

Autonomous Wheelchair for Disabled People

G. Pires, N. Honório, C. Lopes, U. Nunes, A. T Almeida

Institute of Systems and Robotics
University of Coimbra, Polo II
3030 Coimbra, Portugal

Abstract - This paper describes the RobChair assistive navigation system. RobChair project was conceived with the aim to assist disabled people in the difficult task of manoeuvring a powered wheelchair. This paper describes the overall hardware and software architecture including the communication system; a friendly Graphical User Interface (GUI) which also works as a simulator; and introduces a voice Human Machine Interface (HMI). The system's architecture follows a behaviour-based control architecture.

I. INTRODUCTION

The use of powered wheelchairs with high manoeuvrability and navigational intelligence is one of the great steps towards the integration of severely physically disabled and mentally handicapped people. Driving a wheelchair in domestic environments is a difficult task even for a normal person and becomes even more difficult for people with arms or hands impairments. Tetraplegic people are completely unable to operate a joystick unless they use the tongue, which is obviously a very tedious task. Simultaneously blind and paraplegic people deal with a very uneasy situation which couples two problems: locomotion and localisation. The RobChair system is being developed to overcome the problems described above, allowing the end-user to just perform safe movements and accomplish some daily life important tasks.

The organisation of this paper is as follows. Section II describes the hardware and software architecture. Section III presents the GUI. Section IV discusses the behaviour-based control architecture. Section V approaches a Voice HMI which aims to overcome the problems blind and tetraplegic deal with. Section VI describes some practical experiences.

II. IMPLEMENTATION AND SYSTEM CONFIGURATION

Figure 1 shows a diagram with the main modules of the system and a picture of the wheelchair. The wheelchair¹ is a conventional electric wheelchair with two motorised rear wheels and casters in front. There is also a 5th rear wheel connected to the back of the wheelchair with a damper used for stability. A normal analogic joystick is used to control the wheelchair.

The sensorial system is composed by 12 infrared sensors, 4 ultrasonic sensors, a front bumper and optical encoders on wheels. Sensor arrangement is shown in Fig. 5. A real-time operating system runs on a PC based system which allows to perform basic tasks such as wheelchair control, gather sensor information and communicate with client workstation programs.

Build a robust, modular and user-friendly system is very important to guarantee a good performance of the overall system. Communications and software architecture will be explained on the next subsections.

A. Hardware and software architecture

Fig. 2 represents all the physical devices which compose the system and shows how they interact with each other. In the workstation platform a mouse and a joystick can be used to remotely control the wheelchair. In the wheelchair the main devices are: the sensorial system composed by infrareds, sonars and a bumper; the wheels encoders which allow position estimation; and the joystick to directly manoeuvre the wheelchair.

Fig. 3 shows an overview of the software architecture. The system's tasks are modular, which means that they do not directly depend on other tasks and can run on a stand-alone basis. However, they allow communication between them. Each server task provides a service:

- **Vehicle control and position monitoring** - this task sends drive commands to the wheelchair motor controller and "requests" odometric information.
- **Infrared, sonar and bumper readings** - this task takes charge of gathering sensor measurements.
- **User Interface** - this task gives the user the means to perform pre-defined "actions/tasks" by using a mouse or a keyboard and allows the user to communicate with other people sending and receiving messages.
- **Obstacle avoidance algorithms** - this task uses sensor measures information to avoid obstacles.

¹ The wheelchair is a KIPR (KISS Institute for practical Robotics) product.

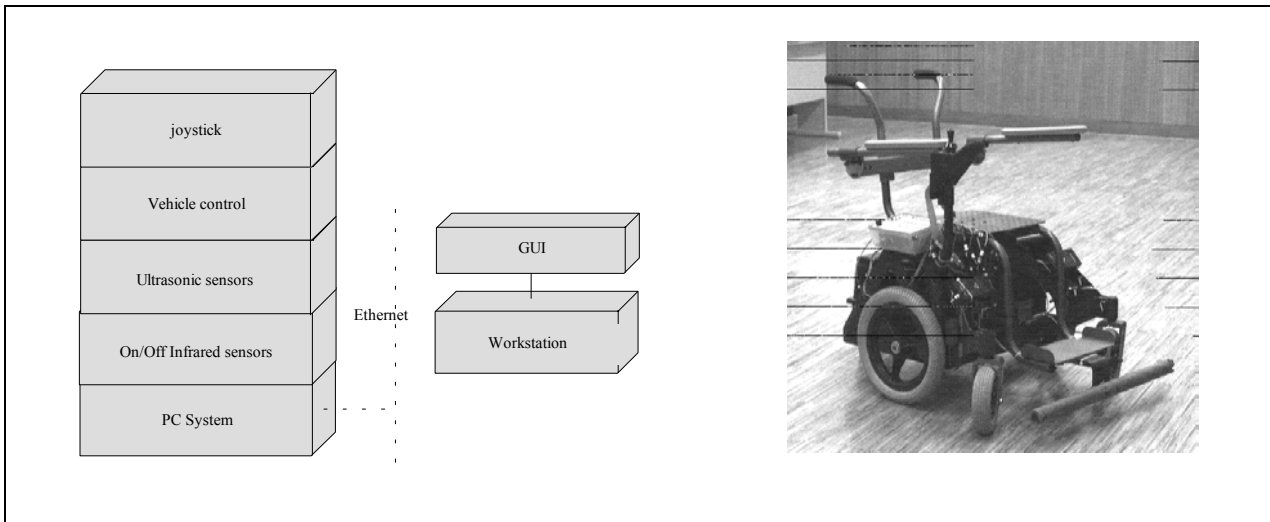


Fig. 1 - left: Modules of the system; right: picture of the wheelchair.

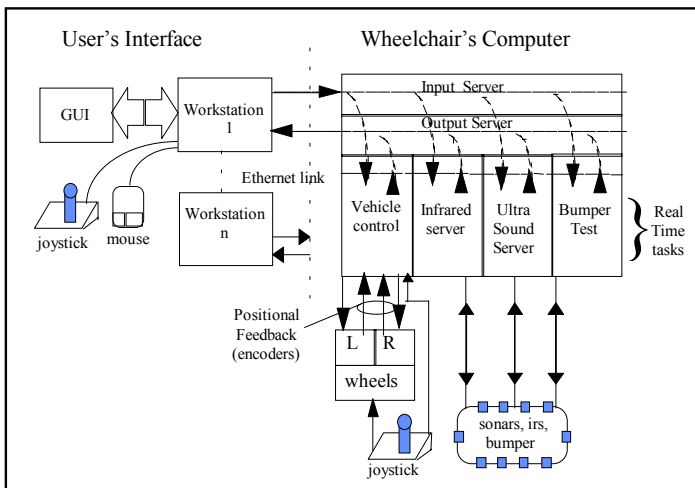


Fig. 2 - Physical devices which compose the system

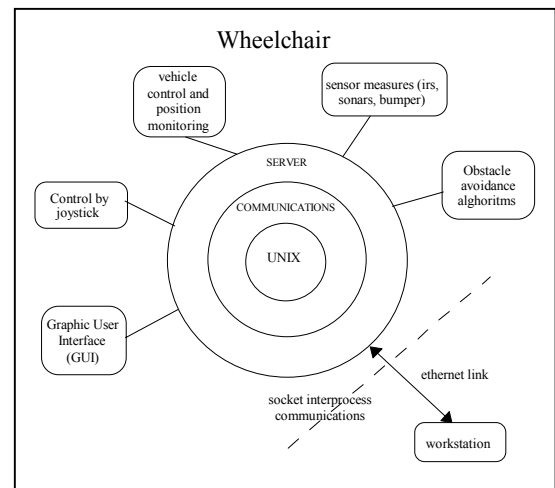


Fig. 3 - Software architecture.

- **Communications** - this task handles the communication between the wheelchair and remote operators.

B. Communications

The communication system (see Fig.1 and 3) is user transparent. It uses a server/client model to exchange information between wheelchair's server and workstation clients. An ethernet card is used to communicate with workstation clients using UNIX socket based inter-process communications. This way the wheelchair can communicate with any computer linked to the network. The wheelchair's server monitors a particular socket address for connections. Connections from valid clients are accepted and a serial stream protocol is provided. To develop client programs the programmer have access to a library of functions which allow an easy implementation of application programs. To simplify the use of these functions the programmer just have to know the IP

(Internet Protocol) address and port of the server (the layer link is completely programmer transparent).

The communication capability is very important since it allows the disabled to easily ask for help to a remote operator and this one can assist the disabled giving him some instructions, or even remotely control the wheelchair. In this phase of the project the communication system is being used to remotely control the wheelchair and receive sensor information.

C. Wheelchair Motion

The wheelchair performs two kinds of movements: straight and pure rotational movements. These two movements executed at a given speed allow to accomplish the tasks we proposed to do. The wheelchair position relies on dead reckoning from wheels encoder readings. Wheelchair displacement and heading angle are obtained from the well known kinematics equations given in [1]. Nevertheless these equations give an accurate method to calculate the

position, it is well known in practice that mobile robotics deal with very poor dead-reckoning. Problems such as wheel slippage and variable surface characteristics contribute to a poor dead-reckoning. David Bell say in [2] about wheelchairs: “Even in straight travel, variations in wheel diameter due to load shifts cause angular accuracy to be an order of magnitude worse than in most mobile robots”. This statement has indeed been confirmed.

III. GUI INTERFACE

The GUI is completely user transparent and allows to perform high level tasks. The GUI was created using the FORMS library and FORM² designer, this is a GUI Toolkit for X, the library uses the services provided by the Xlib and runs on any workstation that have X Window System installed. This toolkit is very useful to create 2D graphical environments (menus, buttons, scrolling, panels, drawhandlers). The Interface is shown in Fig. 4. The main panel (Fig. 4a) displays the world environment and sensor data and has a menu-system to accede to other application panels such as a “joystick” panel, a “record & playback” panel and a “command line” panel. The GUI has the ability to create complex world environments which includes walls and round and square obstacles, allows a zoom in/out capacity and shows sensor data and backtrace positioning. The GUI has two working modes: real mode and simulated mode.

A. Real Mode

In real mode the GUI is connected with the wheelchair. A “joystick” panel (Fig. 4b) provides an easy way to control the wheelchair’s movements (forward, backward, left and right rotation). Estimated position (x,y,θ) returned from the wheelchair allows to represent graphically the wheelchair in the real world environment. Similarly, sensors data are displayed with different colours accordingly with sensor type. This way we can visualise what the wheelchair actually “sees” understanding much better its behaviour. Besides this advantage there is another one which resides on the possibility to remotely control the wheelchair without actually seeing it.

A “record & playback” panel (Fig. 4c) is used to save and load some trajectories such as a path from A to B or a passage through a door. The parameters that can be saved and posteriorly load are position and sensors data.

Another panel provided is the “command line”, this panel is provided to send specific commands to the

wheelchair permitting for instance to change the sensors configuration.

B. Simulated Mode

In this mode there is no connection with the wheelchair. Commands are sent to a simulated wheelchair which performs the same kind of movements performed by the real one. Sensors data are also simulated to be as real as sensor measures. The advantage to build a simulator is that we can easily test obstacle avoidance algorithms and confirm, as a starting-point, their efficiency. Another great advantage of this GUI is that we can build complex environment worlds difficult to replicate in practice.

IV. BEHAVIOUR-BASED CONTROL ARCHITECTURE

The control architecture under development is based on a sub-assumption control architecture [3]. This is a global model composed by a set of modules, each one generator of a behaviour according to an horizontal structure. The behaviours generator modules work in parallel and compete to control the system or a certain set of actuators. Assuming the simplest case, the choice of a behaviour is done by an arbitrage process which works attending to priorities given to the different modules and by an interaction process between modules according to a mechanism delineated before. One of the great problems of this architecture resides on the difficulty to perform autonomously the co-operation between behaviours. The emergency of complex behaviours from simple behaviours co-operation is a typical example of a research problem based on behaviours control architectures. On a general view, it is been accepted that a purely reactive control is not enough whenever the tasks need a global knowledge, for example to mobile robots navigation it is necessary a world model to allow an intelligent, efficient and flexible navigation.

We have approached a two phases development. On the first phase, we aim to develop a purely reactive control giving special attention to co-operation Human-Machine, i.e., the wheelchair must be provided with a reactive capacity to avoid obstacles and have a goal-driven behaviour given by the user, by other words, the cognitive module is provided by the end-user. The system has a low-level obstacle avoidance behaviour purely reactive just based on momentary infrared and sonar sensors information. The final wheelchair movement is the result of the intention of the end-user (given by a joystick and/or

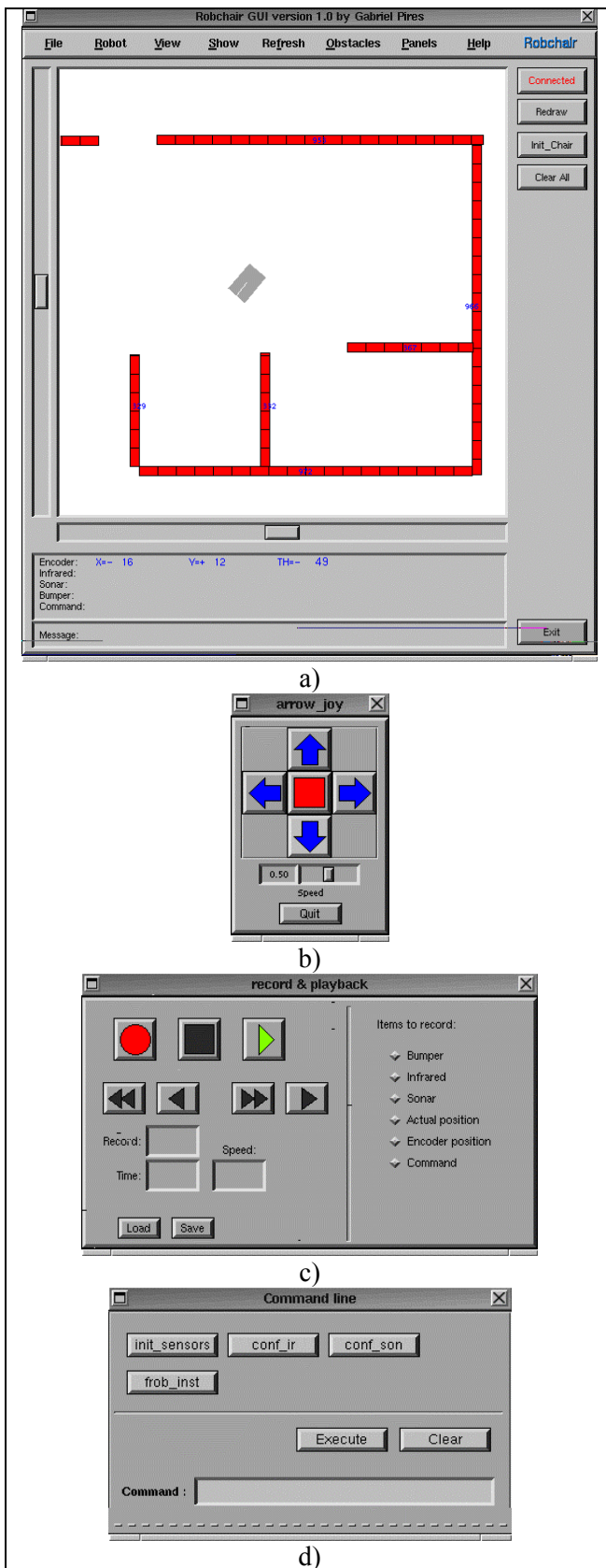


Fig. 4 GUI interface. a) Main panel. b) Record & playback panel. c) Command-line panel.

voice) and the output of the obstacle avoidance behaviour. Having the assistance of a human intelligence we don't think necessary to follow complex collision avoidance algorithms such as Potential Field [4] and Vector Field Histogram (VFH

[5], we will not be concerned with dead-lock problems since the end-user redirects the wheelchair all the time.

On a second phase, even without a completely autonomous navigation such as ordering to go from A to B autonomously, we will consider that the evolution of the system control must be provided with an autonomous cognitive module. Obviously, this cognitive module will have a perception capability (e.g. to perceive a door appearance or have a localisation capability) to map sensor information to cognitive attributes. Even a non fully autonomous wheelchair must have some capabilities to help the user, an example is a passage through a door. A solution to a "door passage" could be based on a specific behaviour to this task as long as supported with perception and cognitive capabilities necessary to this end. There is a nearly optimum way to pass through a door (distance, orientation, etc.). The behaviour module must be provided with a set of information which allows it to accomplish this task with efficiency. This philosophy proposes a cognitive component to plan actions with actuation based on a behaviour-based system.

V. VOICE HMI

Mechanical input devices such as joysticks are very problematic to people who suffers from poor manipulative capability because these devices are as accurate in controlling the wheelchair as the dexterity of the user who operates the device. Using voice commands, like *move forward* or *move right* relieves the user from precise motion control of the wheelchair and gives tetraplegic and blind people a useful way to control it. The user can, with a simple voice command, to perform a set of actions corresponding to a task. Such feature is difficult to implement with mechanical devices.

The user interface will have to interpret fuzzy commands such as *move closer* or *move very slow*, providing the user with a natural way to command the system. The architecture of control will follow the one already explained in section IV.

The user's voice is captured by head microphone and is processed by a voice recognition system³. The system has the capability of being trained which leads to a more recognition accuracy after being used many times. The HMI is still on a early stage of development.

³ Dragon VoiceTools

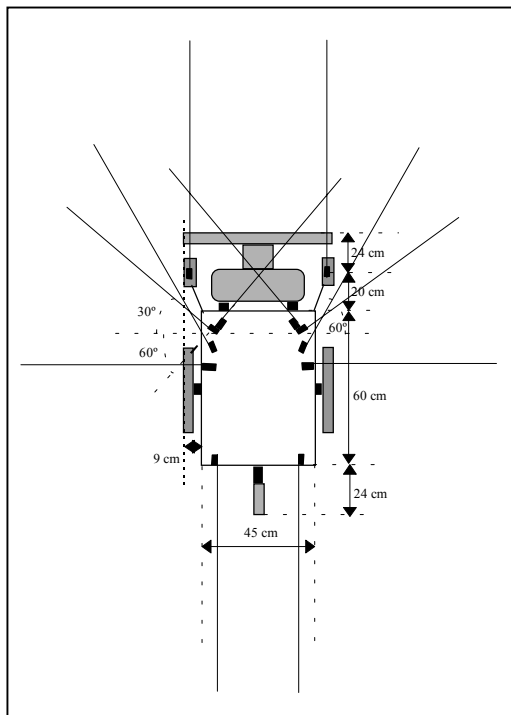


Fig. 5- IR sensor arrangement (top view).

VI. EXPERIMENTATION AND RESULTS

A. Wheelchair Control vs. Sensors

The obstacle avoidance algorithms were based on infrared (IR) sensors measures (Fig. 5 shows sensor arrangement). Two advantages of IR sensors, when compared with sonars, are the absence of cross-talk and the increased speed of obstacle detection. However, they are very directional which means they don't "cover" large areas as sonars do.

The wheelchair's control is performed on a three level behaviour. *First level*: if none of the infrared sensors detects an obstacle, the end-user can drive the wheelchair wherever he wants and with the desired speed; *second level*: if at least one sensor detects an obstacle we come in a half-security zone (represented in Fig. 5) which means to reduce to a moderated speed; *third level*: this corresponds to a binding situation where the speed must be very low.

B. Experiments

It was performed during the experimentation course three different tasks: simple obstacle avoidance, wall following and passage through a door. The philosophy employed relies on the architecture described in section IV. Two different behaviours were used to accomplish these tasks. The end-user drives the wheelchair with a joystick (goal-driven behaviour) and a collision avoidance algorithm (collision avoidance behaviour) provides safe manoeuvres. A passage through a door is faced like the only way between two obstacles (the side walls),

and following a wall is like avoid and round a rectilinear obstacle. These tasks were showed on a TV video system during the conference and have proved the effectiveness of this philosophy.

VII. REFERENCES

- [1] H.R. Everett, *Sensors for Mobile Robots - Theory and Application*, A.K. Petus Ltd., 1995.
- [2] D.A. Bell, J. Borenstein, S. P. Levine, Y. Koren, L. Jaros, "An Assistive Navigation System for Wheelchairs Based upon Mobile Robot Obstacle Avoidance," *Proceedings of the IEEE Conference on Robotics and Automation*, 1994, pp. 2018-2022.
- [3] Rodney A. Brooks, "A Robust Layered Control System For A Mobile Robot," *IEEE Journal of Robotics and Automation*, Vol. RA-2, no. 1, March 1986, pp. 14-23.
- [4] O. Khatib "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", *Int. Journal of Robotics Research*, vol. 5, no. 1, 1986, pp. 90-98.
- [5] J. Borenstein, Yoram Koren, "The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots", *IEEE Transactions on Robotics and Automation*, Vol.7, no. 3, June 1991, pp. 278-288.
- [6] ???K. Kawamura e, "Intelligent User Interface for a Rehabilitation Robot"
- [7] U. Borgolte R. Hoelper, H. Hoyer, H. Heck, W. Humann, J. Nedza, I. Craig, R. Valleggi, A. M. Sabatini, "Intelligent Control of a Semi-Autonomous Omnidirectional Wheelchair", *Proceedings of the 3rd International Symposium on Intelligent Robotic Systems '95*, Pisa, Italy, July 10-14, 1995, pp. 113-120.
- [8] J. D. Yoder, E. Baumgartner, S. B. Skaar, "Reference Path Description for an Autonomous Powered Wheelchair", *Proceedings of the IEEE Conference on Robotics and Automation*, 1994, pp. 2012-2017.
- [9] Ishay Kamon and Ehud Rivlinm "Sensory Based Motion Planning with Global Proofs", *Proceedings of the 3rd International Symposium on Intelligent Robotic Systems '95*, Pisa, Italy, July 10-14, 1995, pp. 249-256.
- [10] Daniel Simon, Paul Freedman, Eduardo Castillo, "Analyzing the Temporal Behaviour of Real-time Closed-loop Robotic Tasks", *Proceedings of the IEEE Conference on Robotics and Automation*, 1994, pp. 841-847.