Managing Limited Sensing Resources for Mobile Robots Obstacle Avoidance

Juan Carlos Alvarez, Rafael C. Gonzalez, Diego Alvarez & Antonio M. Lopez

1. Introduction

Obstacle avoidance is a sensor-based task designed to steer a mobile robot towards intermediate goals while avoiding unexpected local obstacles in the robot's surroundings. Responsiveness requires that the task is computed in short time intervals, fast enough to deal with the robot inertia and dynamics limitations. Simultaneously it has to guarantee detection by gathering enough information to allow the robot to react adequately to any potential danger.

However, throughput and detection guarantee are opposite demands. If we try to assure the latter by augmenting the scanned area in the robot's surroundings, it will increase the time taken to process the sensor data, proportionally decreasing the reactivity to the external world. On the other hand, scanning smaller areas can endanger the robot if some possible trajectories of motion are not swept by the sensors (e.g. the tunnel–vision problem).

The previous trade-off can be addressed in different ways. Some obstacle avoidance algorithms are designed supposing that fixed sensorial information of the robot's surroundings will be available when needed, e.g. range measures in a circular or rectangular robot-centered area (Lumelsky and Skewis, 1990), (Ulrich and Borenstein, 2001). Furthermore, others restrict the required information to the limits imposed by the robot's actual motion, its dynamics and kinematics (Brock and Khatib, 2000).

But the assumption that certain sensorial information will be available when required is not always realistic. For example, the particular field of view of conventional range sensors such as stereovision, sonar or laser, cannot adjust adequately to the specific perception requirements (Laubach and Burdick, 2000). Or the time needed for sensor data processing can have an impact on the real-time motion performance, making some sense modalities more recommendable than others (Kelly and Stentz, 1998). Finally, the availability of each specific sensing device can be an issue whenever the robot's task solicitations exceed the robot's computing capabilities, for example in robots with complex missions or with massive sensorial information. For these cases, treating sensing as an isolated phenomena leads to bottlenecks both oinn computational demand and in real-time performance.

Our approach to the problem of making the required information available when needed is based on two ideas: 1) to define what the (minimum) information required is in order to guarantee the safety of motion for a given motion strategy, and 2) to use sensor management to obtain only the required info. Scanning only the essential area around the robot helps us to adapt to the real time demands about motion reactivity, and managing

Source: Cutting Edge Robotics, ISBN 3-86611-038-3, pp. 784, ARS/plV, Germany, July 2005 Edited by: Kordic, V.; Lazinica, A. & Merdan, M.

the sensor allows us to obtain only the required information, shortening the processing time, by combining the actual usable sensors. The proposed solution has been applied to a real robot motion problem, whose formal formulation is discussed in Section 2. Section 3 is devoted to the analysis of the sensor information requirements for a given robot and motion control policy, in relation to motion safety and performance. Then, and for a given sensor model, we design a sensor management strategy able to gather the required information. In Section 4 we present experiments designed to focus attention on specific points of the problem. Experimental results in realistic environments, in Section 5, show the feasibility of the proposed strategy, allowing a mobile robot to move at high average speeds and with high reactivity (ten cycles per second).

2. Problem Definition

The problem stated in the previous section has two main components: the robot and sensors capabilities and the intended motion control strategy. In the following section both aspects are explained, leading to a specific problem formulation.

2.1 Definitions: Robot and Sensors

We address the problem of a robot moving in a planar Euclidean environment. The robot has a circular shape with radius rr, and its configuration is represented by the coordinates of its center $C_i = (x, y)$ and its orientation θ, relative to a fixed coordinate system. The robot orientation θ is collinear with its instantaneous velocity vector.
The robot is equipped with range sensors, which can sweep an area in front of it called Field of Regard, see Figure 1-left. It is defined by the aperture angle (field of view, ρ), and its maximum range (depth of field, d_f). The sensor is able to pan, represented by a rotation angle γ relative to the mobile frame. The "frontal swept line" (FSL) is chosen to be perpendicular and symmetrical with respect to the mobile frame, and it is defined by parameters (d_f, ρ, γ). This model is applicable for commonly used sensors such as ultrasonic and stereovision cameras. Rotating ultrasonic sensors and stereovision systems are usually mounted over pan and tilt devices permitting the panning operation; for fixed rings of ultrasonic sensors we have a "discrete panning" result of firing only a certain group of sensors.
Notice that, depending on the robot's motion policy and the sensor field of regard, possible motion trajectories are not swept by the sensors, see Figure 1-right. This is known as a tunnel–vision problem, and from that point of view, we are interested in the real-time selection of these three parameters (d_f , ρ, γ) in order to avoid such danger without increasing the sensing effort, while maintaining good reactivity and motion performance.

2.2 Safety, Control Policy and Robot Dynamics

Two robot characteristics will affect the design of the sensing strategy: its mobility limitations and its motion control strategy. Mobility limitations come from the robot's mechanical configuration, dynamics, and other physical limits. Usual steering mechanisms impose a nonholonomic restriction,

$$x = v \cos\theta \qquad y = v \sin\theta \qquad \theta = \omega \tag{1}$$

with v(t) being the robot's forward velocity, and $\omega(t)$ its rate of change in orientation or turn velocity. Robot actuator dynamics limit robot velocities (v, ω) both in magnitude [vM, ωM] and in rate of change or acceleration, [aM, αM]. A convenient model for robots with a synchronous steering mechanism is (Alvarez, et al., 2001):

$$v + kpv = uv \qquad \omega + kh\omega = u\omega \tag{2}$$

kp and kh are constants and (uv, uω) velocity references to the robot actuators. The robot's trajectory for a given control command (uv, uω) can be computed by integrating equations (2) and (1).

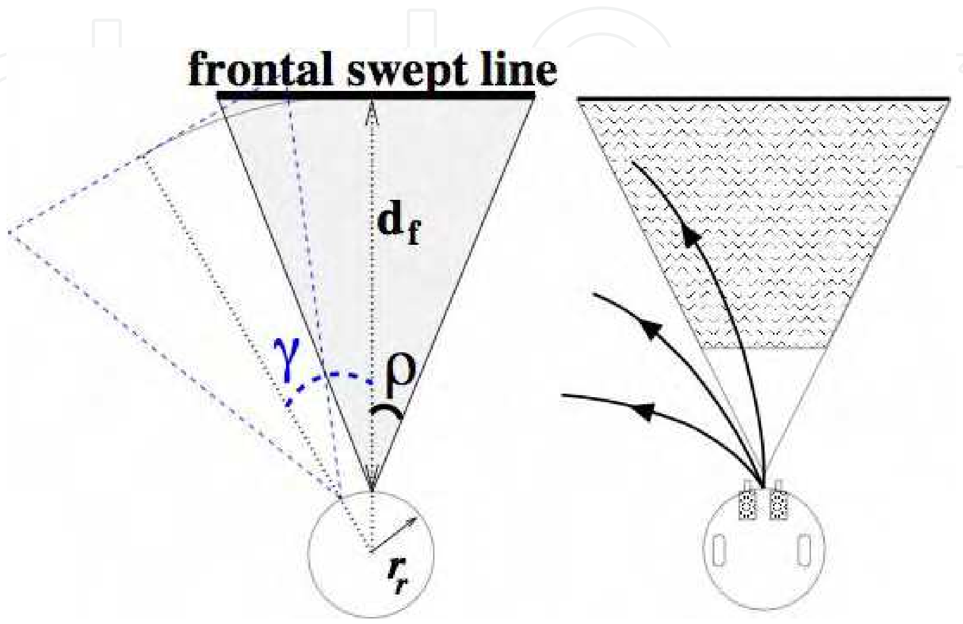


Figure 1. Problem formulation. (Left) The sensor Field of Regard is defined by three parameters: the field of view ρ, the depth of field d_f , and the field orientation γ. (Right) Tunnel vision problem: a fixed Field of Regard can be insufficient to sweep all the possible robot trajectories, with the corresponding safety risk

The motion control strategy is responsible for how the references (u_v, u_ω) are selected from an initial robot position C_i and state (v_0, ω_0), in order to reach an intermediate goal Ti with some optimal criteria (such as minimum time). We are supposing that robot motion decisions are computed in short and fixed time periods T_{cyc}. At every period T_{cyc} a new intermediate goal to reach T_i is provided to the robot (we can think of T_i as the result of a motion planning module). For instance, a reasonable strategy for an "emergency stop" is (u_v, u_ω) = (0, 0). For the rest of the operations we will assume that controls are obtained with a "maximum turn" strategy:

$$u_v = K_{vdobs} \qquad u_\omega = K_\omega(T_i - C_i) \tag{3}$$

with constants K_v and K_ω tuned to maximize velocities, and d_{obs} the sensed distance to the closest obstacle in the intended robot trajectory.

2.3 Overcoming Limited Sensing with Sensor Management

Sensor Management deals with multisensor systems operating in real time, and "how to manage, co-ordinate and integrate the sensor usage to accomplish specific and often dynamic mission objectives" (Luo, et al., 2002). It includes the individual control of each sensor such as direction, pointing, change in frequency, power level, etc. The goal is to reduce unnecessary use of sensors through adequate use of the available resources.

The expected benefit, in our problem, is to be able to deliver the required information when needed (just-in-time). Precisely, the sensor field of regard will be extended in order to maintain motion performance with safety guarantee, by means of a sensor selection strategy. In order to minimize time data processing, such strategy will depend on the actual robot motion conditions at every moment. The solution will be the outcome of an analysis aimed to calculate the minimum sensorial information (field of regard size) needed to assure safety with the proposed control policy.

3. Strategies for Just-In-Time Obstacle Detection

In this section the algorithmic foundation of the proposed solution will be explained. It leads to the definition of a "virtual sensor" able to fulfill the requirements of obstacle avoidance in real time. The detail of the following mathematical analysis has been reported in (Alvarez, et al., 2002).

3.1 Motion at Constant Velocities

Let us consider the safety of motion at constant velocities, that is, for straight and circular motion. When the robot moves along a straight line, the relation between sensor depth and aperture (d_f , ρ) has to be sufficient to guarantee its safety. That is accomplished if the whole area traversed by the robot is previously swept by the FSL. In other words, for a circular robot of radius r_r and a sensor of aperture ρ, the field of regard depth satisfies:

$$d_{f1} = r_r / \tan\rho \tag{4}$$

If d_f is smaller than d_{f1}, safety is not guaranteed. If it is greater, obstacles out of the robot's path will be detected as dangerous, and control (3) will cause an unnecessary reduction of velocity.

A second condition is that sensor field depth has to be large enough to let the robot stop safely –fast enough– if an obstacle is detected in front of it. This distance will be greater as the robot's inertia is increased. Let us call an "emergency stop" the maneuver aimed to completely stop the robot, whatever its initial state (v, ω) = (v_0, ω_0) is. Such maneuver will be implemented by sending the command references (u_v, u_ω) = (0, 0).

The distance traversed before the control command halts the robot, rd, depends on its dynamics and the response time of the robot's control system, t_r,

$$r_d = v_0 t_r + x(\infty) = v_0 t_r + v_0 / k_p \tag{5}$$

being $x(\infty) = v_0/k_p$ the consequence of dynamics, calculated by integrating equations (2) and (1) (Alvarez, et al., 2001). This magnitude establishes a lower limit to the depth of field, $d_f \geq {}_{rd}$, which depends on the robot current status v_0. The worst case analysis $v_0 = v_{max}$ allows us to compute, off–line, a secure distance to stop, for every initial state $(v_0, \omega = 0)_k$:

$$d_{f2} \geq r_{dmax} \tag{6}$$

In conclusion, when moving in a straight line, robot safety and smooth motion is guaranteed by equations (4) and (6). This analysis is condensed in Figure 2.

For constant velocity references (u_v, u_ω) the robot moves along a circular trajectory of radius $r_g = u_v/u_\omega$. following the model defined by (1) and (2). Thus, for a sensor field of view fulfilling conditions (4) and (6), we can identify potentially risky situations, see Figure 3.

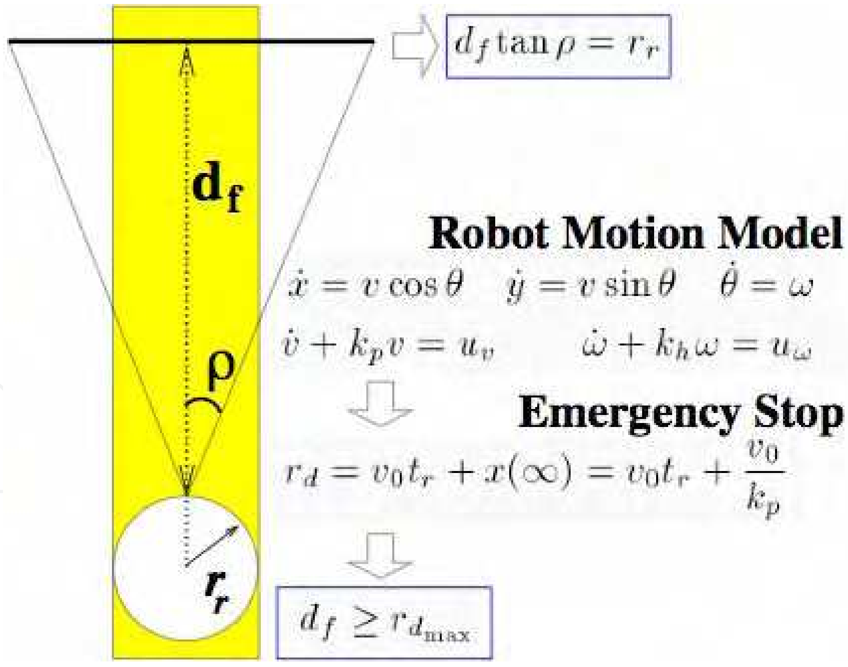


Figure 2. Minimum sensory requirements when a robot moves in straight line. They take into account the robot's dimensions, equation (6), and its dynamics, equation (9)

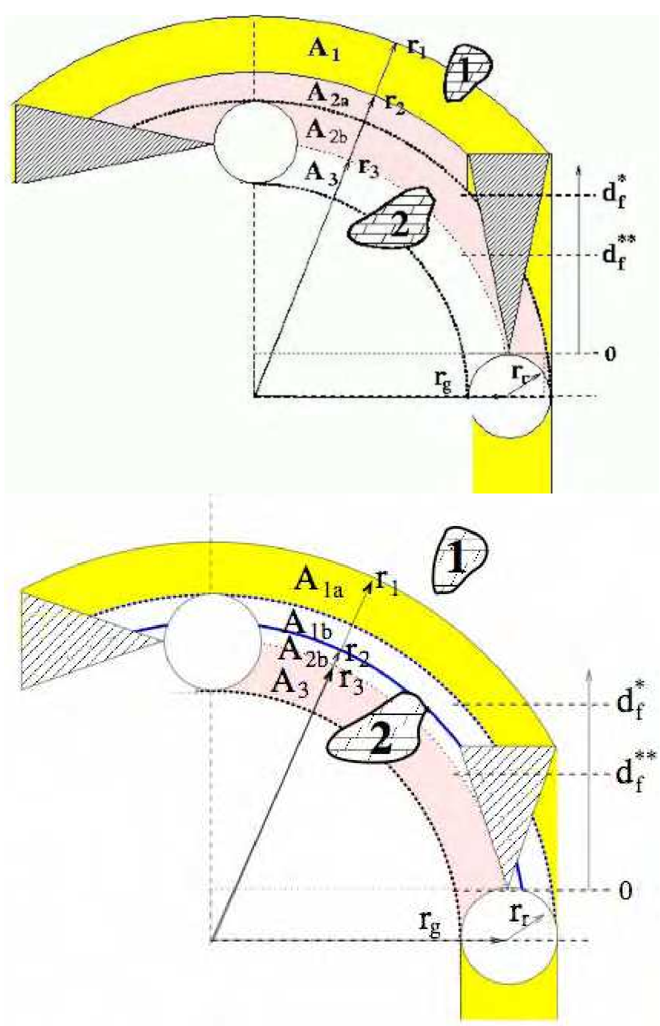


Figure 3. Robot turning with constant velocities. Its sensor Field of Regard fulfills conditions (4) and (6). (Left) Case when $d_f \succ d_f^*$ and the tunnel vision problem: the robot moves where the sensors did not reach. (Right) For $d_f^{**} \succ d_f \succ d_f^*$ the situation is better, but still there are blind zones

The FSL swept action causes three different situations: a zone A_1 of "unnecessary deceleration", a zone of "lateral detection" comprised between sectors of radius r_3 and r_2 (A_2), and a "blind zone" A_3, where obstacles are not detected.
All of them have to be eliminated or reduced as much as possible, setting new conditions on the selection of the sensor field depth d_f . The objective is to sweep in advance with the FSL the exact zone that will be traversed by the robot, as we did in straight-line motion. Notice in Figure 3 that smaller depths d_f reduce them, in opposition to the demand of larger depths–at least fulfilling condition (6)–that increases the options to react to unexpected obstacles. This trade-off depends on two factors (Alvarez, et al., 2002):

1) There is a certain limit depth, d_f^*, which eliminates lateral detection of non-dangerous obstacles (zone A_{2a} in Figure 3-right),

$$d_f^* = 2\sqrt{(r_{rrg})} - r_r \tag{7}$$

This value d_f^* is an upper limit to d_f .

2) Another limit value exists d_f^{**} such as smaller depths $d_f \le d_f^{**}$ completely eliminates lateral detection (zone A_2),

$$d_f^{**} = \sqrt{2}\sqrt{(r_{rrg)}} - r_r \tag{8}$$

A further reduction of blind zones A_3 is only possible by choosing $d_f < d_f^{**}$.
As illustration Figure 4 shows an experiment with a Nomad–200 where an obstacle in a "lateral detection" A_2 zone appears abruptly in the robot field of view. It triggers an emergency stop condition because the robot dynamics do not allow any other avoiding maneuver, and the overall motion performance is negatively affected (Alvarez, et al., 2001).

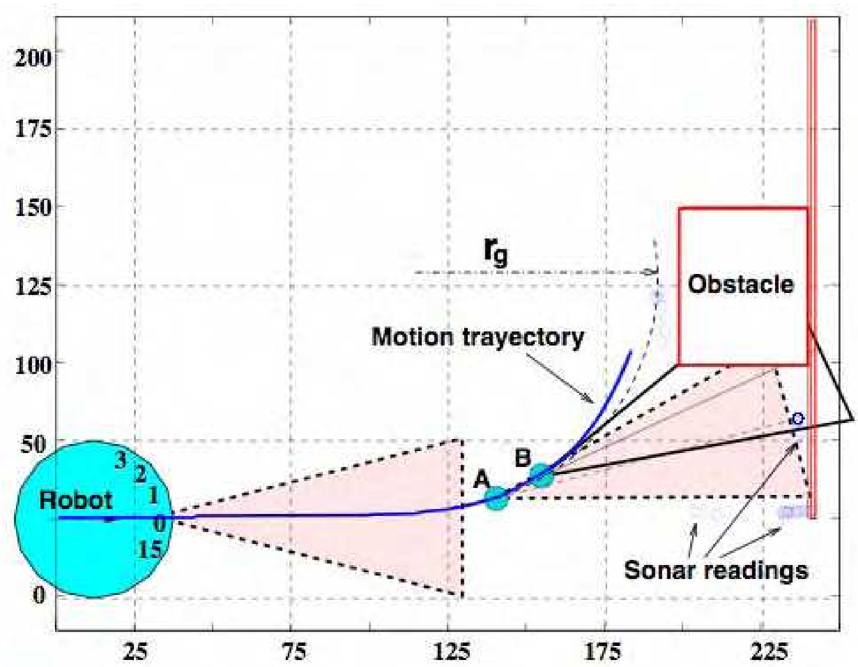


Figure 4. Stop maneuver triggered by an "A_2 zone" obstacle in a real experiment; top view. From point A to point B the robot is turning at maximum speed, and the sensor is measuring a maximum distance d_f ; in B the obstacle corner comes into view, and an abrupt range measure is read, leading to an emergency stop condition (and eventually a collision)

3.2 Improving Detection by Sensor Panning

It is clear that the previous static design is not enough to eliminate the risk of crossing areas of lateral detection or blind zones. An alternative to reduce such dangers is to dynamically accommodate the scanned area to the instantaneous robot motion state. Let us define the scanning swept line by the pair (d_f , ρ), and let us denote γ to the angle, measured from the main robot axis, to direct a new scan action.
We will calculate γ assuming that the scan line has the previously calculated shape d_f , and adding the panning (orientation selection) capability. A first solution consists in finding the sensor orientation γ so that the blind zone A_3 diminishes the maximum possible, or totally disappears. The idea is to make use of the whole FSL to cover the intended robot trajectory. It can be obtained by solving the equation:

$$A \sin \gamma + B \cos \gamma = 1 \tag{9}$$

being A and B being constants for a given field of regard (d_f , ρ). Its solution reveals the greater blind-zone reduction that can be achieved by panning (Alvarez, et al., 2002). Notice that A_2 zones may still appear. To avoid them, an alternative design would be necessary which allows us to modify the three sensor parameters on-line, a much more complicated condition to apply in real sensors.

3.3 Improving Detection by Sensor Selection

Another option is to use the combination of sensors necessary to guarantee robot safety, that is, to reduce sensing to the areas traveled by the robot if a stop maneuver is initiated. We want to check whether the robot will hit an obstacle along that trajectory or not, subject to sensor availability at each cycle. First, we must determine which set of sensors, from the available ones, minimize the difference between the area covered by sensors and the area which has to be inspected. We propose a greedy algorithm to perform this task:

1) Calculate the trajectory followed in an emergency stop initiated after the next command has been executed.
2) Determine the set of available sensors to inspect the desired path and initialize the set of selected sensors to the empty set.
3) From the set of available sensors, select the sensor that covers the points farthest from the stopping trajectory with the best resolution and configure it so that it covers the maximum area.
4) Remove the selected sensor from the set of available ones and add it to the set of selected sensors.
5) Remove from the path to be inspected the area scanned by the selected sensor.
6) Go back to step 3 until the area to be inspected is empty, or no new sensor can be selected.
7) f there is an area that can not be inspected return a not safe condition,
8) Otherwise fire selected sensors and check if all readings are compatible with the minimum free area configured for each sensor.

Greedy algorithms are suboptimal, but in this case, they may give a good approximation in a short time period. To compute the area to be scanned with the sensors the robot motion equations can easily be integrated to obtain the trajectory of the robot until it stops. The stop commands sequence can be arbitrarily chosen every control cycle, and even different combinations of commands may be tested. Figure 5 shows an example of stopping trajectory. Figure 6 shows the result of this covering algorithm in the same environment where the experiments will be carried out (see following section).

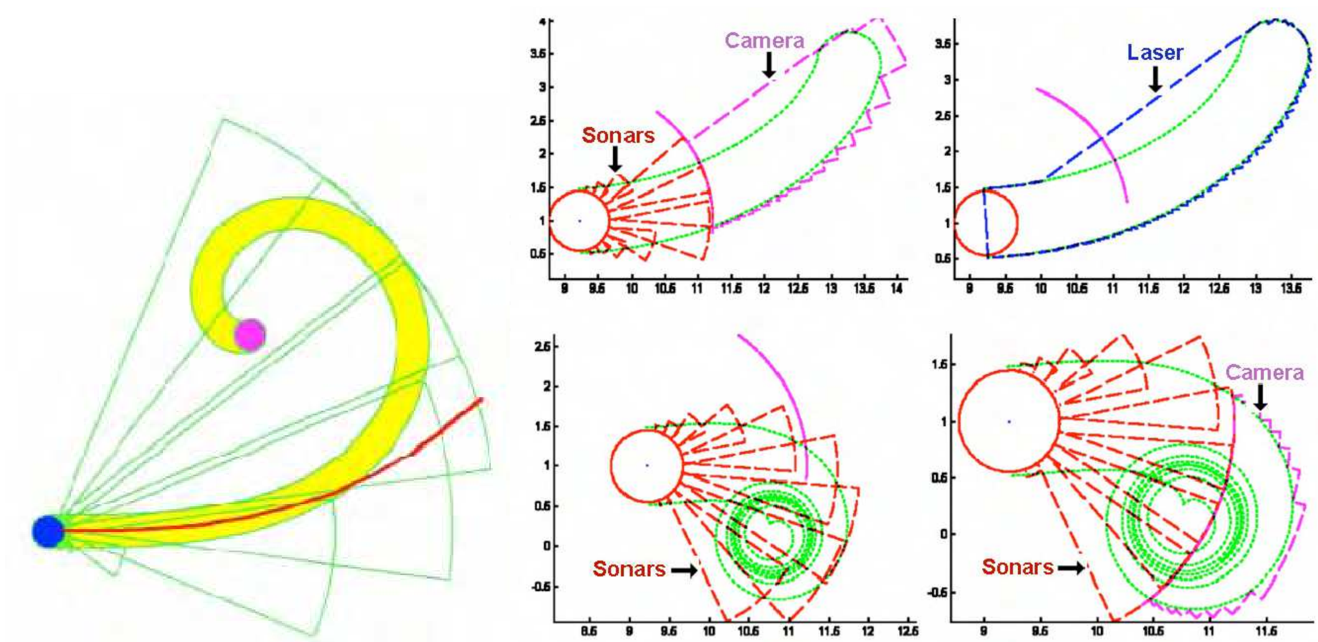


Figure 5. Example of a stopping trajectory and a set of sonars to inspect it. (Left) The continuous line represents the intended trajectory, while the thick path shows the area traversed by the robot for a given stopping procedure. (Right) Different runs of the covering algorithm, for different combinations of stopping trajectories and available sensors

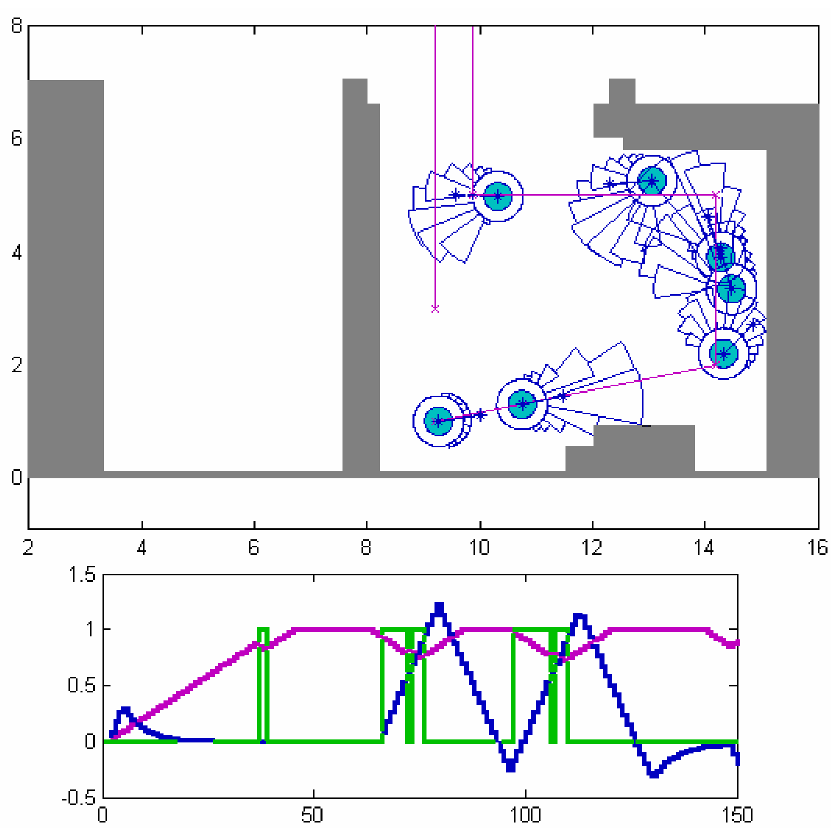


Figure 6. Sensor selection: the minimum set of individual sensors that are able to cover the emergency trajectory within a cycle time. (Above) Robot motion behavior and set of sensors fired in each iteration. (Below) Time velocity profiles: forward and turn velocities (m/s vs. s)

4. Experimental Setup

The previous analysis has been applied to the design of an obstacle avoidance module for a RWI B-21 mobile robot at the University of Oviedo. The local sensing strategy was implemented with a ring of ultrasonic range sensors of 24 equidistant units. Even though it limits the pan angle election to values 15 degrees, different combinations of simultaneous firing of contiguous sonar were used to implement different scan window shapes, as represented in Figure 7.

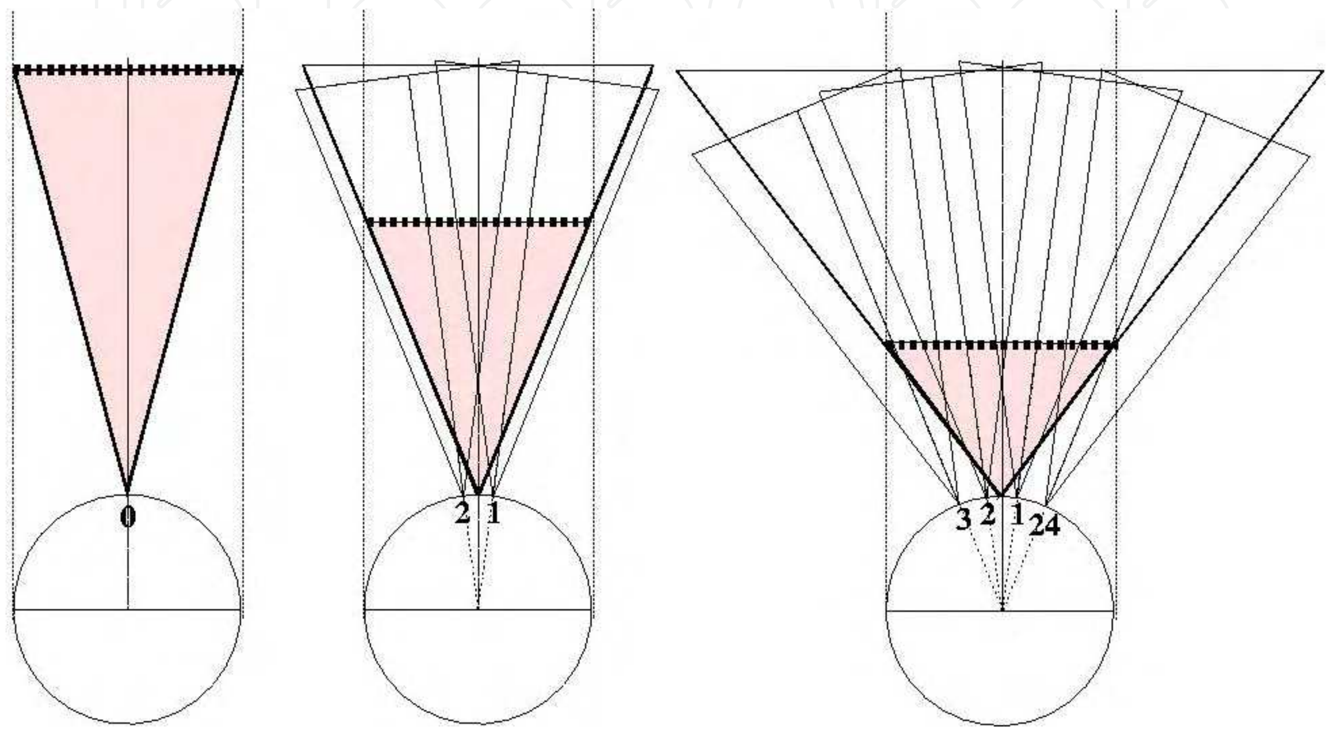


Figure 7. (Left) Virtual sensor implemented with a unique frontal ultrasonic sensor of aperture 15 deg. (Center) Virtual sensor implemented with two sonar sensors, located at ±7.5 degrees from the main robot axis, robot B-21, aperture 22.5 deg. (Right) Virtual sensor implemented with four sonar sensors, located at ±7.5 degrees from the main robot axis, robot B-21, aperture 37.5 deg

Similar experiments to those made with the Nomad–200, as reported in (Alvarez, et al., 2001), have been carried out with the faster and heavier B21 robot, applying the proposed panning strategy to avoid the previous inconvenient situations without loss of reactivity.

Every control cycle the robot is fed with another sub-goal to reach in its workspace, with the condition that the straight line connecting each sub-goal is supposed to be free of obstacles. In our first setup, the sub-goals happen to be located in the corners of a 2x2m square, see Figure 8. The robot does not know which the next sub-goal will be until it reaches the current one. The obstacles labeled in Figure 8 have to be detected and avoided in real time with minimum loss of motion performance.

The B21 robot has a maximum forward velocity of 1 m/s. With a 45 degree beam, the df to cover condition (4) is 64 cm. As this distance is similar to that needed to stop, $d_f \approx d_f^*$, the limiting values (7) and (8) are 59.5 cm and 34.3 cm respectively. The first conclusion is that, with the selected height of d_f =64 centimeters, the sensor will not present lateral detection zones.

When turning, the proposed sensor panning strategy, equation (9), suggests firing the sonar located at 32.5 degrees, which can be physically accomplished with sonar numbers 3

and 4. It eliminates blind zones, and the obstacle is detected with enough time to reduce speed and, eventually, plan a detour.
Figure 8 shows the result of the proposed motion strategy with sensor panning in a cluttered environment. The robot moves counter clockwise from the lower left corner point. Five unexpected obstacles lie in the sub-goals surroundings. When obstacle 3 is not present, obstacle 1 is especially prone to fire emergency stop conditions as described before. Such possibility was eradicated using the panning strategy: the 3-4 sonar pair was fired, allowing smooth obstacle detection and an equivalent velocity reduction proportional to the distance to the obstacle. It produces the flat velocities profiles in Figure 8-right, resulting in a good motion performance and high average velocities.

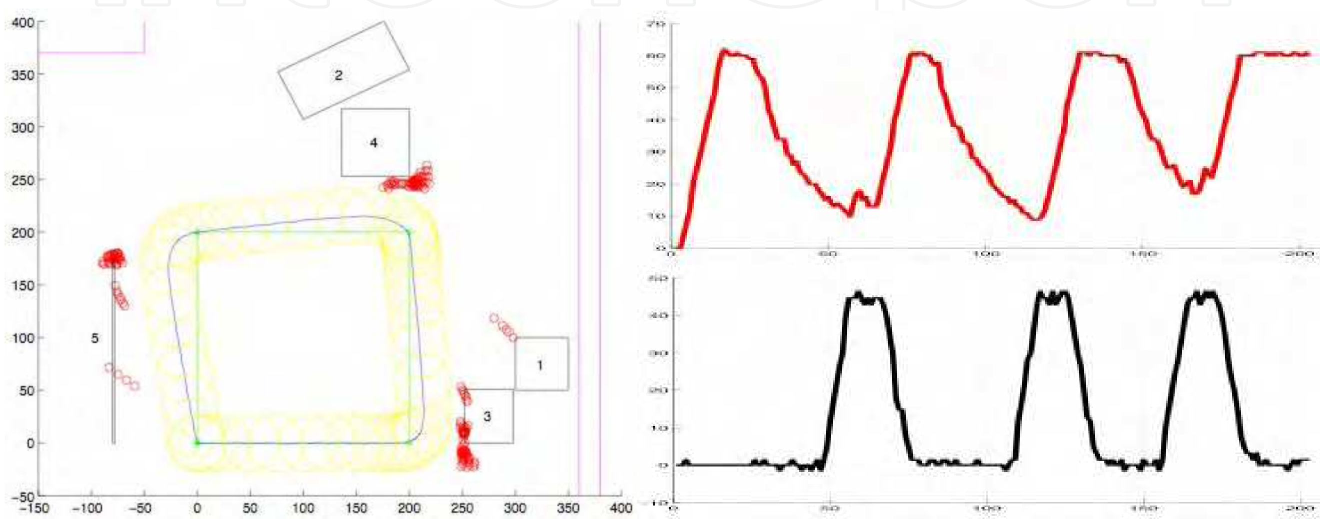


Figure 8. Experiment 4a, in a cluttered environment; top view. (Left) Unexpected obstacles detected during motion, as the sonar readings show, shorten the robot’s trajectory. (Right) Robot forward and turn velocities

Table 1 summarizes the experimental results. The B-21 was programmed with the same dynamic parameters that the Nomad-200 in (Alvarez, et al., 2001), that is, 60cm/s and 45deg/s as maximum velocities, and 25cm/s2 and 50deg/s2 as maximum accelerations. Experiments labeled 4b and 4c correspond to the same setup as 4a, but removing obstacles 4 and 3 respectively. Experiment 1 implements a point-to-point stop-and-turn strategy (notice the total path length of 8 meters), that is safe but of low performance.

		Path Length (cm)	Time (s)	(vM, aM) (cm/s, cm/s2)	(ωM, αM) (deg/s, deg/s2)
Nomad	Exp 1	800	31.6	(60,25)	(45,50)
	Exp 4a	828	22.3	(60,25)	(45,50)
B21	Exp 4a	867	21.0	(60,25)	(45,50)
	Exp 4a	910	19.1	(60,25)	(45,50)
	Exp 4a	939	18.0	(60,25)	(45,50)

Table 1. Comparative experimental results for two mobile robots: a Nomad-200 of Nomadic and a B-21 of Real World Interface

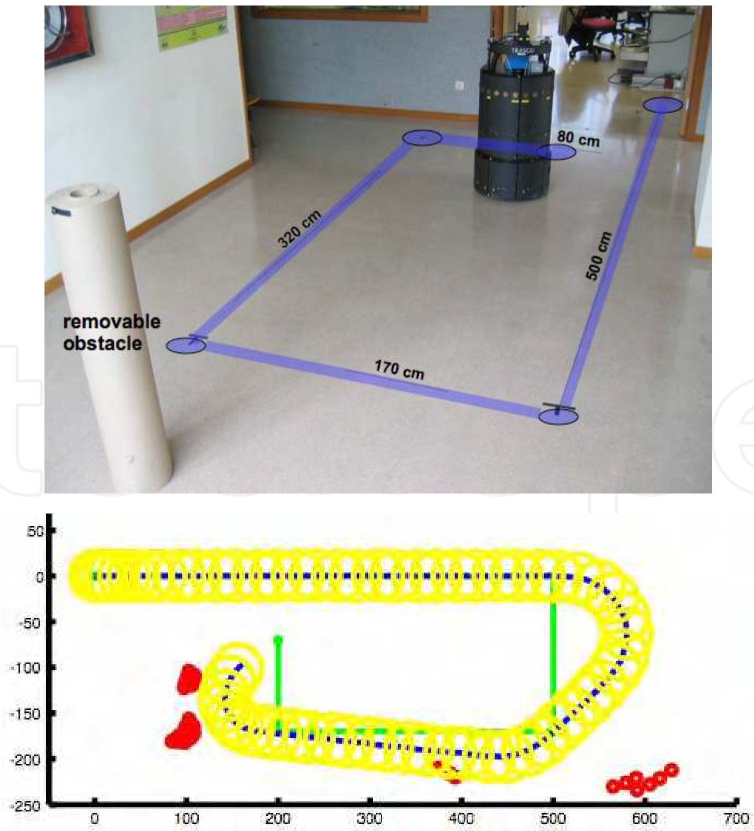


Figure 9. Experimental setup. (Above) The robot has to move as fast as possible between four consecutive goals in an unknown environment. Obstacle detection is implemented by means of two sonars, with a field of view, ρ=22.5 deg and d_f =64 cm, and a control period of 4 cycles/s. The removable obstacle will be located in different situations. (Below) Typical result, top view: the sensor readings inside the scanned window (dots) produce lineal robot decelerations following the strategy in equation (3) (axis in cms)

Figures 10 and 11 show more experimental results, with a different setup plotted on Figure 9. Figure 10 illustrates a situation of unnecessary deceleration, that is solved with the panning strategy. The movable obstacle is located in a zone of unnecessary deceleration A1a, which produces a velocity reduction in t = 12s. The same situation is produced by the wall on the opposite side. In the inferior figure the panning strategy is applied, and the sensors fired are those located at 15 degrees. The movable obstacle in not detected and no deceleration is commanded. A better operation also occurs in the vicinity of the mentioned wall. Figure 11 shows a situation of blind zone, with the consequent collision, and how it is solved by panning. The movable obstacle is located in a blind zone A3, which produces a collision in t = 15s. By firing sensors at γ=15 degrees, the obstacle is detected and a lineal deceleration is commanded. It changes the robot's trajectory enough to avoid the obstacle, and the robot is able to finish the mission.
In Figure 11-below, the covering algorithm is applied instead of the panning strategy. We defined a set of three possible stopping trajectories: to keep turning speed constant, and increasing turning speed to the left/right at maximum allowed rate. We have represented the scanned areas at different positions. In the beginning, the robot accelerates towards its first goal. At the point P_1, the robot decelerates to avoid a column close to the path. This situation is an unnecessary deceleration due to the use of sonar devices. The situation is repeated at the point P_4. As soon as a suitable stopping strategy is found (turning to the interior of the intended path), the robot begins to accelerate again. As the robot reaches P_3, the walls are relatively close. Due to the excessive area scanned by sonar because of its

poor angular resolution some unnecessary decelerations are fired. Once the small passage formed by the obstacle and the wall is left behind, the robot began to accelerate. At point P_5 a wall is too close to the checkpoint, and a new deceleration is fired. The time used to complete the mission is similar to that achieved with the panning strategy, but execution is safer and the robot speed references are smoother.

5. Extended Experiments

The proposed strategy has been tested in longer and more realistic experimental setups. Figure 7 shows the Robotics Lab at the University of Oviedo and the point-to-point mission assigned. The lab area of interest is around 130 square meters, and the point-to-point total distance is 55.6 meters. Figure 8 shows several views of the checkpoints and the obstacles that surround them. Most of these obstacles are removable, in order to present a less cluttered challenge to test the performance degradation with the proposed algorithm. Six experiments are reported and their result compared. Experiments 1 to 3 were carried out removing the obstacles in the checkpoints vicinity, and experiments 4 to 6 were performed in the more cluttered setup in Figures 12 and 13.
Experiments differ from each other on the maximum robot velocities and accelerations applied, and in the length of the scan window d_f adopted (see Table II).

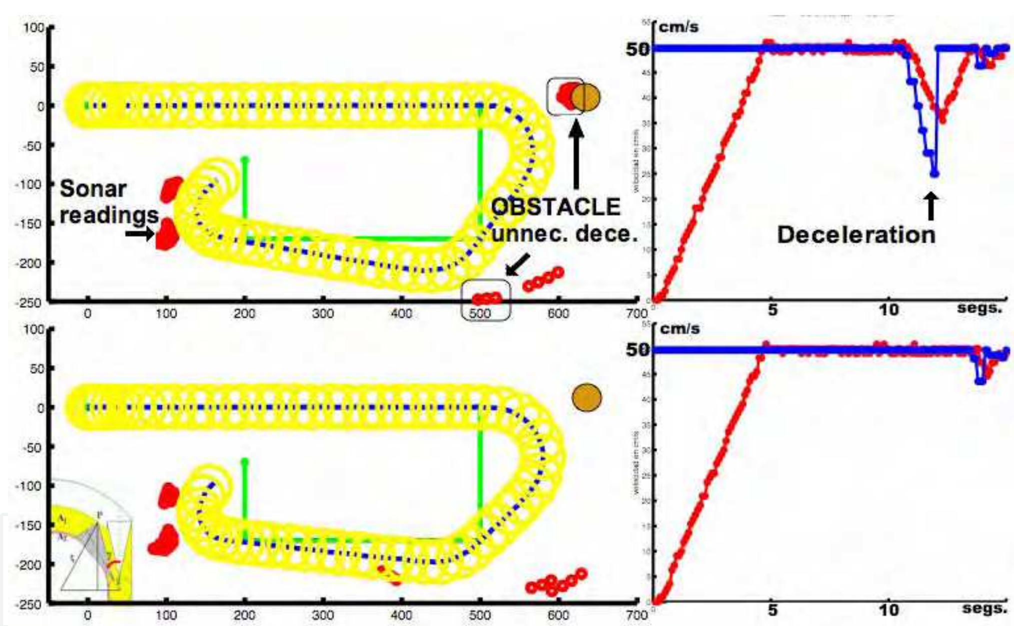


Figure 10. Avoiding situations of unnecessary deceleration by panning. (Above) The movable obstacle is located in a zone of unnecessary deceleration A1a, which produces a velocity reduction in t=12s. The same happens with the inferior wall. (Below) In the first turn the sensors fired are those located at 15 degrees, the obstacle in not detected and no deceleration is commanded. A better operation occurs in the vicinity of the wall too

Experiments	(vM, aM)	(ωM, αM)	df
1,4	(90,50)	(70,50)	170
2,5	(60,25)	(70,50)	170
3,6	(60,50)	(70,50)	144

Table 2. Extended experimental results conditions

Scan length is larger then the minimum, to allow smoother velocity reductions, but it increases the risk of unnecessary decelerations when moving in a straight line as discussed. Sweep aperture was always ρ=22.5 degrees implemented by firing the two frontal sonars. Table III summarizes the results of the experiments. The first line refers to a Stop-and-Turn control strategy that allows moving from point to point without local obstacle detection, supposing that there are no obstacles interfering in the straight-line paths.

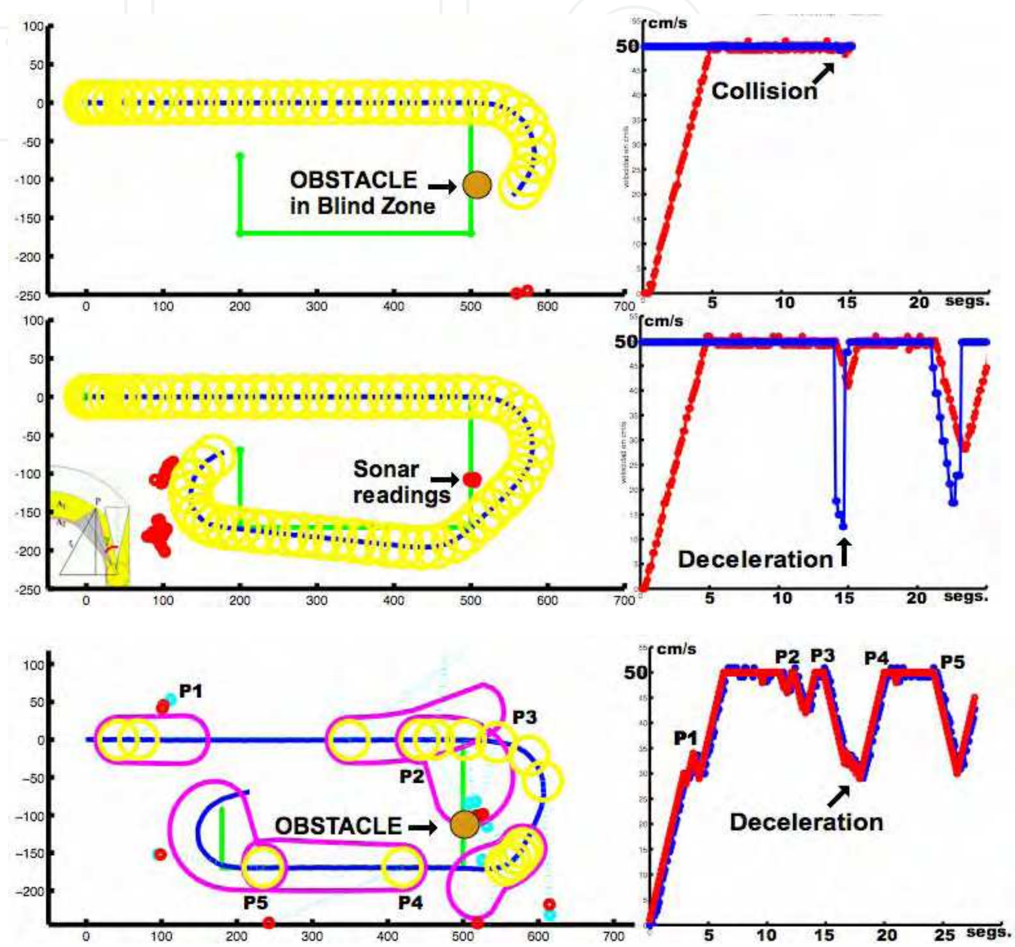


Figure 11. Avoiding blind zones by sensor selection. (Above) The movable obstacle is located in a blind zone A3, which produces a collision in t=15s. (Middle) By firing sensors at γ=15deg. the obstacle is detected and a lineal deceleration is commanded. It changes the robot's trajectory enough to avoid the obstacle, and the robot is able to finish the mission. (Below) Results obtained using the covering algorithm. We can see the real trajectory followed by the robot and some candidate stopping trajectories: straight, turning left and turning right. Notice that the algorithm generates references compatible with the robot dynamics

	dist. (cm)	time (s)	v_a (cm/s)
Stop & Turn	5560	144	38.6
1	6178	84	73.5
2	5899	116	50.6
3	6011	106	56.3
4	5784	89	64.1
5	5641	129	43.5
6	5756	111	51.6

Table 3. Results of the extended experiments

Average velocities v_a are lower in experiments 4-6 than in their corresponding 1-3 because the environment is more cluttered. Also final paths are shorter for the same reason (the robot has to negotiate more carefully at every checkpoint).

Fastest motion happens in Exp. 1 (v_a = 73.5) because the robot has higher velocities and acceleration capabilities to exploit in an easier environment. Experiment 3 has better performance than Experiment 2 only because the robot decelerates faster. Interestingly, being able to see 20% further than "robot 3" does not help "robot 2" to overcome such fact. The same reasoning applies to experiments 5 and 6.

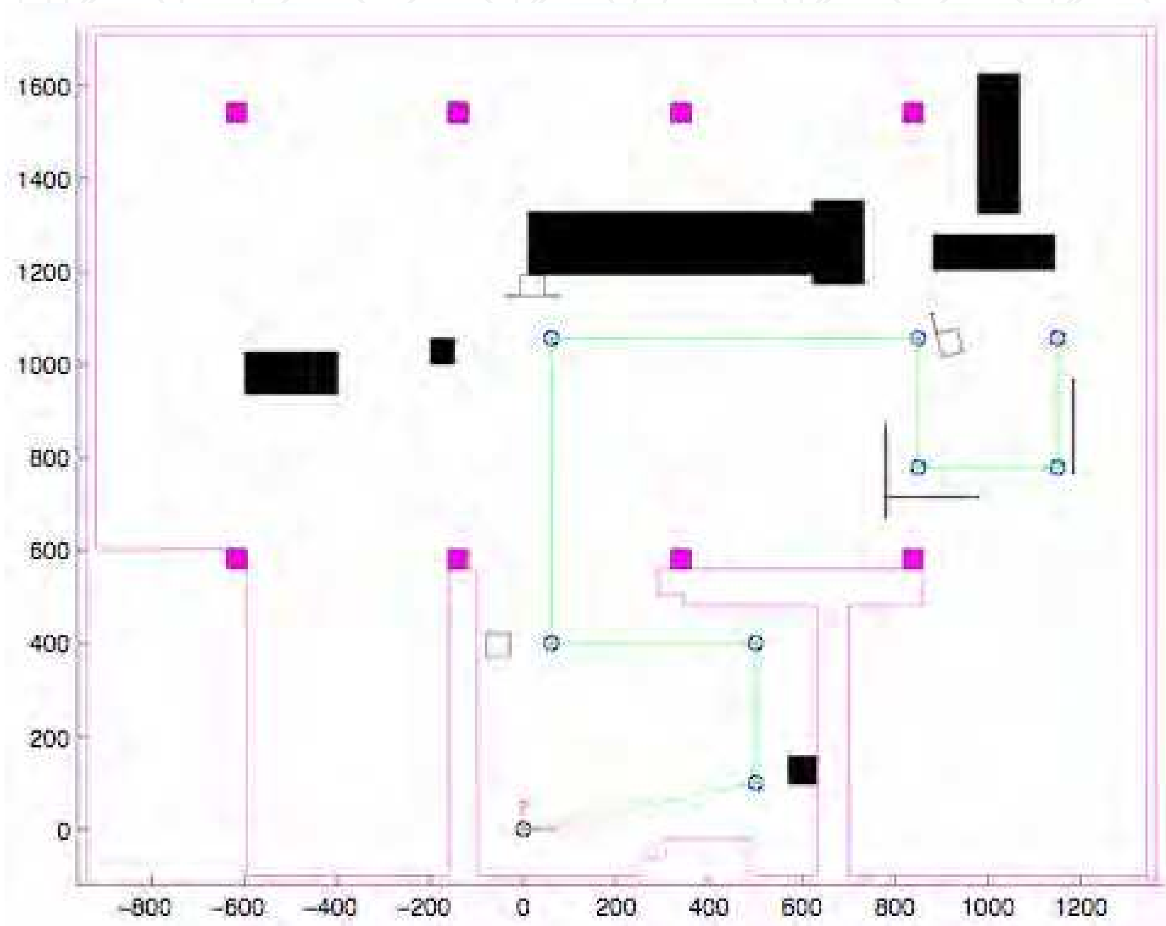


Figure 12. Lab workspace setup: an area of 130 square meters, and a point-to-point mission of a length of 55.6 meters with 12 checkpoints

Figure 13. The experimental setup in Figure 12, and simulated in Figure 6. The images show details of the obstacles around each checkpoint

6. Conclusions

As the sensorial capacity of the robots increases, for example with the availability of MEMS sensors, it will be a requisite for good real-time performance to wisely select and manage them, in order to fulfill the strong time requirements of machines moving in human environments. This work represents a first approach towards solving this kind of problems. An analysis is presented which permits the selection of the sensor requirements for collision avoidance tasks of mobile robots. The design is compatible with motion in real time, as only the indispensable environment zones are explored, avoiding unnecessary velocity reductions. Perception system restrictions can be considered in the strategy. The analysis is deterministic, and the effect of sensor uncertainties is not discussed.
The sensor model presented is general enough to be implemented with various different sensing devices. In practice, some restrictions in the election of the parameters will depend on the specific sensor characteristics, e.g., with a sonar ring, transducers are arranged around the robot with fixed intervals γk, and only those pan angles are permitted. Using computer vision for range measurement gives us more flexibility, but restrictions will exist on the other two parameters. Notwithstanding, the ideas seem to be extensible to more complex models of robots or sensors, and to different motion control policies.

8. References

Alvarez, J. C., et al. (2002) Sensor management for local obstacle detection in mobile robots. New York, IEEE, pp. 2401-2406.

Alvarez, J. C., A. Shkel, and V. Lumelsky. "Accounting for Mobile Robot Dynamics in Sensor-Based Motion Planning." Int. Journal of Robotics and Automation 16, no. 3(2001): 132-142.

Brock, O., and O. Khatib. "Elastic strips: A framework for integrated planning and execution." Experimental Robotics VI 250(2000): 329-338.

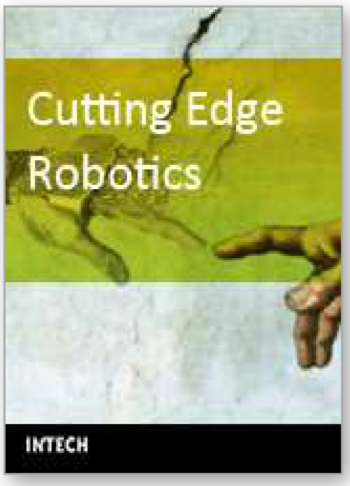
Kelly, A., and a. Stentz. "Rough terrain autonomous mobility - Part 1: A theoretical analysis of requirements." Autonomous Robots 5, no. 2(1998): 129-161.

Laubach, S., and J. Burdick. "RoverBug: Long range navigation for Mars Rovers." Experimental Robotics Vi 250(2000): 339-348.

Lumelsky, V., and T. Skewis. "Incorporating Range Sensing in the Robot Navigation Function." IEEE Transactions on Systems Man and Cybernetics 20, no. 5(1990): 1058-1069.

Luo, R. C., C. C. Yih, and K. L. Su. "Multisensor fusion and integration: Approaches, applications, and future research directions." IEEE Sensors Journal 2, no. 2(2002): 107-119.

Ulrich, I., and J. Borenstein. "The GuideCane - Applying mobile robot technologies to assist the visually impaired." IEEE Transactions on Systems Man and Cybernetics Part a-Systems and Humans 31, no. 2(2001): 131-136.



Cutting Edge Robotics

Edited by Vedran Kordic, Aleksandar Lazinica and Munir Merdan

ISBN 3-86611-038-3
Hard cover, 784 pages
Publisher Pro Literatur Verlag, Germany
Published online 01, July, 2005
Published in print edition July, 2005

This book is the result of inspirations and contributions from many researchers worldwide. It presents a collection of wide range research results of robotics scientific community. Various aspects of current research in robotics area are explored and discussed. The book begins with researches in robot modelling & design, in which different approaches in kinematical, dynamical and other design issues of mobile robots are discussed. Second chapter deals with various sensor systems, but the major part of the chapter is devoted to robotic vision systems. Chapter III is devoted to robot navigation and presents different navigation architectures. The chapter IV is devoted to research on adaptive and learning systems in mobile robots area. The chapter V speaks about different application areas of multi-robot systems. Other emerging field is discussed in chapter VI - the human- robot interaction. Chapter VII gives a great tutorial on legged robot systems and one research overview on design of a humanoid robot.The different examples of service robots are showed in chapter VIII. Chapter IX is oriented to industrial robots, i.e. robot manipulators. Different mechatronic systems oriented on robotics are explored in the last chapter of the book.