ACTIVE SENSING IN SENSOR–BASED MOTION PLANNING WITH DYNAMICS

J.C. Alvarez	A. Shkel	V. Lumelsky
Sys.Eng.& Automation Area	Mech.& Aeros. Eng.Dept.	Robotics Laboratory
ECE Dept., Univ. of Oviedo	Univ.of California-Irvine	Univ. Wisconsin-Madison
33204 Gijón, Asturias, Spain	Irvine, CA 92697, USA	Madison, WI 53706, USA

Abstract: This paper studies active sensing strategies for an autonomous wheeled mobile robot moving on a flat surface among unknown static obstacles. The robot is defined by its geometry, its dynamic motion equations and a control policy. The latter comes from a path planning algorithm which produces changing intermediate goal coordinates to pursue, and motoring signals based on local information. We want to know which is the (minimum) area we need to explore with the robot sensors in order to guarantee that a certain intended motoring signal is safe. The proposed solution depends on a) the sensor system design, b) the robot actual velocity and c) the robot dynamics. All these factors must be taken into account for safety and efficiency reasons. The result is an adaptive scanning procedure based on the robot motion circumstances at every time.

Keywords: sensing algorithm, sensor-based motion planning

1 INTRODUCTION

Real-time sensor-based robot navigation deals with the motion planning problem when only a subset of the workspace is known at each instant [3]. Algorithms of motion planning with incomplete information produce changing intermediate goal coordinates to pursue, and the associate control problem is to generate motoring signals based on local information in order to make the robot move towards the goal.

But three restrictions hinder implementation in practice: 1) robot motion capabilities, 2) sensors characteristics and 3) computation time limits (because control commands must be issued at a fast pace). These are unavoidable experimental limiting factors that are frequently ignored in the algorithms.

It is within this framework that we address the following problem: which is the (minimum) area we need to explore with the robot sensors in order to guarantee the safety of a certain intended motoring signals?. The question is meaningful as it deals with sensorial throughput, which is the major bottleneck when computing motion commands on real-time.

This paper describes a method to decide where to gather this local information about the robot sorroundings, in order to guarantee that a given control command is safe. It provides an algorithm which deals with sensorial throughput making a selective scanning based in the robot motion circumstances at every time.

2 SENSING, MOTION AND DYNAMICS

For the moment, consider the problem of sensor-based navigation as one consisting of two separate problems: a geometric task of path generation (call it Path Planner), for the robot to move in the workspace filled with obstacles, and a control task (call it Controller), which generates motoring commands. The input information to the Path Planner is robot current coordinates, C_i, and the description of the surrounding obstacles. Its output is an intermediate target point, T_i, and a straight-line path segment that leads to it. The Controller input is the current state C_i, current velocity vector v_i, point T_i, and the path segment from the Path Planner which the Controller is expected to execute, Figure 1.

Safety and motion efficiency requires to take into account the system dynamics. Otherwise, conflicts may arise, Figure 2. As the robot arrives at point C_i along the path AC_i, it decides on a new intermediate target, T_i, and a straight-line path segment to it. Since it arrives at C_i with a non-zero velocity, because of the system dynamics, it cannot make a sharp turn suggested by the Path Planner. To preserve continuity

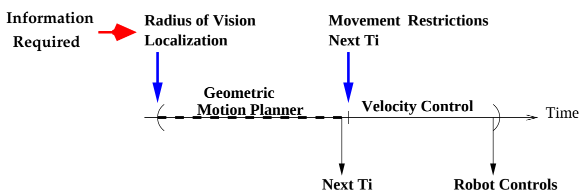

Figure 1: To handle system dynamics, planning and control are tied together within a single-stage cycle.

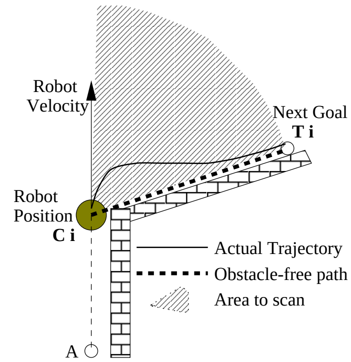

Figure 2: The robot at C_i cannot make a sharp turn suggested by the Path Planner to reach the new intermediate target, T_i.

in velocity, the real path must have a bulge around C_i (shown in Figure 2 in solid line). This curved path could in principle cut through an obstacle, which the Path Planner would not consider because it is off the intended straight line path.

Such dynamic strategies had been developed in previous works [4]. In brief, it operates as follows: 1) compute controls to reach the intermediate goal in minimum time, 2) scan the workspace to check if the computed controls guarantee an (emergency) stopping path in the next robot position, 3) if so, proceed at the maximum speed; otherwise, find suboptimal controls that guarantee the stopping path. The proposed control makes the robot moves with the maximum velocity that is feasible under the circumstances, and with guarantee of no-collision.

The previous scheme of dynamic motion planning relies on the sensors capability to scan a prescribed sector in small time periods (step 2). We are concerned here with such requeriment: what extension around the robot do we need to explore in order to guarantee the safety of an intended motion?. The problem can be stated: *from the robot current state C_i, its velocity vector v_i, and the intended robot commands u_i^* to (optimally) aproach T_i, obtain the minimum sector to explore A_k which, if obstacle–free, guarantees the safety of the resultant motion.* Safety means non collision in period k, and the existence of an eventual stop path τ, if it were neccessary in period $k+1$. We will refer to this minimum scanning area as the Safe-Scan Window, SSW.

In the following, we will focus on a disc shaped robot of radious r_r, moving on a flat surface among unknown static obstacles. Its motion equations involve the robot configuration (its position and orientation), $\mathbf{C}(t) = (x(t), y(t), \theta(t))$, and its velocity vector $\vec{\mathbf{v}} = (v, \theta)$, which is controlled via generic force or torque controls (τ_1, τ_2). How these controls will act on the velocity vector depends on the robot specific drive mechanism. Let us adopt the dynamic model:

$$M\dot{v} + f_v = \bar{\tau}_1 \qquad J\dot{\omega} + f_\omega = \bar{\tau}_2 \tag{1}$$

with M the robot "mass" (its resistence to change its traslational velocity), J its momentum of inertia (its resistence to change its turn velocity, and therefore its orientation, $\dot{\theta} = \omega$), and $f_v = k_1 k_a v$, $f_\omega = k_2 k_b \omega$ lineal functions in the velocity.

Usually forces (τ_1, τ_2) are generated by pre-build high-frequency digital controllers, in response to certain reference inputs (u_1, u_2). Let us assume that controls of forward velocity and orientation are decoupled, $\bar{\tau}_1 = k_1 u_1$, $\bar{\tau}_2 = k_2 u_2$, in order to separate the relations (u_1, v) and (u_2, ω) (as in a synchro–drive mobile robot).

This parametric model allows us to predict robot trajectories from the initial conditions (v_0, ω_0) and the intended motion $(u_1, u_2)^*$. Among the infinite possible selections of $(u_1, u_2)^*$, let us define two canonical operations that cover the worst case of an emergency stop. These are the "Panic Stop" and the "Turn Panic Stop" operations [2].

The *Panic Stop operation* occurs when an emergency stop is required while the robot moves at its higher speed, $(v, \omega)_{PS} = (v_{max}, 0)$. Intuitively, the best controls to apply in order to make it stop are $(u_1, u_2)_{PS} = (0, 0)$. Model (1) predicts a straight line stop path, with an exponential deceleration following equation $v = v_0 e^{-k_1 k_a t/M}$. In a step-by-step version:

$$v_k = \left(1 - \frac{k_1 k_a \Delta t}{M}\right) v_{k-1} \tag{2}$$

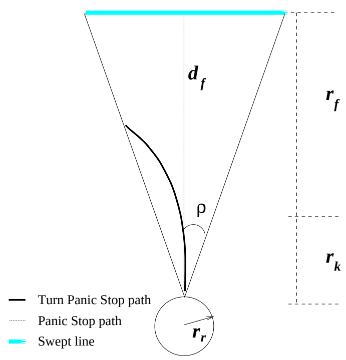

Figure 3: Basic SSW: a complete triangular area is scanned within one cycle; the distance to the closer object inside the area is returned.

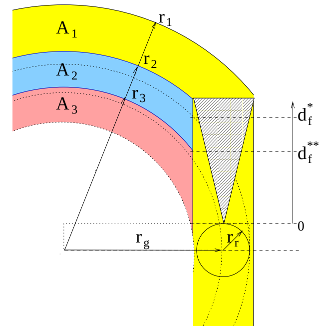

Figure 4: Scanned zones with the swept line of the BSSW when the robot turns with constant velocities, $v = v_k$ and $\omega = \omega_k$. The generated trayectory has a constant turning radious $r_g = v_k/\omega_k$.

The time needed to a complete stop, t_s, makes the robot to travel a distance, r_d,

$$r_d = t_r \cdot v_{max} + \sum_i^{t_s} v_i \Delta t = v_{max}\left(t_r + \frac{t_s}{\frac{k_1 k_a \Delta t}{M}}\right) \tag{3}$$

which includes the distance traversed due to the time delay t_r (response time of the robot control system).

The *Turn Panic Stop operation* corresponds to an emergency stop when the robot is turning at its maximum speed: $(v, \omega)_{TPS} = (v_{max}, \omega_{max})$. The necessary controls will be again $(u_1, u_2)_{TPS} = (0, 0)$. From the robot motion model, the linear velocity will decrease exponentially, with first-order dynamics $\omega = \omega_0 e^{-k_2 k_b t/J}$. As the exact integration is quite complex, an approximate solution is possible assumming that ω is constant during the whole maneuver. Then the orientation will increase linearly, following $\theta = \omega_0 t$. Then, the stop trajectory is obtained from

$$x_k = v_0 \int_{t_0}^{t_1} e^{-p_M t} \cos \omega_0 t \, dt \tag{4}$$

where $p_M = k_1 k_a / M$, leading to

$$\Delta x = v_0 \left(\frac{-p_M \cos(\omega_0 \Delta t) + \omega_0 \sin(\omega_0 \Delta t)}{p_M^2 + {\omega_0}^2} e^{-p_M \Delta t} + \frac{p_M}{p_M^2 + {\omega_0}^2} \right) \tag{5}$$

and analogous for Δy, which describes a spiral curve. The portion of this spiral traversed by the robot depends on the time required to bring its forward velocity to zero.

3 BASIC SSW DESIGN

A basic perception strategy, call it the Basic Safe-Scan Window (BSSW), consist of a sensor swept line traversing a triangular area within one cycle; the distance to the closer object within this area is returned, see Figure 3. Two parameters define the BSSW field of vision, height (d_f) and aperture (ρ). They are selected with the following criteria:

1) The relation between height and aperture (d_f, ρ) has to be such that the robot is not in danger when moving in a straight line. It is fulfilled if the whole area crossed by the robot is previously swept by the front scan zone. That is to say, given the robot dimension (a circle of radius r_r), both parameters satisfy $d_f \tan \rho \geq r_r$.

2) The BSSW height (d_f) has to be big enough to let the robot stop safely (fast enough), if an obstacle appears in front of it. The worst case condition is defined by the canonical Panic Stop operation: being v_k the actual robot velocity, r_k the maximum distance traveled within a control cycle, and r_d the distance needed to stop with maximum brake effort, we get from (3) that

$$r_k = v_k(\Delta t + t_r) \qquad d_f \geq r_k + r_d \tag{6}$$

This magnitude depends on the robot current status v_k. Adopting $v_k = v_{max}$ let us to fix d_f, which now can be computed off–line. With this design, it can be easily seen that, if an emergency stop is issued at instant $(k+1)$, the robot will be able to do a “Panic Stop” no matter what its initial state $(v_k, \omega = 0)_k$ is.

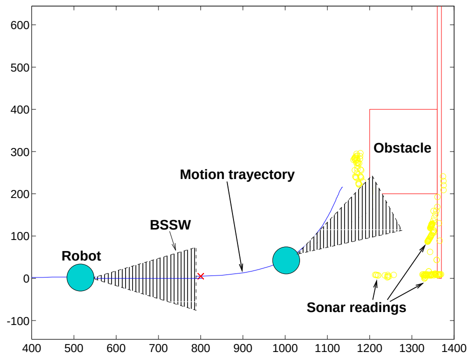

Figure 5: Experiments with the BSSW: stop maneuver triggered by an “A_2 zone” obstacle; top view.

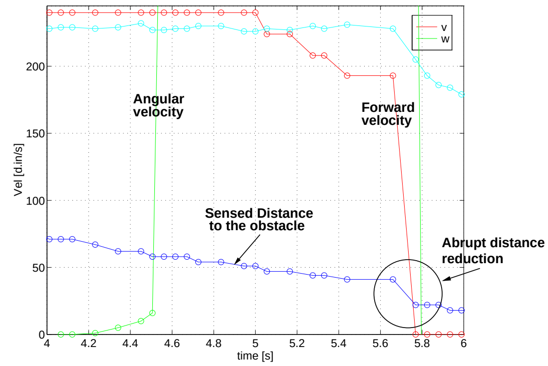

Figure 6: Stop maneuver caused by an “A_2 zone” obstacle in Figure 5; sensed distance and forward & angular robot velocities.

3) As the area covered by the BSSW is getting smaller, the time needed to explore it decreases. Hence d_f will be selected as small as the robot dynamics (condition 2) allows. This fixes the aperture ρ, together with the robot size (condition 1).

This sensing strategy can be implemented by a single frontal sonar device with enough wide beam ρ, and effective maximum range d_f. Experiments carried out with a Nomad-200 mobile robot showed the feasibility of the approach [1].

4 UNSAFE ZONES WITHIN THE BSSW

When moving in a straight line, robot safety is guaranteed because of the first design condition. But when the robot turns ($\omega \neq 0$) the BSSW field of view generates three different situations, as Figure 4 reveals.

Area A_1 (the sector extended between radios r_1 and r_2), is swept by the frontal part of the sensor field. The obstacles inside it will produce a lineal robot velocity reduction. But note that such a velocity reduction was not necessary, as the robot body was not going to cross that zone afterall. It is an area of “unnecesary deceleration”.

In fact, the robot body traverses sectors A_2 and A_3 in Figure 4. The sector comprised beneath radious r_2 and r_3 (A_2 sector) is a zone of “lateral detection”: an obstacle over here will be detected, but it will appear abruptly on the robot field, making a lineal velocity-reduction strategy not useful to avoid collisions. Then, in order to apply this sensing strategy in practice, obstacles in A_2 zones will trigger a stop condition. Figures 5 and 6 show that circumstance in experiments with a real robot (a Nomad-200 of Nomadic Tech.) moving at 60 cm/s and turning at 500 deg/s.

A worse case is A_3, a “blind zone” where obstacles will never be detected, causing a collision; in practice obstacles in A_3 zones must be manually avoided.

A careful selection of d_f will let us to manage those inconvenients. Notice that the BSSW design just fixed a minimum d_f length, enought to take into account robot dynamics limits. Choosing $d_f \to 0$ would eliminate A_2 and (almost) A_3 zones. There is a certain limit height d_f^* such as, if $d_f \geq d_f^*$, the robot body will not cross A_1 zones at all. In that situation, we can consider that the perception effort is useless, and even counter-productive, since it can generate unnecessary velocity reductions. This value d_f^* is such as $r_2 = r_r + r_g$ (see Figure 4), leading to:

$$d_f{}^* = 2\sqrt{r_r r_g} - r_r \tag{7}$$

Similar geometric analysis shows that a smaller height d_f^{**} exists such as, if $d_f \leq d_f^{**}$, completely eliminates A_2 zones. Then, the robot movement will be covered by A_1 and A_3 areas. This value $d_f{}^{**}$ makes $r_2 = r_3 = (r_g - r_r)^2 + 2r_r r_g$, leading to

$$d_f^{**} = \sqrt{2}\sqrt{r_r r_g} - r_r \tag{8}$$

Blind zones like A_3 can be reduced only by choosing $d_f < d_f^{**}$, and only if the robot dynamics allows it. We can quantify the “risk factor” when such reduction is not possible: half the area swept by the robot when turning happens to lie over the A_3 risky zone. Even for a robot without dynamics (able to stop on

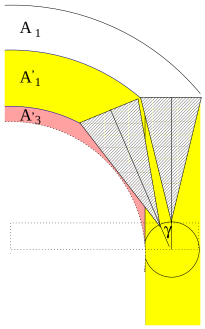

Figure 7: Scannig a fixed area –the same BSSW is selected–, with an orientation γ such as A_1 is besides the previous A_1.

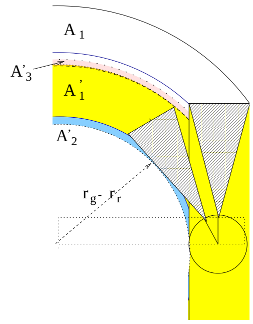

Figure 8: Scannig a fixed area: given the swept line form, (d_f, ρ), the sensor orientation γ is selected in order to eliminate blind zones A_3

site and allowing us to choose $d_f = 0$), the blind zone A_3 does not completely dissapear: notice that A_3 can be subdivided in two areas $A_3 = A_{3a} + A_{3b}$, with: $A_{3a} = f(r_r)$ and $A_{3b} = g(2r_r r_g - {r_r}^2)$. As the robot has a size, there will allways be blind zones $r_r > 0 \rightarrow A_{3a} > 0$. But even without dynamics, the biggest theoretical "risk factor" corresponds to a $r_g = r_r$, and the area not covered is a 25% of the total (notice that as $r_g \rightarrow \infty$, the "risk factor" tends to zero, because the design of the BSSW).

In conclusion: depending on the robot dynamics, and fixing ρ with $d_{stop} \tan \rho = r_r$, three situations might occur:

1) if $d_{stop} > d_f^*$ the robot will be in danger both for A_2 and A_3 reasons, and unwanted velocity reductions may appear.

2) if $d_f^* > d_{stop} > d_f^{**}$ the robot will be in danger for A_3 reasons, while A_2 zone danger and unwanted velocity reductions diminish.

3) if $d_f^{**} > d_{stop} > 0$ the robot will be in danger because of A_3 zone reasons, while A_2 dangers dissapears.

There is no design of SSW which eliminates the risk of crossing areas of lateral detection or blind zones. That is why a different strategy has to be design to avoid them.

5 ACTIVE SSW WITH A FIXED AREA

An alternative to reduce such dangers is to dynamically acomodate 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 axe, to direct a new scan action, Figure 7.

In theory, we have three variables to select in order to cover the robot motion area. In practice, some restrictions will apply, depending on the sensorial system characteristics. For example, with a sonar-like scanning, transducters can be arranged in a ring around the robot, with fixed intervals γ_k. The transducters themselves had usually fixed emision cone ρ and a maximum range d_f. Similar restrictions are applied when the range measures are taken with a stereo par: when mounted over a pan-and-tilt, major limitations with γ dissapear, but they still exist over the other two parameters.

Our objective is to design a scanning line with the size just big enough to cover the robot size while being respectful with its dynamics. A first approach is to calculate γ, assuming that the scan line has the same shape than the BSSW, Figure 7. That is, in the triplet (γ, d_f, ρ) only the first parameter is configurable. This solution would have very good chances to be feasible in practice, as it demands the same performance characteristics from the sensorial system as we previously did in the design of the BSSW (except a turn γ). The only impediment could be not being able of scanning the desired direction γ (v.gr. with a fixed sonar ring system).

Let us suppose that the robot dynamics fix the worst case situation, $d_f \geq d_f^*$. There is a sensor whose scan line is contiguos to the BSSW one, given by

$$\tan \gamma = \frac{2K}{1 - K^2} \tag{9}$$

being

$$K = \frac{r_r}{r_r + d_f} \tag{10}$$

We could expect that the adjacent sensor swept most of A_2 and A_3 zones, but it is not guaranteed that the new risky blind zone A_3 is going to totally dissapear, as the case of Figure 7 ilustrates. The performance is worse as d_f grows.

A better selection consists of finding the sensor orientation γ such us the blid zone A_3 totally dissapears, see Figure 8. For a given BSSW (d_f, ρ), the desired sensor γ can be obtained from

$$\sin\left(\arcsin(\frac{r_r}{h}\cos\gamma) + \frac{\pi}{2} - \gamma - \rho\right) = \frac{r_g - r_r}{h} \quad (11)$$

being $h^2 = r_g^2 + r_r^2 - 2r_g r_r \sin\gamma$. Notice that the whole equation is a relation like $g(\gamma, \rho) = 0$, and does not depend on the height parameter d_f. It means that a new blind zone may appears now on the other side, as marked with A_3^* in the previous Figure 8.

We can calculate D_f such as, if $d_f > D_f$, the new blind zone will not appear. It is the result from the second grade equation:

$$D_f^2 + \left(2r_r\cos^2\rho - 2r_g\cos\rho\sin(\gamma-\rho)\right)D_f - 2r_r r_g(1+\sin\gamma)\cos^2\rho = 0 \quad (12)$$

If this condition is not fullfilled, and if the sensorial system allows it, we can select the new $d_f = D_f$ in order to eliminate blind zones.

In conclusion, for a Scan Window with fixed shape is easy to calculate the adequate scan orientation γ, and allows to eliminate blind zones. But it has the drawback that still A_2 zones may appear. An alternative design, if the three parameters modified on-line, is presented in the following section.

6 CONCLUSIONS

An analysis is presented which allows the on-line selection of the perception process needed to assure safe motion in Sensor-Based Motion Planning. The design takes into account robot dynamics, given by a mathematical model. It is compatible with motion in minimum time, as only the indispensable environment zones are explored, avoiding unnecessary velocity reductions. Two different alternatives, adapted to different perception system restrictions, were described.

References

[1] J. C. Alvarez, A. Shkel, and V. Lumelsky. Accounting for robot motion dynamics on sensor–based motion planning: Experimental results. In *IEEE Int. Conf. on Robotics and Automation*, pages 2205–2209, Leuven, Belgium, may 1998.

[2] Alonzo Kelly and Anthony Stentz. Analysis of requeriments fir high speed rough terrain autonomous mobility. part 1: Throughput and response. In *IEEE Int. Conf. on Robotics and Automation*, pages 3318–3325, 1997.

[3] V. J. Lumelsky and A. A. Stepanov. Dynamic path planning for a mobile automaton with limited information on the environment. *IEEE Trans. Autom. Control*, 31(11), nov 1986.

[4] Andrei Shkel and V. Lumelsky. The jogger's problem: Control of dynamics in real-time motion planning. *Automatica*, 33(7):1219–1233, jul 1997.

Corresponding author:

Assoc. Prof. Dr. Juan Carlos ÁLVAREZ ÁLVAREZ, Electrical & Computer Engineering Dept., Area of Systems Engineering and Automation. Campus de Viesques, Ed. no. 2, 33204 Asturias, España.
Fax Int ++34 985 182 264, Email: juan@isa.uniovi.es.