An Adaptive Scheme for Wheelchair Navigation Collaborative Control

C. Urdiales[1], J.M. Peula[1], C. Barrué[2], U. Cortés[2], F. Sandoval[1], C. Caltagirone[3] and R. Annicchiarico[3]

ISIS Group, ETSI Telecomunicacion, Univ. of Malaga, Campus de Teatinos s/n, 29010-Malaga (Spain)
KEMLg group, Univ. Politecnica de Catalunya, c/Jordi Girona, 1-3, 08034-Barcelona (Spain)
Fondazione Santa Lucia, IRCCS, Via Ardeatina 306, 00179-Rome (Italy)

Abstract

In this paper we propose a system where machine and human cooperate at every situation via a reactive emergent behavior, so that the person is always in charge of his/her own motion. Our approach relies on locally evaluating the performance of the human and the wheelchair for each given situation. Then, both their motion commands are weighted according to those efficiencies and combined in a reactive way. This approach benefits from the advantages of typical reactive behaviors to combine different sources of information in a simple, seamless way into an emergent trajectory

Introduction

Nowadays, population is steadily ageing. This process is provoking an increase in chronic diseases and citizens with special needs. Unfortunately, human resources are not enough to cover the needs of this progressively larger sector. Robotics have proven to be a valuable tool to aid people with disabilities. More specifically, in the field of mobility assistance, robotized wheelchairs and walkers have been frequently used. However, when a wheelchair is fully controlled by a navigation system, the recommended action might go against the user's wishes and cause him/her stress. Also, it has been stated by doctors and caregivers that excessive assistance may lead to loss of residual capabilities, as no effort is required on the user's part. Hence, many approaches to wheelchair navigation focus on cooperation between person and machine (collaborative control). There are many studies on the level of autonomy a robot might have when interacting with a human and viceversa (Bruemmer et al. 2005) (Kofman et al. 2005) (Horiguchi and Sawaragi 2005) (Aigner and McCarragher 2000). Depending on *how* much autonomy the machine has, collaborative approaches can be roughly categorized into i) safeguarded operation; and ii) shared control. In the first case mobiles can be totally controlled by humans, but in some cases the robot makes some decisions when human control is not adequate (Parikh et al. 2005) (McLachlan et al. 2005). In the second case, control may be handled from user to machine depending on the situation at hand. Some of these approaches (Connell

and Viola 1990) (Simpson and Levine 1998) (Miller 1998) (Rao et al. 2002) (Bruemmer et al. 2005)(Horiguchi and Sawaragi 2005) rely on using a basic set of primitives like $AvoidObstacle$, $FollowWall$ and $PassDoorway$ to assist the person in difficult maneuvers, either by manual selection or automatic triggering. In other cases, a subsumption like scheme (Brooks 1986) is used, where detected events trigger one or several behaviours which are merged into an emergent one that is finally executed. MAID (Mobile Aid for Elderly and Disabled people)(Prassler, Scholz, and Fiorini 1999), NavChair (Simpson and Levine 1998), TinMan (Miller 1998), Smartchair (Rao et al. 2002), Wheelesley (Yanco 1998), VAHM (Bourhis and Agostini 1998) follow this approach for assisted navigation. The main difference among them is how behaviours are implemented. In an 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 (Gomi and Griffith 1998)(Simpson and Levine 1997) (Crisman and Cleary 1998) (Nisbet et al. 1995) (Bourhis and Agostini 1998) (Frese, Larsson, and Duckett 2005). Most these systems work like a conventional autonomous robot in the following sense: the user simply provides a destination and the mobile is totally in charge of getting there via a hybrid navigation scheme.

Most commented approaches to shared control rely on swapping control from human to machine according to more or less complex algorithms. However, control swapping provokes curvature changes and discontinuity from the mobile point of view. Furthermore, this could be inconvenient for the persons, who would not know when they could lose control and, hence, not feel in charge or, if it is them who decide when to give up, not make efforts to overcome certain situations and, consequently, lose residual capabilities. Thus, we propose a system where machine and human cooperate at every situation via a reactive emergent behavior, so that the person is always in charge of his/her own motion. Our approach relies on locally evaluating the performance of the human and the wheelchair for each given situation. Then, both their motion commands are weighted according to those efficiencies and combined in a reactive way. It can be observed that, using this approach, the amount of control exerted by human and machine changes in time, but in a smooth way, so that no noticeable swap is perceived by the user. This approach benefits from the advantages of typical

reactive behaviors to combine different sources of information in a simple, seamless way into an emergent trajectory. This approach is commented into section 2, along with some detected drawbacks. Section 3 presents a new method to overcome these problems via learning and adaptation and section 4 presents some experiments and results. Finally, conclusions and future work are given in section 5.

Collaborative control for continuous cooperation

In a conventional power wheelchair, either a joystick or any other pointing device is used to provide the direction in which the user wants to move at each time instant. In our work, the wheelchair also provides its own motion command via a Potential Field Approach (PFA) (Khatib 1986), where obstacles are handled as repulsors and the goal is an atractor. In order to benefit from the PFA properties to easily combine different goals, the user's direction is added as another vector in the potential field at each position. However, it is necessary to determine how to weight the human and robot vectors when adding them. In order to keep navigation safe and smooth, weights are proportional to the efficiencies of human and robot at each time instant. Efficiencies must be locally calculated due to the purely reactive nature of the approach. Consequently, it is necessary to determine which factors can be used to measure efficiency in a punctual way.

Control algorithm

The reactive behavior provides the rotational (v_{rR}) and translational (v_{tR}) velocities the wheelchair supplies as its own motion velocities (v_{rH} and v_{tH}). Shared motion commands (rotational velocity, v_{rS}, and translational velocity, v_{tS}) are defined by:

$$v_{rS} = \eta_R \cdot v_{rR} + \cdot \eta_H \cdot v_{rH} \qquad (1)$$

$$v_{tS} = \eta_R \cdot v_{tR} + 0,5 \cdot \eta_H \cdot v_{tH} \qquad (2)$$

where η_R is the efficiency of robot motion commands and η_H is the efficiency of human motion commands. Both robot commands and human output are added as weighted vectors (Fig. 1), so that humans receive more control as a reward for a better efficiency. The shared motion command efficiency is defined as η_S. Efficiencies range from 0 to 1, being 1 the maximum efficiency. It must be noted that η_S is not equal to η_R nor equal to η_H. Since shared commands linearly combine both robot and human ones, η_S will tend to average η_R and η_H.

As combination is performed at reactive level, efficiencies (η) should only be evaluated in terms of factors having an immediate effect on them. Consequently, three factors are taken into account: smoothness (η_{sf}), directiveness (η_{tl}) and safety (η_{sc}), each of them ranging from 0 to 1. Smoothness reflects how sharp direction changes are undesirable for driving. Safety reflects that it is better to keep away from obstacles. Directiveness tries to reflect that moving ahead to the goal in a straight way leads to shorter paths.

Smoothness (η_{sf}) is locally evaluated as the angle between the current direction of the robot and the provided motion vector. It is meant to take into account that many robots are non-holonomic and that it is better to change heading as less as possible to avoid slippage and oscillations. If C_{sf} is a constant and α_{dif} is the angle difference between the current direction and the command vector, η_{sf} will be:

$$\eta_{sf} = e^{-C_{sf} \cdot |\alpha_{dif}|} \qquad (3)$$

Directiveness (η_{tl}) is locally measured in terms of the angle conformed by the robot heading and the direction towards the next partial goal provided by the global planner. Partial goals in our experiments are obtained as presented in (Urdiales et al. 2003). These partial goals are the extracted from the curvature of the path returned by a deliberative layer, so that reaching from one to the next can be achieved by the reactive layer on its own. Given a goal, partial or not, the shortest way to reach that goal is to make that angle zero. Let C_{tl} be a constant and α_{dest} the angle between the robot heading and the direction towards the next partial goal. Hence, η_{tl} is calculated as:

$$\eta_{tl} = e^{-C_{tl} \cdot |\alpha_{dest} - \alpha_{dif}|} \qquad (4)$$

The third factor, *Safety* (η_{sc}), 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 α_{min} is the angle difference between the current direction and the direction of the closest obstacle, η_{sc} will be:

$$\eta_{sc} = 1 - e^{-C_{sc} \cdot |\alpha_{min} - \alpha_{dif}|} \qquad (5)$$

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

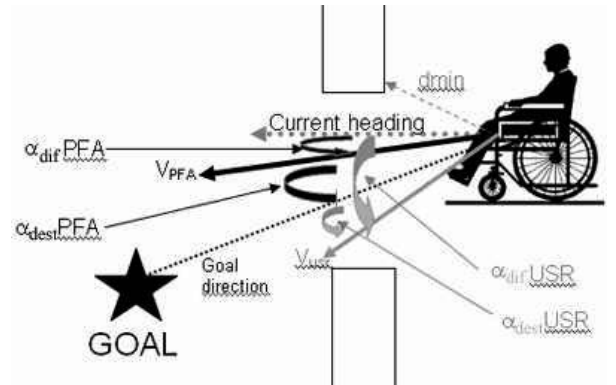$$\eta = \frac{\eta_{sf} + \eta_{tl} + \eta_{sc}}{3} \qquad (6)$$

Figure 1: Local efficiency factors for human and robot

All mentioned factors are reflected in Fig. 1 for human and robot. eq_{eta} is basically used to decide who makes the smarter move at each point, human or wheelchair, so that they get awarded with more control, but not neglected from the emergent behavior nevertheless: if human efficiency is

bigger, motion mostly obeys the driver, whereas if it is small, the wheelchair tends to move on its own.

The proposed approach has several advantages: i) it tends to preserve curvature and to grant safety, as most PFA-based algorithms; ii) humans are in control all the time and they do not perceive sharp control swaps; and iii) humans provide deliberation and avoid local traps.

Experimentation and evaluation

In order to evaluate how adequate the proposed algorithm is for navigation assistance, we performed some tests with volunteering in-patients that already were using a wheelchair in Fondazione Santa Lucia (FSL), a hospital specialized in rehabilitation in Rome (Italy), during July 2007. The experiments were fully described in (Urdiales et al. 2007). Volunteers were characterized in terms of disabilities to have a global idea about their capacities. As it is not easy to determine this from a single index, since Functional Disability (FD) is the result of the interaction of different individual components of compromised functions (Guralnik 1993), several different disability scales have been used to evaluate the state and condition of the in-patients, namely the Mini-Mental State Examination (MMSE) (Crum et al. 1993), Geriatric Depression Scale (GDS) (Yesavage et al. 1983), and Barthel (Mahoney and Barthel 1965). All these indexes were obtained by the specialists at FSL. All in-patients were asked to move from a corridor into a room and viceversa by crossing a door. In the first case, in-patients had to decide when turn left to enter the room. In the second, they were aligned with the door and could freely decide when to turn right to face the corridor. It can be noted that the first test requires more cognitive capabilities than the second, as it is necessary to decide correctly when to turn.

Evaluation metrics   The performances of the in-patients were measured in terms of the local efficiencies calculated by the shared control algorithm. However, we decided later to use captured traces to also measure the quality of the human-computer interaction. In our case, this factor could only be indirectly measured from the available data. We chose to use several metrics. First, the *Intervention level*, defined as the portion of time that the user moves a joystick. It must be noted that in our approach to shared control, a high intervention level is desired, meaning that the system is highly cooperative. We are also interested in knowing if person and machine cooperate in a seamless way. This is evaluated via a parameter named *Disagreement*, which represents the difference between the human output and the robot output. Since both outputs are vectors, we measure Disagreement in terms of angles. A 0 Disagreement means that the robot and the human totally agree. A high Disagreement is expected to be related with effort and frustration. As target population may present cognitive disabilities, it is also important to take into account *Consistency*, defined as the variation of the user output when facing similar situations. A high Consistency is expected to be related to users with good cognitive capabilities, whereas a low one is related to random joystick motion.

Users' performance   Experiments were performed by 36 in-patients, whose FD related indexes are presented in table 1. In average, most persons have an acceptable cognitive capability (see MMSE), even though some of them scored for dementia ($MMSE < 20$). Physically, they had some disabilities ($Barthel < 90$), which ranged from minor to severe. Regarding GDS, they presented significant variations.

Table 1: Disability indexes in the test population

Index	Average	Deviation
MMSE	23.47	8.57
GDS	5.02	3.70
Barthel	64.20	29.66

Using the proposed control strategy, all in-patients managed to finish both requested trajectories. Fig. 2 shows the paths followed by the different users in the first experiment (move from corridor to a room). It can be observed that paths are initially very similar, but they change significantly when turning to cross the door, as showed in detail on the right of the figure. This occurs because no turning point was specified: the users were only instructed to turn left and cross the door.

Efficiencies are briefed in table 2. It can be observed that, as expected, most users perform similarly in terms of efficiencies calculated to share control because, when they behave worse, the wheelchair provides the amount of help necessary to keep performance adequate. It can also be observed that $safety$ is particularly high, due to the nature of PFA, and that $directiveness$ is the lowest local efficiency, as it implies keeping in mind a clear goal and executing the necessary commands to achieve it at all times. Regarding the other factors, it can be observed that deviations are much larger, as they have no effect in control sharing in our algorithm. Consistency is basically correct, as users tend to agree with themselves in most cases. However, they disagree more often with the wheelchair, sometimes very frequently, as evidenced by the disagreement parameter.

Table 2: Disability indexes in the test population

Efficiency factor	Average	Deviation
Smoothness	82.01	9.90
Safety	96.32	0.10
Directiveness	63.52	9.07
Intervention level	88.22	26.6
Disagreement	24.58	23.3
Consistency	12.2	6.3

Divergence and disagreement   If we correlate experimental data (table 2) with in-patients FD (table 2) and go down to case level, several interesting things can be observed:

- MMSE is inversely proportional to Disagreement, meaning that persons with less cognitive capacities agree more with the chair. Obviously, Disagreement is also linked to the time to finish the trajectory: an user with a higher Disagreement takes longer to reach a goal.
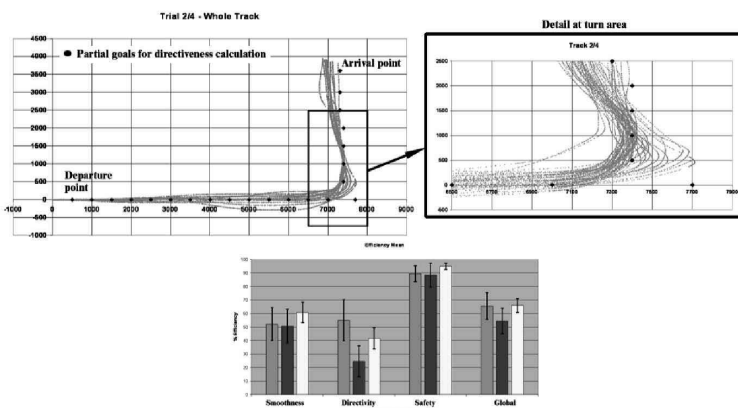
Figure 2: Tracks to move from corridor to room.

- Intervention level depends on Barthel: persons with better physical capabilities tend to be more cooperative and also present a better smoothness, even though the proposed control scheme tends to equalize performances at these levels (smoothness, directiveness and safety)

- If MMSE and Barthel are very low OR very high at the same time, efficiencies are worse than in other cases. This is particularly interesting because it basically means that users with good capacities tend to fight the chair when they realize they are not completely under control.

Adaptation to the user's driving

In order to provide a better adaptation to the user and, hence, reduce Disagreement, we decided to check if we could learn how a given in-patient controlled the chair from his/her trace. As we work at reactive level all through the experiments, we decided to also capture the reactive nature of their driving behavior, meaning that we search for a duplex of wheelchair input readings (range sensors) and output command (joystick). This pair characterizes what the user does at every given situation. In order to capture these pairs, we have used CBR.

Case-Based Reasoning (CBR) is a reasoning, learning and adaptation technique to solve problems by retrieving and adapting past experiences (Aamodt and Plaza 1994). CBR is mostly derived from the *Theory of Dynamic Memory* (Schank 1982), that introduces indexing as the key to use experience in understanding.

A CBR system cycle to solve a new problem consists of four steps: i) *retrieve* the most similar stored *case* or *cases* to the new current *case*; ii) *adapt* its solution to the new current *case*[1]; iii) *evaluate* the results of the proposed solution; iv) *learn* from the new experience. Fig. 3 shows the steps in the well known 4R format. First, the input problem is characterized by means of a number of significant parameters, which, along with the problem solution, conform a case. Thus, redundancy and, hence, computational

---

[1]In this case we do not use case adaptation so as not to modify original values from patients

requirements are reduced. Definition of a correct problem instance is of key importance in this kind of systems. The input instance is matched against all known cases, so that the most similar one in the casebase is retrieved. After this stage, the retrieved case is compared with the input situation to adapt it if necessary. Thus, better solutions can be derived when faced against less experienced situations. Finally, the adapted case is evaluated and stored for future use.

CBR has been used in robot navigation before, but typically for high level planning rather than to accomplish reactive behaviors. CBR has been mostly used for behavior selection and transition and selection of behavioral parameters. However, in these approaches behaviors were not themselves developed via CBR. CBR in navigation has been used for global path planning in static environments (Branting and Aha 1995)(Fox et al. 1998), where cases absorb the structure of the environment. Other CBR based navigation methods focus on non-pure reactive navigation (Likhachev, Kaess, and Arkin 2002) (Ram and Santamaria 1993), but they basically rely on accumulating experience over a time window while navigating in a given environment to obtain an emergent global goal seeking-behavior. The authors proposed a purely reactive CBR based navigation approach in (Urdiales et al. 2006) that has been modified to achieve the goals described above.

Basically, each time a new input instance is detected, our CBR captures a new case, coupling this instance with the user's joystick output and using the local efficiencies (directiveness, smoothness and safety) to rank how good the case is. The input instance describes the relative position of obstacles with respect to the wheelchair and the goal. In our case, these distances are split into bins corresponding to danger/near/medium/far/no influence relative distances between chair and obstacles in order to keep a bounded number of cases learnt, as proposed by the authors in (Urdiales et al. 2006). Then, a clustering algorithm was used to turn this case set into a casebase. This clustering process had several goals. First, it avoids too large casebases, as similar cases are averaged into a prototype. Second, cases associated to similar sensor readings but different joystick outputs tend to provoke oscillations. These cases, too, are averaged into the same prototype. Finally, the clustering process filters punctual cases due to errors and remove them from the casebase.

In standalone mode, an input vector (sensor readings plus goal) is fed to the CBR server. The CBR server access the casebase and looks for the most similar case available and returns it to the client. This case includes also the joystick readings, which is what the emulator assumes that the user would have done in the input situation. In order to determine the likeness between cases, we used a Tanimoto distance (Deichsel and Trampisch 1985). The main difference between this type of distance and an Euclidean one is that it weights the global shape of the instance rather than isolated similarities alone. Hence, matching laser readings presenting low distances are related to similar surrounding geometries because the global shape of the case is weighted.

After the specifics of a given in-patient have been captured into a casebase, the CBR system can be used to replace the PFA, so that the wheelchair drives like the user
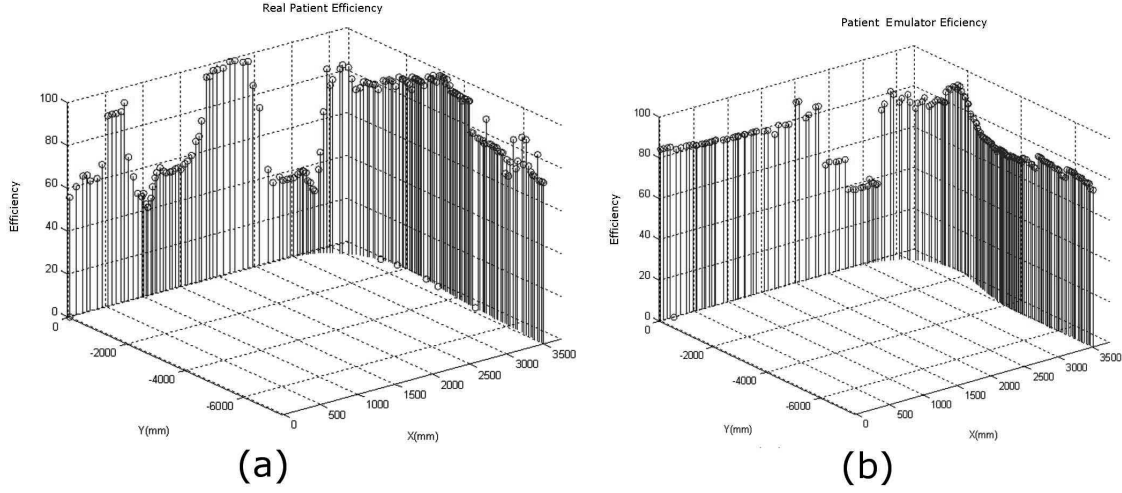
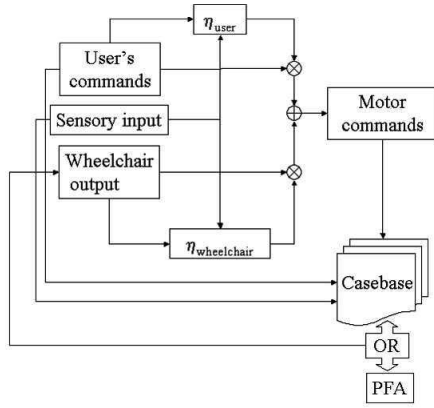Figure 4: Emulated patient (a) and real patient (b) efficiency.

Figure 3: Proposed collaborative control scheme.

and, hence, Disagreement is reduced. It is interesting to note that different users generate different casebases and, hence, different ways of driving. In our experiments, we duplicate how the user drives with collaborative control, because users might not be capable of driving on their own. However, other experiments may be based on traces captured from persons driving on their own.

Fig. 3 presents the proposed collaborative control scheme after CBR learning is included in the architecture. It can be observed that the emerging motor commands are still the linear combination of human and wheelchair commands, which are directly coupled with the sensory input (either human or mechanical) in a purely reactive way. These commands are weighted by their average efficiencies, so that low human efficiencies imply higher help from the machine. The main difference when we include CBR learning is that cases related to specific situations are stored in terms of what an specific person would do given that situation. Reflexes are implicit in people, so the casebase basically stores how the

person drives the wheelchair. Through use, the casebase outputs grow more and more similar to the person's commands. However, if a person drives the wheelchair badly, those cases would also be learnt by the chair. In order to avoid this, we learn cases after they are modulated by either the PFA algorithm or an efficient human command, coming from the PFA block or an already existing case, respectively. Whenever an input situation is detected, the system asks the CBR for the closest case available. If there is none, we use a mix of PFA and human commands as output and learn the corresponding case. If there is any, we use a combination of human and CBR commands as output and, if necessary, learn this combination as well, so that control tends to be progressively more tuned with the person's guidance.

In this work, as learning was done a posteriori, we did not implement a safeguard system to avoid learning non-efficient cases when new human commands are worse than the ones already stored in the casebase. However, this could be achieved by simply comparing the efficiencies of the input human command and the CBR one: if the first one is significantly worse than the second, we use it for the linear combination but do not update the casebase.

Next section presents some of the results of the proposing technique.

Experiments and results

In this section we present some results on the use of CBR to learn a given person's driving specifics. As commented, these experiments are performed with data captured from real patients driving a laser equipped Meyra wheelchair in Fondazione Santa Lucia (Rome) in July, 2007, but it is interesting to point out once more that learning was done a posteriori, so there are no results on the convergence of the whole system to each in-patient yet. These experiments, though, will be done in a near future. It is interesting to note that users were asked to do 6 paths, three from corridor to room and three from room to corridor.

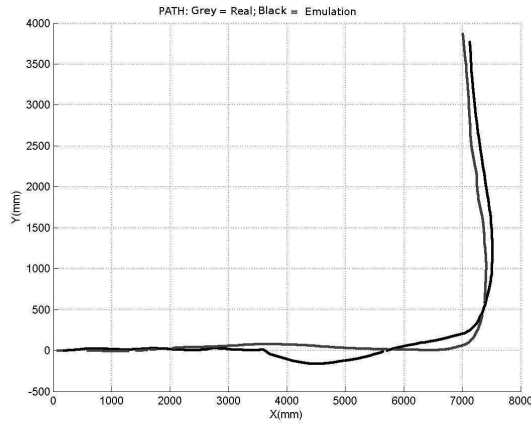First, we capture a given real user trace to move out of a
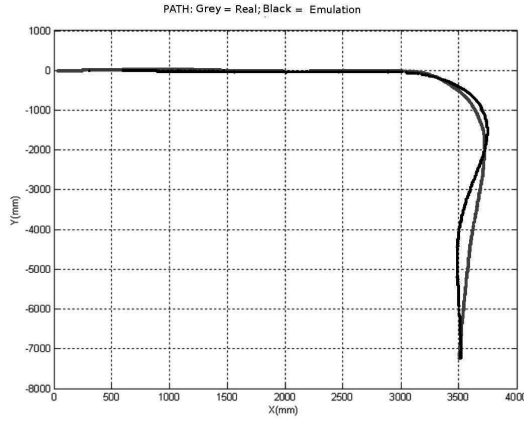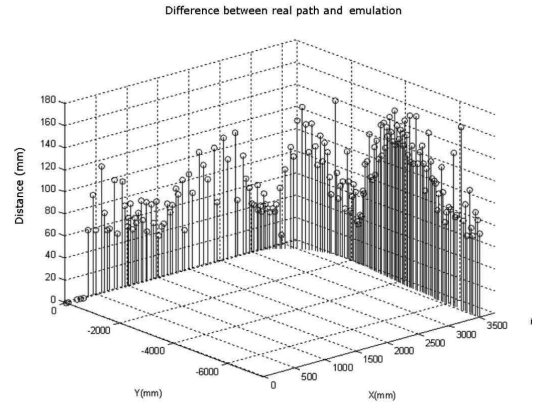
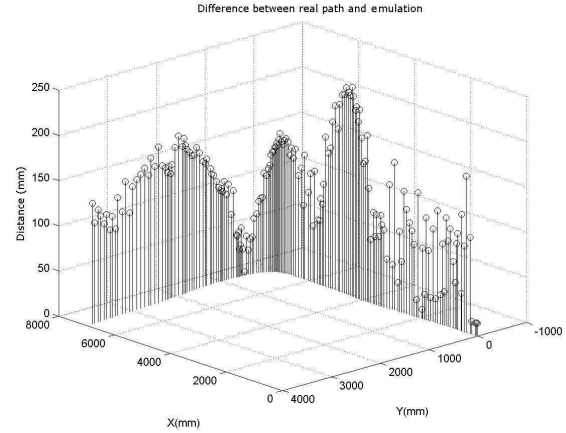Figure 5: (a) Path of the emulation and the real experiment. (b) Differences between real experiment and emulation in millimeters.

Figure 6: (a) Path of the emulation and the real experiment. (b) Differences between real experiment and emulation in millimeters.

room. In order to do that, we chose a random trajectory of that in-patient to fill a casebase. Afterwards, we used that casebase to emulate a similar trajectory, departing from a slightly different location. The resulting trajectory was then compared with another real path, different from the trajectory we use to fill the casebase. Fig. 5.a shows simultaneously a real path (blue) used to capture a patient case base and the emulated path (red) that would result, given the same departure and arrival points, for the same in-patient. It can be observed that both paths are very similar, despite minor differences. In fact, the average deviation is 7.02 cm in an 11 meters long route. These differences correspond to non systemic driving behaviours that result in non repetitive action/reaction pairs not captured by the CBR. If the casebase is trained with a large enough number of trajectories, these errors might be reduced, but, still, some differences would

appear.

In order to prove that emulated patients and real patients behave similarly, Fig. 4 presents the efficiencies of real patients versus emulated ones (76.03% versus 74.39 %). It can be noted that they are quite similar as expected, even in terms of directiveness, which is where more differences are. Larger differences are detected at the beginning of the path, most likely due to initial person insecurity when starting to drive. Efficiency in emulated patients is more homogeneous than the real patients' one, though, as errors are filtered out of the casebase.

If we analyze the second, more complex test -moving through a corridor and, at some point, deciding when to turn to enter a room and move inside-, results are similar than in the previous case. Fig. 6.a shows simultaneously the path of the patient emulator (red) and the real patient one (blue). In
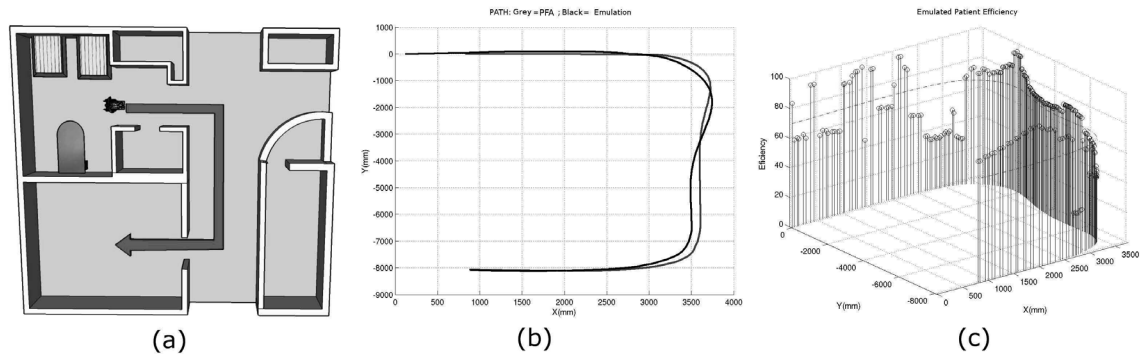
Figure 7: (a) Emulated track; (b) Emulated patient vs. Potential Fields (c) Emulated patient efficiency.

this case, the maximum deviation between the real path and the emulation one is 16.5 cm, and average deviation is 7.02 cm. As before, this deviation is considered small for an 11 meters long trajectory.

Experiments commented above are suitable to analyze results once they have been obtained, but in order to work online, it would be necessary to have a casebase ready whenever the robot is moving. If this requires data acquisition each time a new trajectory is performed, the system is not that useful. However, the reactive nature of the proposed algorithm allows generation of totally new trajectories from data acquired from older, different ones. This is possible because CBR acquires punctual information: the tendency of an user to choose a direction, given a configuration of obstacles around, is more or less the same if he/she plans to turn at the end of a corridor or not. To check this, we emulated a non-existing trajectory (Fig. 7.a) using data from previous experiments. As we have data on corridor navigation and door crossing, this final experiment consists of making the emulated patient leave a room, turn, head for the corridor in a straight way for about 8 meters and then enter into another, different room. The entire trajectory is about 14 meters long, 3 to navigate in/out of each room and 8 in the corridor. Only initial and departure points are provided to the emulator, as usual, but the casebase is learnt both from a room entering and room exiting trajectory from the same patient, because in the first case there were no turns right and in the second there were no turns left.

Fig. 7.b shows the result of this test. Fig. 7.c shows the efficiency of the emulated patient in this emulation. This efficiency is similar to the patient ones in the real experiments: Average efficiency is 75.83 % and typical deviation 12.10 %. As in the previous sections, deviation is lower than in the real patient, as expected.

This final experiment supports that after learning from a few trajectories, we would have a casebase tuned to a person's driving, which is adapted through experience to his/her specifics. Cases would eventually replace the PFA algorithm so that there would be less divergency between the person's decisions and the wheelchair motion commands. This is going to be tested in the next experiments in Santa Lucia.

Conclusions and future work

Assistive navigation will remain as a fertile ground for basic research in the years to come as it not only implies the development of new sensors, new hardware and software but also a deep investigation on the clinical, rehabilitation and ethical issues of the problem. As there are so many researchers from different areas working in the field, standarization and proper evaluation metrics will be of key importance for testing and replicability.

In this paper we have presented a review of most common metrics in the field, as briefed in table 2. The presented metrics range from user's condition medical evaluation to personal questionnaires, including the most common parameters used by engineers to measure performance in new navigation assistance algorithms and Human Computer Interface. Consequently, this review is expected to be useful for doctors and caregivers, engineers and end users all alike.

Acknowledgments

References

Aamodt, A., and Plaza, E. 1994. Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications* 7(1):39 – 59.

Aigner, P., and McCarragher, B. J. 2000. Modeling and constraining human interactions in shared control utilizing a discrete event framework. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans* 30(3):369–379.

Bourhis, G., and Agostini, Y. 1998. The vahm robotized wheelchair: System architecture and human-machine interaction. *Journal of Intelligent Robotic Systems* 22(1):39–50.

Branting, L. K., and Aha, D. W. 1995. Stratified case-based reasoning: Reusing hierarchical problem solving episodes. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence*, 384 – 390.

Brooks, R. A. 1986. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation* 2(1):14–23.

Bruemmer, D. J.; Few, D. A.; Boring, R. L.; Marble, J. L.; Walton, M. C.; and Nielsen, C. W. 2005. Shared understanding for collaborative control. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans* 25(4):494–504.

Connell, J. H., and Viola, P. 1990. Cooperative control of a semi-autonomous mobile robot. In *Proc. of the IEEE Conference on Robotics and Automation*, 1118–1121.

Crisman, J., and Cleary, M. 1998. *Progress on the deictic controlled whellchair*. Springer-Verlag. 137–149.

Crum, R.; Anthony, J.; Bassett, S.; and Folstein, M. 1993. Population-based norms for the mini-mental state examination by age and educational level. *Journal of the American Medical Association* 269(18):2386–239.

Deichsel, G., and Trampisch, H. J. 1985. Clusteranalyse und diskriminanzanalyse. *Gustav Fischer, Verlag* 30(7).

Fox, D.; Burgard, W.; Thrun, S.; and Cremers, A. 1998. A hybrid collision avoidance method for mobile robots. *Proc. of IEEE Int. Conf. on Robotics and Automation* 1238–1243.

Frese, U.; Larsson, P.; and Duckett, T. 2005. A multi-grid algorithm for simultaneous localization and mapping. *IEEE Transactions on Robotics* 21(2):1–12.

Gomi, T., and Griffith, A. 1998. *Developing Intelligent Wheelchairs for the Handicapped*. Springer-Verlag. 150–178.

Horiguchi, Y., and Sawaragi, T. 2005. Effects of probing to adapt machine autonomy in shared control systems. In *Proc. International Conference on Systems, Man and Cybernetics*, volume 1, 317–323.

J. M. Guralnik, E. M. S. 1993. Physical disability in older americans. *Journal of Gerontology* (48):3–10.

Khatib, O. 1986. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research* 5(1):90–98.

Kofman, J.; Wu, X.; Luu, T. J.; and Verma, S. 2005. Teleoperation of a robot manipulator using a vision-based human-robot interface. *IEEE Transactions on Industrial Electronics* 52(5):1206–1219.

Likhachev, M.; Kaess, M.; and Arkin, R. 2002. Learning behavioral parameterization using spatiotemporal case-based reasoning. *Proceedings. ICRA '02. IEEE International Conference on Robotics and Automation* 2:1282 – 1289.

Mahoney, F., and Barthel, D. 1965. Functional evaluation: the barthel index. *Maryland State Medical Journal* (14):56–61.

McLachlan, S.; Arblaster, J.; Liu, D. K.; Valls, J.; and Chenoweth, L. 2005. A multi-stage shared control method for an intelligent mobility assistant. In *Proc. of the 2005 IEEE 9th International Conference on Rehabilitation Robotics*, 426–429.

Miller, D. 1998. *Assistive robotics: an overview*. Springer-Verlag. 126–136.

Nisbet, P.; Craig, J.; Odor, P.; and Aitken, S. 1995. 'smart' wheelchairs for mobility training. *Technol. Disability* 5:49–62.

Parikh, S. P.; Grassi, V.; Kumar, V.; and Okamoto, J. 2005. Usability study of a control framework for an intelligent wheelchair. In *Proc. of the 2005 IEEE International Conference on Robotics and Automation*, 4745–4750.

Prassler, E.; Scholz, J.; and Fiorini, P. 1999. Navigating a robotic wheelchair in a railway station during rush hour. *Int. Journal on Robotics Research* 18(7):760–772.

Ram, A., and Santamaria, J. C. 1993. A multistrategy case-based and reinforcement learning approach to self-improving reactive control systems for autonomous robotic navigation. *Proc. of the 2nd Int. Workshop on Multistrategy Learning* 129 – 150.

Rao, R. S.; Conn, K.; Jung, S. H.; Katupitiya, J.; Kientz, T.; Kumar, V.; Ostrowski, J.; Patel, S.; and Taylor, C. J. 2002. Human robot interaction: Applications to smart wheelchairs. In *Proc. IEEE International Conference on Robotics and Automation*.

Schank, R. 1982. *Dynamic memory*. New York: Cambridge University Press.

Simpson, R., and Levine, S. 1997. Development and evaluation of voice control for a smart wheelchair. In *Proc. Annu. RESNA Conf.*, 417–419.

Simpson, R., and Levine, S. P. 1998. *NavChair: An Assistive Wheelchair Navigation System with Automatic Adaptation*. Springer-Verlag. 235–255.

Urdiales, C.; Bandera, A.; Pérez, E. J.; Poncela, A.; and Sandoval, F. 2003. *Hierarchical planning in a mobile robot for map learning and navigation*. Physica Verlag. 2165–188.

Urdiales, C.; Perez, E.; J.Vazquez-Salceda; M.Sanchez-Marre; and Sandoval, F. 2006. A purely reactive navigation scheme for dynamic environments using case-based reasoning. *Autonomous Robots* 39(5):67–78.

Urdiales, C.; Poncela, A.; Sanchez-Tato, I.; and Sandoval, F. 2007. Efficiency based reactive shared control for collaborative human/robot navigation. In *Proc. of the IEEE Conference on Intell. Robots and Systems (IROS'07)*.

Yanco, H. A. 1998. Wheelesley: A robotic wheelchair system: indoor navigation and user interface, assistive technology and artificial intelligence. *Applications in Robotics, User Interfaces and Natural Language Processing* 256–268.

Yesavage, J.; Brink, T.; Rose, T.; Lum, O.; Huang, V.; Adey, M.; and Leirer, V. 1983. Development and validation of a geriatric depression screening scale: A preliminary report. *Journal of Psychiatric Research* 17:37–49.