many robots are nonholonomic and that it is better to keep a trajectory as soft as possible to avoid slippage and oscillations. If  $C_{sf}$  is a constant and  $\alpha_{dif}$  is the angle difference between the current direction and the command vector,  $\eta_{sf}$  will be

$$\eta_{sf} = e^{-C_{sf} \cdot |\alpha_{dif}|}. \quad (7)$$

Trajectory length could not be globally evaluated at a punctual time instant; thus, we chose to measure it locally in terms of the angle conformed by the robot heading and the direction toward the next partial goal provided by the global planner. Obviously, the shortest way to reach that goal is to make that angle zero. Let  $C_{tl}$  be a constant and  $\alpha_{dest}$  the angle between the robot heading and the direction toward the next partial goal. Hence,  $\eta_{tl}$  is calculated as

$$\eta_{tl} = e^{-C_{tl} \cdot |\alpha_{dest} - \alpha_{dif}|}. \quad (8)$$

The third factor, security, is evaluated in terms of the distances to the closest obstacles at each instant. The closer you get to obstacles, the more risky your trajectory is. Assuming that  $C_{sc}$  is a constant and that  $\alpha_{min}$  is the angle difference between the current direction and the direction of the closest obstacle,  $\eta_{sc}$  will be

$$\eta_{sc} = 1 - e^{-C_{sc} \cdot |\alpha_{min} - \alpha_{dif}|}. \quad (9)$$

Finally, efficiency is obtained through the combination of the three former factors

$$\eta = \frac{\eta_{sf} + \eta_{tl} + \eta_{sc}}{3}. \quad (10)$$

$\eta_{sf}$ ,  $\eta_{tl}$ , and  $\eta_{sc}$  are weighted identically because we are not interested in giving more significance to one of them, although it is not necessary. Indeed, the system can easily include a different weighting process. However, it was not the purpose of this paper.  $\eta$ , defined as in (10), provides a way to evaluate the different options (robot, human, and shared commands) for comparison purposes. Even more, motion commands may be evaluated in terms of factors having an immediate effect on efficiency. It is necessary to note that it is not necessarily advisable

to achieve a local efficiency equal to one. First, some efficiency factors are opposite in presence of obstacles, such as keeping far from obstacles and trying to turn as little as possible. In addition, the layout of the environment may make it necessary not to head to the goal at all times; however, this being a global consideration, the local efficiency cannot contemplate this fact. Finally, in order to move through narrow places, such as doors, security efficiency might be low sometimes.

#### IV. EXPERIMENTS AND RESULTS

In order to test the performance of the proposed shared control algorithm, we have performed several real tests using a Pioneer robot equipped with eight frontal sonar sensors. While navigating, the robot builds an evidence grid [34] to show where obstacles are. In these grids, free space is printed in black, obstacles is printed in white, and unexplored areas are presented in gray. It is necessary to point out that grids are simply used to graphically represent the environment. The trace followed by the robot is superimposed in white color to the evidence grid built during the movement. Tests were performed with a WingMan Attack 2 joystick from Logitech, for right-handed people.

##### A. User Study

Tests were limited to thirteen volunteers, since prior research on collaborative control indicates that a small number of users is sufficient [35]. The users' ages ranged from 23 to 35 years old. The sample space included people familiar and unfamiliar with robots, including professors and students. We also made a point of having a significant group of people not yielding a driving license. Similarly, we included left-handed people in the experiment. Finally, as people drove the robot using a joystick, we included people familiar and unfamiliar with video games as well.

All these people were asked to drive the robot through four different environments based on qualitative guidelines. Fig. 4 shows the four real test environments used for our experiments. From this point on, they will be referred to as follow wall (FW), narrow corridor (NC), double NC (DNC), and pass door (PD). These four tasks have been purposefully chosen to be simplistic to test the performance of the system. This is not a problem since complex tasks, such as exploring an unknown area or moving between two points, can be decomposed into simple tasks, such as WF, CF, or DC [36]–[38], which are the implemented behaviors of our system. Thus, complex tasks may be analyzed in terms of the results obtained for simple tasks. When developing more complex tasks, we rely on a top-down approach where the performance of the system is clearly coupled with task decomposition. The idea of using the proposed scheme is to provide more control at the reactive level for unexpected situations without having to feed back information to higher levels if unnecessary. Thus, we avoid replanning and, consequently, save processing time.

All volunteers performed each test five times, resulting in 20 trajectories, in no particular order. We did not set the same order for all because we wanted to test if drivers starting with more

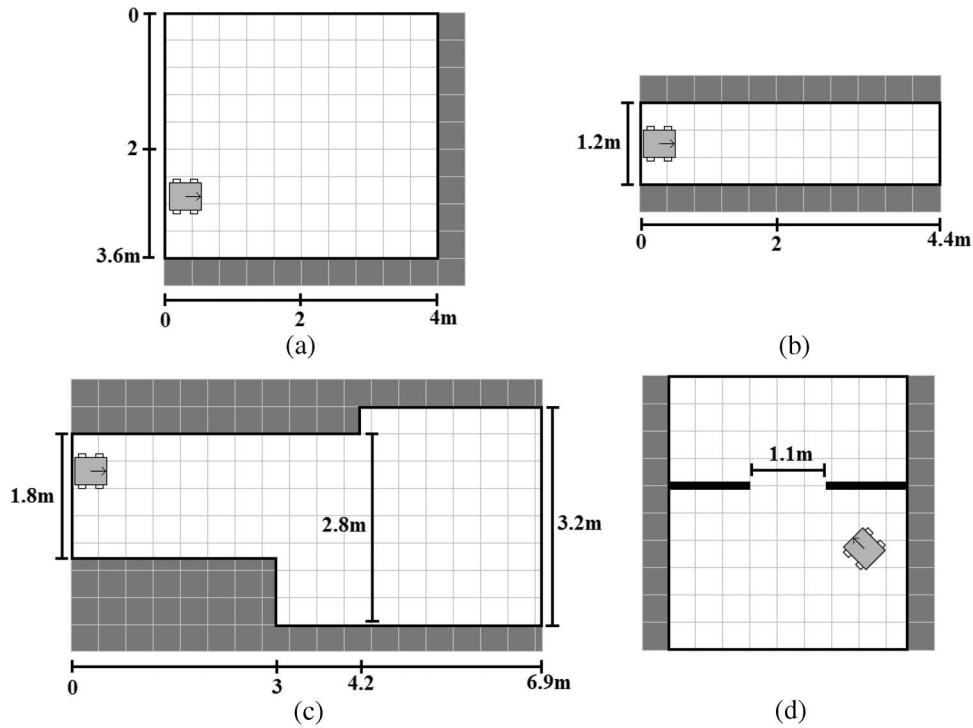


Fig. 4. Real environments configurations. (a) FW. (b) NC. (c) DNC. (d) PD.

complex trajectories outperformed the ones starting with easier ones or, on the contrary, if they did worse.

Volunteers were not informed about shared control; they were simply asked to follow some guidelines, as if all control was theirs. However, at some point of the tests, most of them realized they were having some help with their trials. During the experiments, we wanted to test several things: 1) if results using shared control were better than those using either human or robot guided trajectories alone; 2) if humans managed to learn safely how to guide the system; and 3) if they felt comfortable with this shared control. To achieve this, we established some metrics, explained in Section III-B, and provided a questionnaire to check the previously commented features and the personal opinion of users on the system performance. Table I presents the model of the questionnaire.

Users' characteristics are presented in Table II, based on the data from the questionnaires. It must be pointed out that the sample space consists of five men and eight women, with one man and one woman being left handed. As commented, we have purposely introduced five people without car driving license. It is also significant that nine people never or rarely play video games. Finally, there are eight people with no or little relation to robotics, as we were looking for.

### B. Robot Performance

The requested trajectories for FW, NC, DNC, and PD environments—as performed by the robot alone—are shown in Fig. 5. As stated before, the trace of the robot is superimposed in white color to the evidence grid. In Fig. 5(a), the trajectory was supposed to keep a fixed distance from the nearest right wall all the time. In Fig. 5(b), the robot moved autonomously in the middle of the corridor. The same had to be done in Fig. 5(c);

TABLE I  
USER QUESTIONNAIRE

<i>Subject ID:</i>			
1. Age:			
2. Sex:			
<input type="checkbox"/> Man		<input type="checkbox"/> Woman	
3. I drive:			
<input type="checkbox"/> Bicycles		<input type="checkbox"/> Motorbikes	<input type="checkbox"/> Cars
4. My pulse is:			
<input type="checkbox"/> Bad		<input type="checkbox"/> Not too bad	<input type="checkbox"/> Good
<input type="checkbox"/> Very good			
5. I usually wear glasses:			
<input type="checkbox"/> Yes		<input type="checkbox"/> No	
6. I play video games:			
<input type="checkbox"/> Never		<input type="checkbox"/> Rarely	<input type="checkbox"/> Weekly
<input type="checkbox"/> Almost daily		<input type="checkbox"/> Daily	
7. My relation to robotics is:			
<input type="checkbox"/> None		<input type="checkbox"/> Little	<input type="checkbox"/> Enough
<input type="checkbox"/> A lot			
8. Mark the difficulty of each test (1 minimum, 5 maximum):			
FW test	NC test	DNC test	PD test
<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1
<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2
<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4
<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5
9. Mark your grade of control for each test (1 minimum, 5 maximum):			
FW test	NC test	DNC test	PD test
<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1
<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2
<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4
<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5
10. Mark your grade of stress for each test (1 minimum, 5 maximum):			
FW test	NC test	DNC test	PD test
<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1
<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2
<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3
<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4
<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5



TABLE II  
USER CHARACTERISTICS

Sex	Man	5
	Women	8
Car driving license	Yes	8
	No	5
Pulse	Not too bad	5
	Good	8
Wearing glasses	Yes	7
	No	6
Playing video games	Never	3
	Rarely	6
	Weekly	3
	Almost daily	1
Relation to robotics	None	7
	Little	1
	Enough	1
	A lot	4

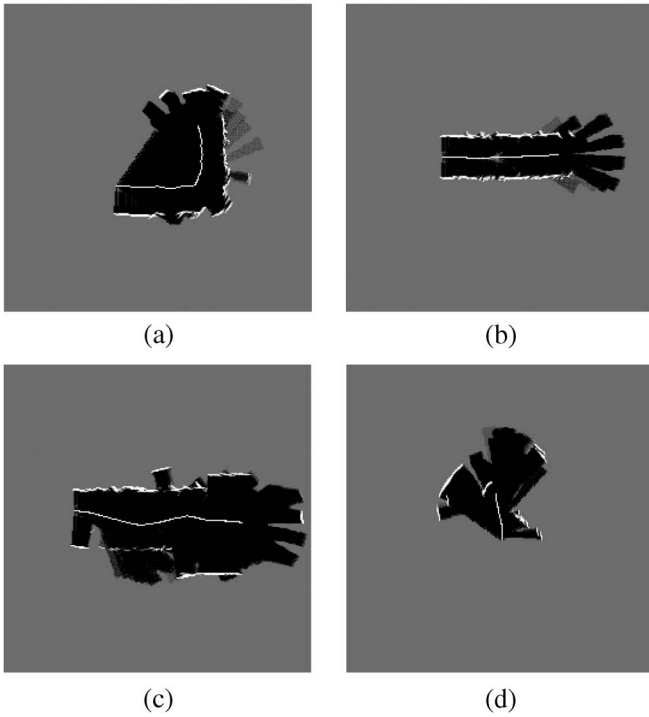


Fig. 5. Trajectories performed by the robot alone. (a) FW environment. (b) NC environment. (c) DNC environment. (d) PD environment.

however, the corridor widened, and the end and the trajectory had to be corrected to achieve the same effect. Finally, the goal in Fig. 5 was to move through a narrow door. As commented in Section II, the control system in the robots works in a hierarchical way; the deliberative layer sets partial goals for the reactive one, which is based on a PFA. Consequently, the local performance of the trajectory is affected by the same problems than the PFA, namely, oscillations and refusal to move close to obstacles. Local minima, however, is dealt with by the deliberative layer.

In order to evaluate efficiency  $\eta$  all through a trajectory, we used both a graphic representation and simple statistics. Graphically, each one of the factors involved in efficiency is assigned to a red–green–blue color channel: red for softness  $\eta_{sf}$ , green for trajectory length  $\eta_{tl}$ , and blue for security  $\eta_{sc}$ . To that purpose, a 3-D representation is employed, where the  $XY$  plane

represents space, while the  $Z$ -axis is assigned to combined efficiency at each point. When the trajectory is deployed onto the  $XY$  plane, it is possible to know how good each control parameter rated at each trajectory point depending on its color. If projected onto  $XZ$ , the height of the point may represent, at the same time, the combined efficiency at the point. This representation proved to be useful, as it should be recognized, from a simple look at their colors, how trajectories behaved at certain areas.

Figs. 6 and 7 show how the robot behaved in its four trajectories when moving completely autonomously. It may be appreciated that trajectories present typical oscillations of approaches based on PFA. It is shown in Figs. 6(c) and (d) and 7(c) and (d) that, in general, the global efficiency of the reactive control  $\eta_R$  is quite high. Exactly, average efficiencies are 74.13%, 91.90%, 80.84%, and 58.84% for the FW, NC, DNC, and PD tests, respectively (Table III). It is expected, however, that efficiency factors depending on oscillations, such as smoothness and straightness to goal, decrease when the agent moves close to obstacles [Figs. 6(c) and 7(d)]. Security, instead, should be adequate, depending on the threshold used by behaviors. Its value  $\eta_{sc}$  is above 75% for all the tests, as the robot tries to avoid obstacles while performing.

Average efficiencies presented in Table III rate the robot performance along its whole trajectory. However, there are situations with lower efficiency than the average. One such situation happens in the FW test. Global efficiency decreases to an average less than 50% while turning in the corner, Zone A in Fig. 6(a), (c), and (e). It belongs to a safe trajectory where smoothness and straightness to goal have a low contribution to global efficiency. In fact, average softness and trajectory length are below 25% while average security factor is above 95%. Table III presents, as well, the sum and the variance of the curvature function obtained from the trajectory performed by the robot alone. These values evaluate global factors on how good a given trajectory is from a qualitative point of view, no sharp turns, and conservation of trajectory. For a given trajectory  $y = f(x)$ , its curvature function is defined by

$$\rho = \frac{\frac{d^2 y}{dx^2}}{\left[1 + \left(\frac{dy}{dx}\right)^2\right]^{3/2}} \quad (11)$$

being calculated as in [39]. Due to the turn in the corner [Fig. 6(e)], the sum of the curvature function should be of  $90^\circ$  in the FW test. It is close to  $100^\circ$ , since the mobile robot oscillates after the turn, considering that the robot behavior, in this experiment, is based on a PFA. Furthermore, oscillations can also be observed in the DNC test [Fig. 5(c)], while the robot tries to center in the corridor. These oscillations are concerned with two zones of the trajectory, Zones B and C in Fig. 7(a), (c), and (e). In both cases, global efficiency is less than 68%, i.e., significantly lower than the average (80.84%). If we inspect the three factors involved, the following are shown in both cases: 1) Trajectory is safe since the average security factor is above 91%, and 2) average softness and trajectory length are below 60%.

From this point on, it must be remembered that commands to the robot motor system at any time instant will be a linear

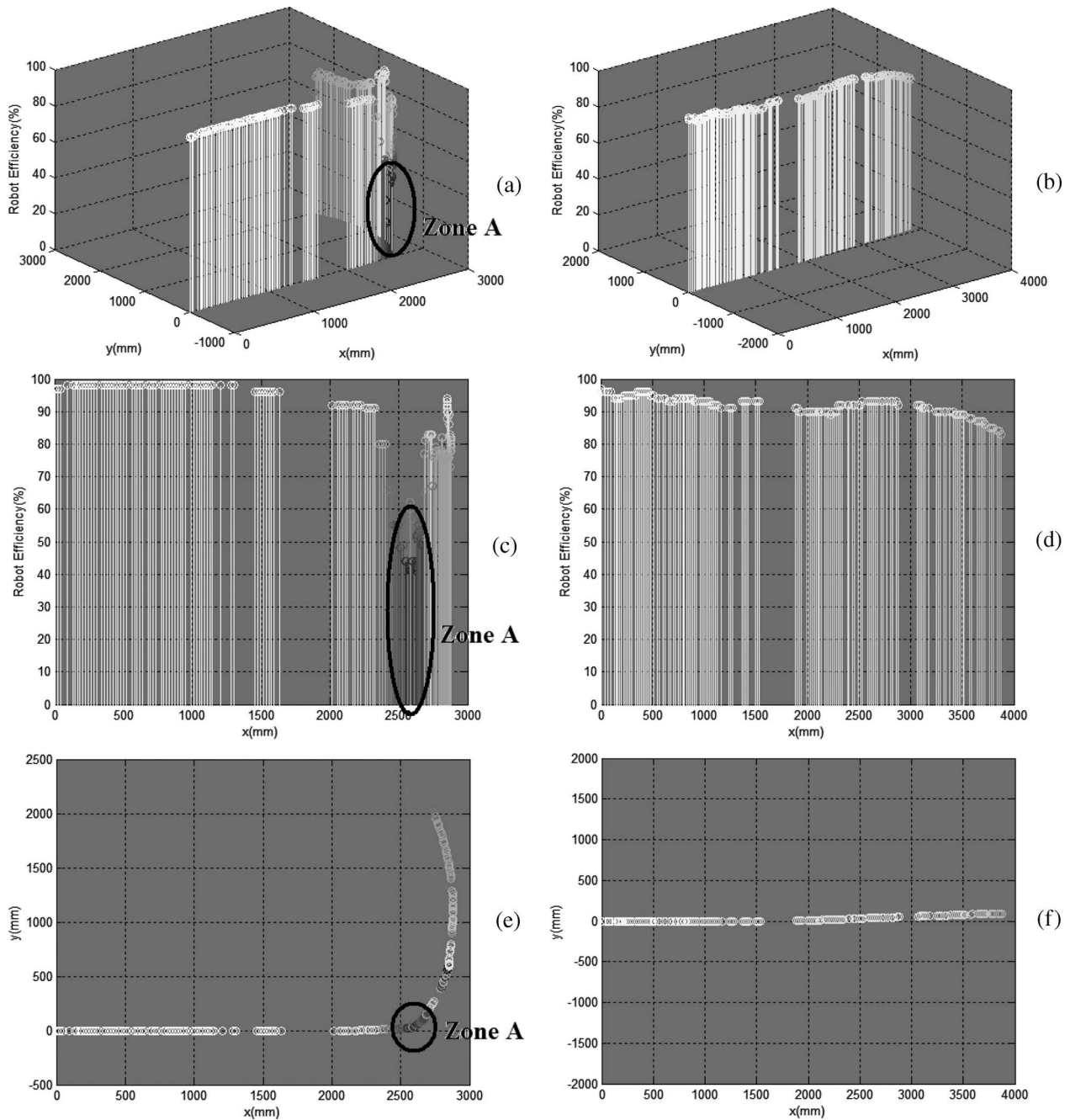


Fig. 6. Robot performance in FW and NC test environments. (a) Three-dimensional representation of robot efficiency in FW. (b) Three-dimensional representation of robot efficiency in NC. (c) XZ projection of robot efficiency in (a). (d) XZ projection of robot efficiency in (b). (e) XY projection of robot efficiency in (a). (f) XY projection of robot efficiency in (b).

combination of the user input and the autonomous control architecture output. It is necessary to note, however, that the system commands will not necessarily turn into the trajectories in Fig. 5, considering that they depend on the instance of the robot at any point and that such an instance is affected by the human driver in the next experiments.

### C. Shared Control Performance

When experiments with people were performed, shared control was compared with human and robotic controls at each instant, as the robot had to contemplate human guide all the

time. Tables IV–VI show the results of the shared control performance for the thirteen users, being similar to that of the robot performance (Table III). For each user and each experiment, the average along the five trials is presented.

A first interesting fact is that all people involved in robotics performed similarly in all four experiments. In average, they achieved 79%, 91%, 79%, and 66% shared efficiency in the FW, NC, DNC, and PD tests, respectively. Individuals registered some differences, however, in control smoothness. We evaluated the variation of the curvature at each point, calculated simply as the difference between the current heading of the robot and the one desired by the user. All users performed



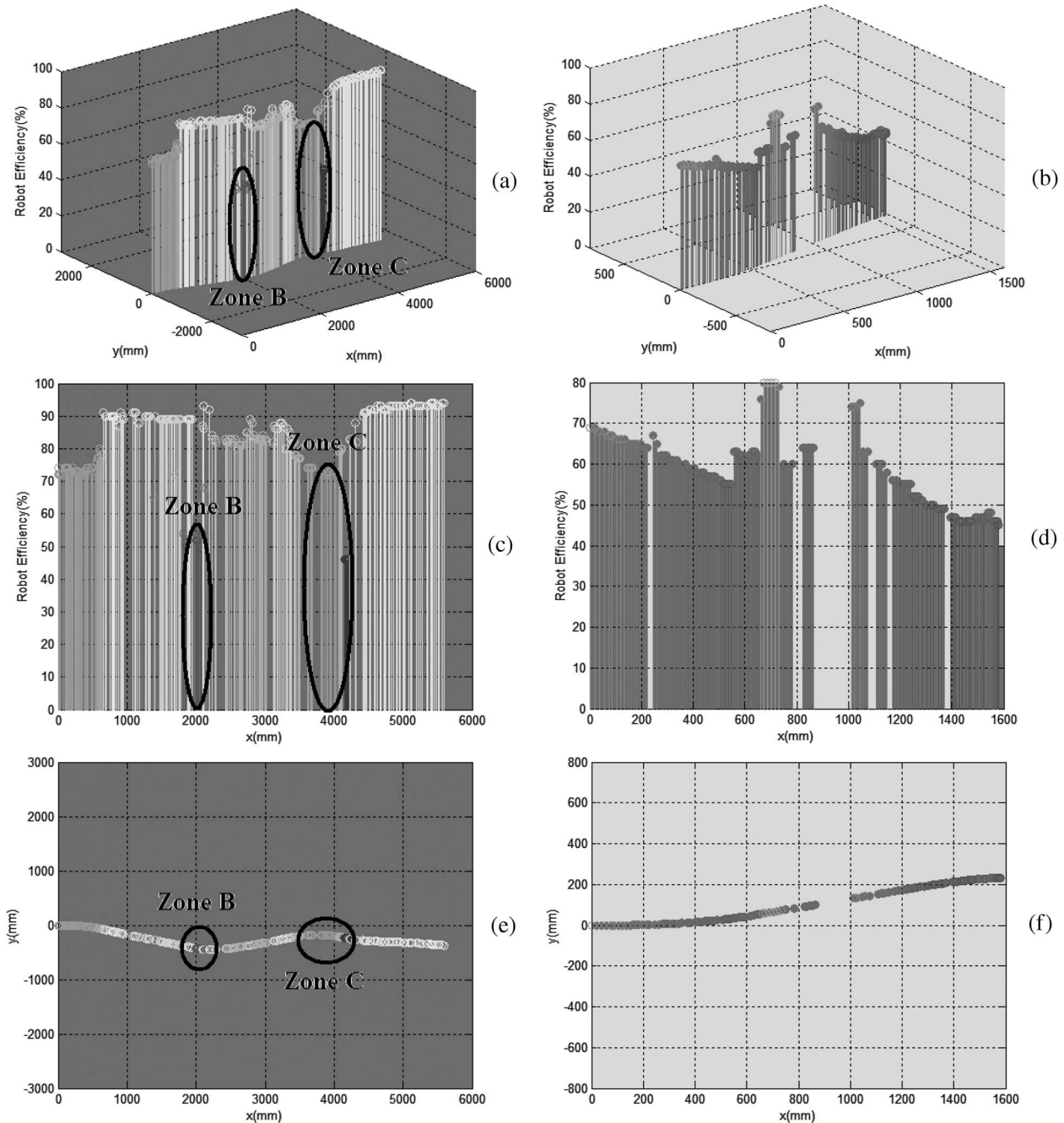


Fig. 7. Robot performance in DNC and PD test environments. (a) Three-dimensional representation of robot efficiency in DNC. (b) Three-dimensional representation of robot efficiency in PD. (c) XZ projection of robot efficiency in (a). (d) XZ projection of robot efficiency in (b). (e) XY projection of robot efficiency in (a). (f) XY projection of robot efficiency in (b).

TABLE III  
STATISTICAL RESULTS OF ROBOT PERFORMANCE

Test	FW	NC	DNC	PD
Average Efficiency (%)	74.13	91.90	80.84	58.84
Average $\eta_{sf}$ (%)	63.44	97.24	73.69	64.54
Average $\eta_{tl}$ (%)	63.57	88.59	72.02	34.84
Average $\eta_{sc}$ (%)	94.83	90.00	96.84	77.09
Variance Efficiency (%)	21.89	2.79	11.81	9.01
Sum Curvature Function (°)	101.15	1.68	-1.22	2.64
Variance Curvature Function (°)	5.89	0.56	8.98	7.02

similarly in this aspect in NC, where the trajectory was supposed to be a straight line. There was not much variation either in their performance to move through the door (PD). However,

we registered significant differences in their performance in the FW and DNC experiments, mostly because they had more freedom to decide when to start to turn the corner in FW or to correct the trajectory in DNC. It is necessary to note that, thus far, curvature was local; hence, it only measures reactivity to perceived errors in desired trajectories, namely, jerkiness.

In order to evaluate real differences in global decisions, we also measured the variance in the complete trajectory curvature. This factor depends on medium term guiding, meaning that the human decides to start turning earlier or later depending on his/her global perception of the environment to avoid sharp movements. These decisions are mostly influenced by the way people drive and all subjects in this case not only had a driving

TABLE IV  
STATISTICAL RESULTS OF SHARED CONTROL  
PERFORMANCE FOR USERS 1–5

Test	FW	NC	DNC	PD
<b>User 1</b>				
Average Efficiency (%)	77.93	91.53	80.51	65.18
Average $\eta_{sf}$ (%)	68.23	92.93	74.54	54.86
Average $\eta_{tl}$ (%)	70.01	92.26	71.00	59.06
Average $\eta_{sc}$ (%)	95.94	88.37	95.85	81.96
Variance Efficiency (%)	15.27	3.88	12.22	7.04
Sum Curvature Function ( $^{\circ}$ )	99.56	-0.13	3.32	-30.23
Variance Curvature Function ( $^{\circ}$ )	5.13	0.62	8.11	8.49
<b>User 2</b>				
Average Efficiency (%)	80.19	91.85	77.93	66.24
Average $\eta_{sf}$ (%)	68.49	95.33	72.31	58.24
Average $\eta_{tl}$ (%)	77.27	92.12	65.05	58.47
Average $\eta_{sc}$ (%)	95.10	89.17	96.25	82.32
Variance Efficiency (%)	14.70	4.82	11.23	6.94
Sum Curvature Function ( $^{\circ}$ )	103.65	1.50	6.55	-27.03
Variance Curvature Function ( $^{\circ}$ )	5.25	0.52	10.07	7.61
<b>User 3</b>				
Average Efficiency (%)	78.62	90.77	79.91	63.87
Average $\eta_{sf}$ (%)	68.64	91.16	75.54	51.53
Average $\eta_{tl}$ (%)	71.61	90.41	67.35	57.11
Average $\eta_{sc}$ (%)	96.05	90.59	96.66	82.98
Variance Efficiency (%)	15.17	4.63	11.10	9.20
Sum Curvature Function ( $^{\circ}$ )	100.18	3.53	5.68	-39.05
Variance Curvature Function ( $^{\circ}$ )	6.77	1.08	6.89	8.88
<b>User 4</b>				
Average Efficiency (%)	80.86	93.81	88.37	65.41
Average $\eta_{sf}$ (%)	70.53	96.69	85.70	63.12
Average $\eta_{tl}$ (%)	73.85	94.48	83.09	52.11
Average $\eta_{sc}$ (%)	98.15	90.28	96.13	80.96
Variance Efficiency (%)	16.62	2.04	6.74	5.57
Sum Curvature Function ( $^{\circ}$ )	103.93	1.01	-0.67	-31.21
Variance Curvature Function ( $^{\circ}$ )	6.08	0.25	4.96	9.21
<b>User 5</b>				
Average Efficiency (%)	75.19	90.12	77.21	61.66
Average $\eta_{sf}$ (%)	63.50	92.00	68.23	46.65
Average $\eta_{tl}$ (%)	64.51	87.97	66.43	51.79
Average $\eta_{sc}$ (%)	97.75	90.28	96.87	86.61
Variance Efficiency (%)	17.25	5.65	12.13	11.18
Sum Curvature Function ( $^{\circ}$ )	102.23	1.97	4.93	-31.05
Variance Curvature Function ( $^{\circ}$ )	6.56	1.37	8.39	12.35

TABLE V  
STATISTICAL RESULTS OF SHARED CONTROL  
PERFORMANCE FOR USERS 6–9

Test	FW	NC	DNC	PD
<b>User 6</b>				
Average Efficiency (%)	80.61	89.14	72.96	58.74
Average $\eta_{sf}$ (%)	70.48	90.80	61.63	45.26
Average $\eta_{tl}$ (%)	75.08	87.93	60.71	49.45
Average $\eta_{sc}$ (%)	96.42	88.83	96.35	81.45
Variance Efficiency (%)	15.84	8.21	16.16	9.90
Sum Curvature Function ( $^{\circ}$ )	97.47	4.36	3.64	-24.75
Variance Curvature Function ( $^{\circ}$ )	5.96	2.24	8.16	10.63
<b>User 7</b>				
Average Efficiency (%)	71.28	92.29	79.41	61.34
Average $\eta_{sf}$ (%)	65.91	96.28	73.51	50.39
Average $\eta_{tl}$ (%)	61.03	89.33	69.37	50.08
Average $\eta_{sc}$ (%)	87.15	90.72	95.36	83.75
Variance Efficiency (%)	16.48	4.06	11.67	8.40
Sum Curvature Function ( $^{\circ}$ )	105.27	0.52	5.00	-25.74
Variance Curvature Function ( $^{\circ}$ )	8.65	1.84	9.34	11.87
<b>User 8</b>				
Average Efficiency (%)	79.77	92.63	77.40	65.18
Average $\eta_{sf}$ (%)	66.95	95.62	67.60	53.17
Average $\eta_{tl}$ (%)	74.50	93.87	67.27	61.57
Average $\eta_{sc}$ (%)	97.63	88.35	96.66	80.93
Variance Efficiency (%)	16.14	2.21	13.34	8.45
Sum Curvature Function ( $^{\circ}$ )	97.50	3.71	2.67	-37.46
Variance Curvature Function ( $^{\circ}$ )	5.61	0.61	9.41	10.92
<b>User 9</b>				
Average Efficiency (%)	79.25	91.29	78.04	66.43
Average $\eta_{sf}$ (%)	66.85	92.38	69.29	53.32
Average $\eta_{tl}$ (%)	72.41	91.21	67.94	57.07
Average $\eta_{sc}$ (%)	98.34	89.90	96.90	88.86
Variance Efficiency (%)	15.42	8.30	11.56	8.21
Sum Curvature Function ( $^{\circ}$ )	94.22	1.35	5.18	-34.12
Variance Curvature Function ( $^{\circ}$ )	5.04	0.38	9.07	9.59

license but also knew well how to drive the robot. We compared these global curvature changes to the reactive trajectories and found out that, in all cases, global smoothness is better in the aforementioned humans than in the autonomous control for the FW experiment. This was only expected, as this case is prone to oscillations. In the corridor tests, however, emergent curvature variations were similar to the robot or, in some cases, slightly worse, meaning that humans tried to correct their curvature all the time to keep in the middle of the corridor, whereas the system managed to reach equilibrium and not oscillate. It is interesting to note, as well, that drivers performing worse than the autonomous control system in this respect had less jerky trajectories, meaning that they kept their heading as much as possible but had to smoothly correct it all the time. Humans outperforming the machine in these cases tended to turn sharp when needed but did not correct their trajectories the rest of the time. Nevertheless, in this group of drivers, the average shared efficiency of the human was better or similar to the robot in all cases. The only explanation we found to this driving likeness was that, knowing about how the robot

operates, drivers suspected what it would do and, purposefully or not, tried to compensate for the robot weaknesses only when necessary. Fig. 8 shows a comparison of such a situation with a user involved in robotics and a user noninvolved in it. Fig. 8(a) and (b) shows the 3-D representation of the shared efficiency along the path and the trajectory, respectively, in the DNC test for a user involved in robotics. Fig. 8(c) and (d) show the same graphics for a user noninvolved in robotics. The person involved in robotics turns sharp at the beginning of the trial to center the robot in the corridor. Once the robot is centered, this trajectory is not corrected, since the user applies his/her experience in the field to compensate only when necessary. The user noninvolved in robotics tries to center the robot in the corridor in a smoother way. However, the user must correct the trajectory of the robot since he/she has not taken into account the robot itself. Average shared efficiency for the user involved in robotics is 80.4%, similar to the robot alone (80.84%), while the same average shared efficiency is slightly worse for the user noninvolved in robotics (77.5%). The not-too-large difference occurs because the robot helps the user all the time. Robot help occurs even for users involved in robotics, since, in tests, it has been qualitatively noticed that they entrust their experience, and thus, the system actually helps them.

It must be noted that efficiency is better at the beginning of the trajectory in Fig. 8(a) (Zone D) than in Fig. 8(c) (Zone E). Furthermore, the average efficiency in this area is 75.1% for the user involved in robotics, while it is only 66.7% for the

TABLE VI  
STATISTICAL RESULTS OF SHARED CONTROL  
PERFORMANCE FOR USERS 10–13

Test	FW	NC	DNC	PD
<b>User 10</b>				
Average Efficiency (%)	75.45	88.33	79.62	68.71
Average $\eta_{sf}$ (%)	64.34	89.25	73.22	60.43
Average $\eta_{tl}$ (%)	64.48	88.01	68.77	59.11
Average $\eta_{sc}$ (%)	97.60	87.89	96.71	86.55
Variance Efficiency (%)	18.33	14.83	12.79	6.96
Sum Curvature Function ( $^\circ$ )	95.92	1.68	3.78	-30.63
Variance Curvature Function ( $^\circ$ )	5.79	0.55	8.40	8.53
<b>User 11</b>				
Average Efficiency (%)	78.33	91.46	76.67	56.88
Average $\eta_{sf}$ (%)	67.11	93.22	66.92	43.53
Average $\eta_{tl}$ (%)	71.11	96.76	65.85	43.13
Average $\eta_{sc}$ (%)	96.76	89.21	97.04	83.79
Variance Efficiency (%)	14.70	4.61	13.04	9.06
Sum Curvature Function ( $^\circ$ )	100.96	2.14	7.47	-9.26
Variance Curvature Function ( $^\circ$ )	9.13	0.57	9.16	11.77
<b>User 12</b>				
Average Efficiency (%)	79.06	89.45	76.54	71.58
Average $\eta_{sf}$ (%)	70.15	90.79	66.27	62.28
Average $\eta_{tl}$ (%)	70.82	86.95	66.69	66.71
Average $\eta_{sc}$ (%)	96.47	90.03	96.65	85.92
Variance Efficiency (%)	18.06	6.64	12.41	9.35
Sum Curvature Function ( $^\circ$ )	100.37	1.85	4.65	-12.75
Variance Curvature Function ( $^\circ$ )	7.03	0.63	9.78	6.21
<b>User 13</b>				
Average Efficiency (%)	80.57	92.87	78.58	61.46
Average $\eta_{sf}$ (%)	70.38	95.09	70.39	46.11
Average $\eta_{tl}$ (%)	74.48	93.34	68.95	50.78
Average $\eta_{sc}$ (%)	97.35	89.71	96.39	87.29
Variance Efficiency (%)	16.68	3.41	12.10	11.25
Sum Curvature Function ( $^\circ$ )	96.26	1.92	3.90	-26.72
Variance Curvature Function ( $^\circ$ )	6.02	0.50	10.67	11.25

noninvolved one. Even more, the efficiency yields a softness factor ( $\eta_{sf}$ ) of 67.9% and 49.5%, respectively. Thus, the person who knows about the performance of the robot alone leads to a smoother trajectory even when he/she has turned sharp at the beginning. On the other hand, the user noninvolved in robotics tries to drive through a smooth path. However, control applied by the robot decreases the efficiency at the beginning, with a lower smooth factor.

To evaluate the learning process during a test, we define the learning rate  $LR$ . It defines the relation between average efficiencies of human motion commands and robot motion commands through the whole trajectory to estimate the percentage of control exerted by both the human and the robot

$$LR = \frac{\text{Average}(\eta_H)}{\text{Average}(\eta_R)}. \quad (12)$$

Fig. 9 shows the learning rate of a user involved in robotics along the five trials of the FW test.  $LR$  ranges from 0.89 to 0.97. The highest learning rate occurs at the first trial. Furthermore, there are no significant differences among the five trials since  $LR$  presents a nearly constant value. Such a situation points out that the user knows how the robot operates from the first to the last trial. Thus, as expected, the learning process is not noticeable, since the user takes into account shared performance from the beginning.

On a second instance, we also observed that people not having a driving license, not fond of video games, and who were not related to robots either tended to perform better than the rest, particularly regarding adaptation, presenting a neat learning curve. We believe that they were more careful in their driving and, although they typically drove slower, they made sure to keep to the premises of the experiment as tight as possible. In average, this group of people achieved 80%, 90%, 79%, and 65% shared efficiency for the FW, NC, DNC, and PD tests, respectively. These results may seem to be very similar to the experienced group of users; however, there is a significant difference in their learning curves. Moreover, in one of the individuals, the average shared efficiency of the human was above the machine all the time (80.8%, 93.8%, 88.4%, and 65.4% in the FW, NC, DNC, and PD tests, respectively). In no other case have we detected something like this. The rest of the individuals did, in this respect, very much like the rest, averaging above the machine in the FW and PD experiments and slightly worse in the corridors, as usual. Although users felt comfortable with the system, some of them qualitatively realized that the machine exerted a certain degree of control over the global performance in the FW and PD tests but did not feel this control in the NC and DNC experiments. Fig. 10 shows a trial in the FW test where the user feels the machine is partially controlling the system. When this user was performing the FW experiment, he noticed that the robot was partially controlling the movement. This situation happened while turning in the corner (Zone F) in Fig. 10(a). In this test, the user was not experienced with the system, since the user tried to approach too close to the wall before turning [Fig. 10(b)], with an average joystick command efficiency of less than 38%. In such a context, the robot performs its behavior on its own, following the wall at a predefined distance, with an average reactive efficiency of 50%. Hence, the shared efficiency is about 44%, where reactive commands have a higher weight. Although the robot moves far enough from obstacles, the user does not perform the task correctly, considering that he/she should have moved the robot not so close to the wall but, instead, should have driven the agent at a larger predefined distance. Furthermore, the users feel the system is helping them particularly in the FW and PD tests while, actually, this help is larger while centering in a corridor (NC and DNC tests).

Third, we compared the performance of people used to video games not having a driving license and vice versa and concluded that video gamers performed slightly better in average, specifically because the global curvature was better preserved by this group. We believe that this is mostly influenced by the robot interface—a joystick—and because people were not actually mounted on the robot but were guiding it from a distance. In order to extract a more reliable conclusion to this respect, we would need to perform a similar experiment on a robotized car or wheelchair and/or to change the interface to a driving wheel.

Regarding the order of the different experiments, the most obvious observation was that people starting with simpler experiments, such as FW or NC, performed better in the hardest one (PD) in terms of curvature and efficiency from the very first try than those starting with DNC or PD. In this sense,



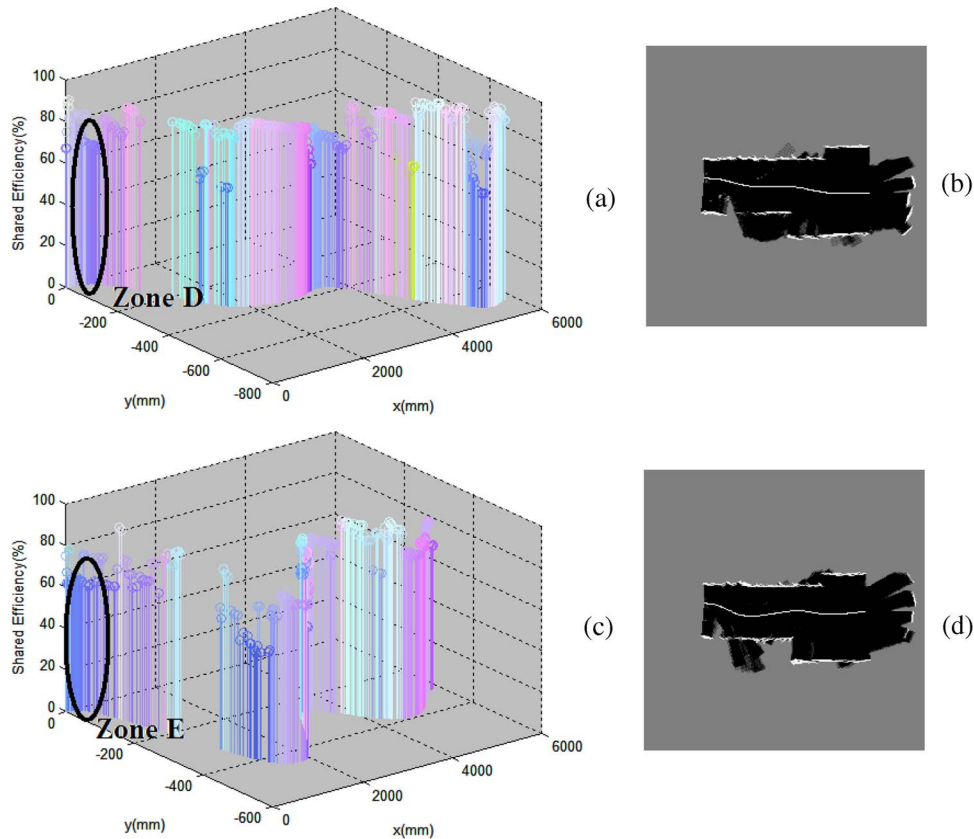


Fig. 8. Person involved in robotics. (a) Three-dimensional representation of shared efficiency in DNC. (b) Trajectory. Person noninvolved in robotics. (c) Three-dimensional representation of shared efficiency in DNC. (d) Trajectory.

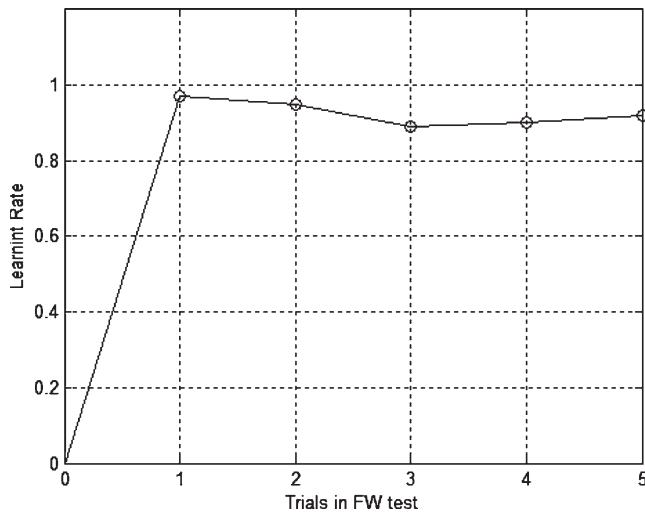


Fig. 9. Learning rate in FW test.

Fig. 11 shows the learning process of a user in the PD test. This user started with the simplest experiments (FW and then NC) followed by the DNC test. The last experiment was the hardest one (PD). Fig. 11 graphically shows the learning rate of the user along the five trials of this PD test.

It is shown in Fig. 11 that the first trial  $LR$  is 0.73, rising as the user performs more trials until an  $LR$  of 1.14 in the fifth trial. It must be noted that the  $LR$  increases from the first toward the last trial. This fact makes the learning process of the

user clear. Furthermore, in the last trial, the human is even more efficient than the robot ( $LR = 1.14$ ). Hence, the human exerts more control of the mobile than the robot, thanks to the learning process the human has experienced from the first trial in the FW test to the last trial in the PD experiment.

When a user starts with one of the simplest experiments (FW and NC tests), the situation is different. Fig. 12 shows the learning process in the NC test for a user starting with the hardest experiment (PD test), followed by the FW and DNC tests.  $LR$  ranges from 1.02 to 1.10. The largest  $LR$  (1.10) occurs at the first trial, yielding a constant rate along the last four trials. The following must be pointed out: 1) The user exerts more control of the mobile than the robot in the five trials of the experiment, and 2) the learning process does not exist in this trial, since  $LR$  yields an approximately constant value. Due to the previous learning process while performing the most difficult tests, the user has learned about the peculiarities of the robot and the shared control. Thus, in this test, the user has a higher percentage of control in the system than the robot.

Finally, we did not observe major differences from an efficiency point of view between left- and right-handed people who participated in the experiments, except in the FW tests, which seemed to be easier for right-handed people. Fig. 13 presents the trajectory followed in the FW test for a left-handed [Fig. 13(a)] and a right-handed [Fig. 13(b)] user. Since the joystick is adapted for right-handed people, Fig. 13(a) shows the problems that left-handed people are faced up to the FW test. First, they have difficulties maintaining the robot parallel

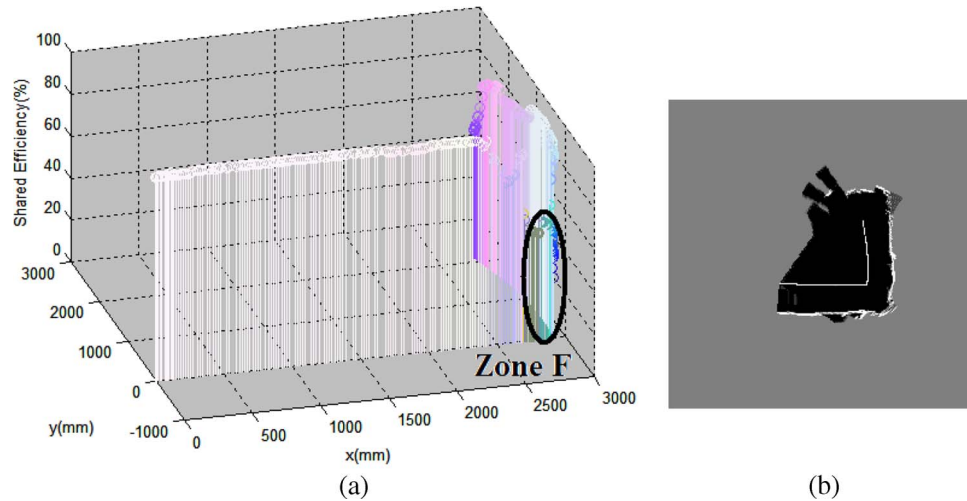


Fig. 10. User notes the control of the machine. (a) Three-dimensional representation of shared efficiency in FW test. (b) Trajectory.

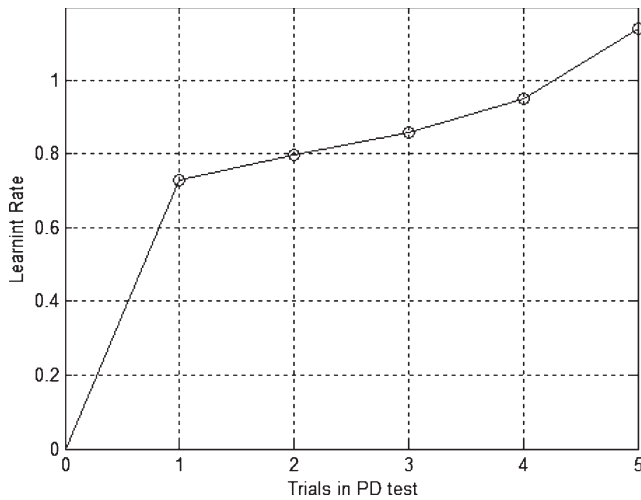


Fig. 11. Learning rate in PD test.

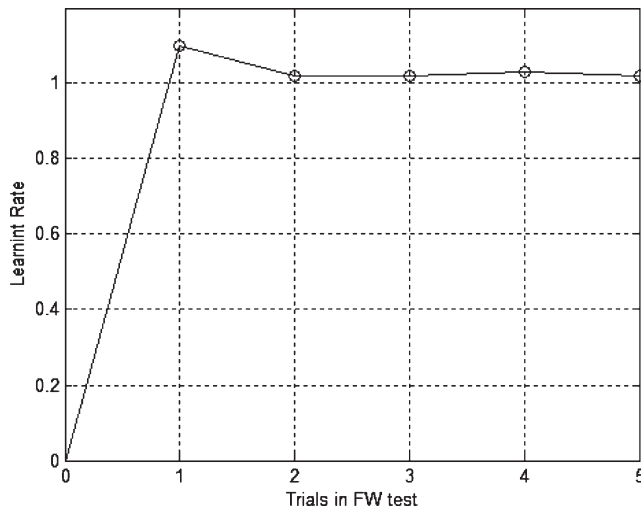


Fig. 12. Learning rate in NC test.

to the right wall, whereas right-handed people do not face this problem. Second, the left turn in the corner is not performed as correctly for left-handed people than it is realized for right-

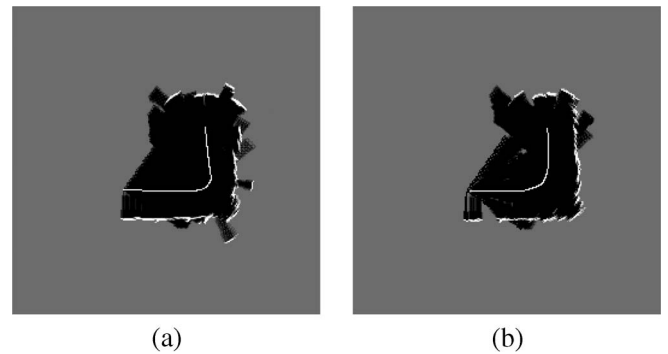


Fig. 13. Trajectory followed by the mobile robot in the FW test. (a) Left-handed user. (b) Right-handed user.

handed people. Finally, as shown in the trajectory of Fig. 13(a), left-handed people do not move the robot at a predefined distance, while right-handed people do, even after turning the corner [Fig. 13(b)]. From a subjective point of view, left-handed people seemed to be forced through the FW test. However, all right-handed people seemed to be comfortable while doing this test. Furthermore, left-handed people qualitatively stated their problems in these tests; however, none of the right-handed people expressed any difficulty.

Although a combination of both human and robot control commands is a positive feature in our shared control system, there are situations where this approach could not be valid because of an impending danger. Such a situation appeared when a user moved close to the wall while turning in the FW test. However, the system can deal with this problem. The user qualitatively said that the robot did not pay attention to his/her joystick orders. This was true since he/she was not performing correctly the task in this test, highly decreasing his/her efficiency at this point of the trajectory. Thus, shared control gives more control to the robot commands than human ones. Since robot commands provides a safe behavior due to  $f_{WFavoid}$  force, the impending danger situation is overcome.

During the experiments, some failures were observed. When the robot performed the four trajectories alone, it did not become stuck, even in the PD test where it oscillated in

front of the door. When experiments with users were performed, some failures occurred in the shared control mode. All of these situations appeared in the PD test. There were seven collisions, all of them performed by users unfamiliar with robots. If we remember that tests were performed by thirteen users and that each test was carried out five times, seven failures mean just 10% in the PD test and 2.7% for the whole four tests. It must be pointed out that failures were made by six users. This implies that all of them made one failure and one user made two failures. It must also be noticed that all of the one-failure users failed in the first trial of the test, while the two-failure user consecutively failed in the first two trials. Failures are due to their lack of experience in robotics. However, those users learned how to perform the task in the PD test after their failures. Furthermore, if the robot tends to collide because of the user's driving, in general, trajectories are corrected by the system, allowing the user to have more time to react to that situation.

## V. CONCLUSION

This paper has presented a new approach to shared control for autonomous robots and humans. The key idea of this paper is to measure the efficiency of both at each time instant from a reactive strategy point of view. Using the two measures, the orders of human and robot can be weighted and linearly combined into a single motory command. Continuous combination of these commands returns an emergent behavior that does not match exactly the robot or human standalone performance but tends to combine their advantages in a seamless way. We have tested the system with 13 volunteers presenting different features and a Pioneer AT robot in four different scenarios. Initially, they were not informed about the robot assisting their driving. However, they eventually realized they shared control with the machine but most were not able to say exactly when they had more help. We have developed a representation strategy to visually evaluate the efficiency of the trajectory at each point in terms of the different evaluated factors. In almost every case, the combined performance of person and machine improved the driving of the human, even when he/she did not realize it. In many cases, also, it improved the performance of the robot, particularly in situations where pure reactive control has been reported to fail, such as doors or close obstacles. Future work will focus on extending this paper to mobility assistance for power wheelchairs.

## REFERENCES

- [1] D. J. Brummer, D. A. Few, R. L. Boring, J. L. Marble, M. C. Walton, and C. W. Nielsen, "Shared understanding for collaborative control," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 25, no. 4, pp. 494–504, Jul. 2005.
- [2] J. Kofman, X. Wu, T. J. Luu, and S. Verma, "Teleoperation of a robot manipulator using a vision-based human–robot interface," *IEEE Trans. Ind. Electron.*, vol. 52, no. 5, pp. 1206–1219, Oct. 2005.
- [3] Y. Horiguchi and T. Sawaragi, "Effects of probing to adapt machine autonomy in shared control systems," in *Proc. Int. Conf. Syst., Man, Cybern.*, HI, Oct. 2005, vol. 1, pp. 317–323.
- [4] P. Aigner and B. J. McCarragher, "Modeling and constraining human interactions in shared control utilizing a discrete event framework," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 30, no. 3, pp. 369–379, May 2000.
- [5] A. Morris, R. Donamukkala, A. Kapuria, A. Steinfeld, J.T. Matthews, J. Dunbar-Jacob, and S. Thrun, "A robotic walker that provides guidance," in *Proc. IEEE Int. Conf. Robot. Autom.*, Taiwan, Sep. 2003, pp. 25–30.
- [6] D. Kortenkamp, R. P. Bonasso, D. Ryan, and D. Schreckenghost, "Traded control with autonomous robots as mixed initiative interaction," in *Proc. AAAI Spring Symp. Mixed Initiative Interaction*, Stanford, CA, 1997.
- [7] S. P. Parikh, V. Grassi, V. Kumar, and J. Okamoto, "Usability study of a control framework for an intelligent wheelchair," in *Proc. IEEE Int. Conf. Robot. Autom.*, Barcelona, Spain, Apr. 2005, pp. 4745–4750.
- [8] S. Parikh, V. Grassi, V. Kumar, and J. Okamoto, "Incorporating user inputs in motion planning for a smart wheelchair," in *Proc. IEEE Int. Conf. Robot. Autom.*, New Orleans, LA, Apr. 2004, pp. 2043–2048.
- [9] S. McLachlan, J. Arblaster, D. K. Liu, J. Valls, and L. Chenoweth, "A multi-stage shared control method for an intelligent mobility assistant," in *Proc. IEEE 9th Int. Conf. Rehabil. Robot.*, Chicago, IL, Jul. 2005, pp. 426–429.
- [10] C. Urdiales, A. Bandera, E. J. Pérez, A. Poncela, and F. Sandoval, *Hierarchical Planning in a Mobile Robot for Map Learning and Navigation*. New York: Physica-Verlag, 2003, pp. 2165–2188.
- [11] J. S. Albus, "Outline for a theory of intelligence," *IEEE Trans. Syst., Man, Cybern.*, vol. 21, no. 3, pp. 473–509, May/Jun. 1991.
- [12] H. Hu and M. Brady, "A parallel processing architecture for sensor-based control of intelligent mobile robots," *Robot. Auton. Syst.*, vol. 17, no. 4, pp. 235–257, Jun. 1996.
- [13] J. Borenstein, B. Everett, and L. Feng, *Navigating Mobile Robots: Sensors and Techniques*. Wellesley, MA: A.K. Peters, Ltd., 1996.
- [14] U. Frese, P. Larsson, and T. Duckett, "A multigrid algorithm for simultaneous localization and mapping," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 1–12, Apr. 2005.
- [15] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robot. Autom.*, vol. RA-2, no. 1, pp. 14–23, Mar. 1986.
- [16] E. Coste-Manière and R. Simmons, "Architecture: The backbone of robotic systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, San Francisco, CA, Apr. 2000, pp. 67–72.
- [17] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, "An architecture for autonomy," *Int. J. Robot. Res.*, vol. 17, no. 4, pp. 315–337, 1998.
- [18] R. G. Simmons, "Structured control for autonomous robots," *IEEE Trans. Robot. Autom.*, vol. 10, no. 1, pp. 34–43, Feb. 1994.
- [19] R. P. Bonasso, "Integrating reaction plans and layered competences through synchronous control," in *Proc. Int. Joint Conf. Artif. Intell.*, Sydney, Australia, Aug. 1991, pp. 1225–1233.
- [20] A. Poncela, E. J. Pérez, A. Bandera, C. Urdiales, and F. Sandoval, "Efficient integration of metric and topological maps for directed exploration of unknown environments," *Robot. Auton. Syst.*, vol. 41, no. 1, pp. 21–39, Oct. 2002.
- [21] B. Kuipers, "The spatial semantic hierarchy," *Artif. Intell.*, vol. 119, no. 1/2, pp. 191–233, May 2000.
- [22] D. Jung and A. Zelinsky, "Grounded symbolic communication between heterogeneous cooperating robots," *Auton. Robots*, vol. 8, no. 3, pp. 269–292, Jun. 2000.
- [23] J. H. Connell and P. Viola, "Cooperative control of a semi-autonomous mobile robot," in *Proc. IEEE Conf. Robot. Autom.*, Cincinnati, OH, 1990, pp. 1118–1121.
- [24] R. Simpson and S. P. Levine, *NavChair: An Assistive Wheelchair Navigation System With Automatic Adaptation*. Berlin, Germany: Springer-Verlag, 1998, pp. 235–255.
- [25] D. Miller, *Assistive Robotics: An Overview*. Berlin, Germany: Springer-Verlag, 1998, pp. 126–136.
- [26] R. S. Rao, K. Conn, S. H. Jung, J. Katupitiya, T. Kientz, V. Kumar, J. Ostrowski, S. Patel, and C. J. Taylor, "Human robot interaction: Applications to smart wheelchairs," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, 2002, pp. 3583–3588.
- [27] R. Simpson and S. Levine, "Development and evaluation of voice control for a smart wheelchair," in *Proc. Annu. RESNA Conf.*, Washington, DC, 1997, pp. 417–419.
- [28] J. Crisman and M. Cleary, *Progress on the Deictic Controlled Wheelchair*. Berlin, Germany: Springer-Verlag, 1998, pp. 137–149.
- [29] P. Nisbet, J. Craig, P. Odor, and S. Aitken, "'Smart' wheelchairs for mobility training," *Technol. Disability*, vol. 5, pp. 49–62, 1995.
- [30] G. Bourhis and Y. Agostini, "The VAHM robotized wheelchair: System architecture and human–machine interaction," *J. Intell. Robot. Syst.*, vol. 22, no. 1, pp. 39–50, May 1998.
- [31] N. L. Katevas, N. M. Sgouros, S. G. Tzafestas, G. Papakonstantinou, P. Beattie, J. M. Bishop, P. Tsanakas, and D. Koutsouris, "The autonomous mobile robot SENARIO: A sensor-aided intelligent navigation system for



powered wheelchairs," *IEEE Robot. Autom. Mag.*, vol. 4, no. 4, pp. 60–70, Dec. 1997.

- [32] H. A. Yanco, "Wheesley: A robotic wheelchair system: Indoor navigation and user interface," in *Assistive Technology and Artificial Intelligence: (Applications in Robotics, User Interfaces and Natural Language Processing)*. Berlin, Germany: Springer-Verlag, 1998, pp. 256–268.
- [33] B. Fernández, A. Poncela, C. Urdiales, and F. Sandoval, "Collaborative emergent navigation based on biometric weighted shared control," in *Proc. IWANN*, San Sebastián, Spain, Jun. 2007, pp. 814–821.
- [34] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Mag.*, vol. 9, no. 2, pp. 61–74, Jul./Aug. 1998.
- [35] T. Fong, C. Thorpe, and C. Baur, "Robot, asker of questions," *Robot. Auton. Syst.*, vol. 42, no. 3/4, pp. 235–243, Mar. 2003.
- [36] M. Mucientes, D. Moreno, A. Bugarín, and S. Barro, "Evolutionary learning of a fuzzy controller for wall-following behavior in mobile robotics," *Soft Comput.*, vol. 10, no. 10, pp. 881–889, May 2006.
- [37] N. Rahman and A. R. Jafri, "Two layered behaviour based navigation of a mobile robot in an unstructured environment using fuzzy logic," in *Proc. IEEE Int. Conf. Emerging Technol.*, Islamabad, Pakistan, Sep. 2005, pp. 230–235.
- [38] H. Zhang, S. Liu, and S. X. Yang, "A hybrid robot navigation approach based on partial planning and emotion-based behavior coordination," in *Proc. IEEE/RSJ Int. Conf. IROS*, Peking, China, Oct. 2006, pp. 1183–1188.
- [39] A. Poncela, C. Urdiales, C. Trazegnies, and F. Sandoval, "A new sonar landmark for place recognition," in *Proc. 6th Int. Conf. FLINS*, Blankenberge, Belgium, Sep. 2004, pp. 456–462.



**Alberto Poncela** was born in Spain, in 1975. He received the M.Sc. degree in telecommunication engineering from the Universidad de Málaga (UMA), Málaga, Spain, in 1999.

During 2000, he worked in a research project under a grant by the Spanish Comisión Interministerial de Ciencia y Tecnología. Since 2000, he has been an Assistant Professor with the Departamento de Tecnología Electrónica, Escuela Técnica Superior de Ingenieros de Telecomunicación, UMA. His research is focused on robotics and artificial vision.



**Cristina Urdiales** received the M.Sc. degree in telecommunication engineering from the Universidad Politécnica de Madrid, Madrid, Spain, and the Ph.D. degree from the Universidad de Málaga (UMA), Málaga, Spain.

She is currently a Lecturer with the Departamento de Tecnología Electrónica, Escuela Técnica Superior de Ingenieros de Telecomunicación, UMA. Her research is focused on robotics and computer vision.



**Eduardo J. Pérez** was born in Barcelona, Spain, in 1974. He received the M.Sc. degree in telecommunication engineering and the Ph.D. degree from the Universidad de Málaga (UMA), Málaga, Spain, in 1999 and 2006, respectively.

During 1999, he worked in a research project under a grant by the Spanish CYCIT. Since 2000, he has been a Lecturer with the Departamento de Tecnología Electrónica, Escuela Técnica Superior de Ingenieros de Telecomunicación, UMA. His research is focused on robotics, artificial intelligence, and

computer vision.



**Francisco Sandoval** (M'91) was born in Spain in 1947. He received the M.Sc. degree in telecommunication engineering and the Ph.D. degree from the Technical University of Madrid, Madrid, Spain, in 1972 and 1980, respectively.

From 1972 to 1989, he was engaged in teaching and research in the fields of optoelectronics and integrated circuits with the Universidad Politécnica de Madrid, as an Assistant Professor and a Lecturer successively. Since 1990, he has been with the University of Málaga, Málaga, Spain, as Full Professor

with the Departamento de Tecnología Electrónica, Escuela Técnica Superior de Ingenieros de Telecomunicación, starting his research on artificial neural network (ANN). He is currently involved in autonomous systems and foveal vision and in the application of ANN to energy management systems and broadband and multimedia communications.