Tracking a partially predictable target with uncertainties and visibility constraints ⋆

P. Fabiani [a] H.-H. González-Baños [b] J.-C. Latombe [b] D. Lin [b]

[a] *Onera-Cert/ DCSD, Control & Decision Group, 2 av. Édouard Belin, BP4025, Toulouse, F-31055, FRANCE.*

[b] *Stanford University, Department of Computer Science, Robotics Laboratory, Stanford, CA 94305-9010, USA.*

Abstract

Planning the motions of a mobile robot tracking a partially predictable moving object (a *target*) in a workspace cluttered by obstacles is a challenging multifold planning problem. The planner must take into account both the visibility and motion constraints imposed by the obstacles. It must also deal with the uncertainties in both the robot's position and the target's future trajectory. This paper proposes a framework based on Game theory to solve this multifold problem. At each time step, a probability distribution models the uncertainties on the robot and target positions. A utility function represents the reward associated with the possible goal states of the motion decision problem. This framework allows the simple modeling of various tracking strategies, one of which was implemented and tested successfully with two mobile robots. By considering simultaneously target visibility and position uncertainty, the robot takes advantage of landmarks scattered in the workspace to localize itself more precisely so as to better track the target afterward. Our experiments have highlighted the relations between the robot's limited computing resources and the real time constraints imposed by the tracking task. There appears to be a planning horizon that achieves the best balanced compromise between adaptivity and robustness in the tracking robot's behavior. This trade-off strongly depends on the target's escaping capabilities compared to the robot's sensing, planning and moving capabilities. In this work Geometry and Game theory have been successfully combined. By extension, classical planning formalisms and symbolic inference tools could be usefully introduced in order to allow an on-line adaptation of the planning horizon.

Key words:
Game theory, motion planning, visibility constraints, uncertainty, Geometry

⋆ This research was funded in part by ARO MURI grant DAAH04-96-1-007. P.Fabiani was Visiting Scholar at Stanford and supported by French Onera and DGA.

1 Introduction

Dealing with uncertainty is a major problem in Robotics (e.g. see [9] for an overview), which raises and combines a number of issues often associated with estimation, planning or execution control. It is usually unrealistic to assume that a complete and absolutely precise model of the environment can ever be obtained. Yet, a robot is expected to compute and execute robust plans. While Computational Geometry offers tools to deal with geometric constraints (collision, visibility) arising from the robot's workspace, Artificial Intelligence techniques make it possible to make decision in the presence of uncertainty. But the techniques from those two fields have rarely been combined and in fact are seldom compatible.

In this paper we show how methods developed independently can be combined to solve the target tracking problem presented in § 2. In this problem, a mobile robot must keep a moving target into its field of view. The workspace is cluttered by obstacles that create motion and visibility constraints. Related work is presented in § 3. Uncertainties in the robot's self-localization and in the target's behavior add to the complexity of the problem. Game theory [21] offers a convenient framework for both decision making under uncertainty [24] and robot motion planning [17]. The game-theoretic formulation is presented in § 4 and § 5 and the benefits of combining it with geometric algorithms are discussed in § 6. This combination is specifically proposed for the on-line computation and execution of target-tracking motion strategies in the presence of uncertainties (§ 7). We have implemented our techniques and tested them with real robots. Characteristics of the produced motion strategies are discussed in § 8 and § 9. Some remaining limitations and shortcomings are studied in § 10 in connection to the necessary trade-off between adaptivity and robustness. This discussion leads us to propose several extensions in § 11.

2 Tracking problem

A mobile robot equipped with visual sensors, called the *observer*, operates in a workspace cluttered with static obstacles. Its task is to keep another moving object, the *target*, in its field of view while avoiding collision with the obstacles.

All the obstacles create visual occlusion. Hence, the observer must move in such a way that the target is never hidden behind an obstacle. We assume that the workspace is realistically described by a 2-D map (planar layout) of the obstacles. Both the observer and the target are modeled as disks. An accurate map of the obstacles is given to the observer before target tracking begins. The observer uses a camera mounted on a turret to detect the target.

The turret rotates fast enough to make it possible to consider that the observer has omni-directional vision, independent of its current heading. The observer detects and recognizes the target whenever it has a sub-cone of sight to the entire target's disk not obstructed by any obstacle. In our implementation, a black-and-white geometric pattern is mounted on the target to simplify visual detection. There exist more sophisticated vision techniques to detect natural targets (e.g. see [6]).

The target is mostly unpredictable, that is, its future trajectory is not known in advance, but it moves collision-free and its velocity has a known upper bound. The observer localizes itself relative to the workspace using both odometric and landmark techniques. Our *landmarks* are simple geometric patterns placed at the ceiling of the robot environment; they are detected and recognized using a camera mounted vertically on the observer's platform [1]. Natural landmarks could be used as well [8, 15], but would require more sophisticated vision techniques. Proximity sensor readings could be integrated for grid-based Markov localization [2, 13], but would require more elaborate filtering techniques. Whenever the observer sees a landmark, it localizes itself in the workspace with a given precision (on the order of 1-2in) that has been experimentally established. When the robot does not see any landmark, the imprecision of its location estimate grows linearly with the distance traveled (dead-reckoning). To keep our work realistic and compatible with the use of natural landmarks, the artificial landmarks are sparsely distributed over the workspace. The two vision systems used for target tracking and landmark detection, respectively, are independent and operate concurrently.

The observer uses the vision system that tracks the target to estimate the target's position relative to itself and (through a simple transform) to the workspace. Note that the observer cannot track the target by only using their position relative to one another. It must also estimate both its own position relative to the workspace to avoid colliding with the obstacles and the target's position in the workspace in order to move in such a way that it prevents the target from hiding behind an obstacle. The decision loop is sketched in Fig.1.

3 Related work

This tracking problem is a challenging combination of different planning problems that have previously been studied in Robotics, but separately. Visibility and collision constraints must be satisfied in the presence of uncertainties in the positions of the observer and the target.

The problem of maintaining visibility with the target while avoiding collision with the obstacles is addressed in [18], but position uncertainties are not taken into account. When the target is fully predictable, that is, when its future trajectory is completely known in advance, a dynamic programming approach [3] can be used to compute a trajectory of the observer that has minimal length. As shown in [18], this approach becomes untractable in practice if the target is only partially predictable. Then, the approach can be applied to choose a motion command that aims to maximize the likelihood that the target will remain visible during a short interval of time in the future. The observer iterates this computation while tracking the target and updates its motion heading at each iteration. This is essentially the approach taken in this paper (see also Fig.1). The main difference is that we take into account the uncertainties in the observer's and target's positions. This difference will motivate the embedding of this approach into the game-theoretic framework presented in § 4.

Taking position uncertainties into account naturally leads to planning the observer's motions in order to take advantage of the landmarks in the workspace to reduce the imprecision of the estimate of the observer's position (and consequently that of the target), whenever this does not immediately conflict with keeping the target in the observer's field of view. Robot navigation based on active Markov localization was also achieved using only raw proximity sensors [2, 13] in which case the robot may plan motions for position disambiguation. Landmark-based navigation has been addressed from different points of view in the literature (e.g. see [4, 23, 25, 19]). The principle is simple: if the robot primarily localizes itself relative to landmarks, the planner must guarantee that the robot will see landmarks often enough along its path [14].

To our knowledge, no previous work on landmark-based navigation nor on active localization simultaneously deals with achieving a visual task such as maintaining visibility with a moving target [11].

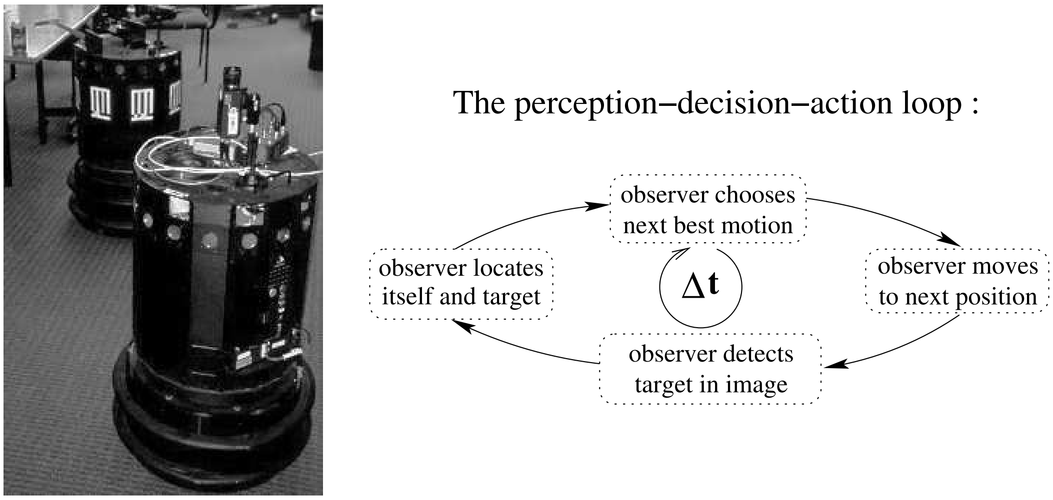

Fig. 1. The observer tracks the target.

4 Game-theoretic approach

Different discrete processes intervene in the behavior of the observer. During each iteration, the observer chooses its next move according to the last estimate of the target's position, executes the selected move, localizes itself (using a landmark, if one is visible ; using pure dead-reckoning, otherwise), detects the target in the new video image and computes a new estimate of the target's position. Meanwhile, the target is moving to another location.

Let t_1 be the initial time point, Δt the duration of the time interval between two sampling time (the duration of the global decision loop - see Fig.1), and $k \geq 1$ an index corresponding to the time point $t_k = t_1 + (k-1) * \Delta t$. The observer and the target move in the same workspace, but since their respective shapes and sizes may differ, their free configurations belong to two different configuration spaces, defined with standard parameterizations [16] and denoted by, respectively $\mathcal{C}^{obs}_{free}$ and $\mathcal{C}^{tgt}_{free}$.

Let q^{obs}_k and q^{tgt}_k denote the respective configurations of the observer and the target at time k. The global state at time k is given by $\omega_k = (q^{obs}_k, q^{tgt}_k) \in \mathcal{C}^{obs}_{free} \times \mathcal{C}^{tgt}_{free}$.

Let $\Omega = \mathcal{C}^{obs}_{free} \times \mathcal{C}^{tgt}_{free}$ denote the set of all possible states. Let $\mathcal{V}^{tgt}(q^{tgt})$ designate the *visibility region* of the target at configuration q^{tgt}, i.e., the set of all configurations q^{obs} from which the observer has an unobstructed cone of sight to the entire body of the target. Let $\mathcal{R}^{obs}_{k+1}(q^{obs}_k)$ (resp. $\mathcal{R}^{tgt}_{k+1}(q^{tgt}_k)$) stand for the set of all the configurations in $\mathcal{C}^{obs}_{free}$ (resp. $\mathcal{C}^{tgt}_{free}$) that are reachable without collision from q^{obs}_k (resp. q^{tgt}_k) at time $k+1$.

Game theory provides a convenient framework to express decision problems in the presence of uncertainty. Yet, most work in that field has focused on the probabilistic part of the theory, especially on the Bayesian computation of the probability distributions. As a matter of fact, probabilistic inference in Bayesian Networks [22], Markov Decision Processes (MDP's) [20] or Partially Observable MDP's (POMDP's [28]) of quite reasonable size may be a task of high computational complexity [5]. Game theory rests on the axiomatization of a relation of preference defined over the possible choices. This relation rests on the definition of both a probability measure and a utility function over the possible consequences. We think that the utility part of the theory is as important as the probabilistic part.

Let us define a decision problem over the triplet $\{\Omega, \mathcal{D}, \mathcal{K}\}$, where Ω is the set of atomic *states*, $\mathcal{D}$ is the set of possible *decisions*, and $\mathcal{K}$ is the set of possible *consequences* $k_{ij} = d_j(\omega_i)$. Each possible decision d_j is a function $\{d_j : \Omega \longrightarrow \mathcal{K}\}$.

A decision problem also requires the definition of a *relation of preference* $\succeq$ over $\mathcal{D} \times \mathcal{D}$. For each $(d, d') \in \mathcal{D} \times \mathcal{D}$, $d \succeq d'$ must be read "d is preferable than d'". Von Neumann's theorem tells us that this relation of preference can be defined as follows [27]:

- Let P be a *probability distribution* over the set Ω of states,
- Let U be a *utility function* over the set $\mathcal{K}$ of consequences,
- Posing:

$$\mathrm{E}_P U(d) \;=\; \sum_{\omega \in \Omega} P(\omega).U(d(\omega)),$$

($\mathrm{E}_P U(d)$ is the expected utility of choosing $d \in \mathcal{D}$) we have:

$$d \succeq d' \quad \text{if and only if} \quad \mathrm{E}_P U(d) \geq \mathrm{E}_P U(d').$$

With this formulation, the best decision $d^\star$ is:

$$d^\star \;=\; arg \max_{d \in \mathcal{D}} \{\mathrm{E}_P U(d)\} \tag{1}$$

In our target tracking problem, suppose that at time $k+1$ the target is at configuration q_{k+1}^{tgt} and the observer has reached q_{k+1}^{obs} after the motion decision d_k^{obs} was made in state (q_k^{obs}, q_k^{tgt}). The new state (q_{k+1}^{obs}, q_{k+1}^{tgt}) is a consequence of d_k. The need that q_{k+1}^{obs} be inside the visibility region $\mathcal{V}^{tgt}(q_{k+1}^{tgt})$ can be expressed in the utility function of the problem.

More precisely, our tracking problem is a sequence of motion decision problems (one at each time k) that are all defined on the same state space $\Omega = \mathcal{C}_{free}^{obs} \times \mathcal{C}_{free}^{tgt}$. The set $\mathcal{K}$ of possible *consequences* is also fixed : as the observer may collide with obstacles by mistake (due to localization errors), there is an additional consequence *coll* (standing for "collision"). The resulting set of consequences is $\mathcal{K} = \{\mathcal{C}_{free}^{obs} \cup \{coll\}\} \times \mathcal{C}_{free}^{tgt}$. At each time k the motion decision problem can be defined as follows:

- The observer's and target's positions are not precisely known at time k. Let P_k^{obs} (resp. P_k^{tgt}) denote the probability distribution of the observer's (resp. target's) position over $\mathcal{C}_{free}^{obs}$ (resp. $\mathcal{C}_{free}^{tgt}$). The joint distribution $P = (P_k^{obs}, P_k^{tgt})$ is the probability distribution over Ω.
- Let $\tilde{q}_k^{obs}$ be the current estimate of the position of the observer according to P_k^{obs}. This estimate can be a singleton or a set of configurations.
- Let $\mathcal{D}_k^{obs}$ be the set of possible motion decisions for the observer from the current estimated configuration $\tilde{q}_k^{obs}$. In practice, $\mathcal{D}_k^{obs}$ is finite.
- The utility function U_k over $\mathcal{K}$ (here, the set of possible consequences at time $k+1$) can be simply defined to be 1 if the observer sees the target at time $k+1$ and 0 if it does not see the target or has collided with an obstacle.

More formally:

$$
\begin{aligned}
U_k(coll,\, q^{tgt}_{k+1}) &= 0 && (2)\\
U_k(q^{obs}_{k+1}, q^{tgt}_{k+1}) &= 1 \;\text{ if }\; q^{obs}_{k+1} \in \mathcal{V}^{tgt}(q^{tgt}_{k+1}) && (3)\\
&= 0 \;\text{ otherwise} && (4)
\end{aligned}
$$

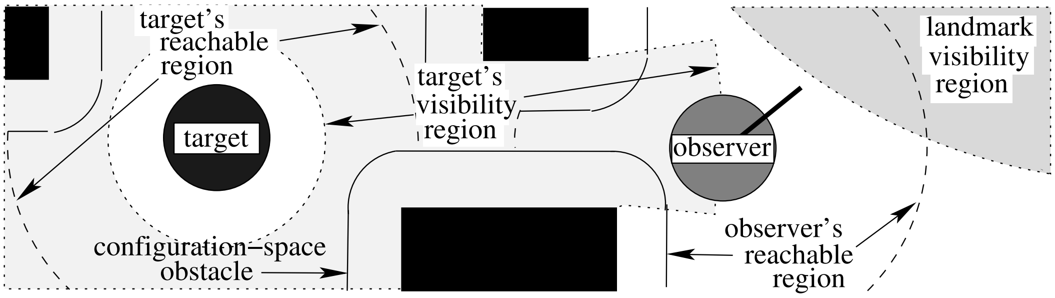

Fig. 2. Geometric constraints and utility regions

Geometric constraints illustrated in Fig.2 directly affect the evaluation of the above utility function. Is the observer colliding with obstacles, or not? Is the observer in the visibility region of the target? As we will see in the next section, they also affect the computation of the probability distributions P_k and P_{k+1}, hence that of the best motion decision defined by Equ. (1): can the observer see a landmark?

In our experiments, we used the utility functions U_k defined by Equ. (2)-(4), but more generally, U_k can be chosen to give a higher penalty to collision, to penalize states where the observer is close to loosing the target and/or where the observer's turret needs to rotate by a large amount. U_k can also be designed to take into account the speed of the observer at stage k in order to advantage future configurations that correspond to small accelerations of the observer in either direction.

5 Modeling uncertainties

In our work, we only consider uncertainties caused by inaccuracies in sensing, in motion control and in lack of prior knowledge of the target's trajectory. Uncertainty in the observer's location depends critically on whether it sees a landmark, or not : if so, then the observer updates its position and its positioning uncertainty decreases. Fig.3 is based on snapshots of the graphic display of our planner's output (see also Fig.8). The arrows and circles show the evolution of the observer's position and uncertainty depending on which option is chosen : on the left the observer moves without updating its position, on the right the observer updates its position thanks to a landmark. Continuous circles of variable radius represents the uncertainty on the position of the center of the observer (see also Fig.7 & 8). Any intersection of these circles with the configuration-space obstacles (see Fig.2) corresponds to a risk of collision. The dashed circles drawn on each snapshots with a dashed tail represent the observer's next configuration.

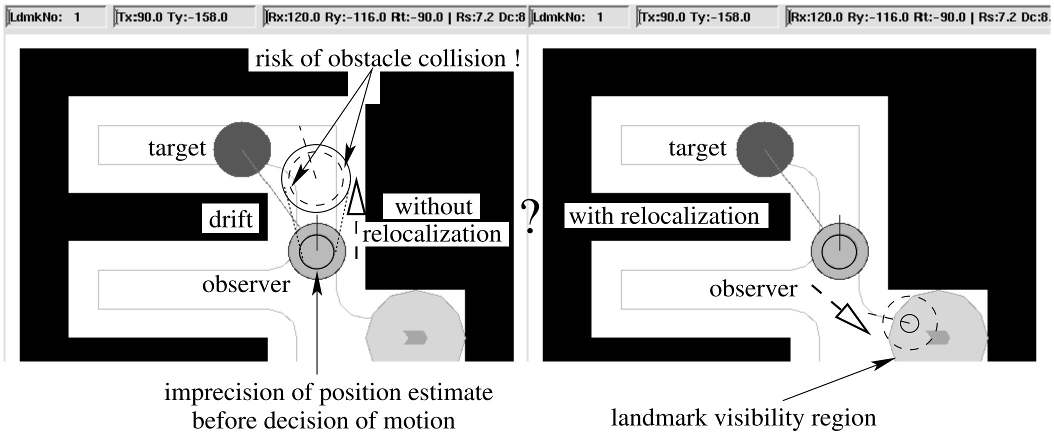

Fig. 3. The localization dilemma

Whenever the observer moves and no landmark is visible, dead-reckoning errors accumulate. Consequently, the uncertainty on the position of the observer increases (see the snapshot on the left in Fig.3) and the observer may risk collision along a supposedly collision-free path. We model the configuration actually reached at time $k+1$, q^{obs}_{k+1}, as a random variable that is conditionally dependent upon both the observer's configuration at time k, q^{obs}_k, and the decision made at time k, d^{obs}_k.

The conditional probability $Pr(q^{obs}_{k+1} \mid d^{obs}_k, q^{obs}_k)$ is given *a priori* and allows the computation (at time k) of the predictive probability distribution $\hat{P}^{obs}_{k+1}$ at time $k+1$ which models the imprecision of dead-reckoning.

$$\forall q^{obs} \in \mathcal{C}^{obs}_{free},\ \hat{P}^{obs}_{k+1}(q^{obs} \mid d^{obs}_k) = \sum_{\substack{q^{obs}_- \in \mathcal{C}^{obs}_{free} \\ d^{obs}_k \in \mathcal{D}^{obs}_k}} Pr(q^{obs} | q^{obs}_-, d^{obs}_k) P^{obs}_k(q^{obs}_-) \quad (5)$$

$$\hat{P}^{obs}_{k+1}(coll \mid d^{obs}_k) = 1 - \sum_{q^{obs} \in \mathcal{C}^{obs}_{free}} \hat{P}^{obs}_{k+1}(q^{obs} \mid d^{obs}_k) \quad (6)$$

If no landmark is visible at time $k+1$, the predictive probability distribution $\hat{P}^{obs}_{k+1}$ cannot be updated and the best estimate of the observer's location at time $k+1$ is the one given by dead-reckoning, that is $P^{obs}_{k+1} = \hat{P}^{obs}_{k+1}$.

If the observer sees a landmark, the localization uncertainty decreases at time $k+1$. Let loc^{obs}_{k+1} be a boolean variable equal to 1 whenever the observer reaches a position at time $k+1$ where it detects a landmark, and 0 otherwise (case where no landmark is visible or the observer fails to detect the landmark). When a landmark is detected, the positioning accuracy that is obtained after the updating process at time $k+1$ depends both on the predictive probability law $\hat{P}^{obs}_{k+1}$, and on the experimentally established precision of the localization. This is modeled by the updating function $\mathcal{L}^{obs}$ in the following equation:

$$
\begin{aligned}
P^{obs}_{k+1} &= \mathcal{L}^{obs}(\hat{P}^{obs}_{k+1}) \quad \text{if} \quad loc^{obs}_{k+1} = 1 && (7)\\
&= \hat{P}^{obs}_{k+1} \qquad\quad \text{if} \quad loc^{obs}_{k+1} = 0 && (8)
\end{aligned}
$$

Consequently, the probability distribution P^{obs}_{k+1} at time $k+1$ depends both on $\hat{P}^{obs}_{k+1}$ and on the observation of a landmark, if a landmark has been detected. P^{obs}_{k+1} is given by equations (5) through (8) as a function of the probability distribution P^{obs}_k, the decision d^{obs}_k made at time k, and the value of the variable loc^{obs}_{k+1}. This is summarized in the following *evolution equation*, where the function $\mathcal{F}^{obs}$ models both the possible motion errors and the reduction of positioning imprecision whenever a landmark is detected.

$$
\begin{aligned}
P^{obs}_{k+1} &= \mathcal{F}^{obs}(P^{obs}_k, d^{obs}_k, loc^{obs}_{k+1}) && (9)\\
P^{obs}_{k+1}(coll) &= 1 - \sum_{q^{obs} \in \mathcal{C}^{obs}_{free}} P^{obs}_{k+1}(q^{obs}) && (10)
\end{aligned}
$$

Similarly, the observer can only obtain an imprecise estimate of the target's position relative to its own current position (see Fig.4). Let $\tilde{P}^{tgt}_k$ denote the probability distribution over the possible configurations of the target after detection by the observer at time k. The uncertainty on the position of the target in the workspace combines the uncertainty on the target's position relative to the observer and the uncertainty on the observer's position in the workspace. Hence, $\tilde{P}^{tgt}_k$ depends on both P^{tgt}_k and P^{obs}_k (see Fig.4).

The target's partial predictability between times k and $k+1$ can be expressed by $P^{tgt}_{k+1} = \mathcal{S}^{tgt}(\tilde{P}^{tgt}_k)$. Since the target's trajectory is not known in advance, $\mathcal{S}^{tgt}$ depends only on the known target's maximal velocity. This means that the uncertainty on the target's position at time $k+1$ (as predicted at time k) may be much greater than for the observer. On the other hand, the target is observed at every time step before the prediction stage, which considerably reduces uncertainty. The evolution equation for the target takes into account how the probability measure P^{tgt}_k is updated at time k by the observer and how it evolves from time k to time $k+1$ due to partial predictability of the target:

$$
P^{tgt}_{k+1} = \mathcal{F}^{tgt}(P^{tgt}_k, P^{obs}_k) \tag{11}
$$

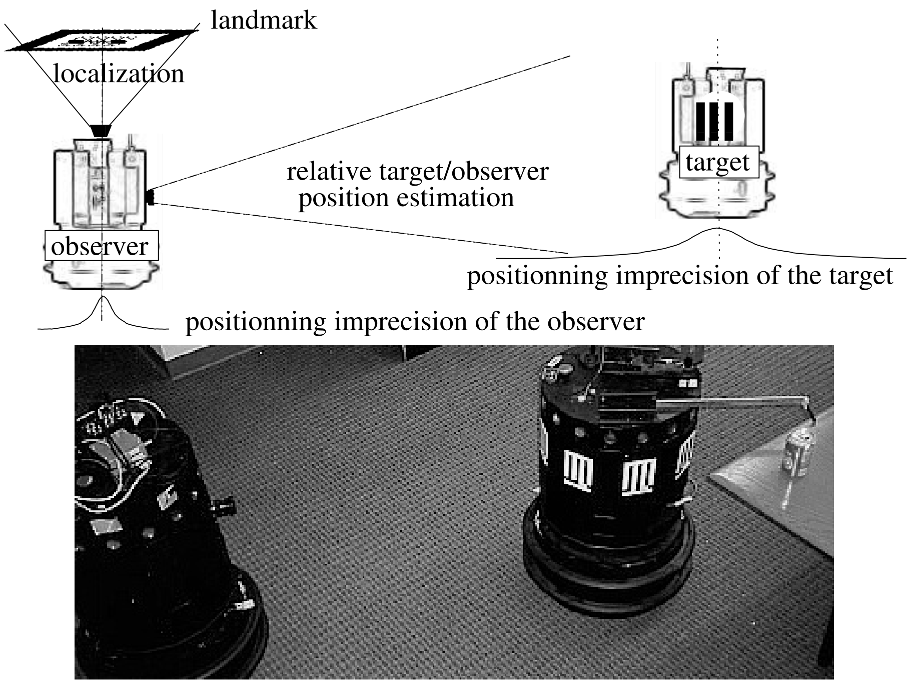

Fig. 4. Position imprecisions

6 Benefits of combining Geometry and Game theory

From a game-theoretic point of view, $P_k = (P_k^{obs}, P_k^{tgt})$ is the *information state* of the observer at time k : it represents the observer's beliefs according to which a decision can be made. The evolution equation for P_k follows from Equ. (9) and (11) :

$$P_{k+1} = \mathcal{F}(P_k, d_k^{obs}, loc_{k+1}^{obs}) \tag{12}$$

This equation is classical in Markov Decision Processes (MDP) and Partially Observable Markov Decision Processes (POMDP) [28]. But, as is well known, MDPs and POMDPs are highly dependent on the choice of the discretization of the workspace, and they rapidly become untractable in high-dimensional state spaces. Here, the observer's and target's workspaces are basically 2-D continuous spaces and Equ. (12) does not depend on any discretization choice. In practice, however, both workspaces are systematically discretized to allow the application of numerical techniques.

Nevertheless, a change in the resolution of the discretization only affects computation times by a polynomial factor and does not change the problem definition. The discretization must be selected fine enough to fit our precision needs for the representation of the robot and the target. The coarser acceptable discretization in our experiments (see § 8) gives approximately 1500 possible free configurations for both the target and the observer.

For each free configuration of the observer, there is an average of 40 states in which the observer can see the target. Assuming perfect localization of both the target and the observer, the tracking problem could easily be turned into an MDP, but the size of the state space of this MDP would be around 60000 states. Adding localization uncertainties, the dimension of the information space of the corresponding POMDP would then be around 60000, which, to our knowledge, is out of reach of the existing resolution algorithms [5].

In order to reduce our tracking problem to a tractable POMDP, one could define fewer larger states and precompute motion primitives for the observer and the target to navigate within and between these states. Then, the corresponding matrices of transition probabilities between states and the matrices of observation probabilities (encoding localization imprecisions) would have to be derived experimentally, or by learning techniques. Such a simplified POMDP-like formulation of our tracking problem seems difficult to extract automatically from the initial problem description, though this has already been done for simpler indoor navigation problems.

Instead, we proposed in [11] to exploit the geometry of the problem to reduce its dimensionality. The utility functions need not be evaluated on every state, but rather on geometric regions of the state space, essentially visibility and reachability regions. This subsequently reduces the regions over which probabilities have to be computed. Each region comprises several states with homogeneous or easy-to-compute properties. Thus, despite the number of discretized states, probability and utility functions are obtained through a relatively small amount of calculation (Equ. (2)-(4)). Besides, thanks to the fact that all obstacles are static and known in advance, the visibility and reachability regions are precomputed to reduce on-line computation times .

7 Tracking strategies

According to the problem formulation given in § 4-6, at each time step, the observer decides where to move according to both probability distributions P_k^{tgt} and P_k^{obs}. If $P_k = (P_k^{obs}, P_k^{tgt})$ denotes the information state of the observer at time k, a *strategy* π for the tracking problem with finite time horizon h ($h \geq 1$) is a sequence $\{ \pi_k \}_{k \in \{1,..,h\}}$ of *decision rules* $\pi_k(P_k) = d_k^{obs}$. An optimal strategy can be computed by discrete dynamic programming [28]. Let denote $V^h(P_1)$ the *optimal value function* for the h-stage tracking problem beginning with an initial information state P_1 at time t_1, i.e. the maximum expected reward that the observer can get with this initial condition.

Bellman's optimality equations give :

$$\forall k \in \{1, h\}, \quad V^{h-k+1}(P_k) = \max_{d \in \mathcal{D}^{obs}(P_k)} \left\{ \mathrm{E}_{P_{k+1}} U_k(d) + V^{h-k}(P_{k+1}) \right\} \tag{13}$$

where $V^0(.) = 0$ and $P_{k+1} = \mathcal{F}(P_k, d, loc^{obs})$.

Unfortunately, the cost of computing an optimal strategy is exponential in the horizon h. This difficulty is well-known in the field of POMDPs [28] where dynamic programming is nevertheless widely used. In our target tracking problem, a greater h would yield better strategies if the computation time was negligible [11, 7]. But since this is not the case and the target moves while the observer computes its next motion, h is necessarily limited. This problem is further commented on in § 10.

8 Experimental setup

Fig. 5. Video sequences of the robots in their environment

Experiments were conducted to test the game-theoretic approach presented above : the observer and the target are two Nomad-200 mobile robots of Nomadic Technologies (Fig.5). Each Nomad-200 has a rigid cylindrical body that allows its representation as a disk in the 2-D workspace (Fig.3). The observer detects landmarks placed at the ceiling of the environment. In Fig.5, little stickers are visible on the floor ; each of which corresponds to the position of a landmark at the ceiling. The observer can track the target regardless of its current heading, thanks to a camera that is kept aimed at the target by servoing the observer's turret appropriately.

All the obstacles in the environment are modeled by polygons in the 2-D workspace. They obstruct both motion and visibility. Actually, some obstacles in the environment (e.g. the ladder in Fig.5) do not obstruct visibility completely. However, in practice they disturb the observer's vision modules and cause the observer to lose track of the target if they are not treated as view-obstructing obstacles. More sophisticated geometric models for the obstacles are possible. Simple probabilistic models of the robots were used in the evolution equations (Equ. (9)-(11)). Their good behavior in practice results from the fact that they are rather conservative. Although they could be refined, it can be argued that little improvement is to be expected from more sophisticated probabilistic models, since so little is known about the target's future motion anyway.

Our experiments with the Nomad 200 robots were conducted in the Robotics Laboratory at Stanford. Fig.6 shows a full screen snapshot of the graphic displays of our experimental setup. The same screen has been taped during the dynamic demonstration of the observer's behavior with our planner. Sample sequences of the video tape were digitalized : mpeg movies are available at *http://underdog.stanford.edu/* or *http://www.cert.fr/dcsd/cd/stanford/* .

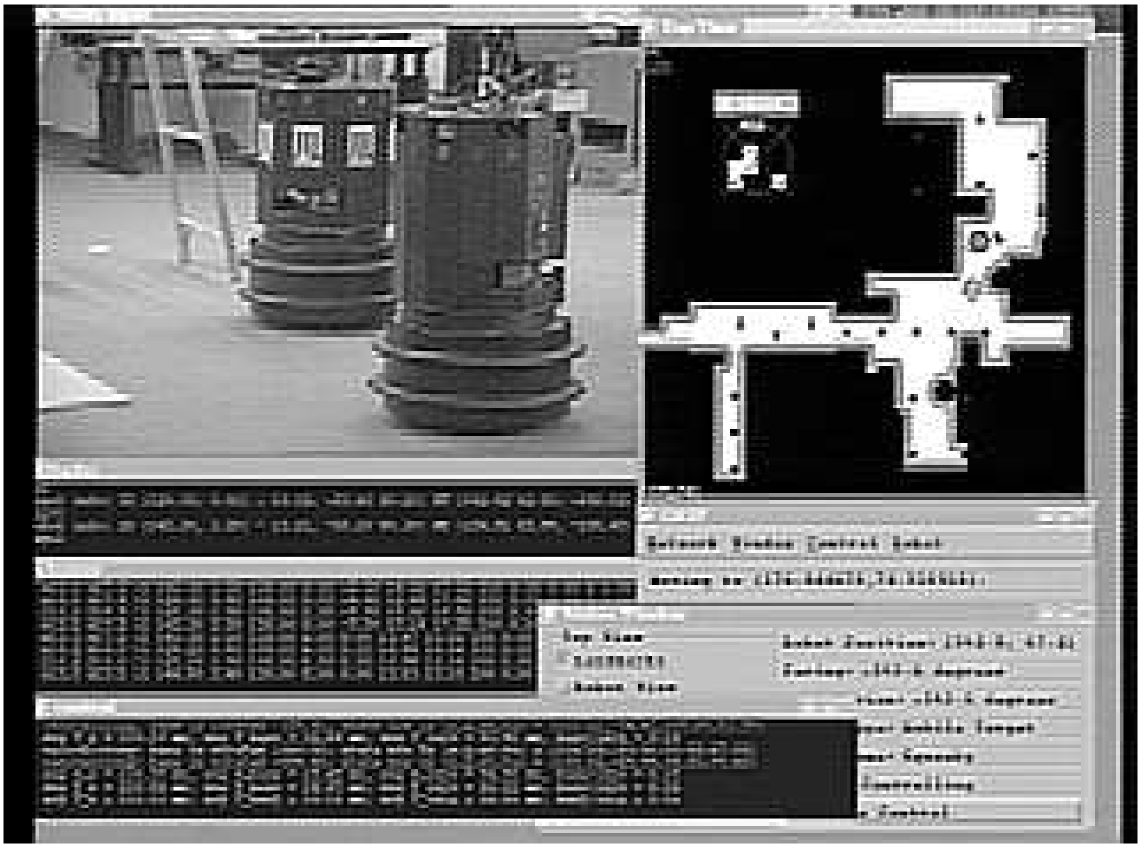

Fig. 6. Graphic display of the dynamic demonstration (*http://underdog.stanford.edu/* or *http://www.cert.fr/dcsd/cd/stanford/*)

In Fig.6 the upper left window shows a live video from a camera tracking the evolutions of the observer and the target in the main hall of the laboratory. The target is easily recognizable thanks to the black-and-white geometric patterns mounted on it. The obstacles sketched out in the 2D lay out can be visually identified with the one appearing in the video images. The lower right windows correspond to the control panels of the graphic displays. The lower left windows show the real time outputs of the different processes running on the robots : planning, image processing and motion controls.

The upper right window is the graphic display of the observer's current belief state. It shows the observer's current estimate of the respective position of each robot in the laboratory. The 2-D layout includes the obstacles, the free space and the positions of the ceiling landmarks. The small subwindow in the upper left quarter of this window is a picture of the last recognized ceiling landmark, i.e. a black-and-white geometric pattern that is barred with a circled cross indicating that the observer does not see the landmark any longer.

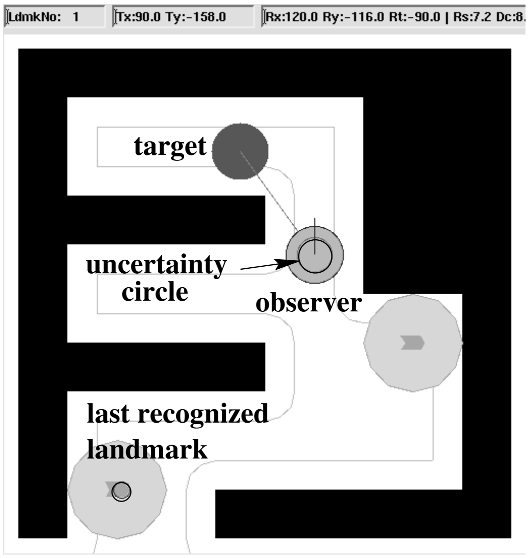

Fig. 7. Snapshot

The display of the 2-D layout and the observer's belief state can be seen in more details in the sample snapshot of Fig.7 where the same graphic conventions are used. In both images Fig.6 and Fig.7 the target appears as a dark disk and the observer as a light-grey disk with a little tail. The other static light-grey disks represent landmark regions (regions in which the observer's center must lie in order to detect a landmark). The location of the last recognized landmark is marked with a small circle. The observer's positioning uncertainty is depicted by a continuous circle with variable radius and with the same center as the observer's disk (these circles had to be highlighted manually to be more clearly visible in Fig.8). The landmark that was detected the most recently, recognized and used for location updating by the observer is marked with a small circle in the center.

Some of these features are hardly perceivable in the movies and even less in Fig.6. We use Fig.8 to comment on the observer's behavior with our planner. The arrows describe each agent's moves. Please note that the *target's* moves are represented by arrows reflecting the *target's next move* in the scenario, whereas for the observer the arrows reflect the *subsequent move* that is performed by the observer in response to the *observed move* of the target at the *previous stage*.

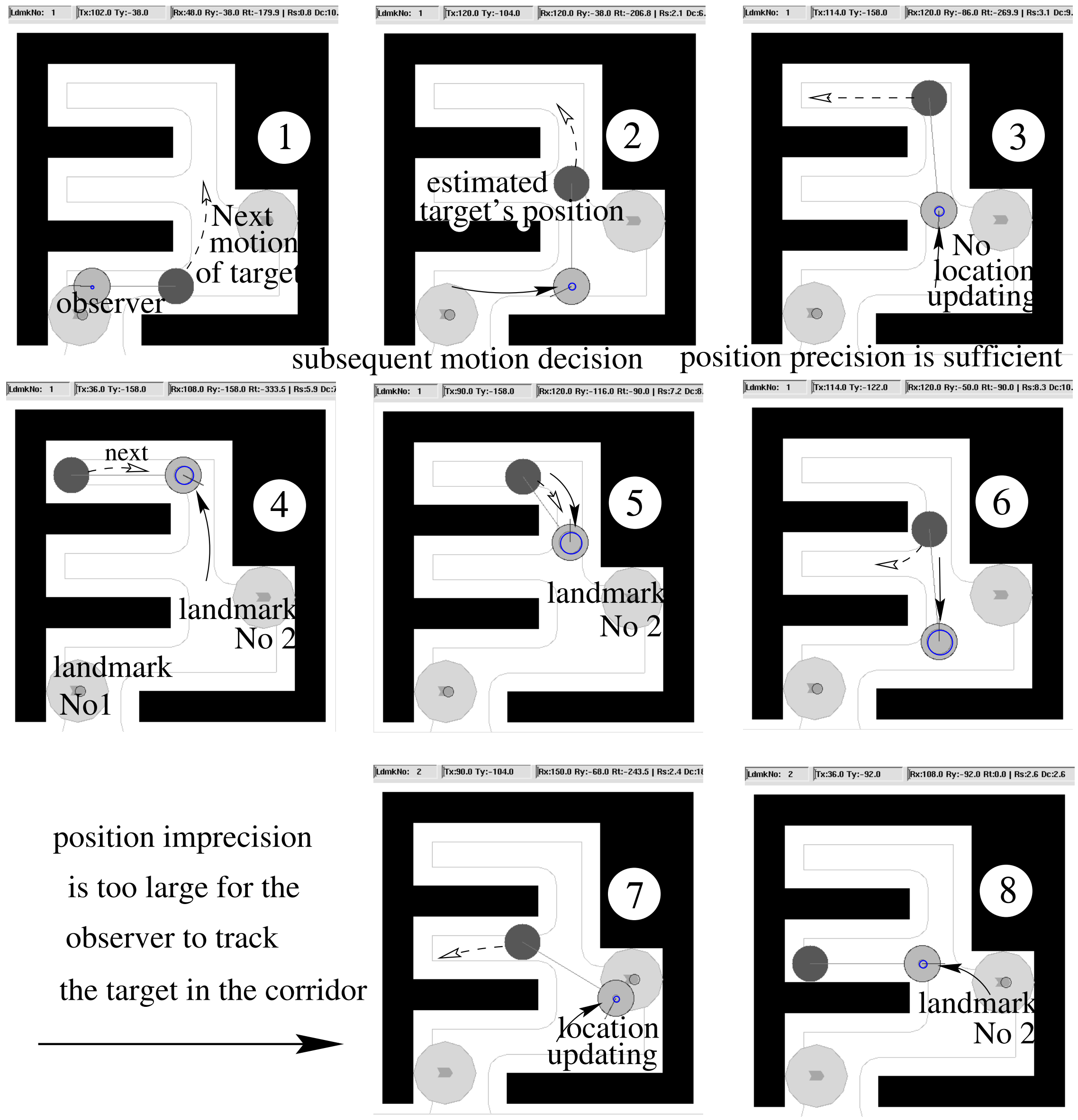

Fig. 8. Sample sequence of motion decisions

9 Experimental behavior

Our experiments successfully validated our approach. The observer was able to manage properly its position uncertainty. Whenever a landmark visibility region is reachable, the motion decision corresponds to the best compromise between the tracking task and the need for location updating. The observer localizes itself when useful in order to better track the target afterward.

More specifically, Fig.8 shows an 8-step decision sequence generated by the observer's planner in response to the target's moves. This sequence is illustrated by snapshots of the graphic display of the planner's output.

The following features should be noted in the observer's behavior in Fig.8:

- The observer takes its position uncertainty into account before choosing its next move: it localizes itself when useful, but does not need to do so when it can travel in a wide enough free area (stage 1-6);
- The observer goes and localizes itself so as to reduce its positioning imprecision (if possible) prior to entering a narrow region (stages 7-8);
- The observer needs to minimize its probability of collision in order to maximize the probability of visibility of the target at next time step;
- In practice the observer's probability of collision remains equal to zero whenever possible (stages 1-8 here), unless the observer gets trapped by the target in a narrow region where it cannot reach any landmark visibility region in order to localize itself.

Thus, our planner outputs a robust decision (with probability 1 of keeping the target in sight) whenever there exist one on the planning horizon and optimizes a trade-off otherwise. Given the imposed reactivity constraints in our initial experiments [11], our planner produces robust solutions on the short term, but sometimes not on the longer term.

10 Further experiments : robustness vs. reactivity

Further experiments were conducted in order to study and improve the balance of deliberation and reactivity [7] with our planner.

Balancing deliberation and reactivity is a very important issue for autonomous systems acting in uncertain and changing environment. In our case, achieving the best balance between deliberation and reactivity consists in determining the best value of Δt, if any. A correlated motivation for further experimenting was to assess the impact of having a limited planning horizon.

10.1 *Limitations*

The dynamic behavior of the observer with our planner was further tested with different values of the planning horizon h and the response time Δt, here assimilated to the deliberation time (see Fig.1 - $h.\Delta t$ is the temporal horizon of prediction of the planner). When changing the values of these two parameters, the observer is given more or less reactivity to the target's moves. As a matter of fact, increasing Δt gives more deliberation time for the planner between two observations of the target. However, it also gives more time for the target to move meanwhile. When Δt is increased, the number of reachable states for the robots grows exponentially faster than the number of possibilities that the planner is able to take into account.

10.2 *Adaptivity*

We call *adaptivity* the planner's ability to return a solution that is in *adequacy* with the true situation of the robots at execution time. In other words, it is the ability to provide a plan quick enough so that the prediction hypotheses about the environment (here the target's position) are still valid. In that sense, we show here that there is an optimal value for Δt with respect to *adaptivity* (Fig.9 and Equ.14).

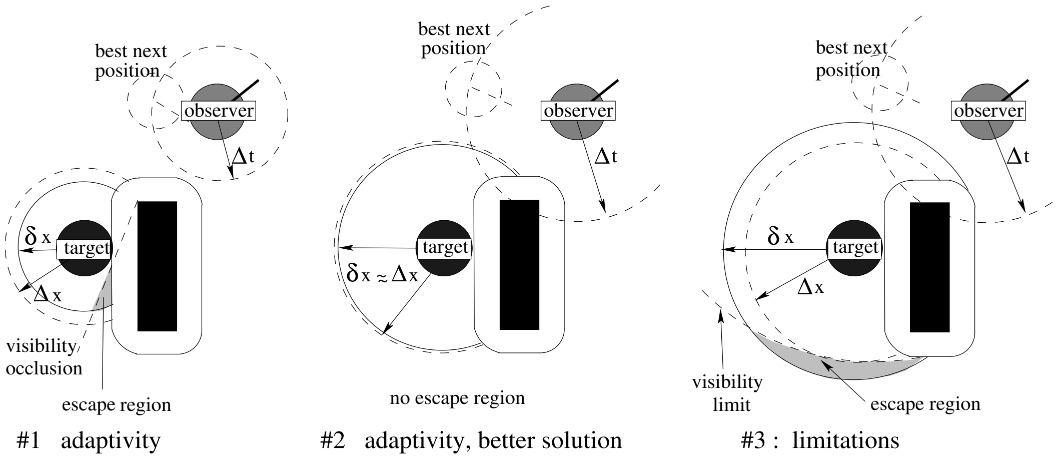

Fig. 9. *Adaptivity* as the ability to provide adequate plans within time constraints.

Two time scales are competing here :

- Δx is the distance that the target is predicted to travel during deliberation time Δt, $\Delta x = V_{max}.\Delta t$, if V_{max} is predicted to be the target's speed limit ;
- δx is the distance that the target can actually travel during the time $\delta t_{plan}(h, \Delta t)$ that is actually needed by the planner to take into account all the predictable moves within the temporal horizon of prediction.

In Fig.9, the dashed circles correspond to the planner's predictions and the continuous circles to what can actually happen : the observer's reachability region is dashed and limited by a Δt radius, which must be understood as the distance traveled within this time. Meanwhile, the target is expected to stay within Δx but can actually travel δx. In case #1 of Fig.9 the planning time δt_{plan} is short enough to take into account all the actually possible moves of the target ($\delta x < \Delta x$). Yet, h or Δt could be increased in order to find a better next position for the observer, as shown in case #2. In case #3, the planning time is too long and the target has enough time to escape out of the reachable region predicted by the *planner* : $\delta x > \Delta x$. In this case, part of the actually reachable region is even outside the visibility region of the next position of the observer, which creates an *escape region* (in grey).

For the planner to be *adaptive*, we actually need (and it is enough) to have :

$$\delta t_{plan(h,\Delta t)} < \Delta t \tag{14}$$

10.3 Robustness

We call *robustness* the planner's ability to produce a plan that will guarantee the achievement of its goal whatever the unpredictable or uncontrollable events that can occur in the environment (here the target's moves) during the completion of the *perception-decision-action* loop. In that sense, a *robust* decision, is a decision that guarantees that the observer will keep the target in sight with probability 1.

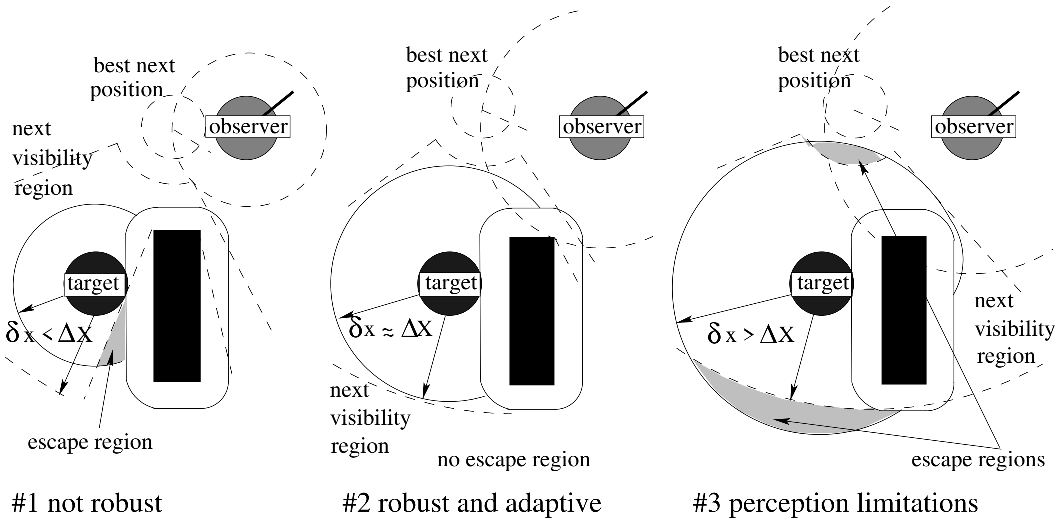

Fig. 10. Increasing the planning time may lead to find (more) robust solutions, until perception or time limitations interfere.

In order to find *robust* solutions to the tracking problem, the planner should use all the available CPU-time [7] within the limits given by Equ.(14).

However, the limited perception capabilities of the observer introduce additional limitations : increasing the deliberation time Δt enlarges the *reachable region* of the target up to a point at which it *"cannot fit in"* the observer's field of view anymore : then the observer needs more time in order to scan the target's reachable region. In case #1 of Fig.10, the planning time δt_{plan} should be increased in order to allow a larger planning horizon and find the robust position for the observer that is shown in case #2. In case #3, the planner uses extra *deliberation* time, yet with no better result : there appear two new escape regions due to perception limitations.

When the observer's perception capabilities are limited, two time scales are competing :

- ΔX is the maximum radius of the region of the state space in which the observer is *"certain to find the target in one observation"*, i.e. to keep track of the target at next time step - ΔX is limited by the perception capabilities of the observer ;
- ΔT is the time for the target to move by a distance of ΔX ;
- $\delta t_{plan(\Delta X)}$ is the time for the planner to compute the best moving decision taking into account all possible target's moves of length $\leq \Delta X$.

For the planner to be *robust*, we need :

$$\delta t_{plan(\Delta X)} \quad < \quad \Delta T \tag{15}$$

It should be noted that, Equ.15 is only a necessary condition : if $\delta t_{plan(\Delta X)} < \Delta T$, then the observer is not guaranteed to keep track of the target, because the target's reachable region may also be partially occluded by obstacles as in cases #1 of Fig.10 and Fig.11.

10.4 Discussion : toward perception planning

During our experiments, the on-line computation speed of our planner first had to be improved, and the planning time consequently reduced, before we could really observe and study the impact of both the robustness and the adaptivity constraints on our system. Then, examples were experimentally observed where the solution found by the planner with a longer deliberation time is actually worse than with a shorter one : this phenomenon was especially due to the perception limitations and the unpredictability of the target's moves (Fig.11). Hence, the question of the impact of a limited planning horizon finds a very pragmatic answer : there may be cases were there exist a robust solution on a shorter temporal horizon of prediction and no better (nor even satisfactory) one on a longer horizon.

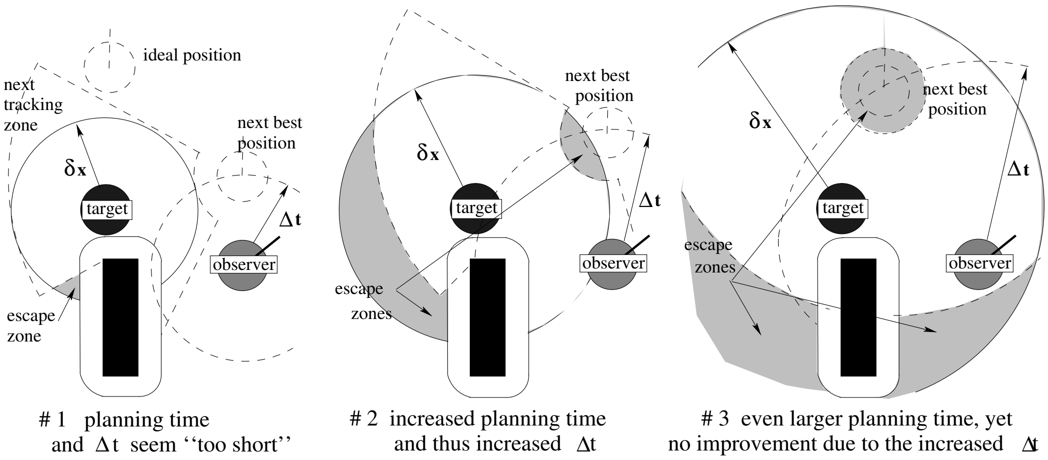

Fig. 11. Sketch of an observed case where increasing the planning time worsens the obtained solution.

Our experiments highlighted the existence of an optimal deliberation time and temporal horizon of prediction with respect to a compromise between adaptivity and robustness. The best balance between deliberation and reactivity is obtained with the maximal deliberation time such that both adaptivity and robustness constraints are satisfied :

$$(h, \Delta t) = \arg\max\Big\{\delta t_{plan(h,\Delta t)} \ / \ \delta t_{plan(h,\Delta t)} \ < \ \min(\Delta T,\ \Delta t)\Big\}$$

Our experiments show the importance of achieving the right balance between reactivity and deliberation in autonomous systems acting in an uncertain and changing environment. The above equation shows that the optimal deliberation time depends upon various parameters that could also change with time, in which cases the planning horizon should be adapted *on-line*.

This problem seems very important to us even if many difficulties are to be overcome in the general case. The question would deserve to be studied in its full generality, which is not the purpose of this paper. As a matter of fact, searching for the best planning horizon may demand to reason with general knowledge about the planning problem itself : the best planning horizon in order to obtain a robust decision may not be the same as for the optimization of a trade-off. Besides, optimizing the planning horizon requires both to take into account the robots limitations and to plan for actions of perception (target detection and robot localization in our case). Perception planning capabilities are then required, which can lead to a number of open problems in the general case [26] but also to very promising developments. Techniques involving a combination of both symbolic (or geometric) reasoning and decision-theoretic planning seem interesting to investigate further in that perspective.

11 Perspectives : between satisfaction and optimization

Autonomous systems acting in uncertain and changing environments need action and perception planning capabilities either to achieve a specified goal in a state space, or to perform the best level of service on a given time period.

Geometry, symbolic reasoning or classical (STRIPS-like) planning techniques preferably apply to the first category of problems. With such *symbolic* planning approaches, the goal is to find a solution achieving a specified goal given a number of constraints, i.e. a problem of *satisfaction* of a sequence of transition conditions leading to a termination condition. In other cases, the problem may be to find the best plan a given temporal horizon and according to a given criterion, i.e. a problem of *optimization*. Our tracking problem lies in the second category. Yet, producing a robust plan (e.g. conformant planning), or even a robust closed loop strategy (e.g. contingent planning) is closer to a *satisfaction problem*, like in classical (STRIPS-like) planning. In our case, the problem is eventually solved by planning intermediate goals (the limited planning horizon), falling thus a little more in the first category. This allowed the use of some symbolic techniques by combining Geometry and Game theory.

As a matter of fact, most of the time, robots have to deal with situations where a robust solution may not exist on the planning horizon imposed by adaptivity constraints and searching for a robust plan on a longer planning horizon may not be tractable. In these cases a trade-off has to be achieved. Symbolic planning approaches cannot produce such *"optimized compromises"*. In the latter case, decision-theoretic planning provides a powerful framework, but leads to search for optimal solutions in high-dimensional information state spaces, which may become untractable. Besides, when there exists a robust solution, this approach is far less efficient than symbolic techniques that exploit efficiently the problem structure.

In our tracking problem, we were able to reduce considerably the computational complexity of the planning task by combining successfully geometric techniques and game-theoretic planning. Similar benefits can be expected in more general domains, still by combining the efficiency of both symbolic and decision-theoretic approaches. Such a combination should allow the planner to apply respectively its reasoning and optimizing capabilities at the right level of representation of its environment ([12] describes some preliminary work in that direction). For example, the challenge can be to determine at planning time the best horizon on which a robust decision can be made or an adapted plan can be optimized (i.e. within the time constraints). Ideally, a robust solution should be found whenever there is one (completeness), and a compromise should be optimized on the longest tractable horizon otherwise.

12 Conclusion

The nice experimental behavior of the designed planner shows that the choice of modeling this problem in a game-theoretic framework was more than a theoretical exercise: it should allow geometric algorithms to be efficiently combined with algorithms designed for POMDPs, Bayesian estimation, Markov localization or Bayesian Networks. Still, this would need to be further formalized with more generality, so to be applicable to a wider range of problems.

Furthermore, the proposed game-theoretic framework allows dealing with symbolic constraints (e.g. *"a robot cannot see/travel through walls"*) and reasoning with probabilistic uncertainty to be efficiently combined through the computation of well-adapted utility functions. The introduction of reasoning mechanisms, such as in Fuzzy Sets theory [23] or Possibility theory [10] seems to be a promising approach for reasoning with utility functions.

On the other hand, classical planning approaches could be usefully combined with Game theory in order to boost our planner's performances when searching for a robust solution (satisfaction of the goal with certainty), while keeping the current abilities to optimize compromises when facing more uncertainties.

The same kind of ideas actually appear in [5] about MDPs and POMDPs. In that sense, such combinations with more classical action and perception planning formalisms and symbolic inference tools should aim at automatically allowing an on-line adaptation of the planning horizon and deliberation time to the current situation. In that perspective, the conducted experiments provide interesting ideas for future developments.

Lastly, our experiments led us to considerations that are more related to situation assessment and updating problems : for instance, whenever the observer happens to estimate the position of the target on a configuration that is not collision-free. This may be due to location and detection errors, but also to the fact that some obstacles in the modeled workspace of the robots do not exist in the real environment. Our 2-D layout of the workspace actually is a very conservative model of the real indoor environment. Dealing with an incomplete and inaccurate map in full generality is a really tough challenge. The proposed game-theoretic approach should provide an interesting framework in order to release progressively the hypothesis of a perfectly known environment.

References

[1] C. Becker, J. Salas, K. Tokusei, and J.C. Latombe. Reliable navigation using landmarks. In *ICRA*, 1995.

[2] M. Beetz, W. Burgard, D. Fox, and A.B. Cremers. Integrating active localization into high-level robot control systems. *Robotics and Autonomous Systems*, 23(Special issue for SIRS'97):205–220, 1998.

[3] D.P. Bertsekas. *Dynamic programming : deterministic and stochastic models*. Prentice-Hall, 1986.

[4] B. Bouilly and T. Siméon. A sensor-based motion planner for mobile robot navigation with uncertainty. In Dorst et al. [9].

[5] G. Boutilier, T. Dean, and S. Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.

[6] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 8–15, 1998.

[7] R. Chatila. Deliberation and reactivity in autonomous mobile robots. *Robotics and Autonomous Systems*, 16(2-4):197–211, 1995.

[8] M. Devy, R. Chatila, P. Fillatreau, S. Lacroix, and F. Nashashibi. On autonomous navigation in a natural environment. *Robotics and Autonomous Systems*, 16(1):5–16, 1995.

[9] L. Dorst, M. van Lambalgen, and F. Voorbraak, editors. *Reasoning with Uncertainty in Robotics*. Springer, mar 1996.

[10] D. Dubois and H. Prade. Possibility theory as a basis for qualitative decision theory. In *IJCAI95*, pages 1924–1930. Montreal, 1995.

[11] P. Fabiani and J.-C. Latombe. Dealing with geometric constraints in game-theoretic planning. In *IJCAI'99*, pages 942–947, Stockholm, aug. 1999. Morgan Kaufmann.

[12] P. Fabiani and Y. Meiller. Planning with tokens: an approach between satisfaction and optimisation. In *PUK'2000 14th Workshop "New Results in Planning, Scheduling and Design"*. ECAI'00, aug 2000.

[13] D. Fox, W. Burgard, and S. Thrun. Active markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(3-4):195–207, 1998.

[14] T. Fualdes and C. Barrouil. Perception planning for an U.G.V. *Robotics and Autonomous Systems*, 11(2), 1993.

[15] P. Hébert, S. Betgé-Brezetz, and R. Chatila. Probabilistic map learning: Necessity and difficulties. In Dorst et al. [9], pages 307–321.

[16] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, 1991.

[17] S.M. LaValle. *A Game-Theoretic Framework for Robot Motion Planning*. PhD thesis, University of Illinois, Beckman Institute, 1995.

[18] S.M. LaValle, H. Gonzalez-Banos, C. Becker, and J.C. Latombe. Motion strategies for maintaining visibility of a moving target. In *ICRA*, 1997.

[19] A. Lazanas and J.C. Latombe. Motion planning with uncertainty: a land-

mark approach. *Artificial Intelligence*, 76(1-2), 1995.

[20] M.L. Littman, T. Dean, and L.P. Kaelbling. On the complexity of solving markov decision processes. In *UAI'95*, pages 394–402, Providence, jul. 1995.

[21] R. D. Luce and H. Raiffa. *Games and Decisions.* John Wiley & Sons, New York, 1957.

[22] J. Pearl. *Probabilistic reasoning in intelligent systems : networks of plausible inference.* Morgan Kaufmann, San Mateo, 1988.

[23] A. Saffiotti and L.P. Wesley. Perception-based self-localization using fuzzy locations. In Dorst et al. [9], pages 368–385.

[24] L.J. Savage. *The Foundations of Statistics.* Dover, New York, 1972.

[25] H. Takeda, C. Facchinetti, and J.C. Latombe. Planning the motions of a mobile robot in a sensory uncertainty field. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(10), 1994.

[26] PLANET Robot Planning TCU. Open workshop on perception planning. In *PLANET Newsletter.* http://www.cert.fr/dcsd/cd/planet, aug 2000.

[27] J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior.* Princeton University Press, 1944.

[28] D.J. White. *Markov Decision Processes.* John Wiley & Sons, Chichester, 1993.