BARBANETA – A MODULAR AUTONOMOUS VEHICLE [1]

Pedro Aparício, João Ferreira, Pedro Raposo, Pedro Lima

Instituto Superior Técnico/Instituto de Sistemas e Robótica,
Av. Rovisco Pais – 1, 1096 Lisboa Codex
PORTUGAL
email: jpp@isr.ist.utl.pt

Abstract:
This paper presents a small autonomous mobile robot designed and implemented by the authors. The vehicle is modular, in the sense that new functionalities may be added if needed. Therefore, it can be used as a testbed for research and development on autonomous mobile robots. An open architecture was designed and implemented to allow the integration of the different modules. Three modules were implemented in the current stage: motor controllers, guidance control loop and ball catcher/selector. These allow the vehicle to follow an optical track and recover the track if it is suddenly interrupted (using retroreflectors), while collecting and discriminating billiard balls of different colors placed along the path.

Keywords: Architectures, Guidance Systems, Sensor Fusion, Vision.

1. INTRODUCTION

New sensors, actuators and control algorithms for mobile robots are introduced every day (Jones and Flynn, 1993) (McComb, 1987) (Everett, 1995). Design methodologies and vehicle architectures are therefore needed to integrate the sensing, control and actuation functions, as well as to compare different solutions for the same application.

This paper presents the design and implementation of a small, autonomous, modular mobile platform. The platform was designed to be used as a testbed for motor control, vehicle guidance and localization methods for mobile robots. Currently, the vehicle has the ability to follow a stripe painted on the floor and recover from a path loss. It can also collect and select billiard balls of different colors, placed on the floor along the track. The design process included the development of a mechanical structure, electronic solutions for speed controllers and motor drivers, image pro-

Fig. 1. Barbaneta in action

[1] This work has been supported by the following entities: Alcatel, EDP, Celcat, Renault, French Embassy, CD/IST, DEEC/IST, DEEM/IST, JNICT FACC - PROC 435.40.50 9/98/6/0226.

cessing software, guidance algorithms and a ball catcher/selector.

The platform has an on-board PC motherboard, with a i486 processor, a floppy drive and a video card used for software debugging. Three microcontrollers (PIC16C74) are also used. Two of them interface the i486 processor through the motherboard racks/ISA bus. Hardware expansion is possible by adding independent modules or new interfaces with the PC bus.

The vehicle competed in the 1996 edition of the "*Festival des Sciences et Technologies*" mobile robot contest, held annually at La-Ferté Bernard – France.

2. MECHANICAL STRUCTURE AND KINEMATICS

The mechanical structure is divided in two main blocks: the chassis and the transmission.

Chassis - The platform chassis is made of aluminum and was designed with modularity in mind, *i.e.*, the chassis provides a solid support for the traction module (motors), for the electronic boards, for the batteries and for any other item to be added in the future. The chassis is 67 centimeters long by 40 centimeters wide and 8 centimeters tall.

Transmission - The vehicle transmission is based on a differential drive system, consisting of two 12 V (6 Watt) DC motors, one per wheel. Each of them has a gearbox that delivers 40 rpm, 1.8 Nm torque. Mechanical compliances were inserted between the gearbox output shaft and the wheels. This avoids damage to the gearbox in case of axis misalignment. To obtain an angular wheel speed of 60 rpm, a 2:3 belt drive links the gearbox output shaft and the wheel.

Figure 1 presents the vehicle in its current stage.

Kinematic Structure - The vehicle has a differential drive structure. This leads to the following kinematic equations: (Borenstein and Everett, 1996)

$$\begin{cases} v(t) = \dfrac{1}{2} \cdot (v_{left}(t) + v_{right}(t)), \\ \\ \dot{\theta}(t) = \dfrac{1}{\ell} \cdot (v_{left}(t) - v_{right}(t)) \end{cases} \quad (1)$$

where $v(t)$ is the vehicle linear velocity along its longitudinal axis at time t, $\dot{\theta}$ is the vehicle angular velocity, v_{left} and v_{right} are the linear velocities of the left and right wheel, respectively, and ℓ is the distance between wheels (49 cm).

3. POWER SUPPLY

The vehicle power supply is divided in two blocks: the signal power (electronics) and the motor power. This is required to avoid spikes introduced by the motors in the power lines (McComb, 1987). These spikes, although short in time, have severe consequences in the signal electronics, leading to malfunctions.

The electronics power supply comes from two 12 V NiCad battery packs assembled such that +12 V, GND and -12 V are accessible. Note that these packs draw 1.7 Ah, which gives about 30 minutes autonomy if 3 A are drawn from the batteries in a continuous mode operation. To get 5 V and -5 V for the electronics boards, voltage regulators were designed and built.

For the motors power supply, only +12 V are required, because the motor PWM servo amplifiers are based on H-bridges. The power supply for this module comes from a 12 V, 1.5 Ah NiCad battery pack. NiCad batteries were used due to their good size/autonomy rate.

To provide a +5 V voltage for the motor drivers, a 7805 regulator is used. When the motors are stalled, the surged current tends to rise and overcome its nominal values. To avoid that, current limiters were also designed and built, limiting the motor current to 500 mA per motor.

4. MOTOR CONTROLLERS

Closed loop motor velocity control is accomplished by coupling encoders to the motor shafts and processing the velocity error signal by a PI (proportional-integral) controller. Two motor controllers are required because a differential drive solution is used. Both of them are implemented in the same microcontroller PIC16C74.

For control purposes, digital controllers have significant advantages over the analog ones, namely: higher gains can be used; reliability is higher; they are easy to interface with microprocessors; several controller types can be tested in order to have best performance. On the other hand, the adjustment of the sampling period is critical. The proposed control scheme is presented in Figure 2.

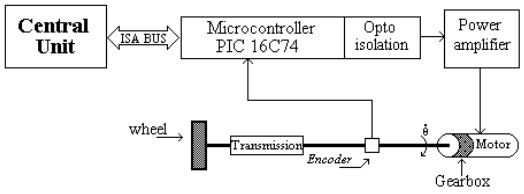

Fig. 2. Speed controllers

Each controller input is the difference e between the desired and actual wheel velocities, it i.e., $e = v_d - v$.

The actual wheel velocities are proportional to the number of encoders pulses in a given time interval.

The controller mathematical model (Aström, 1984) is the following:

$$u(t) = k_p \cdot e(t) + T_s \cdot k_i \cdot \sum_{k=0}^{t} e(k), \qquad (2)$$

where T_s represents the sampling period, t is a sampling time, k_i the integral gain and k_p the proportional gain.

Expression (2) is implemented in the micro-controller. The controller gains were determined experimentally.

Note that, when using digital controllers, one must assure that all variables stay in the boundaries imposed by the digital representation in the micro-controller. To avoid overlapping and saturation, an antiwindup scheme was implemented in the controller.

5. GUIDANCE AND RELATIVE LOCALIZATION SYSTEMS

In this particular application, the developed vehicle is equipped with two types of guidance systems (Everett, 1995): a track follower and a passive beacon follower. Both sensing devices and guidance algorithms were designed to cope with the rules of the contest where the robot competed.

The two sensors are described in the first two subsections. Subsection 5.3 presents the guidance algorithm.

5.1 *Track Sensor*

The track follower is a module that gives information on the vehicle deviation relative to a path represented by a stripe painted on the floor below the platform. A typical path is presented in Figure 3.

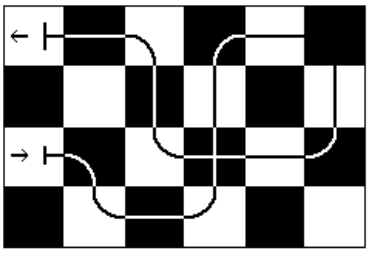

Fig. 3. Typical path

Infrared Sensors (IR) were used do detect the painted stripe. These were designed to behave as digital sensors so that when a sensor is above white floor, the output is a digital one. A digital zero appears in the output when the sensor is above a black floor. These sensors were first presented in (Fernandes *et al.*, 1996).

An infrared sensor line, shown at the top of Figure 5, was designed to measure the deviation of the vehicle longitudinal axis with respect to the track, at each sampling instant. This line is composed of 16 infrared emitter/receiver pairs of LEDs and covers a 36cm line. A digital word (16 bit) is read from the receivers and processed by a PIC microcontroller. Processing includes noise filtering and computation of the actual deviation (in cm) from the desired trajectory. When the central i486 processor requests new data, it sends a request signal to the path sensor module and then reads the deviation value.

5.2 *Beacon Sensor*

In the contest where the robot was presented, the track was sometimes interrupted. Under such a situation, there was a passive beacon (retroreflector) above the place were the path resumed.

This sensor is based on computer vision. It uses a low-cost digital camera (324 x 242 pixels) interfacing the PC board through the ISA bus. The camera is placed at the same height of the retroreflector. The main advantage of using such a camera is its low cost. The main disadvantage is the slow image transfer to memory.

Image processing comprehends thresholding, edge detection, and mass center computation (Jain, 1989). Only a small part of the image (which contains the beacon) is processed. After being acquired, the image is thresholded to extract the relevant information. With the correct exposure time and diaphragm aperture, the beacon appears as a very light object in a dark scene. From the thresholded image, edges are detected to extract the beacon boundaries. Edge detection is done both in horizontal and vertical directions, but only edges with a slope of the corresponding straight line higher than a given value are used, to reduce image noise. After edge detection, the mass center of the edge image is computed. The difference between the center of the image and the x coordinate of the mass center is proportional to the deviation of the vehicle axis from the position of the beacon mass center, scaled by an appropriate factor which reflects the perspective transformation. This deviation is the input of the guidance controller.

5.3 *Guidance Control Loop*

The guidance strategy favors a constant common mode velocity (v_{cm}). Based on the vehicle's differential drive kinematic structure, common mode velocity can be computed from the wheel velocities as:

$$v_{cm} = \frac{v_{left} + v_{right}}{2}, \tag{3}$$

where v_{left} and v_{right} represent the left and right wheel linear speed, respectively.

At time t, the speed setpoints for the right and left motors are given by:

$$\begin{cases} v_{left\_d}(t) = v_{cm} + \delta v(t), \\ v_{right\_d}(t) = v_{cm} - \delta v(t) \end{cases} \tag{4}$$

where $\delta v(t)$ is a correction factor which ensures that v_{cm} is kept constant. If one makes

$$\delta v(t) = \frac{\ell f(d(t))}{2}, \tag{5}$$

where $d(t)$ represents the measured distance between the vehicle longitudinal axis and the track (or the beacon mass center) at time t, the vehicle kinematics (1) leads to

$$v = v_{cm} \quad and \quad \dot{\theta} = f(d(t)), \tag{6}$$

where $f(d(t))$ is some control function of the deviation. For instance, if one uses a PD control law

$$f(d(t)) = k_p \cdot d(t) + k_d \cdot (d(t) - d(t - t_s)), \tag{7}$$

the error dynamics is such that, when $t \to \infty$, the deviation over straight lines ($\dot{\theta} = 0$) goes to 0 at a rate dependent on k_p and k_d. Analogously, the deviation over curves ($\dot{\theta}$ = constant) tends to a constant value, dependent on the curve radius.

The sampling time t_s of the guidance control loop is usually made $t_s = nT_s$, with $n = 4$ or 5, where T_s is the sampling time of the motors control loop. This is a rule of thumb to ensure that the motors have enough time to settle at the reference velocity required by the guidance controller. When using the track sensor, $d(t)$ is a discrete measure of the distance between the track and the vehicle longitudinal axis, as shown in Figure 4. If $d(t)$ is small, it is a good approximation to the orientation of the platform with respect to the track, i.e., $d \simeq \beta$ in the figure. The problem is that, due to the sensor quantization error, the vehicle must have an orientation error $d(t)$ corresponding to approximately 4 degrees to detect a misalignment with the track. Therefore, small oscillations around the path may occur.

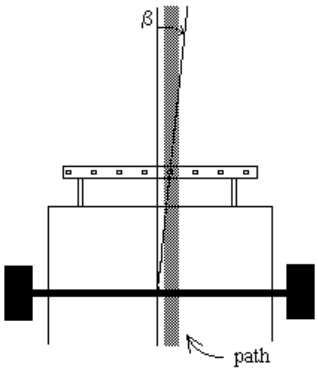

Fig. 4. Angle measure

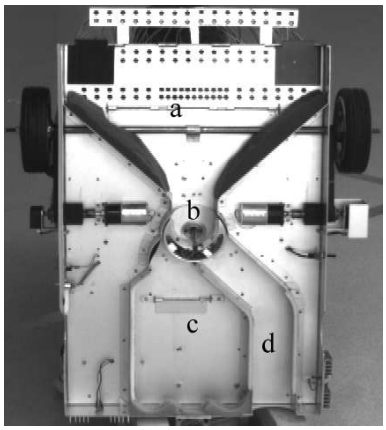

Fig. 5. Vehicle Bottom

6. BALL CATCHER/SELECTOR

The ball catcher/selector was specifically designed for this competition. This module catches billiard balls placed along the track and rejects some of them, depending on their color. Our solution takes advantage of the forward motion of the vehicle.

A bottom view of the platform can be seen in Figure 5. In this figure, the four main sections of the ball catcher/selector are shown: a) ball entrance, b) ball selector, c) container and d) ball exit corridor.

The ball entrance directs the balls to the ball selector. The ball selector is composed of an hollow cylinder, with an opening that allows the balls entrance. The cylinder can hold a billiard ball inside, it is suspended from the top and may be rotated in both directions. The ball container is an area under the vehicle, designed such that the catched balls get in but do not get out.

The central unit of this module is a PIC16C74 micro-controller, which does the sensor readings and the stepper actuation. IR LEDs to determine the ball colors were installed inside the cylinder. There are also four micro-switches in the system, used to calibrate the stepper initial position and limit its excursion, as well as to detect ball entrances. When a ball enters the ball selector,

its color is examined, and, based on the result, the cylinder rotates towards the container or the exit corridor, resuming the original position afterwards.

In the competition, black balls should be rejected and red balls should be collected. The system showed an almost 100% discrimination rate.

7. ARCHITECTURE

The main goal of the platform system architecture is modularity *i.e.*, module insertion/deletion should be easy to do.

The vehicle has an open architecture that favors hierarchically distributed control solutions, e.g., the "low level" motor velocity controllers and track sensor processing are autonomous and execute their task in parallel and also in parallel with the "higher level" guidance controller.

Figure 6 shows the functional and hardware architectures of the platform for the described application, pointing out the distribution of the functional modules by the hardware modules. Communications between different modules are done through the ISA bus of the PC motherboard.

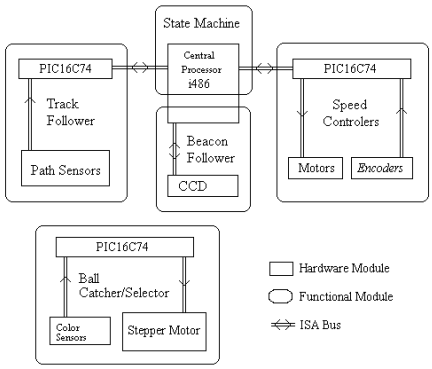

Fig. 6. Functional *vs* Hardware architecture

Note that the ball catcher/selector is independent and therefore works in parallel with the others (it runs on one independent processor).

The state machine which coordinates the vehicle function execution is depicted in Figure 7.

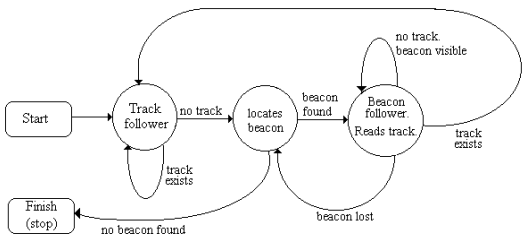

Fig. 7. Control state machine

When the vehicle is turned on, it starts in the track following state. While there is a track to follow, it stays in that state. If the track is lost, the beacon search state is started. Once the beacon is found the vehicle moves towards it until the track is recovered or the beacon is lost. In that case the beacon search module is re-started. After track recovery, the track following state is re-started. If the beacon is not found the vehicle stops and suspends all operations.

8. RESULTS

To make the error correction more effective for high orientation errors (when d is no longer $\simeq \beta$), a "non-linear" PD controller was implemented in the guidance control loop:

$$\delta_v(t) = k_p \cdot sgn(d(t)) \cdot \sqrt{|d(t)|} + k_d \cdot (d(t) - d(t - t_s)), \quad (8)$$

When following a straight line, the vehicle deviation from the track is plotted in Figure 8. It can be seen that the vehicle shows a high frequency oscillation and a low frequency oscillation around a zero mean deviation. The high frequency oscillations are due to the quantization step of the track sensor, explained in Section 5, while the guidance controller is responsabile for the low frequency oscillations.

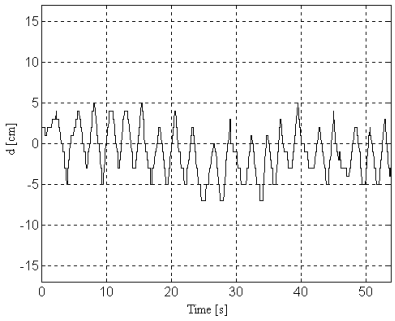

Fig. 8. Results obtained on a straight line

Figure 9 shows the vehicle deviation when it is moving along a curve. The same small oscillation around the nominal trajectory is observed, but now the mean deviation is approximately 5 cm, as expected from the explanation in Section 5.

Figure 10 presents results obtained when the vehicle follows the retroreflector, after losing the track.

Note that the same guidance controller was used for track and beacon following. The results show that the vehicle deviation from the setpoint (track or beacon mass center) is smaller when a camera is used as the sensor (beacon follower). This is due to the almost zero quantization error in this case, if compared to the 16 IR line track sensor.

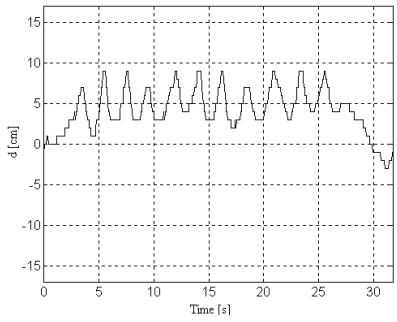

Fig. 9. Results obtained on a curve

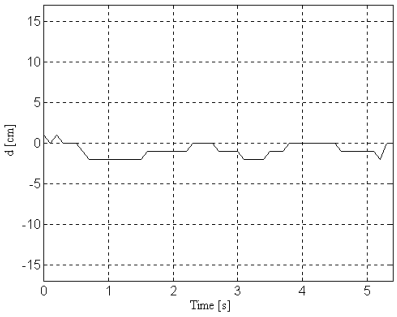

Fig. 10. Results obtained when heading to the retroreflector

9. CONCLUSIONS

This paper presented the design and implementation of a small autonomous mobile platform, with an open control architecture that integrates the different functional and hardware modules. The platform was developed to be used as a testbed for research and development on mobile robotics. In the application described here, track follower, beacon follower, and ball catcher/selector modules were added to the original mechanical chassis and motor velocity control loop. From the results obtained, it can be concluded that digital IR track sensors have the disadvantage of a considerable quantization error, despite their simplicity and low price. Currently, the vehicle is under an upgrade process which will replace the IR sensors by a CCD camera for track following. Relatively low-priced digital CCD cameras can be found in the market, with the advantages of small quantization error and improved information on the localization of the vehicle with respect to the track: not only the deviation, but also the relative orientation can be computed using such a sensor, at the cost of a small increase in sampling time.

10. REFERENCES

Aström, Karl J. (1984). *Computer Controlled Systems.* Prentice-Hall. London.

Borenstein, J. and H.R. Everett (1996). *Where am I? – Sensors and Methods for Mobile Robot Positionig.* University of Michigan. Michigan.

Everett, H.R. (1995). *Sensors for Mobile Robots.* AK Peters. Massachusserts.

Fernandes, Dinis, Luis Farrolas, Pedro Brito and Pedro Lima (1996). Progress on the design of a small flexible automated guided vehicle. In: *Proc. of CONTROLO 96, 2nd Portuguese Conference on Automatic Control.* Vol. 1. pp. 283–289.

Jain, Anil K. (1989). *Fundamentls of Digital Image Processing.* Prentice-Hall. London.

Jones, Joseph L. and Anita M. Flynn (1993). *Mobile Robots – Inspiration to Implementation.* AK Peters. Massachusserts.

McComb, Gordon (1987). *The Robot Builders Bonanza.* TAB Books - McGraw-Hill. New York.