Proceedings of the 2002 IEEE
International Conference on Robotics & Automation
Washington, DC • May 2002

Blacky, an Interactive Mobile Robot at a Trade Fair

Diego Rodriguez-Losada, Fernando Matia, Ramon Galan, Agustin Jimenez.
UPM-DISAM. Universidad Politecnica de Madrid.
Jose Gutierrez Abascal, 2. E28006-Madrid (SPAIN)
{drodri,matia,rgalan,ajimenez}@etsii.upm.es

Abstract— **This paper presents the first approach towards the main goal of developing a completely autonomous robot that serves as a local and remote guide at a trade fair.**

Innovative solutions are provided to solve several problems found in this environment. Reactive-perceptual behaviors are executed to provide motion while a low level controller avoids collisions. A virtual corridor map, simulated perception, and an Extended Kalman Filter for localization, are used to overcome the lack of perception. Voice synthesizing is proved to be an effective aid for navigation, as well as for the overall success and acceptance of the system.

A Denning Mrv4 robot called *Blacky* was used to carry out experiments in actual environments on three occasions, and the obtained conclusions sound promising for future research.

I. INTRODUCTION.

The objective of the current work is the creation of a fully autonomous system with high robot-people interaction capabilities, able to navigate robustly and safely in an environment as complicated as a trade fair.

Development and lab testing work started in December 2000 and ended in March 2001 at *Indumatica 2001*, a fair organized at our University in which a Denning Mrv4 robot (Section II) had to be shown working. During the next two months two more field experiments were performed, where the system had been installed and tested at a robot-contest and at another trade fair.

The new environment presents new problems and innovative solutions are provided. The whole system is built from scratch, dealing with every detail in mobile robotics: It is intended to implement a no collision low level controller, reactive behaviors, an Extended Kalman Filter (EKF)[1] for continuous localization and a virtual geometric-topological navigation map (Sections III, IV, V). Voice synthesizing is used as a navigation aid and for interacting with people [8][10][11] (Section VI). A hybrid control architecture is carried out and the software is running concurrently using different OS (Section VII, VIII, IX).

There are several references about mobile robots working in populated environments and interacting with people, but there are few of them with similar functionality. Rhino[4], Minerva[11] and Sage[8] robots from Bonn University and CMU, which are designed to serve as local and remote (via www) guides for museums, have been working successfully on several occasions.

Nevertheless, no reference has been found about any robot navigating at a trade fair, and this new environment is firstly approached in this paper.

The experience gained and the conclusions obtained (Section X) in this first approach, are very useful for future developments.

II. BLACKY, THE ROBOT.

A Mrv4 robot from Denning Branch, Inc. was used for the experiments. Its diameter is 70cm and its height is 140cm. A ring of twenty four sonars is located 50cm above the ground.

The 3 wheel synchro-drive system is equipped with optical incremental encoders, and allows the robot to move with no restrictions in the XY plane, while no heading movement is possible. Communication with the drive system and the sonars hardware is performed through an ISA card inside the on board Pentium II PC.

Fig. 1. Blacky, our Mrv4 robot.

The top platform supports a horizontal rotating laser system called *LaserNav* communicated with the on board PC by serial port. It can detect, measure the angle and identify up to 32 different (bar coded) passive targets[2]. Powerful auto-amplifying loudspeakers are also placed on this platform.

The red cover of the LaserNav is provided with funny eyes and mouth. A pirate hat completes the personality of our robot that is named *Blacky*.

This work is funded by Spanish Minister of Science and Technology (URBANO project: DPI2001-3652-C02-01) and the European Union 5th Framework R&D Programme (WebFAIR: IST-2000-29456).

III. Movement control.

Robot movement control is divided in two parts: A) A low-level controller which serves as an intelligent interface to the robot hardware, and B) several reactive behaviors, that implement simple patterns of movements, are easily used by a supervisory system to get complete tasks achievement.

A. No collision low level controller.

Historically, collision avoidance methods have been implemented embedded in the reactive algorithms that provide robot motion[7]. In this approach, collision avoidance has been embedded inside the low level controller, making it easier to develop new algorithms based on it. No autonomous motion is performed by the controller and it should continuously receive movement commands, but it is able to reduce the speed and to stop the robot completely if necessary to avoid any collision. It does not matter whether a direct teleoperation is performed, or it is a reactive behavior control, or a communication failure occur. It is never going to collide because the low level controller is continuously working. The speed reduction is computed from the current sonar readings, robot speed and direction of travel, taking into account robot geometry. Definition of security parameters is allowed to fit different environments as the lab or a fair.

The low level controller has also this functionality:

- Asynchronous communication with robot hardware.
- Movement control and regulation of position of steer (angle) and speed of drive. Limits and range management.
- Dead reckoning, continuous integration of odometry and encoder noise filtering.
- Elementary security handling controlling robot state, position and control loop timing.

B. Hybrid reactive-perceptual behaviors.

These behaviors use the low level controller to command movements to the robot based on the current position estimation and sonar readings. We have implemented several reactive behaviors called *"Follow corridor"*, *"Go to point"*, *"Escape from minimum"*, *"Border by the right or the left"* and *"Intelligent escape"*. It has been noticed that the observation of the perception applied to obtain the desired control action may be useful to provide behavior execution information to the supervisory control. Usually, the application of a reactive behavior (e.g. *"Follow corridor"*) in an incorrect place (a room, the corridor ends in a wall or disappear in a cross, or maybe the corridor is blocked) leads to incoherent movement, and the supervisory control has to continuously check using perceptual abilities the status of the reactive behavior being used. In this approach, both reactive movement and simple perceptual abilities are joined to simplify supervisory control. Status codes are returned within the function of reactive control, while robot is stopped if necessary. For example if the robot is forced to move in an open space using the behavior *"Follow corridor"*, it does not move at all and a code is returned telling that the robot is not in a corridor. This approach does not try to despise the decision capabilities of a supervisory control, but it is very useful to group both things into a single behavior.

Only two of these behaviors were actually applied at the fair:

- *"Follow corridor"* is a very simple motion pattern that tries to move along a corridor towards a defined direction, carrying out lateral displacements to avoid obstacles. Several previous approaches in the literature try to center the robot in the corridor, but a fair corridor is wide enough to allow traveling along a parallel line to the corridor axis, moving away from corridor borders only if it is too close. Local minima can not be handled by this behavior and a "robot blocked" code is returned if the robot path is not free. It is supposed (and it is almost always true) that the only obstacles at a fair corridor are humans. Supervisory control uses this hypothesis to ask for free way with voice synthesizing, solving the problem. "Corridor end" and "Corridor does not exist" codes are also returned to supervisory control if necessary.
- *"Intelligent escape"* is an useful behavior for making oral presentations as it provides a semi-random movement avoiding minimum and searching open spaces, looking like autonomous intelligent movement achieved by humans while speaking to crowds.

IV. Environment perception and modeling.

A. Environment perception problems.

The new environment presents several perception problems related to the kind of sensors used (sonars) and their horizontal distribution:

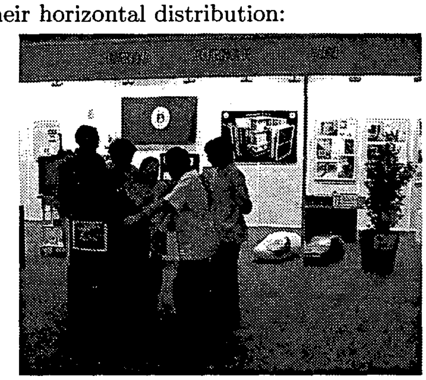

Fig. 2. Perception problems in a fair

• *People.* The attractiveness of the robot invite people to surround it, making it impossible to "see" anything with the sonars but people. Beam divergence and relatively low number of sensors[2], implies that few people are necessary to block completely the sight of the robot. This problem could be partially solved by the use of laser range finders and map based algorithms as in[11], to filter corrupted measures due to people. Nevertheless this solution is not enough to solve the following problem.
• *Environment virtuality.* The stands of a trade fair rarely have physical walls as represented by the fair maps. Instead the stands are areas commonly defined by aluminum structures and/or a step on the floor, a platform, floor colors, hanging posters, furniture, fences, etc. None of these items are visible to the robot sonars, as well as many of the exhibits inside the stands. This is a serious problem as the robot can enter a stand without sensing it and it may collide with any of the above invisible items. Although the problem of invisible obstacles has been handled before[6] for obstacle avoidance, the manual definition of their existence is not practical at all. Many of these objects can be easily moved and a stand can be reconfigured from day to day.
• *Environment size.* The huge size of the environment results that most of the time, the objects that could be a navigational reference (e.g. the back wall of a stand) are out of the sensors range. *Coastal planning*[9] is not a suitable solution as all the "invisible objects" are close to the possible references. Automatic occupancy map building is not possible at all, and manual (remote operation) map building is not practical either.

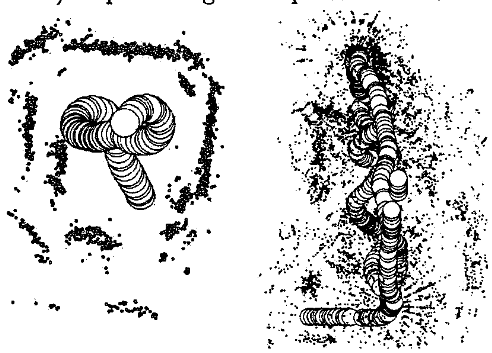

Fig. 3. Lab sonar map (left), fair sonar map (right).

Figure 3 shows the difference between the perception achieved in the lab where obstacles and furniture can be very easily identified and the perception in a fair corridor (over a standard, medium populated experiment), where clouds of dots do not correspond to stands or objects but to immobile groups of people. Any actual object (possible reference) could be identified in any experiment with people.

It is made the assumption that environment perception, map building or localization are not feasible with sonar information.

B. Virtual corridors model.

Since there are big forbidden areas that the robot almost does not "see", a position constraint model is required. Virtual corridors are defined geometrically in a global coordinate system as orientated rectangles corresponding to corridors at the fair or to a possible way in/out of a stand. The robot must only move within these corridors, trusting on a good position estimation. These corridors are theoretically free of objects (if the virtual corridor is correctly defined this assumption is almost always true) and the only obstacles found are people close to the robot. Corridors must intersect completely with other corridors to achieve an easy navigation (fig. 4).

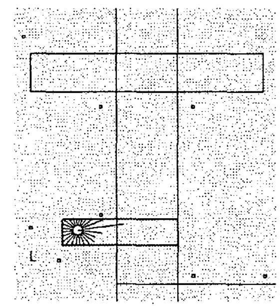

Fig. 4. Virtual corridors map.

C. Simulated environment perception.

Because the robot cannot sense the virtual corridor, the previously described reactive algorithms cannot be directly applied. Instead of using the raw sonar data as the input to reactive algorithms, a simulation is first made. Sonars are modeled as straight half lines and intersections distances with the current corridor borders are computed. The minimum measure from the actual and the simulated one is selected, replacing the original one, making feasible to the robot to "see" both the virtual corridor and the obstacles inside it.

A good robot position estimation is needed to achieve a coherent movement inside the actual corridor and the virtual one simultaneously.

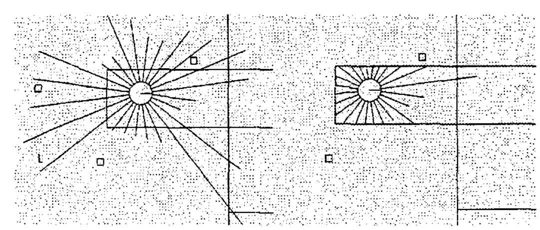

Fig. 5. Actual(left) and simulated(right) sonar perception.

D. *Task planning and execution.*

Task planning and execution using this model is very simple. A complete plan is defined by the corridors and their respective direction by which the robot should move, in a similar way to directions given to a human who wants to go to another stand: "walk along this corridor in this direction, take this other corridor in this direction, etc".

To execute a plan is called continuously the behavior *"follow corridor"*, while monitoring when the following corridor in the defined plan is reached. When the robot enters the intersection between the current corridor and the next one in the plan, the robot changes its current corridor and begins to travel on the new one in the defined direction, and so on.

The execution of the plan is accomplished by the supervisory control. Although automatic task planning has not been implemented yet, and manual predefined plans are used to reach different goals, the planner algorithm is believed to be very simple, just a heuristic search through the corridors tree, minimizing total travel length (sum of corridors length).

V. Localization.

As described above, the environment virtuality requires a good position estimation to obtain correct results. The sonars were not useful at all for localization, so the sensors used for this task were only the encoders and the *LaserNav*. Only continuous localization algorithms were provided, and absolute localization problem was not solved. An initial rough position and orientation estimation must be introduced manually.

It must be considered that the laser target landmarks are located in a horizontal plane at the height of the sensor (120 cm), so they are frequently obstructed by people. It has been roughly estimated that the percentage of time(%) that the robot "sees" 0 targets is 44%, 1 target 25%, 2 targets 28%, 3 targets 3% and 4 or more targets 0%(never), in a medium populated corridor with at least 4 theoretically "visible" targets from any location inside the corridor.

Random noisy measurements are frequent, and can be classified in two groups: angle errors (small gaussian or large non-gaussian errors for an actual target) and identifier errors (actual target defective identification or non existing targets identification). To solve these problems an EKF algorithm with the following characteristics was used:

- As explained before, the robot movement is constrained to 2 dimensions (x-y) while heading movement is theoretically impossible. The matter is that an uncontrolled heading movement is performed very slowly while moving the robot, so the heading accumulates a considerable error after a couple of minutes. This movement is not random at all and the angle error always grows with the same sign. This extra knowledge is applied restricting the filter heading correction to the cases when 3 or more landmarks are seen, because EKF gaussian distribution assumptions (false in this case) yield to erroneous correction of the heading when only one or two landmarks are seen.
- The *a priori* robot position estimation is obtained from the low level controller that integrates encoders measures to compute robot position. This integration is used for the a priori position estimation instead the one computed by a robot kinematics model, because it is much more accurate, despite inherent odometry errors.
- Process (position and heading) noise is estimated to be proportional to the movement performed.
- The observation is composed by a vector of landmark identifiers and the angles measured using the *LaserNav* sensor, and this is the information used to correct the odometry *a priori* position estimation.
- Observed landmarks are compared against a prediction based on the*a priori* robot position estimation and a previously stored landmark position map. Incorrect observations due to bad identifiers or bad angles are rejected. The rest of the measures are subtracted from the predicted ones to obtain the innovation vector and the EKF is applied.

VI. Interacting with people. Voice.

Although a lot of work is to be done is this area, some simple interacting capabilities have been added to the robot. An on board laptop computer was used for voice synthesis, using IBM TTS-ViaVoice (Spanish) libraries.

Pre-defined codes are used for common actions. Thus, when the voice program receives a code, it selects randomly one of its predefined sentences and synthesizes it. There are several codes defined for greetings, welcome messages, self presentations, saying goodbye, etc., but the most useful and commonly used are the "ask for free way" codes. Supervisory control monitors corridor blocking frequency and consequently issues "polite", "insistent" or "insulting" codes, as if the robot changes its mood to face up to annoying people.

Any sentence written by a supervisor in the remote control interface can be synthesized.

Oral presentations based on text files can be synchronized with movement. Supervisory control issues synchronization codes while executing a task, achieving complete guided tours. Presentations are automatically interrupted to ask for free way and resumed afterwards, resulting in a high degree of intelligence and autonomy appearance.

VII. HYBRID CONTROL ARCHITECTURE.

Adopted control architecture corresponds to a hybrid scheme[5], but some special characteristics should be highlighted:

- The low level controller ability of avoiding collisions is typical of Subsumption[3] architecture.
- Reactive behaviors running in parallel with continuous localization is typical of a layered reactive control. These behaviors embed the avoidance collisions ability of the low level controller.
- The supervisory control monitoring reactive behaviors and taking into account the information of the geometric-topological virtual map is the hierarchical module of the architecture.
- The influence of the virtual map over the reactive module, carried out by the sonar simulation, is a non typical information flow due to the virtual nature of our approach.
- Voice synthesis is considered as an actuator, and it is used in two different ways: a reactive one (ask for free way) and a high level one (oral presentations, guided tours and mood change).

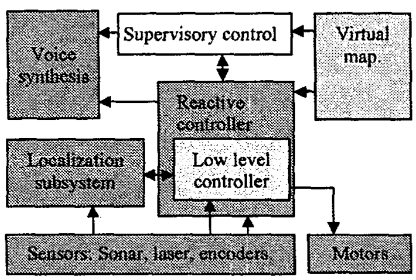

Fig. 6. Control architecture.

VIII. ASYNCHRONOUS MULTITHREAD CONTROL.

The modules of the navigation and control system have been implemented looking for simplicity of use for future users. The low level controller and the laser localization subsystems have their own control thread that is launched, continuously executed, automanaged and stopped at program end, in a transparent way to the user. This mechanism releases the user of having to look after details when building high level controllers and completely encapsulates the functionality that requires continuous execution. The communication is achieved with functions that share variables with the control loop.

The low level controller contains a thread because it implements the regulation of steer position (angle) and drive speed, so continuous execution of a control loop is required to reach desired references. Collision avoidance is built within this loop, so collisions never happen in spite of any applied reference. This thread is also useful for avoiding unnecessary delays when interfacing with the robot due to the relatively slow (250 msec) hardware refresh rate.

The localization laser subsystem also owns a thread that manages communications and status of the *LaserNav* sensor, trying to solve any problem (typically communication disorders) by automatically reinitializing the system. The supervisory control is not informed of any failure until "sensor dead" is confirmed. This thread also continuously executes the localization algorithms: the landmark angles (using *LaserNav*) and the robot position estimation (from low level controller) are read, then the EKF is applied and the robot position is corrected in the low level controller. Thus, reactive behaviors always can obtain the corrected robot position from the low level controller.

The execution of reactive behaviors is controlled by a timer of the main executable. Another secondary timer is used if robot data storing is activated.

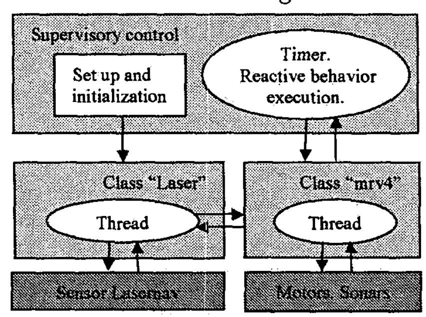

Fig. 7. Multithread implementation.

IX. DISTRIBUTED SOFTWARE APPLICATION.

Three computers have been used for developments:

- The on board control PC (Pentium II running Linux Red Hat 5.1.) runs the low level controller, the laser localization subsystem and communicates through serial port to the laptop computer containing voice synthesis. A sockets server is built on top of these components to provide remote access to them.
- The on board laptop computer (Pentium II running Windows 98) runs the voice synthesis, using the loudspeakers to gain volume, and communicates with the on board control PC by serial port at 9600 bauds, using our own ASCII protocol.
- A remote PC (Pentium II running Windows NT 4.0 SP6) contains a sockets client of the above server that connects to it for many purposes: robot management, reactive behavior execution, data storing, virtual map definition, task definition and execution, graphical interfaces and robot teleoperation. It communicates with the onboard control PC through Wireless Ethernet at 10Mbits/s. The flow of information between the TCP/IP sockets server and client is controlled by our own ASCII protocol.

X. Experiments, results and conclusions.

The robot Blacky has been working successfully in actual environments on 3 occasions, accumulating 7 days of intensive use:

- *Indumatica 2001*, trade fair organized by UPM, in March 2001.
- *Cybertech 2001*, robot contest organized by DISAM in April 2001.
- *Madrid for the Science II*, trade fair organized by IFEMA in May 2001.

Low level controller ability of avoiding collisions has been extremely useful for an easy development and a reliable navigation through crowded environments, but it has been noticed that temporarily suspension of this ability may be useful to find a way through immobile crowds of people.

Simple reactive behaviors with voice synthesis assistance is a good solution for traveling along fair corridors. Two different human behaviors happen when the robot ask for free way: 1)Instantaneous clearing of the way. 2)Robot intelligence testing with continuous path blocking. This test is not a big problem because if the test is not passed, the person would get bored and "give way". A simple reactive behavior and voice synthesizing is a better approach than trying to develop a complex algorithm able to deal with all possible problems. Face orientation is more important than it seems at first glance, because people often clear the way the robot is looking at, even if the robot wishes to move in another direction. Head orientation and facial expression capabilities[10] are considered important for future developments.

Although the continuous localization system has shown an acceptable working, it failed in several occasions and it is the critical part of this approach. Loss of geometric information within the EKF and the difficulty in rejecting noisy laser measurements were the main problems found.

Multithread and distributed software implementation has been running very robustly. Neither system crashes nor program failures have occurred. Communication problems have never happened either. Multithread functionality, transparent for the user encapsulation, is considered to be an effective solution for building and testing new high level controllers with extreme simplicity.

The virtual corridors map and the perception simulation for reactive behavior execution represent a correct solution to the problems found in a fair. But it is considered that lack of perception problem should not be counteracted with a geometric position estimation and model acceptance. The size of the environment is a problem for landmark placing and mapping. Landmarks as a line in the middle of the corridors are considered to be conceptually more correct as they physically mark the corridor, so the robot is able to "see" it, instead of computing its position in it. Possible errors are eliminated, as reference line placing is immediate and manual mapping is not required as can be done automatically.

The friendliness and personality of *Blacky* have gained high attention and reached great success. Change of the robots mood due to the persistent path blocking is very funny, the more angry the robot becomes the more funny it is. The robot is also an excellent advertisement distributor. People always obey robot orders so they will be delighted to take any item (letter, paper, advertisement, etc.) if asked by the robot to do so. It has been enjoyed the chance to observe people interacting with the robot, and fun has been part of daily work.

Future research will be focused in increasing perception capabilities through the use of artificial vision and laser range finders. Three dimensional perception will be used. We would like to advance in automatic artificial and natural landmarks identification and mapping in big size populated environments, and new localization systems will be studied. We also pretend to expand the interaction abilities of the robot as a main goal to the overall success of the system.

References

[1] G. Bishop, G. Welch, *An Introduction to the Kalman Filter.*, Course 8 Presented at ACM SIGGRAPH 2001.

[2] J. Borenstein, B. Everett, and L. Feng, *Navigating Mobile Robots: Systems and Techniques.*, A. K. Peters, Ltd, Wellesley, MA, 1996.

[3] R.A. Brooks, *A Robust Layered Control System for a Mobile Robot.*, IEEE J. Robotics&Automation. Vol2(1986).

[4] W. Burgard, A.B. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. *The interactive museum tour-guide robot.*, 15th National Conference on Artificial Intelligence, Madison, Wisconsin, July 1998.

[5] J.L.Crowley. *Mathematical Foundations of Navigation and Perception for an Autonomous Mobile Robot.* In Proceedings of the RUR, Amsterdam, Dec. 1995

[6] D. Fox, W. Burgard, S. Thrun, and A.B. Cremers. *A Hybrid Collision Avoidance Method For Mobile Robots.* IEEE Inter. Conf. on Robotics and Automation, 1998

[7] D. Gachet, M.A. Salichs, J.R. Pimentel, L. Moreno, A. De la Escalera, *A software Architecture for Behavioral Control Strategies of Autonomous Systems.*, Proc. IECON 1992

[8] I. Nourbakhsh, J. Bobenage, S. Grange, R. Lutz, R. Meyer, A. Soto. *An Affective Mobile Educator with a Full-time Job.*, Artificial Intelligence, Vol. 114, No. 1 - 2, Oct,99.

[9] N. Roy, W. Burgard, D. Fox, and S. Thrun, *Coastal Navigation - Mobile Robot Navigation with Uncertainty in Dynamic Environments.*, Proc. ICRA'99, May, 1999.

[10] N. Roy, G. Baltus, D. Fox, F. Gemperle, J. Goetz, T. Hirsch, D. Margaritis, M. Montemerlo, J. Pineau, J. Schulte, and S. Thrun, *Towards Personal Service Robots for the Elderly.*, Workshop on Interactive Robots and Entertainment (WIRE 2000), 2000.

[11] S. Thrun, M. Bennewitz, W. Burgard, F. Dellaert, D. Fox, D. Haehnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. *MINERVA: A second generation mobile tour-guide robot.*, Proc. ICRA'99, May, 1999.