

Learning-Based Adaptation for Personalized Mobility Assistance

Cristina Urdiales, Jose Manuel Peula, Manuel Fernández-Carmona,
and Francisco Sandoval

ISIS group, ETSI Telecommunications, University of Malaga, 29071 Malaga, Spain
acurdiales@uma.es

Abstract. Mobility assistance is of key importance for people with disabilities to remain autonomous in their preferred environments. In severe cases, assistance can be provided by robotized wheelchairs that can perform complex maneuvers and/or correct the user's commands. User's acceptance is of key importance, as some users do not like their commands to be modified. This work presents a solution to improve acceptance. It consists of making the robot learn how the user drives so corrections will not be so noticeable to the user. Case Based Reasoning (CBR) is used to acquire a user's driving model reactive level. Experiments with volunteers at Fondazione Santa Lucia (FSL) have proven that, indeed, this customized approach at assistance increases acceptance by the user.

1 Introduction

Mobility is of key importance for persons to carry their Activities of Daily Living (ADL). People affected by a disability may require mobility assistance to remain autonomous. Lack of human resources has led to research in mobility assistive devices, like a robotic power wheelchair. These wheelchairs are not a traditional robot in the sense that it is controlled, at least partially, by its user. This approach is known as shared control. Furthermore, doctors and caregivers have reported that excessive assistance may lead to loss of residual skills, whereas an active profile is reported to improve rehabilitation. Hence, in these cases it is desirable to give the user as much control as possible.

There are different approaches to shared control. In *safeguarded navigation*, for example, robots are always under human control, except when a potentially dangerous situation is detected. In these cases, the robot takes over [1][2] and a reactive algorithm is used to avoid such a danger. Other shared control approaches[5][6] rely on a basic set of primitives like *AvoidObstacle*, *FollowWall* and *PassDoorway* to assist the user in difficult maneuvers, either by manual selection or automatic triggering. In extreme, the user just points a destination and the robot does the rest [3]. In order to avoid sharp control switches from human to robot and to prevent loss of residual skills by not letting the user participate at all in complex tasks a third approach to shared control is collaborative control [7][8], where user and robot commands are mixed in a continuous way so that people may contribute their best to any situation.

The authors proposed a collaborative control method in [7] consisting on weighting robot's and user's commands by their respective driving efficiencies at each situation and adding them at reactive level. Thus, the most efficient agent was awarded with more control, yet the least efficient one always contributed to emergent motion. This approach was tested at Fondazione Santa Lucia (FSL), Rome, by a large number of volunteers presenting different degrees of functional disability. Surprisingly, we found out that people with a better functional disability profile actually performed worse than people with worse diagnosis in a significant number of cases. Further analysis proved that this group actually rejected assistance, i.e. tried to counteract robot commands, as soon as they acknowledged that the wheelchair was not doing exactly what they commanded.

To solve this problem, we propose to use a CBR based approach to let the robot learn how the user drives. Rather than choosing the most efficient command, the robot will try to provide the most similar one to what the person would do at each situation within established safety constraints. Since the robot's commands become much more similar to the user's, acceptance is improved and global efficiency grows, as proven by further experiments with volunteers at FSL. To achieve this, the wheelchair learns how a given user drives using CBR.

2 Collaborative Navigation System

Our basic approach to collaborative navigation – fully explained in [7] – is based on reactive navigation. Reactive schemes implicitly deal with several sensors and goals at a time, so we can simply handle user and robot commands as two different goals. Let \vec{v}_U and \vec{v}_R be the user and robot command vectors respectively. \vec{v}_U is extracted from a joystick and \vec{v}_R is calculated via the simplest pure Potential Fields Approach (PFA) [9], where goals and obstacles are modelled as attractors and repulsors, respectively. We can combine \vec{v}_U and \vec{v}_R linearly into a collaborative command \vec{v}_C . However, we do not want user and robot to have the same weight in the emerging decision. Instead, we want assistance to adapt to the user's needs. Hence, we weight \vec{v}_U and \vec{v}_R by the local efficiency of user and robot, respectively: the more efficient a user is at solving a given situation, the less assistance he/she receives.

$$\vec{v}_C = \eta_R \cdot \vec{v}_R + \eta_U \cdot \vec{v}_U \quad (1)$$

Motion efficiency η needs to be calculated locally, because in a purely reactive (i.e. memoryless) approach, global factors like trajectory length or completion time cannot be used. We have identified three local factors, ranging from 0 to 1, with an impact on η : *smoothness* (η_{sm} in Eq. 2), *directness* (η_{dir} in Eq. 3) and *safety* (η_{sf} in Eq. 4), corresponding respectively to how smooth the wheelchair is driven, how efficient it is to reach a target and how close it moves to obstacles. Global efficiency η is the average of these three efficiencies, that roughly correspond to the properties of a navigation function:

$$\eta_{sm} = e^{-C_{sm} \cdot |\alpha_{sm}|} \quad (2)$$

$$\eta_{dir} = e^{-C_{dir} \cdot |\alpha_{dir}|} \quad (3)$$

$$\eta_{sf} = 1 - e^{-C_{sf} \cdot |\alpha_{sf}|} \quad (4)$$

α_{sm} being the angle between the heading direction and command direction; α_{dir} being the angle between goal direction and command direction; α_{sf} being the angle between obstacle direction and command direction; and C_{sm} , C_{dir} and C_{sf} constants to decide how much impact have each angle on its respective local efficiency. As most of situations require uniform efficiency factor changes, these constants are set to 1 by default.

The main advantages of our approach [7] are that: i) it tends to preserve curvature and safety, as most PFA-based algorithms do; ii) users contribute to control chair all the time because η_U is never equal to 0; and iii) humans provide (when possible) a deliberation level to the system to avoid local traps.

The main problem with the proposed approach was that $\bar{\eta}_C$ (collaborative command efficiency) turned out to be lower for people with better functional profiles than for people with severe disabilities in a significant number of cases [7]. We observed that a number of these users seemed to be fighting robot control, so we developed a new metric that we called Disagreement. Disagreement is equal to the angle between the user's command \vec{v}_U and the emergent one \vec{v}_C and it represents how similar emergent motion is to what the user expects. Disagreement in our tests for people with good functional profile and low driving efficiency was very high: around 40-45%. Even though there is always a baseline Disagreement when a person drives a vehicle depending on its dynamics and kinematics -we measured it to be around 15% in our wheelchair-, it becomes obvious that it is not comfortable for a person to drive like this.

To solve this problem, we propose to replace commands provided by PFA by commands learnt from previous experience on how the user drives, as described in next section. Thus, the differences between commands proposed by the user and by the robot should be quite lower and acceptance improves.

3 Robot Adaptation to User

3.1 CBR-Based Collaborative Control

CBR has been used in navigation before, typically for global path planning in static environments [10][11] rather than for reactive navigation. There are also approaches for global planning in dynamic environments [12]. However, in [12] new opportunities cannot be discovered when the environment changes unless the topological map, which is based on, is regularly reorganized. Kruusmaa [13] proposed a grid-based CBR global path planning method to overcome the aforementioned problem. However, she concluded that CBR-based global navigation is beneficial only when obstacles are large and dense and only a few solutions exist. Otherwise, the solution space may become too large. Some CBR-based methods

focus on reactive navigation [14][15], but they all rely on accumulating experience over a time window while navigating in a given environment to obtain an emergent global goal seeking-behavior. Hence, they are environment-dependent. The authors already proposed a purely reactive navigation layer based on CBR in [7] for autonomous robots. Its original purpose was to create ad-hoc reactive navigation strategies via supervised learning and adapt them to different robot structures via learning by experience. Hence, we could avoid kinematics and dynamics calculations. In the present work, a similar strategy is used to make the robot learn how the user drives to improve acceptance.

Learning a reactive navigation behavior basically consists of associating whatever the driver is doing to the situation at hand. The user is already taking into account the vehicle kinematics and dynamics, as well as the relative position of obstacles and goal and any local consideration, like floor condition, mechanics, etc, that he/she intuitively adapts to through practice. This knowledge is implicitly added to the case base. After a while, all this information is encoded into a set of cases, that can be evaluated with our local metrics: η_{sm} , η_{dir} and η_{sf} . Eventually, the case base stabilizes to the best average solutions the user gives to any input situation. This happens after acquired cases are clustered to obtain valid prototypes, so that duplicates and least efficient cases are removed from the case base as explained below. This case base is not environment-dependent because there are not so many situations one can face from a local point of view [16]: the only relevant information at reactive level is how far we are from close obstacles, where we are heading and where we would like to go.

Obviously, if our system cloned exactly what the user does, it would provide no assistance except to correct punctual errors. However, if we combine a CBR reactive navigation module with the proposed collaborative control approach, advantages are more obvious: we use learnt cases when possible and receive assistance when needed. Each time the robot retrieves a case from the case base, its efficiency (η_{CBR}) is checked. If η_{CBR} is over a given threshold U_η (in our case, 0.7), the retrieved case solution becomes \vec{v}_R . Otherwise, \vec{v}_R is obtained from PFA, even if η_{CBR} is bigger than η_{PFA} . Then \vec{v}_R and \vec{v}_U are weighted by their respective efficiencies as usual to calculate \vec{v}_C , which is stored in the case base for future reference.

3.2 CBR Implementation

The number of different situations that a robot can locally find is not too large [16], so the number of cases to acquire a given motion strategy is not large either (150-200 cases in our previous tests [7]). Hence, we can use a flat structures and the usual feature-value vector representation. Our cases are compared using a nearest-neighbour (NN) algorithm. After several tests, we chose to work with the Tanimoto distance T_s . For cases C_1 and C_2 :

$$T_s(C_1, C_2) = \frac{C_1 \cdot C_2}{|C_1|^2 + |C_2|^2 - C_1 \cdot C_2} \quad (5)$$

We found that similar environment arrangements result in a lower Tanimoto distance, whereas other metrics, e.g. Euclidean distance, reward higher partial similarities in nearby obstacles than the environment as a whole [7]. Our input instance includes all commented reactive factors: goal position, laser readings (nearby environment) and wheelchair heading. Our output includes the user motion command (\vec{v}_{CBR}) and our evaluation measure is its global efficiency (η_{CBR}). Fig. 1 shows our case structure. Since minor differences between sensors readings usually correspond to slight robot shiftings rather than to different situations, we have discretized the space sampled by the robot into 8 equal sized arc regions and also discretize laser readings into 5 non equal intervals that have proven to be valid for a typical indoor environment [7]: i) critical (0-20 cm); ii) near (20-50 cm); iii) medium (50-100 cm); iv) far (100-150 cm); and v) no influence (more than 150 cm).

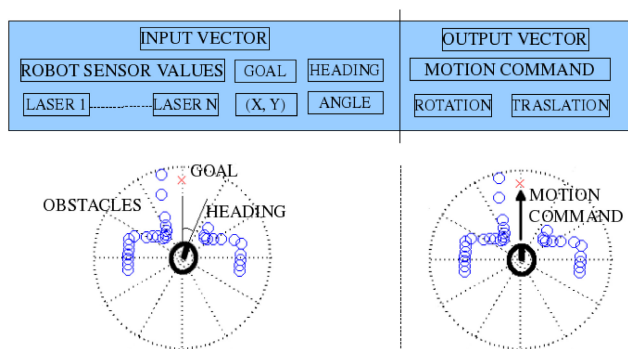


Fig. 1. CBR case structure and example

Our system uses both *learning by observation* and *learning by own experience*. If there is any recorded trace of a given user driving the wheelchair without assistance, it is used to initialize the case base. As soon as the user starts driving, his/her commands are combined with the robot's and learnt by the system. If the user is unable to provide an efficient way of solving a problem, the solution will be mostly based on PFA. However, the robot's commands tend to be more and more similar to the user's as a whole. It needs to be noted that we include no case adaptation stage in our CBR cycle to preserve user's commands as much as possible. Instead, adaptation is implicitly provided by collaborative control.

Finally, a MaxMin clustering algorithm is applied to the case base on a regular basis to group similar cases into a single cluster prototype (CP). The resulting CP is equal to the average of all cases in the cluster weighted by their own efficiencies. Thus, a given CP is not the most frequent response to a situation, but the most efficient response the user may produce on a regular basis. Also, low efficiency cases weight too little in CP calculation, so in practice they are removed unless there is no learnt alternative to cope with the related situations.

Our final case base is composed only of CPs and we only compare new instances with those Cps, but we preserve all learnt cases for future rearranging. This whole process is performed offline each time the number of newly acquired cases exceeds a threshold. Typically, it needs to be performed at least twice, once for the learning by observation stage and another for the first run while learning by own experience. Later on, the number of acquired cases decreases significantly and it is no longer necessary to cluster cases after each experiment. This process has, mainly, four targets: i) to remove duplicated cases, ii) to bind the number of cases in the database, iii) to avoid oscillations between similar cases, and iv) to clean spurious solutions from the database.

4 Experiments and Results

The proposed CBR system was built on CARMEN (Collaborative Autonomous Robot for Mobility ENhancement), a modified a Runner Meyra wheelchair (Fig. 2.a), donated by Sauer Medica S.L. and equipped with an industrial PC running Linux OS and a frontal Hokuyo laser URG04-RX for localization and obstacle detection. Moving backwards was not allowed due to lack of rear sensors and mirrors in the wheelchair. All experiments took place -after approval by FSL Ethical Committee- in Casa Agevole, a 60 m² fully furnished, standard-compliant test house built in the FSL complex in Rome¹. All tests were performed by 18 volunteering inpatients presenting different degrees of disability (Left or Right Hemiparesis, Ischemic Stroke, Spinal cord injury, Cerebral Hemorrhage). Their cognitive and physical skills ranged from good to low, according to the minimal state examination (MMSE)[17] (1-30), the Barthel Index [18] (0-100) and the Instrumental ADL (IADL) [19] (0-5/8). In our volunteers, these scales were MMSE: 3-29, Barthel: 8-100, IADL: 0-8. Volunteers were divided into 3 groups depending on their diagnosed profile: 1) minor; 2) mild; and 3) severe physical/cognitive disabilities. Fig. 2.b shows the approximate path that volunteers were asked to perform. It can be observed that it involves door crossing, narrow areas and significant turns. This path was suggested by our medical staff because it includes most situations faced in ADL.

Each volunteer performed at least three runs (autonomous mode, shared control using PFA, and shared control using CBR). First of all, they drove the wheelchair without assistance (for benchmarking). In this mode, only a safeguard layer is active to prevent collisions. Many volunteers did not manage to complete the path in this mode. Our second mandatory run was PFA-based collaborative control navigation. Most users managed to do the run in assisted mode at first attempt, but one group of inpatients – group 3 – had notable exceptions. During this run, CBR was active and cases were acquired for the next run. Finally, all volunteers tried at least a CBR-based collaborative control runs, using their own case bases to assist navigation. During these runs, the case base kept acquiring new cases on a need basis.

¹ <http://www.progettarepertutti.org/progettazione/casa-agevole-fondazione/index.html>

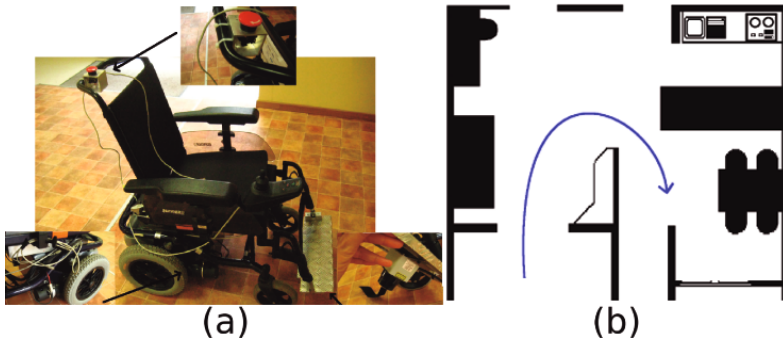


Fig. 2. a) Robot wheelchair (CARMEN) b) Proposed path at Casa Agevole

Table 1. Average results for all inpatients in collaborative control tests

| | PFA Tests | | CBR Tests | |
|------------------------|-----------|-------|-----------|-------|
| | Mean | Dev. | Mean | Dev. |
| Global efficiency (%) | 65.23 | 20.10 | 71.42 | 18.74 |
| Smoothness (%) | 64.18 | 27.80 | 69.05 | 25.54 |
| Directness (%) | 39.31 | 28.93 | 53.60 | 26.73 |
| Safety (%) | 92.25 | 17.60 | 91.73 | 16.06 |
| Intervention Level (%) | 79.17 | - | 75.59 | - |
| Disagreement (%) | 41.83 | 27.75 | 26.11 | 21.87 |
| Joystick variation (%) | 1.62 | 4.69 | 1.64 | 4.47 |
| Inconsistency (%) | 9.53 | 10.93 | 6.59 | 9.28 |
| Completion time (sec) | 48.45 | - | 43.46 | - |

Table 1 shows the average results of our experiments. Standalone results are omitted, as they are possibly biased by the users’ learning curve. Our task metrics include all efficiency factors η , η_{sm} , η_{sf} and η_{dir} and total time. Our psych metrics include: *Intervention Level*, defined as the portion of time that the user moves a joystick [20] and showing if the user presents an active profile; ii) *Disagreement*, as previously commented, related to effort and frustration; it needs to be noted that due to mechanical issues like inertia, response time, joystick sensitivity, etc., our lowest wheelchair disagreement seemed to settle around 20% in standalone mode for our wheelchair; iii) *Inconsistency*, defined as the variation of the user’s commands when facing similar situations; and iv) *Joystick variation*, which measures changes over 10% in the position of the stick and has been used as an indirect measure of workload [21][22].

It can be observed that efficiency in CBR-based collaborative mode is higher – specially in terms of directness – and its deviation is lower. The most important issue, however, is that *disagreement* decreases from 41.83% (PFA-based mode) to 26.11% (CBR-based mode), meaning that users are more comfortable with

the wheelchair. This was, in fact, the main target of the proposed approach. Consistency is also better.

Table 2 shows the global results of the experiments separated into our 3 groups. It can be observed that the CBR approach improves and homogenizes performance significantly. However, disagreement does not decrease equally for all groups. People in group 3 do not benefit from this method, probably because their consistency is low and learnt commands are not too efficient.

Table 2. Anova Test of global efficiency and disagreement for groups: 1) Good cognitive and physical 2) Good cognitive and low physical 3) low cognitive and physical

| | Group 1 | Group 2 | Group 3 |
|---------------------------|---------|---------|---------|
| PFA Global efficiency (%) | 66.77 | 70.75 | 68.86 |
| CBR Global efficiency (%) | 74.50 | 76.20 | 76.33 |
| ANOVA (pvalue) | 0.000 | 0.018 | 0.000 |
| PFA Disagreement (%) | 32.87 | 31.85 | 25.66 |
| CBR Disagreement (%) | 25.10 | 19.40 | 31.8 |
| ANOVA (pvalue) | 0.000 | 0.004 | 0.000 |
| Patients in group | 3 | 3 | 12 |

4.1 A Case in Detail

The problem our CBR approach meant to solve is clearly represented by volunteer 1, a 56 years old female affected by multiple sclerosis with good physical and cognitive skills (MMSE=26, Barthel=100, IADL=8). This person could move with the help of a walker, but had no previous experience with power wheelchairs. Table 3 briefs her performance in her five tries: standalone, PFA-based (x3) and CBR-based collaborative control. This person repeated the PFA test three times and her third try was the worst of all. In fact, she failed to finish her last two PFA-based paths and reported that "the wheelchair was not working" while trying to move it into a wall.

In her first standalone run, only the safeguard layer was active – to prevent imminent collisions –, hence the minor differences between human and collaborative performance in this mode (table 3). Her standalone global efficiency was equal to 67.56%, and her worst feature was directness, probably due to lack of practice with power wheelchairs. Intervention Level (table 3) was very high because the wheelchair did not move unless there was some human input (99.73%). Besides, her Joystick variation was very low (0.09 %). Her standalone trajectory was quite smooth and efficient, even at door crossing, except at the second turn, when she got too close to the walls and had to steer right sharply. In brief, her standalone run was quite good and she reached her goal in just 30.67 seconds. In order to observe what this volunteer actually lacked, we clustered her commands according to the relative position of wheelchair, goal and obstacles and realized that she had trouble adjusting turns.

Table 3. Results for inpatient 1 trials in all modes. In the CBR column, data without parenthesis represents the PFA value and data with them the CBR value.

| Control type | | User | PFA | PFA | PFA | CBR |
|---------------------|--------|-------|-------|-------|-------|--------------|
| η_{global} (%) | Robot | — | 63.12 | 65.73 | 58.97 | 68.61(69.23) |
| | User | 67.51 | 65.17 | 60.7 | 57.05 | 70.6 |
| | Shared | 67.56 | 68.4 | 67.88 | 64.0 | 74.5 |
| η_{sm} (%) | Robot | — | 56.45 | 57.37 | 36.81 | 52.2(78.73) |
| | User | 66.4 | 65.91 | 56.4 | 61.26 | 66.62 |
| | Shared | 66.33 | 65.1 | 63.17 | 61.96 | 74.6 |
| η_{dir} (%) | Robot | — | 43.85 | 51.6 | 48.75 | 61.98(34.13) |
| | User | 44.28 | 37.46 | 38.42 | 22.0 | 48.61 |
| | Shared | 44.39 | 45.75 | 44.85 | 37.0 | 52.73 |
| η_{sf} (%) | Robot | — | 88.9 | 88.24 | 91.39 | 91.81(94.92) |
| | User | 91.95 | 92.34 | 87.41 | 87.94 | 96.55 |
| | Shared | 92.03 | 94.45 | 95.59 | 92.96 | 96.22 |
| Intervention Level | % | 99.73 | 70.73 | 77.85 | 97.95 | 82.83(82.56) |
| Disagreement | % | 25.18 | 21.37 | 29.87 | 47.31 | 20.89(25.11) |
| | dev | 25.18 | 14.96 | 18.02 | 27.7 | 18.72(26.63) |
| Joystick variation | % | 0.09 | 0.2 | 0.07 | 0.06 | 2.38 |
| | dev | 1.12 | 2.37 | 1.08 | 0.99 | 3.65 |
| Inconsistency | % | 6.73 | 7.7 | 8.58 | 4.91 | 6.69 |
| | dev | 6.73 | 7.7 | 8.58 | 4.91 | 6.69 |
| Total Length | m | 6.7 | 6.68 | 4.98 | 4.8 | 6.4 |
| Completion time | sec | 30.67 | 38.2 | 31.61 | 32.22 | 30.63 |

Her next 3 runs were performed with PFA-based collaborative control: not only did η_C not increase but even decreased in the last run whereas the volunteer tried to collide into a wall to check if the wheelchair obeyed her. Obviously, the wheelchair tried to correct her and they struggled for a couple of seconds. Her *directness* dropped to 22% and η_C decreased sharply. In this particular run, *disagreement* became as high as 47.31% – meaning that half the time, user and emergent commands were in conflict –.

Fig. 3 shows η_H , η_R and η_C for the third PFA-based run. We have chosen to represent η_{sm} , and η_{dir} and η_{sf} as the red, green and blue channels of the RGB colorspace, respectively, for visibility in our efficiency plot. Hence, pink efficiency means loss of directness (steering areas) and purple-blue means loss of directness and smoothness (sharp direction changes). Fig. 4.b shows how *disagreement* was highly correlated with η_R , i.e. it grew when the robot had more impact on emerging commands. Fig. 4.a shows how disagreement grew after crossing the door – when η_R began to increase and the user became aware of the robot intervention–. After that, the rest of the trajectory was a fight for control. Although the robot tried to compensate the user’s motion commands (cyan η_R area), emergent ones were affected and, as a result, she failed to reach the goal. The inpatient’s *inconsistency* in this run shows that she was driving far from her usual skills: from the 43 different commands clusters that she typically used

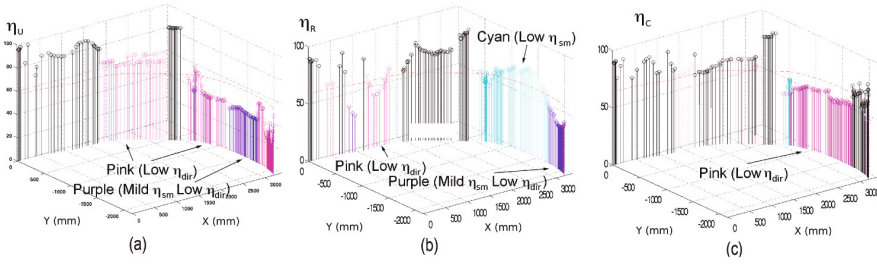


Fig. 3. Inpatient 1 results for PFA-based collaborative mode: a) human efficiency; b) robot efficiency; c) collaborative efficiency

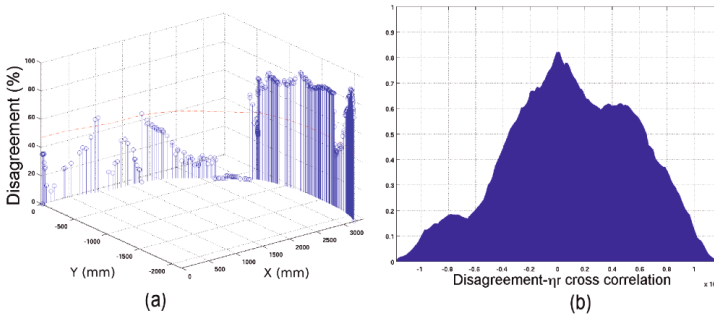


Fig. 4. Inpatient 1 results for PFA-based collaborative mode: a) *disagreement* for PFA-based collaborative control; b) cross correlation $\text{disagreement}/\eta_R$

to solve the proposed trajectory in standalone mode, 25 – corresponding mostly to steering decisions – had a very large standard deviation in this run. This basically means that her decisions were statistically erratic during the struggle.

Fig. 5 shows two clusters corresponding to locations involving steer correction and strong *disagreement*. The robot -in the center of the plot- is heading in the 0 degrees direction, obstacles are represented with circles and the goal is marker with an x. On the right of each plot, we can see the joystick shadow for human and robot. Cluster 16 corresponds to the beginning of the trajectory, when the first steering decision needed to be taken. As usual, inpatient 1 delayed her steering command, so the robot initiated it, and provoked a noticeable turn to the right to avoid getting too close to the wall. The emerging command, no longer equal to the user's, was practically a 90° right turn, only much slower than the user's command (shorter vector). At this point, the user became fully aware of the robot's influence and tried to fight it, but, eventually, the struggle became so intense that we got clusters like cluster 34, with a goal on the right side of the wheelchair, obstacles on the left and yet, the user pushing the joystick hard into the obstacles. Since her efficiency was very low at the point, the robot was dominant and the combination was a slow forward motion. This eventually got the wheelchair so close to an obstacle that further safe manoeuvre was not possible and inpatient 1 failed to finish the trajectory.

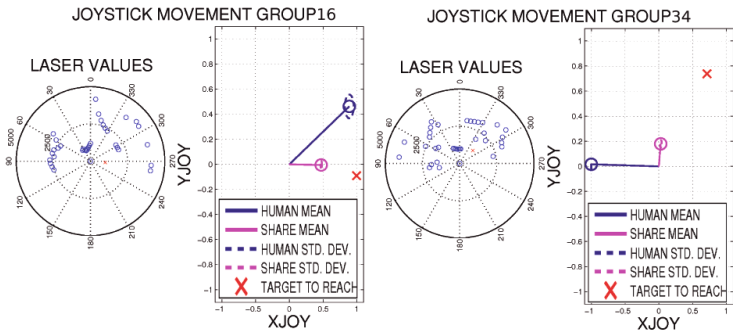


Fig. 5. Clusters representation of two situations with a strong *disagreement* between inpatient 1 and robot in PFA-based collaborative control

Fig. 6.a shows the path in CBR-based collaborative control. As commented, the case base was filled with user’s data coming from all the previous runs, but in this case mostly included data from the standalone and first PFA runs, because efficiency grew worse later. During execution, if a situation is not similar enough to the output case or η_{CBR} is too low, we use a PFA command instead in collaborative control at that specific location (areas marked in Fig. 6.b with a dot). We can observe that *disagreement* was specially high when PFA was used instead of CBR. Nevertheless, *disagreement* was no longer correlated with η_R because most learnt cases were efficient enough to be extensively used through the trajectory. In this case, *inconsistency* was similar in average to standalone mode and there were less, more homogeneous command groups than in PFA mode.

Results of this test are briefed in the last column of table 3. As commented, we calculate what PFA would do all the time, but PFA commands are not used unless η_{CBR} goes under a safety threshold or the retrieved case instance is too far from the current situation. Table 3 shows how η_C in the CBR-based mode case increased to 74.5%, higher than in any of the previous runs, including standalone mode, and higher than each of the components separately: PFA (68.61%),

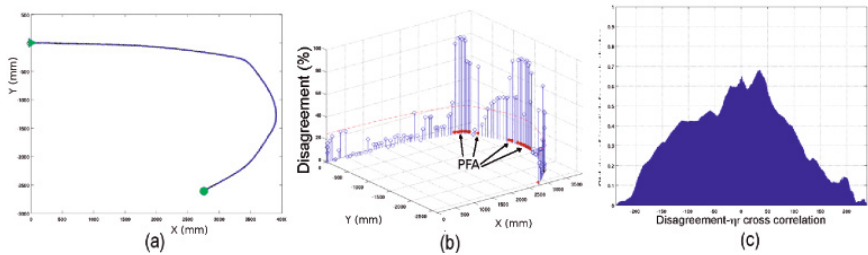


Fig. 6. Inpatient 1 results for CBR-based collaborative mode: a) path; b) disagreement (PFA and CBR); c) correlation between disagreement and system efficiency

CBR (69.23%) and user (70.6%). More specifically, *smoothness* was boosted up to 74.6%, even though CBR was in this case, by definition, worse than PFA (78.73% to 52.2%). This happened because: i) the case base was still not complete enough; and ii) PFA are designed to preserve smoothness, whereas CBR produces mildly sharp direction changes at case switching. Inpatient 1 had practically the same smoothness than in standalone mode, i.e. she was not fighting the machine anymore. *Directness*, though, was quite low for CBR with respect to PFA, but the user compensated this and, at locations where η_C was too low, PFA took control of the situation, so that combined *directness* raised to 52.73%. *Safety*, as commented, was preserved by all combinations of control in the experiment, but it was a bit higher here. We have experimentally checked that this happens when users are comfortable with control and drive smoothly, as, in these cases, they tend not to get too close to obstacles.

To illustrate how CBR decisions are closer to human commands, Fig. 7 shows two clusters corresponding to approximately the same situations regarding the relative position of obstacles and goal. Fig. 7.a corresponds to PFA-based collaborative mode, whereas Fig. 7.b is obtained in CBR-based collaborative mode. It can be observed that PFA-based collaborative mode corrected *directness* and reduced variability, but CBR-based collaborative mode reduced *disagreement* as well.

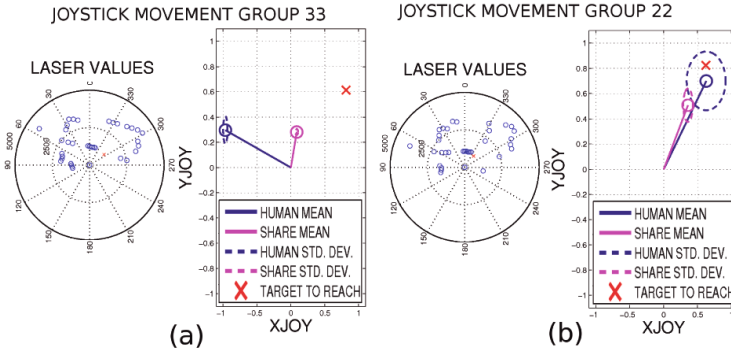


Fig. 7. Inpatient 1 cluster comparative for the same location for PFA-based (a) and CBR-based (b) collaborative mode

5 Conclusions and Future Work

This paper has presented a CBR-based collaborative control technique to reduce user's stress and assistance rejection by adapting help to the user via learning. Collaborative control is based on reactively combining the contribution of both human and robot, weighting them by their respective local efficiencies, to obtain an emergent collaborative navigation behavior. The robot learns how the user drives via CBR and contributes to control with more familiar commands. Thus, differences between user and emergent commands are less perceivable by the

user. However, if CBR commands are not efficient enough, and due to safety reasons, PFA is used instead of CBR.

The system was tested by 18 inpatients at FSL in a home-like environment. Average efficiency was higher in CBR-based collaborative mode than in PFA-based one or standalone mode. Besides, all users managed to finish a mildly complex trajectory in CBR-based collaborative mode. As expected, CBR-based navigation mimicked the user's way of driving and, in most cases – good or mild cognitive skills –, reduced disagreement between user and machine. Persons with very low cognitive skills did not provide enough efficient patterns to build a valid user model, so in those cases the system typically behaved like a PFA-based collaborative one.

The main drawback of the proposed system is that most users agreed that the wheelchair moved a bit brusque. This is provoked by case swapping, since the case base has a limited number of cases, and also by swaps from CBR to PFA-based collaborative control. This problem can be solved by adding some temporal inertia and future work will focus on this.

Future work will focus too on checking if the contents of a given user's case base can be correlated with the person's condition and, if so, on predicting the amount of help required by a specific user at a given situation instead of just providing it in a completely reactive fashion. This would allow us to include temporal inertia and reduce the commented problems related to case switching.

Acknowledgement. This work has been partially supported by the Spanish Ministerio de Educacion y Ciencia (MEC), Project TEC2011-29106-C02-01. The authors would like to thank Santa Lucia Hospedale and all volunteers for their kind cooperation and Sauer Medica for providing the power wheelchair.

References

1. Parikh, S.P., Grassi, V., Kumar, V., Okamoto, J.: Usability study of a control framework for an intelligent wheelchair. In: Proc. of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, pp. 4745–4750 (April 2005)
2. McLachlan, S., Arblaster, J., Liu, D.K., Valls, J., Chenoweth, L.: A multi-stage shared control method for an intelligent mobility assistant. In: Proc. of the 2005 IEEE 9th International Conference on Rehabilitation Robotics, Chicago, USA, pp. 426–429 (July 2005)
3. Frese, U., Larsson, P., Duckett, T.: A multigrid algorithm for simultaneous localization and mapping. *IEEE Transactions on Robotics* 21(2), 1–12 (2005)
4. Mandel, C., Huebner, K., Vierhuff, T.: Towards an autonomous wheelchair: Cognitive aspects in service robotics. In: Proceedings of Towards Autonomous Robotic Systems (TAROS 2005), pp. 165–172 (2005)
5. Bruemmer, D.J., Few, D.A., Boring, R.L., Marble, J.L., Walton, M.C., Nielsen, C.W.: Shared understanding for collaborative control. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans* 25(4), 494–504 (2005)
6. Horiguchi, Y., Sawaragi, T.: Effects of probing to adapt machine autonomy in shared control systems. In: Proc. International Conference on Systems, Man and Cybernetics, Hawaii, USA, vol. 1, pp. 317–323 (October 2005)

7. Urdiales, C., Fernandez-Carmona, M., Peula, J., Annicchiaricco, R., Sandoval, F., Caltagirone, C.: Efficiency based modulation for wheelchair driving collaborative control. In: Proc. of 2009 IEEE Conf. on Robotics for Rehabilitation (ICRA 2010), Anchorage, USA (2010)
8. Carlson, T., Demiris, Y.: Human-wheelchair collaboration through prediction of intention and adaptive assistance. In: IEEE International Conference on Robotics and Automation, ICRA 2008, pp. 3926–3931 (May 2008)
9. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research* 5(1), 90–98 (1986)
10. Branting, L.K., Aha, D.W.: Stratified case-based reasoning: reusing hierarchical problem solving episodes. In: IJCAI 1995: Proceedings of the 14th International Joint Conference on Artificial Intelligence, Montreal, pp. 384–390. Morgan Kaufmann, San Mateo (1995)
11. Fabrizi, E., Oriolo, G., Panzneri, S., Ulivi, G.: Mobile robot localization via fusion of ultrasonic and inertial sensor data. In: Proc. of the Sixth Midwest Artificial Intelligence and Cognitive Science Conference, Carbondale, USA, pp. 32–36 (1995)
12. Haigh, K.Z., Veloso, M.: Route planning by analogy. In: Aamodt, A., Veloso, M.M. (eds.) ICCBR 1995. LNCS, vol. 1010, pp. 169–180. Springer, Heidelberg (1995)
13. Kruusmaa, M.: Global navigation in dynamic environments using Case-Based Reasoning. *Autonomous Robots* 14, 71–91 (2003)
14. Likhachev, M., Arkin, R.C.: Spatio-temporal case-based reasoning for behavioral selection. In: Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 1627–1634 (2001)
15. Santamaria, J., Ram, A.: A multistrategy case-based and reinforcement learning approach to self-improving reactive control systems for autonomous robotic navigation. Tech. Rep. (1993)
16. Minguez, J., Osuna, J., Montanor, L.: A divide and conquer strategy based on situations to achieve reactive collision avoidance in troublesome scenarios. *IEEE Trans. on Robotics* (2009)
17. Crum, R., Anthony, J., Bassett, S., Folstein, M.: Population-based norms for the mini-mental state examination by age and educational level. *Journal of the American Medical Association* 269(18), 2386–2391 (1993)
18. Mahoney, F., Barthel, D.: Functional evaluation: the barthel index. *Maryland State Medical Journal* (14), 56–61 (1965)
19. Lawton, M., Brody, E.: Assessment of older people: self-maintaining and instrumental activities of daily living. *Gerontologist* (9), 179–185 (1969)
20. Cooperstock, J., Pineau, J., Precup, D., Atrash, A., Jaulmes, R., Kaplow, R., Lin, N., Prahacs, C., Villemure, J., Yamani, H.: Smartwheeler: A robotic wheelchair test-bed for investigating new models of human-robot interaction. In: Proc. of the IEEE Conference on Intell. Robots and Systems (IROS), San Diego, USA (2007)
21. Clarke, D., Yen, S., Kondraske, G.V., Khoury, G.J., Maxwelle, K.J.: Telerobotic network workstation for system performance and operator workload monitoring. NASA JSC, Houston, TX. Tech. Rep. 91-013R (1991)
22. Khoury, G.J., Kondraske, G.V.: Measurement and continuous monitoring of human workload associated with manual control devices. NASA JSC, Houston, TX. Tech. Rep. 91-011R (1991)