Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT

ELSEVIER

Cognitive Systems Research 4 (2003) 243–282

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

Map-based navigation in mobile robots: I. A review of localization strategies

Action editor: Risto Miikkulainen

David Filliat[a,*], Jean-Arcady Meyer[b]

[a] *DGA/Centre Technique d'Arcueil, 16 bis Av Prieur de la Cote d'Or, 94114 Arcueil Cedex, France*
[b] *AnimatLab-LIP6, 8 rue du Capitaine Scott, 75015 Paris, France*

Received 29 June 2002; received in revised form 29 January 2003; accepted 24 February 2003

Abstract

For a robot, an animal, and even for man, to be able to use an internal representation of the spatial layout of its environment to position itself is a very complex task, which raises numerous issues of perception, categorization and motor control that must all be solved in an integrated manner to promote survival. This point is illustrated here, within the framework of a review of localization strategies in mobile robots. The allothetic and idiothetic sensors that may be used by these robots to build internal representations of their environment, and the maps in which these representations may be instantiated, are first described. Then map-based navigation systems are categorized according to a three-level hierarchy of localization strategies, which respectively call upon direct position inference, single-hypothesis tracking, and multiple-hypothesis tracking. The advantages and drawbacks of these strategies, notably with respect to the limitations of the sensors on which they rely, are discussed throughout the text.

Keywords: Autonomous mobile robot; Map-based navigation; Localization strategies

1. Introduction

In a recent review, Guillot and Meyer (2001) emphasized the animat contribution to cognitive systems research. In particular, they stressed the capacity of this approach to integrate both the body and the control in the quest for understanding intelligence in living systems, a discourse that a steadily growing number of researchers elaborate according to various modalities (Brooks, 1991a,b; Clark, 1999; Hara & Pfeifer, 2000; Pfeifer & Scheier, 1999; Varela & Rosch, 1991).

An animat is a simulated animal or a real robot that permanently interacts with its environment through its sensors, its body and its actuators, and that must continuously cope with many concurrent, and possibly contradictory, needs and goals (Meyer & Wilson, 1991; Meyer, Roitblat, & Wilson, 1993; Cliff, Husbands, Meyer, & Wilson, 1994; Maes, Mataric, Meyer, Pollack, & Wilson, 1996; Pfeifer, Blumberg, Meyer, & Wilson, 1998; Meyer, Berthoz,

[*] Corresponding author.
E-mail addresses: david.filliat@etca.fr (D. Filliat), jean-arcady.meyer@lip6.fr (J.-A. Meyer), http://animatlab.lip6.fr (J.-A. Meyer).

1389-0417/03/$ – see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S1389-0417(03)00008-1

Floreano, Roitblat, & Wilson, 2000; Hallam, Floreano, Hallam, Hayes, & Meyer, 2002). Animat designers, wishing to contribute to our understanding of human intelligence and cognition, approach such endeavors in an evolutionary perspective—according to which intelligence and cognition are supposed to be rooted in basic adaptive capacities inherited from animals. Therefore, their effort to investigate the interactions between an animat and its environment particularly focuses on the animat's aptitude to survive in unexpected environmental circumstances.

In the light of these characteristics, the animat approach is complementary to that of traditional artificial intelligence. Instead of directly modeling human-specific and isolated capacities—like problem solving, natural language understanding or logical reasoning—it addresses basic adaptive capacities that man shares with other animals—like those of perception, categorization and motor control—in both a bottom-up and an integrative perspective.

This article will review various models where such capacities are integrated within a framework highly important in the animat literature, i.e. that of navigation tasks. Indeed, the ability to navigate—making it possible to reach any goal place from any starting point, while avoiding passing through unwanted places—is probably the most basic requirement for an animat's survival. Clearly, without such an ability, the animat would not be able to reach energy sources, to avoid bumping into damaging obstacles, or to escape from dangerous hazards. However, among many navigation models and strategies that animals and robots may use to this end (see Trullier, Wiener, Berthoz, & Meyer, 1997; Franz & Mallot, 2000, for recent reviews), this article will focus on map-based navigation, in which internal representations of the spatial layout of the animat's whole environment will be used, thereby making detour behavior and goal-directed movement planning possible. Basically, such models will cover the three last navigation strategies described by Trullier et al. (1997), namely those of place-recognition-triggered response, topological navigation and metric navigation. Likewise, they will call upon the three varieties of knowledge that, according to Werner, Krieg-Bruckner, Mallot, & Schweizer (1997), are involved in spatial cognition of both humans and robots: landmark, route and survey knowledge.

Map-based navigation seems quite natural to humans because using a map is a very convenient way to describe an environment and to share it with other people. However, the human use of a map requires a lot of high-level cognitive processes in order to interpret the map and to establish the correspondence with the real world. Often, this correspondence is made easier by modifying the environment, for example, by writing names on posts indicating subway stations.

The first research efforts in robotic map-based navigation were mainly inspired by these cognitive processes, assuming that errors in sensing and acting may be detected and corrected by a high-level cognitive process, or using some sort of environmental modification to make the navigation process easier. However, some ethological studies suggested that animals also made use of maps for navigation (for example the *cognitive map* hypothesized by Tolman, 1948). Such hypotheses gained support with the identification of *place cells* in rodent brains (O'Keefe & Dostrovsky, 1971). These place cells are neurons, found notably in a part of the brain called the hippocampus, that have an activity correlated with the rat's position in its environment. Experimental studies show that the activity of these cells largely depends on visual cues, but that they are also sensitive to animal motion, as they are still active in the dark. This kind of map-based navigation is a much more appealing paradigm for robot map-based navigation, as it does not presuppose high-level cognitive processes and is able to work in natural and unmodified environments. Because several robotic navigation models inspired by these biological examples have been designed, we will include them in this review, along with robotic navigation models designed without such an inspiration, so as to point out their respective capacities and differences.

Basically, map-based navigation calls upon three processes (Levitt & Lawton, 1990; Balakrishnan, Bousquet, & Honavar, 1999).

- Map-learning, the process of memorizing the data acquired by the robot during exploration in a suitable representation.
- Localization, the process of deriving the current position of the robot within the map.

- Path planning, the process of choosing a course of action to reach a goal, given the current position.

The third process is simply dependent on the first two, as the current position and the map of the environment between this position and the goal are required to plan actions toward this goal. The first two processes are also closely related. The *chicken-and-egg* nature of their relation (Yamauchi, Schultz, & Adams, 1999; Kurz, 1995) arises from the fact that the map is needed to estimate the position and that, conversely, the position is needed in order to build the map. This relation makes the problem of simultaneous localization and map-learning[1] very difficult.

Conceiving a review on map-based navigation also entails tackling the above-mentioned chicken-and-egg problem. To solve this we have chosen to limit the scope of this article to localization strategies that rely on a fully-known map of the environment. The issues that arise when localization and map-learning are tackled simultaneously, together with a survey of planning methods, are the subject of another review to be published in a companion paper (Meyer & Filliat, 2003). The reason for this order is that most of the literature on navigation in animals and humans addresses localization but not map-learning. Our first goal is therefore to provide elements about localization methods in robots that could be compared with those of animals and humans, before trying to describe how the corresponding underlying maps have been built and how they are used.

This article will start with the description of what sort of information may be used by a robot to localize itself and of how this information may be stored in a map. Then, it will review localization strategies and sort them according to their use of allothetic and idiothetic information. Incidentally, it should be noted that, because the field is the subject of considerable research efforts, the choice of these models is not exhaustive but, rather, seeks to be representative of the numerous strategies currently implemented in the field.

2. Useful information for map-based navigation

Two distinct sources of information may be used for map navigation. The first is the idiothetic source, which provides internal information about the robot's movements. The second is the allothetic source, which provides external information about the environment.

Idiothetic information may concern speed, acceleration, leg movement for animals, or wheel rotation for robots. A straightforward integration of these data results in a position estimate for the robot in a 2D space. This process is referred to as dead-reckoning or path-integration. The term idiothetic data used in this review is borrowed from biological literature. It corresponds to what is called odometry in robotics.

Allothetic information may be derived from vision, odor, or touch, for animals, and laser range-finder, sonars or vision, for robots. Here again, the word allothetic is borrowed from biology; it corresponds to expressions like observation, perception or sensor data in robotics literature. There are two main uses of this information source.

- Data may be used to directly recognize a place or a situation. In this case, any cue—such as sonars time-of-flight, color or odor—may be used.
- A metric model may be used to convert raw allothetic data into information expressed in the 2D space related to the idiothetic data. In this case, geometric properties of the environment, such as object positions, are inferred (Fig. 1). This may be straightforward, as is the case for laser-range finders or sonars, or more complicated, as for stereo-vision.

Using a model to convert raw allothetic information into a 2D space may be quite difficult. In fact, sensor measurements depend not only on the intrinsic characteristic of the sensor, but often also on the local properties of the environment (for example sonar sensor values depend on the material of the walls). This makes reliable metric models quite difficult to obtain, as these models depend on the properties of both the local environment and the sensor.

It must be noted, however, that such a dependence

[1] Often referred to as the SLAM—Simultaneous Localization And Mapping—problem.

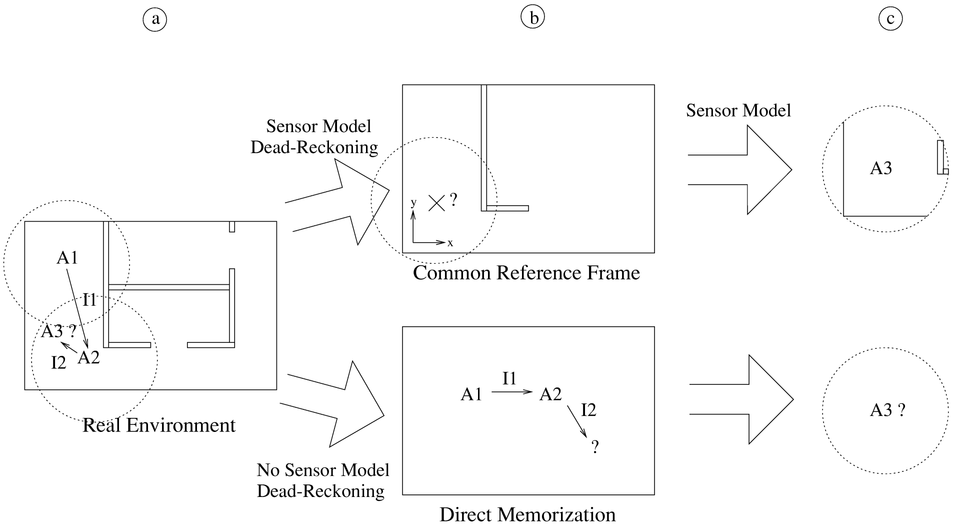


Fig. 1. A sensor model may be used to infer allothetic cues which should be available in unvisited places. In this example, allothetic cues $A1$ and $A2$ are collected in two places related by idiothetic cues $I1$ (part *a*). Using a metric model for allothetic sensors allows information to be fused in a common reference frame where objects are represented (part *b*, top), whereas, without a metric sensor model, only a set of places characterized by allothetic cues and related by idiothetic information is memorized (part *b*, bottom). Using a sensor model, these data may then be used to infer allothetic cues $A3$ in a new position related to a previous position by idiothetic cues $I2$ (part *c*, top). Without such a model, only visited places may be recognized, and no inference can be made about unvisited places (part *c*, bottom).

on the environment only becomes an issue when trying to infer metric information about the environment. When using raw sensory data to characterize a position, the fact that a given sensor response depends on the local properties of the environment is fully integrated in the place definition and is not problematic.

The gain obtained with such a metric model is quite frequently worth the effort. Indeed, the first consequence of the use of a metric model is that it makes it possible to fuse allothetic and idiothetic information in a common geometric reference frame which is quite natural and expressive for human operators (Fig. 1b).

Another important consequence is that a metric model makes it possible to infer allothetic information about parts of the environment that have not been physically explored by the robot, but that have been sensed from other locations. This property arises from the fact that a sensor model allows allothetic information to be inferred from the map for different locations (Fig. 1c). Thus, if a robot senses a wall 3 m ahead and moves forward 1 m, it can estimate that the wall is now 2 m away. A related consequence is that the relation between allothetic information acquired in two places can be used to infer the relative position of these places (Fig. 2). Thus, if a robot senses a wall first 3 m ahead and subsequently 2 m ahead, it can estimate that it has moved forward 1 m.

The drawbacks and advantages of these two sources of information are complementary. Indeed, the main issue raised by idiothetic information is that, because it involves an integration process, it is subject to *cumulative error*. Its quality accordingly decreases continually, so that such information cannot be trusted over long periods of time. On the contrary, the quality of allothetic information is stationary over time, but it suffers from the *perceptual aliasing* problem, i.e. the fact that, for a given sensory system, two distinct places in the environment may appear the same. A related problem raised by allothetic cues is that of *perceptual variability* (Kuipers & Beeson, 2002) which occurs when a

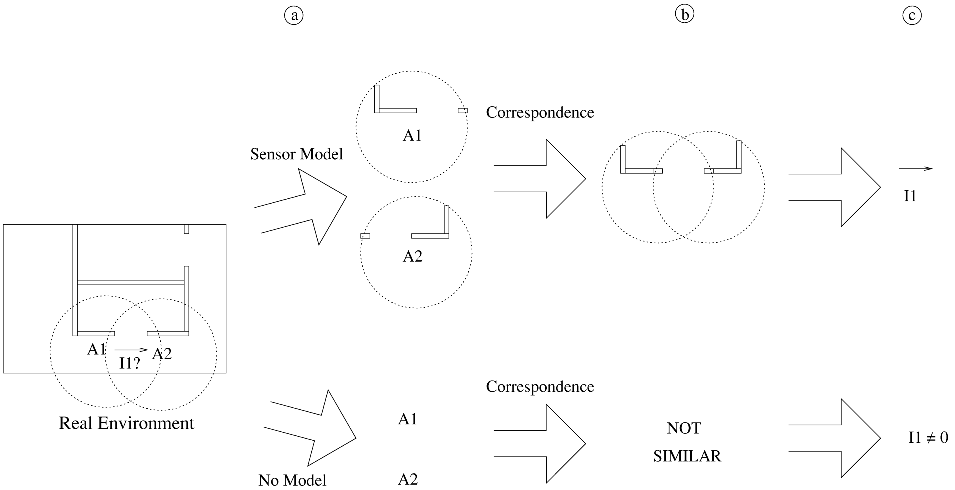


Fig. 2. A sensor model may be used to infer the relative position $I1$ of two places where two sensory situations $A1$ and $A2$ were recorded (part *a*). This is achieved by comparing the two allothetic situations. When using a sensor model, finding this correspondence first entails finding an object which is common to the two situations (part *b*, top). Using this common object makes it possible to infer the relative metric positions of the two places (part *c*, top). When no sensor model is used, only the similarity of the two situations can be assessed (part *b*, bottom). It is then only possible to conclude whether the two situations are the same or not and whether $I1$ is null or not (part *c*, bottom).

given place looks different over time because, for example, of changes in light conditions. This second problem often trades off against the first one because using only features that are not subject to perceptual variability often leads to poor discrimination between places, and hence to perceptual aliasing. On the contrary, trying to characterize more precisely a given perception to avoid perceptual aliasing often makes the system sensitive to perceptual variability. In this review, we concentrate on the perceptual aliasing problem, assuming that, in most models, the choice has been made to design perceptual systems that are as resistant to perceptual variability as possible. However, some models explicitly take the problem of perceptual variability into account (Kuipers & Beeson, 2002), while other models (such as most of those of Section 4.5) are designed to be robust to noise in the perceptions, and are consequently little concerned by it.

The consequence of these properties is that, in order to reliably navigate for long periods of time, the two sources must be combined (Cox, 1991). In other words, allothetic information must compensate for idiothetic information drift, while idiothetic information must allow allothetic information to be disambiguated.

3. Map representations

Given the allothetic and idiothetic sources of information, there are many ways to integrate them in a representation useful for robot navigation. Classically, the corresponding models are separated into two categories resorting to either metric or topological maps. In metric maps, the positions of some objects, mainly the obstacles that the robot can encounter, are stored in a common reference frame. On the contrary, within topological maps, allothetic definitions of places the robot can reach are stored, along with some information about their relative positions (Fig. 3).

3.1. Metric maps

In the metric framework, the environment is represented as a set of objects with coordinates in a 2D space. As idiothetic information makes it pos-

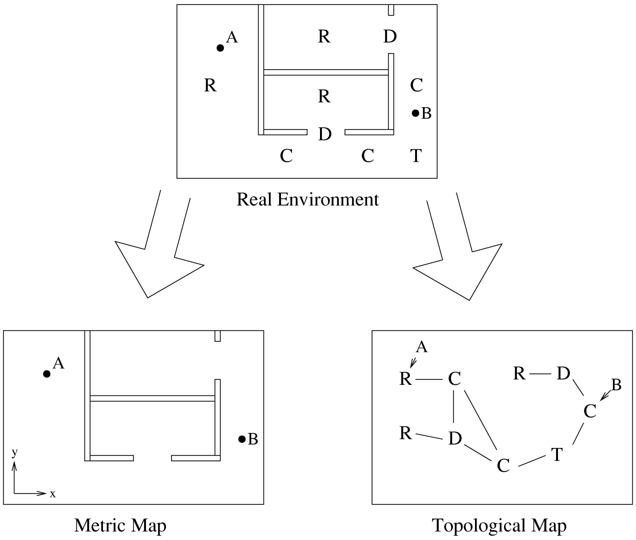


Fig. 3. Illustration of the classical distinction between metric and topological maps. In the metric framework, object positions are inferred and represented in a common reference frame. Two positions, *A* and *B*, are represented in this map by their coordinates in this reference frame. These coordinates make it possible to infer their distance. In topological maps, places are stored with their spatial relations. Two positions *A* and *B* in the environment may be recognized as being part of places *R* and *C*. This makes it possible to infer that position *B* can be reached from position *A* via places *C*,*C* and *T* (in this figure, R = room, C = corridor, D = door and T = turn).

sible to directly monitor the robot position in this space, this source of information is usually important in this representation. As explained in Section 2, allothetic information is stored after transformation in the 2D space by means of a metric model. This transformation yields a set of objects, or obstacles, along with their positions relative to the robot. The key difference with respect to topological maps stems from this use of sensor models, which allows the fusion of idiothetic and allothetic information in a common reference frame.

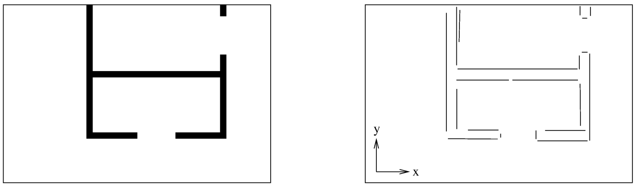
The position estimate is continuous in the 2D space and therefore usually much more precise than it is in the topological framework. Moreover, metric maps display the layout of the environment in a way similar to an architectural sketch, which is easy to read for humans. This objective view of the environment, which is rather independent of any given robot, also makes it easy for different robots to reuse such maps. Metric maps are easier to build than topological maps because of the non-ambiguous definition of locations afforded by their coordinates (Thrun, 1999).

The difficulty in obtaining a reliable sensor model is an important issue in metric map-building. As mentioned in the previous section, this difficulty arises from the fact that such a model may depend not only on a given sensor, but also on the local properties of the environment. Metric map-building also often heavily depends on the quality of the position estimated by idiothetic cues. The drift of this estimation is difficult to correct without assumptions about particular properties of the environment (such as orthogonal walls). Moreover, path planning is often computationally expensive because, contrary to topological maps, no natural high-level environment discretization is available.

3.1.1. Feature representation

Metric maps can explicitly store features that may be perceived by a robot, along with their positions (Fig. 4). There is a wide choice in such represented features, as well as in the abstraction level of the representation.

Points, or objects considered as punctual, may be used (Levitt & Lawton, 1990; Prescott, 1995; Feder, Leonard, & Smith, 1999). This choice corresponds to the intuitive definition of a landmark as a reference point. However, as the perception of a single point does not allow the robot's position to be inferred,



Real environment

Feature map

Fig. 4. Example of a feature map containing segments detected on the obstacles' boundaries.

several points must be perceived. Moreover, uniquely identifying a landmark may be quite challenging. To overcome this difficulty, it is possible to record points scattered over object surfaces (Lu & Milios, 1997; Gutmann & Konolige, 2000; Thrun, Bugard, & Fox, 2000). Such points are usually gathered using laser range-finders that are able to detect several points on obstacle surfaces. Then, uniquely identifying each point becomes useless, since only the spatial configuration of a set of points is necessary to define an object.

In order to gain a more precise contribution to position estimation from a single feature, corners extracted from laser-scans may be used (Borghi & Brugali, 1995), thus providing additional orientation information thanks to the two lines that define the corner. Hébert, Betgé-Brezetz, & Chatila (1996) and Smith, Self, & Cheeseman (1988) also provide models where objects are represented by points with an associated orientation.

Obstacles or object boundaries may also be represented in a metric map. Lines defining polygonal boundaries may be used. These lines are often extracted from sets of points detected by sonars sensors (Dudek & MacKenzie, 1993; Gasós & Martin, 1997), or by laser range-finders (Moutarlier & Chatila, 1990; Einsele, 1997; Castellanos, Montiel, Neira, & Tardos, 1999). Cylinders and planes detected by sonars sensors are used by Leonard, Durrant-Whyte, & Cox (1992) and higher-level features, such as planes containing lines detected by stereo-vision, are used by Ayache & Faugeras (1989).

Representing uncertainty in the values of such features is crucial for many systems, as it plays an important role in deciding whether a measure corresponds to a feature or not. This is often achieved by estimating the variances of the object parameters (Smith et al., 1988; Ayache & Faugeras, 1989; Moutarlier & Chatila, 1990; Leonard et al., 1992; Hébert et al., 1996; Feder et al., 1999; Castellanos et al., 1999). Leonard et al. (1992) also assign a credibility value to each feature in order to model the confidence that a given object is really present in the environment and not the result of a perception error. Fuzzy sets may also be used to represent feature position uncertainty (Gasós & Martin, 1997).

3.1.2. Free-space representation

Instead of using a set of features to represent objects in the map, it is possible to represent the portion of the environment that is accessible to the robot. The most popular approach to this idea is the *occupancy grid* (Moravec & Elfes, 1985; Thrun, 1999; Yamauchi et al., 1999) (Fig. 5). In this case, the environment is discretized into a regular high-resolution grid. Each cell of the grid is assigned a probability of being occupied by an obstacle. As this method entails using a lot of memory to represent large environments, irregular space discretization may be used (Arleo, del Millán, & Floreano, 1999). The great advantage of these methods is that they can directly use sensor data without the need for feature extraction, often either computationally expensive or brittle.

3.2. Topological maps

In the topological framework, the environment is represented by a set of *distinctive places* (Kuipers & Byun, 1991) and by the way a robot can go from one place to another. Place definitions call upon allothetic information available at the corresponding position in the environment. Some idiothetic information collected while going from one place to another is also usually stored in the links of the

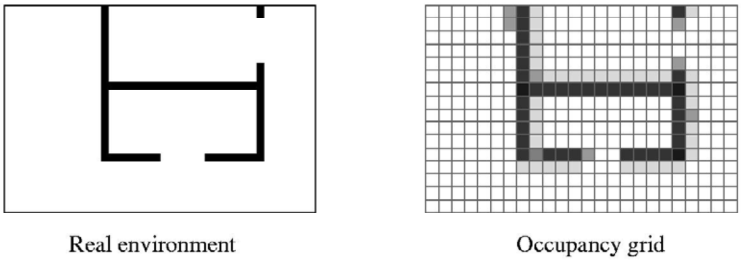


Fig. 5. Example of occupancy grid. The environment is discretized in regular cells and the grey value of a given cell indicates the probability for the considered position of being occupied by an obstacle, from 0 for white to 1 for black.

graph relating the different places. In its common sense, a topological map is therefore a sparse representation of the environment that only represents key places for navigation using allothetic data, on the one hand, and connexions between these places using idiothetic data, on the other hand. In this paper, the definition of a topological map will be extended to every map that records allothetic and idiothetic data separately, and that is not used to infer relative positions of places thanks to the allothetic data. This definition clearly covers all the maps classically defined as topological, but also includes some maps which call on a very fine and regular discretization of the environment. Within the framework of this paper, this definition is justified by the fact that similar localization and map-learning methods are associated with all these maps, which are different to techniques used with metric maps.

The first advantage of topological maps is that they do not require a metric sensor model to convert allothetic data in a common 2D reference frame. The only requirement is a method for storing place definitions and for recognizing places given a sensory situation, which, incidentally, may be conveniently achieved through a metric model. Topological maps are closely related to the robot's perceptual capacities and do not require extracting an objective representation of the environment. Moreover, this memorization of the environment as a set of places calls upon a discretization of the spatial layout that is often directly useful for higher-level processes. For example, this representation is very convenient for planning or problem-solving, because the size of the corresponding search-space is small compared to the set of possible trajectories in a continuous 2D space. This discretization may also be very natural when it calls upon places defined by humans, such as corridors and rooms. This allows problems to be described and solved in a way meaningful to humans (for example, giving the order *go to room 10*, instead of *go to point x,y*).

However, these advantages are offset by the disadvantage that allothetic information is only available for places physically explored by the robot, thus requiring a more exhaustive exploration of the environment when higher precision for position estimation is needed. Another difficulty lies in the definition of places, which may be hard in case of unreliable sensors or in a dynamic environment. This definition is made even more difficult in case of perceptual aliasing (see Section 2). As a consequence, topological maps may be hard to build in large scale environments, because erroneous place recognition results in faulty map topology, which may be hard to detect and correct. The lack of an objective description of the environment may also turn out to be a problem if the goal is to let humans or other robots reuse this map.

3.2.1. Node definition

As a topological map provides an intrinsic discretization of the environment, the first issue when designing a mapping system is to choose which places to represent, i.e. when to perform localization and map updates. This decision may be conditioned by human choices, or may be completely dependent upon the particular environment the robot will be confronted with.

3.2.1.1. Operator-defined nodes

Places that must be represented may be directly defined by a human operator. In such cases, the robot is endowed with procedures able to detect predefined types of places, which it must locate in its environment. The most common choice is to use corridors, doors and intersections (Kunz, Willeke, & Nourbakhsh, 1997; Dedeoglu, Mataric, & Sukhatme, 1999; Shatkay & Kaelbling, 2002; Hertzberg & Kirchner, 1996). In this situation, a lot of perceptual aliasing occurs (when, for example, individual intersections cannot be recognized); therefore correct localization and mapping rely heavily on internal information (Section 4.4.1).

3.2.1.2. *Nodes defined at canonical places*

Instead of completely specifying nodes, the designer can specify where places have to be detected, leaving the robot the task of precisely defining each place. This is the case in the *distinctive places* approach pioneered by Kuipers & Byun (1991). In their model, places are defined as areas from which a hill-climbing control law is able to guide the robot toward a locally unique point (for example, a corridor intersection place is defined as the area from where a unique control law can guide the robot to a precise point, at the junction of two corridors). A given place is then characterized by the allothetic situation at the point reached by the control law. This technique affords a solution to the point-of-view problem, which arises from the discretization of the real world by humans into places which are not characterized by a unique allothetic situation from the robot point-of-view.[2] Engelson & McDermott (1992) exploit the same idea, of which Kortenkamp & Weymouth's (1994) *gateways* are another instance. Gateways are defined as places detected by sonar sensors where the robot can change its travel direction, which are therefore important for navigation (such as doors leading from one room to the other, or such as corridor intersections).

Within such a framework, a sensory situation is recorded at each canonical place encountered in order to more precisely define the corresponding node. Place definitions are quite general in Kuipers and Byun's model; however, these authors report that simply using the distances of neighboring obstacles measured by sonar sensors is not precise enough to disambiguate all places and that the additional memorization of an occupancy grid representing the surroundings of the distinctive place proves to be necessary. Engelson and McDermott suggest recording signatures of images taken at canonical places to define each node. An image signature is a set of measurements (such as the color distribution) made on an image and that characterize it. Such a compact representation of the image (compared to the original image size) allows efficient storage and image comparison. Similarly, Kortenkamp & Weymouth (1994) use a low-dimension representation of images, named *Abstracted Scene Representation*, to characterize nodes in the map.

3.2.1.3. *Automatically defined nodes*

A third method for defining nodes is to define them as areas where perceptions are similar, regardless of the robot's position in such areas. This method naturally circumvents the point-of-view problem. Place definitions are simply obtained through unsupervised categorization of allothetic cues, the category of a perception corresponding to the place where the robot is located. Such node definition is based on the robot's perceptual capacities alone and does not depend on the human definition of what a place should be, which make places easier to recognize, given a single perception. Moreover, such space discretization is the natural choice of models which try to mimic some animal navigation capabilities, because it does not presuppose any high-level definition of places (such as door or corridor). Places can by defined by the local configuration of landmark directions or distances (Levitt & Lawton, 1990; Sharp, 1991; Burgess, Recce, & O'Keefe, 1994; Bachelder & Waxman, 1994; Gaussier, Joulain, Banquet, Lepretre, & Revel, 2000; Balakrishnan et al., 1999; Touretzky, 1994), by the values of proximity sensors (Nehmzow & Owen, 2000; Mataric, 1992; Kurz, 1995; Duckett & Nehmzow, 1997), or by some characteristics of panoramic images (Arleo & Gerstner, 2000; Franz, Scholkopf, Georg, Mallot, & Bulthoff, 1998; Ulrich

[2] The point-of-view problem can make map-learning quite difficult because of the possible definition of places by multiple allothetic situations. The reason is that an allothetic situation never encountered before does not always correspond to a new place, but may correspond to a known place under a different point-of-view.

& Nourbakhsh, 2000; Von Wichert, 1998). These models are detailed in Section 4.3.1.

When such a strategy is chosen, in the absence of any a priori definition of what the places to represent are, a criterion has to be designed to determine when the robot has reached a new place. As places are supposed to represent almost constant sensory situations, the obvious choice is to monitor the variations of the sensory situation and to consider that a new node is reached when this variation exceeds a given threshold. This is used in several models (Mataric, 1992; Kurz, 1995; Franz et al., 1998; Nehmzow & Owen, 2000; Gaussier et al., 2000), but requires that the situation be monitored in real-time, which may be difficult on real robots. Other models simply define a new place when the distance from the previous place exceeds a threshold (Touretzky, Wan, & Redish, 1994; Yamauchi & Beer, 1996; Von Wichert, 1998; Arleo & Gerstner, 2000).

Note that in all these node-definition methods a metric model for allothetic cues may be used for convenience. However, in this topological framework, such a model is not used *to infer information about relative positions of places* (see Section 2), which depends on idiothetic cues only, as will be explained in the next section.

3.2.2. Link definition

3.2.2.1. Adjacency

The main information provided by a link is that the two nodes it connects are adjacent, i.e. it is possible to move directly from one to the other (Kortenkamp et al., 1994; Nourbakhsh, Powers, & Birchfield, 1995; Hertzberg & Kirchner, 1996; Franz et al., 1998; Gaussier et al., 1998; Ulrich & Nourbakhsh, 2000).

3.2.2.2. Metric relations

Additional information gained from idiothetic sensors may be stored in the map. This is often achieved by recording the relative metric position of the nodes (Kuipers & Byun, 1991; Engelson & McDermott, 1992; Simmons & Koenig, 1995; Shatkay & Kaelbling, 2002; Kunz et al., 1997; Von Wichert, 1998; Nehmzow & Owen, 2000; Hafner, 2000). This approach has the advantage of bounding the error of the idiothetic information, as it is reset whenever a node is encountered. This use of local metric information leads to a so-called *diktiometric map* (Engelson & McDermott, 1992) (Fig. 6a).

3.2.2.3. Assigning nodes a position

Each node may also be assigned a position in a global reference frame (Mataric, 1992; Touretzky et al., 1994; Kurz, 1995; Yamauchi & Langley, 1997; Duckett & Nehmzow, 1997; Oore, Hinton, & Dudek, 1997; Von Wichert, 1998; Arleo & Gerstner, 2000; Balakrishnan et al., 1999; Dedeoglu et al., 1999). This makes it possible to retrieve the general layout of the environment, albeit at the price that long-term error in idiothetic cues must be compensated for. Such metric information may be substituted for any link information, or added to links coding for adjacency. As the *absolute* position of each node is defined in a 2D space, in the remainder of this paper this kind of map will be called an *absolute diktiometric map* (Fig. 6b).

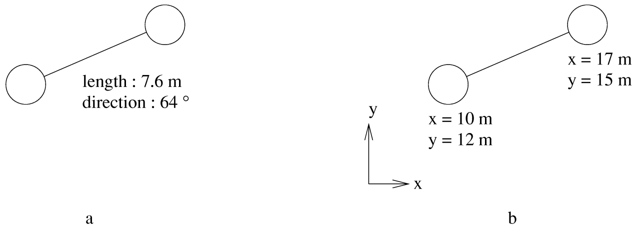


Fig. 6. Two different types of maps obtained when metric information is added to a topological map. When the metric information about relative positions of the nodes is stored in the links between nodes, the map is said to be *diktiometric* (part *a*). When an absolute metric position is stored in each node, the map is said to be *absolute diktiometric* (part *b*).

3.2.2.4. Implicit links

Some varieties of allothetic information implicitly define links between nodes in the map representation. This is the case, for example, when punctual landmark distances or directions are used to characterize nodes. Common landmarks detected from different places can then be used to infer information about place adjacencies, and the fact that nodes have landmarks in common can be used as implicit links (Levitt & Lawton, 1990; Sharp, 1991; Burgess et al., 1994).

4. Localization strategies

In this section we consider that a complete map of the environment is provided to the robot. We will be presenting different localization strategies and the way they integrate the idiothetic and allothetic information.

4.1. Localization capacities

There are two different localization capacities. The first one is the capacity to provide a new position estimate, given a previous position estimate and new idiothetic or allothetic information. This capacity may be called *local localization* or *position tracking*. It is useful as long as the robot has an estimate of its initial position and that the said estimate does not become too different from the robot's actual position.

In the case where the robot has no estimation of its previous position, a second, more powerful, capacity is required, because the robot must locate itself without capitalizing upon information about where it was before. This situation is referred to as the *lost-robot problem* or the *drop-off problem*. The corresponding capacity may be called *global localization*.

When metric maps are used, these two localization capacities have dual properties (Piasecki, 1995). Position tracking is a rather continuous problem, with a quantitative solution: given an allothetic situation, the robot's position should be corrected in order to best reflect its perception. On the contrary, global localization is a rather discrete and qualitative problem: given an allothetic situation, the robot must identify the object or the region of the environment that accounts for its perceptions. As several objects or regions widely scattered in the environment may have produced this perception, the issue is more to choose among distinct separate position hypotheses than to correct a previous position estimate. In the context of topological maps, however, these capacities are more similar, because they both entail finding the node which best reflects the robot's position. However, position tracking is simpler than global localization because only a small set of nodes must be discriminated between.

It should be noted that the position of a robot is not only defined by its two coordinates in a plane, but also by its orientation. However, throughout this review, we will only refer to localization in a 2D space because most of the topological localization strategies do not directly estimate the robot orientation. This estimation usually relies on a dedicated sensor, i.e. on a compass, or on a dedicated procedure that evaluates directions independently from position coordinates. It should nevertheless be clear that most of the approaches using metric maps directly estimate the robot's orientation along with its position.

4.2. Classification used

In the following, we will classify the localization strategies into three categories.

- Strategies which *directly infer the position from allothetic cues* without requiring idiothetic cues. The corresponding models make the assumption that the processing of the available allothetic information is powerful enough to allow the robot's position to be recognized. They therefore allow global positioning in environments where there is no perceptual aliasing.
- Strategies which *track a single hypothesis* about the robot's position using both idiothetic and allothetic cues. The corresponding models solve the perceptual aliasing problem by selecting the most credible position using a previous position estimate and idiothetic cues. However, if the previous position estimation turns out to be totally wrong (e.g. because the robot has been moved by the experimenter to a new place), it cannot be correctly updated, and the robot gets lost. In any

case, these models do allow position tracking in environments with perceptual aliasing.
- Strategies which *track multiple-hypothesis* about the robot's position using both idiothetic and allothetic cues. Instead of tracking only the most credible hypothesis, the corresponding models maintain a set of hypotheses which are all updated in parallel. These strategies allow alternative position estimates to be maintained, even if the most credible estimation at one time turns out to be totally wrong. Therefore, they make global positioning possible in environments with perceptual aliasing.

Within these categories, models may be broken down into sub-categories which correspond to the classical metric/topological distinction. However, we will make a distinction between position representation and map representation, which leads to three categories (see Fig. 7).

- The map is represented in a topological framework, and the position is represented as a node in this map or as some activity distribution over the nodes of this map.
- The map is represented in a topological framework, but the position is represented in a 2D metric framework.
- Map and position are both represented in a 2D metric framework.

A fourth category, where the map would be represented in a metric framework whereas the position would be represented by a topological node, is not considered here. In fact, no models fall into this category as far as localization is concerned. The reason seems to be that once a metric map is available, the loss of precision in localization resulting from the conversion to a topological representation is not compensated for. Note however that the extraction of a topological map from a metric one may be useful for path planning. In the corresponding models, however, the robot position is represented in the metric space and the topological position is inferred from this metric estimate.

In the first category, topological maps may contain metric information, but this information is not used in an absolute 2D space for position estimation. In the second category, however, topological maps must be *absolute diktiometric* maps in order to make it possible to infer the robot's metric position from the currently recognized node. The different categories,

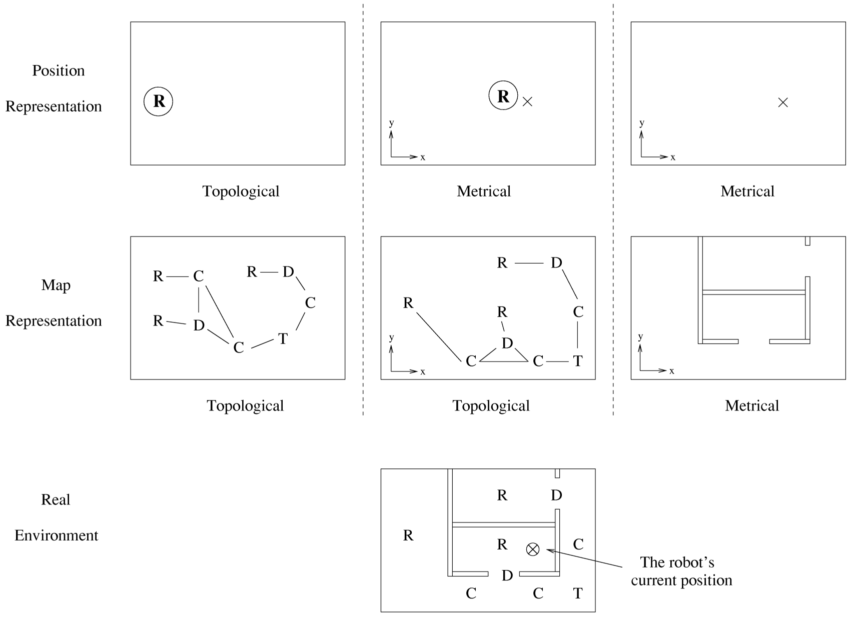


Fig. 7. The three categories for map and position representation (see text for details).

Table 1
Main attributes of three ways to represent maps and positions

	Topological map/ topological position	Topological map/ metric position	Metric map/ metric position
Direct position inference	**Hyp.:** NPA **Input:** AC **Capacity:** GP, DPE	**Hyp.:** NPA **Input:** AC **Capacity:** GP, CPE	**Hyp.:** NPA, SM **Input:** AC **Capacity:** GP, CPE
Single-hypothesis tracking	**Hyp.:** None **Input:** AC, IC, PE **Capacity:** PT, DPE	**Hyp.:** None **Input:** AC, IC, PE **Capacity:** PT, CPE	**Hyp.:** SM **Input:** AC, IC, PE **Capacity:** PT, CPE
Multiple-hypothesis tracking	**Hyp.:** None **Input:** AC, IC **Capacity:** GP, DPE	**Hyp.:** None **Input:** AC, IC **Capacity:** GP, CPE	**Hyp.:** SM **Input:** AC, IC **Capacity:** GP, CPE

NPA: no perceptual aliasing; SM: sensor model; IC: idiothetic cues; AC: allothetic cues; PE: position estimate; GP: global positioning; PT: position tracking; DPE: discrete position estimation; CPE: continuous position estimation.

together with their main attributes, are summarized in Table 1.

4.3. Direct position inference

Allothetic sensors in robots may be limited or noisy. Moreover, environments may be very regular, with only a few distinguishable features. These two facts put together may render allothetic cues nearly useless for localization. However, one could argue that, in many situations—for example a robot using a camera in a common office environment—there is enough available information in the current view of the environment to determine its position. This section presents models which basically rely on this assumption. The matter at issue is to find efficient methods that make it possible to extract enough information from the available allothetic cues so that the resulting position estimation becomes unambiguous. This section also describes procedures that are used in an initialization step to produce a rough position estimate in models that allow position tracking and that will be described later, in Section 4.4.

4.3.1. Topological map/topological position

When using a topological framework for direct position inference, each node of the map must represent a different sensory situation (Fig. 8).

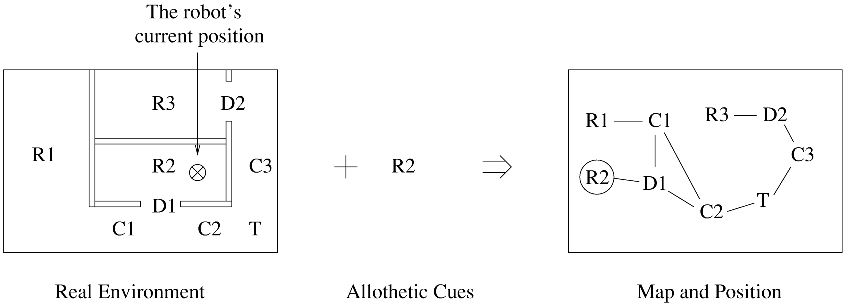


Fig. 8. Direct position inference in a topological framework requires that no perceptual aliasing occur, i.e. that each node encodes a different sensory situation. Localization then simply entails finding the node that best corresponds to the current sensory situation. In this example, the robot positions itself at node R2.

Localization then simply entails finding the node that encodes the sensory situation the most similar to the current allothetic cues. In order to do that, some models rely on metric sensor models, but this is not mandatory.

In some models, map positions correspond to canonical places in the environment (e.g. Kuipers, 2000), which are recognized using a hill-climbing control law. Once such a place is reached, the sensory situation is simply compared to all the stored situations, the closest one being recognized as the current place. Lee (1996) implements this strategy using a local occupancy grid to define nodes and an occupancy-grid matching procedure for place recognition. Kuipers & Beeson (2002) define a place by a set of views that correspond to the robot's possible directions in this place. Each view in this model is a laser range-finder scan of the environment in front of the robot. In a similar way, Kortenkamp & Weymouth (1994) recognize places by the local wall configuration detected by sonar sensors, and by an *Abstracted Scene Representation* (ASR) extracted from images in eight fixed directions around the robot. An ASR is a low-resolution representation of an image in which the direction, length and distance of vertical edges detected by stereo-vision in 25 regions of the image are stored. In the model of Franz et al. (1998), panoramic images, sub-sampled to a low resolution representation of the surroundings using only 78 pixels, are stored in each node. However, the authors acknowledge that such a simple place representation must contend with the perceptual aliasing problem, which leads to a restriction of the size of the environment in which their system can operate.

Other models use direction or distance of punctual landmarks for place definition instead of a representation of the robot's whole surroundings. Among these models, some are bio-mimetic models, like those that have been designed to emulate neuronal activity in the hippocampus of rats. As these models have not been implemented on real robots, such landmarks are rather idealized.

For instance, Sharp (1991) and Burgess et al. (1994) use the distances and directions of perfectly recognizable landmarks as their primary source of information. Their models rely on artificial neural networks designed to emulate *place cells* of the rat hippocampus in computer simulations. Inputs to their networks are provided by sets of neurons whose activity depends on the distance to, or on the direction of, specific landmarks. The activities of these neurons are fed to successive layers of neurons where competitive mechanisms result in the activation of a few nodes in the output layer which correspond to the robot's position (Fig. 9). Trullier and Meyer (2000) also simulate hippocampal activity but without the use of any intermediate layer or competitive mechanisms. In their model, place cells directly respond to a combination of landmark distances via a mathematical function which is maximum when the current landmark distribution is identical to the memorized one.

Some of the models using punctual landmarks have also been implemented on robots and therefore use different allothetic cues, namely landmark direc-

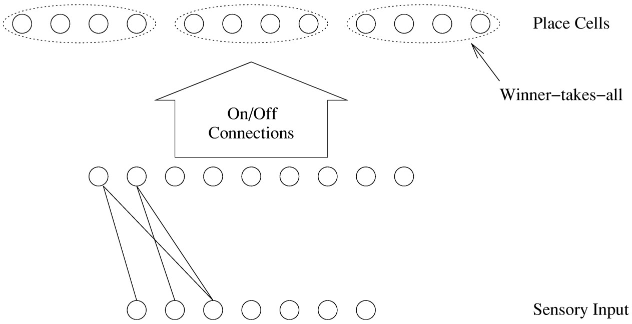


Fig. 9. In Burgess et al.'s model (1994), allothetic cues are direction and distance to distant punctual landmarks. A three-layered neural network takes this information as input and is responsible for categorizing allothetic cues. The output values of the neurons in the last layer emulates place-cell activity in the hippocampus of rats and are correlated to the robot's position (see text for details).

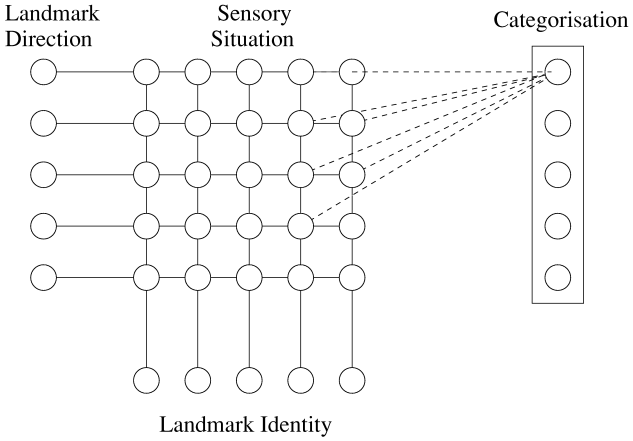


Fig. 10. In the models of Gaussier et al. (2000), and of Bachelder & Waxman (1994), the identity of distant landmarks—the 'what' information—is encoded on a linear set of neurons (bottom of the figure), while their direction—the 'where' information—is coded on a second linear set of neurons (left of the figure). The activities of these two sets are used to calculate the activity of a 2D set of neurons that encodes the current sensory situation. The activity of this last set is then classified so as to recognize the node corresponding to the current sensory situation (see text for details).

tions, instead of landmark distances, which are more difficult to extract from robot sensor data. In Bachelder & Waxman's (1994) model, the robot is placed in an environment with easily recognized artificial landmarks, while the model of Gaussier et al. (2000) is designed to work in an unmodified environment using local views extracted around *focus points* (e.g. corners). In both models, a procedure extracts landmark identities and directions from panoramic images. These two categories, named the 'what' and the 'where', are coded on two 1D arrays of neurons. The product of the activity of these two arrays gives the activity of a 2D array that encodes the local sensory view of the robot (Fig. 10). The activity of this array is then fed into a classifier that categorizes these patterns, each category corresponding to a node in the map.

In all these models, the output is not only the most activated neuron but also an activity distribution among map nodes. This activity distribution may be used in order to produce a finer position estimate, for example through population vector coding.[3]

Levitt & Lawton's (1990) model also makes use of the direction and distance of landmarks, but not in a biologically-inspired fashion. These authors provide two kinds of place representation for mobile robot localization. The first one is the *orientation region*, which is defined by the identity and order of landmarks around the robot without distances, nor directions. This results in regions delimited by lines joining two landmarks and provide a coarse segmentation of the environment. The second type of region is called a *view-frame* and defines a region by coarse estimates of landmark directions and distances. Such regions are used when a more precise position estimate for the robot is useful.

4.3.2. *Topological map/metric position*

Recognizing a node in the models of this section is also simply performed by comparing the current allothetic data with the data stored in each of the nodes. The use of an absolute diktiometric map, in which all nodes encode different sensory situations, makes it possible to infer the robot's metric position (Fig. 11). However, the precision of metric position estimates is limited by the size of the places encoded in the map. Population vector coding may however be used in order to refine this estimate. A metric sensor model is not mandatory but may also be used

[3] Population vector coding (Georgopoulos et al., 1986) is a widely-used method for computing a robot's position, given an activity distribution over a set of possible locations. The computed position is simply the mean of the positions of each location, weighted by the activity of each location.

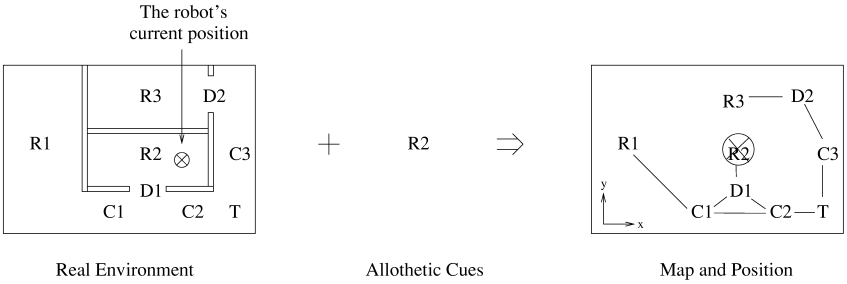


Fig. 11. When an *absolute diktiometric map*, in which each node encodes a unique sensory situation, is available, recognizing the node which corresponds to the current sensory situation allows the robot's metric position to be derived.

to improve the position estimate in the area coded by one node.

In the models of Arleo & Gerstner (2000) and in Balakrishnan et al. (1999) (to be described in greater detail in Section 4.4.2), when no previous position estimate is available, global localization procedures rely on the assumption that some nodes represent allothetic situations which are unique in the environment. When such a unique situation is detected, the robot's metric position may be inferred using allothetic cues alone. In Balakrishnan et al.'s model, which is evaluated in simulation, the allothetic information is that of the distances, directions and identities of landmarks, relative to the robot, with an absolute reference direction. This information is categorized by an unsupervised classifier that recognizes the current sensory situation and therefore the current node. In Arleo and Gerstner's model, allothetic cues are a set of four images taken by a linear or 2D camera in four fixed directions in order to obtain 360-degree information. These images are processed using different kinds of filters, in order to extract some characteristic features which are then used to derive the activity of a set of *snapshot cells*. Such cells are connected to a second set of cells that carry out an unsupervised classification of the sensory situation, leading to an activation of each node in the map that is related to its similarity to the current sensory situation. When this situation is unique in the environment, the result is a coherent activity distribution over the cells of the map centered around one point. This coherent activity allows population vector coding to be used to estimate the robot's position.

When the nodes of the map are sparse, the precision of position estimation can be enhanced by computing the robot's position relatively to the node center after having recognized the correct node. To achieve this, Yamauchi & Langley (1997) use a map where each node stores a local occupancy grid built around one point in the environment, along with the position of this point in a 2D space. Although occupancy grids are metric maps, they are used in this model for place recognition only. Localization entails building a local occupancy grid and searching out the node to which the most similar grid is attached in the map. Once the node corresponding to the current perceptions is recognized, this use of local metric maps allows a finer localization, because the relative position of the robot inside the area covered by the node is derived through an occupancy grid matching feature.

4.3.3. Metric map/metric position

When the positions of some landmarks or objects of the environment are known within a metric map, and when a metric sensor model is used, the robot's metric position may be inferred relatively to the detected objects, instead of just recognizing the local allothetic view (Fig. 12). The methods used to estimate this position are numerous and may be classified into three categories.

- Methods belonging to the first category compute the robot's position given the positions of some detected landmarks relative to the robot. This may be accomplished by *triangulation*, in virtue of the fact that the position of a point in a 2D space is defined uniquely if the distances or directions to

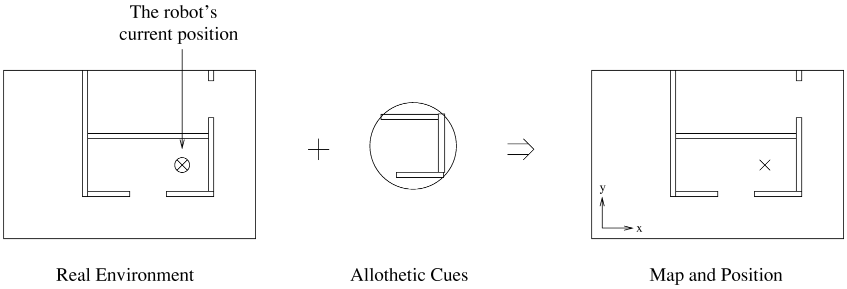


Fig. 12. Localizing a robot within a metric map requires a metric model for the sensors. Given this model, the robot's position may be inferred from its position relative to known landmarks.

three characteristic landmarks are known. The procedures for detecting such landmarks may be similar to those of Section 4.3.1.

Betke & Gurvits (1994) describe an efficient algorithm for triangulation given the position of at least three distinguishable punctual landmarks. Such an algorithm is robust with regard to noisy data and makes it possible to cope with incorrectly recognized landmarks. This approach calls upon an omnidirectional linear camera that makes it possible to detect dark regions that are used as landmarks. As a consequence, landmarks are usually distributed over the whole surroundings, which affords good precision for position estimation. However, if only three very close landmarks are perceived, the triangulation becomes very brittle, and may lead to poor position estimation. This problem is tackled by Madsen, Andersen, & Rensen (1997) who describe another triangulation algorithm that uses an estimate of the robot's position to select landmarks and avoids the above-mentioned brittleness problem (see Section 4.4.3 for details).

With punctual landmarks, other techniques for position estimation may be used. For example, the landmarks used by Wijk & Christensen (2000) are corners detected by sonar sensors. The corresponding localization process evaluates different matches of corner pairs to recognize the detected corners and to estimate the robot's position (Fig. 13). A very similar algorithm is used by Borghi & Brugali (1995) using corners extracted from laser rangefinders.

Instead of using punctual landmarks, other models use spatially extended landmarks which allow the robot's position to be computed from the detection of a single one. Gomes-Mota & Ribeiro (2000), for example, use sets of pairs of non-parallel lines extracted from a laser-scan as landmarks. Such pairs, called *frames*, are defined by the two lines and their intersection point. The consequence is that landmark identity is very weakly defined, as only the similarity of two frames may be assessed. In such a framework, localization entails extracting from the metric map a set of frames, used as reference, and matching a pair of frames taken in the map and in the current robot's perceptions (Fig. 14). An extension of this approach makes it possible to incorporate an initial approximate estimation of the robot's position in order to simplify computations (see Section 4.4.3). Arsenio & Ribeiro (1998a) use similar landmarks, but position estimation is computationally simplified through the use of a so-called *visibility cells* decomposition (Guibas, Motwani, & Raghavan, 1997) that segments the map in regions where the same landmarks are detectable. Visibility cells allow a fast processing of landmarks by quickly detecting impossible matches of sensed and memorized landmarks. This approach is also simplified when an initial approximate position estimate is available for the robot (see Section 4.4.3).

A different approach is that of Sim & Dudek (1999) who use small images extracted around *focus points* in camera images as landmarks. The map contains a set of landmarks, along with their appearance from different positions distributed over the whole environment. Localization entails taking an image, extracting landmarks from this image and

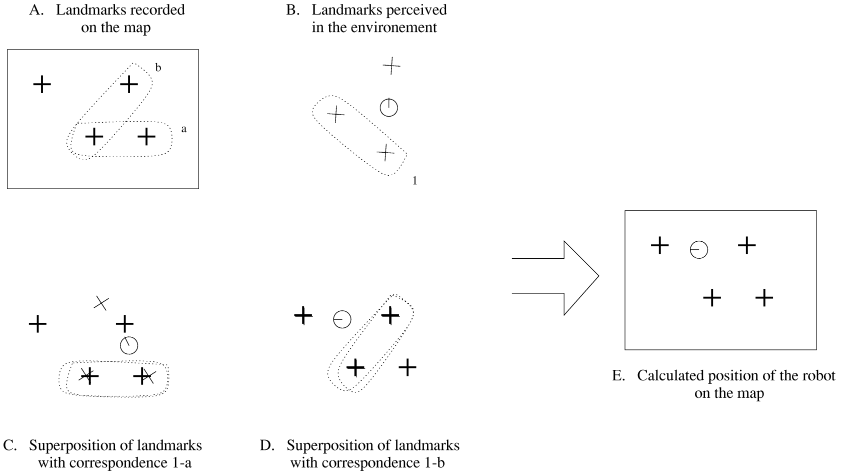

A. Landmarks recorded on the map
b
a
B. Landmarks perceived in the environement
1
C. Superposition of landmarks with correspondence 1-a
D. Superposition of landmarks with correspondence 1-b
E. Calculated position of the robot on the map

Fig. 13. (A) In the localization procedure used by Wijk & Christensen (2000), a first pair of landmarks is chosen in the map (for example pair *a*). (B) A second pair is chosen among the perceived landmarks (for example pair 1). (C, D) The robot's position in the map where the two pairs would correspond is calculated. This position is then used to estimate where the remaining perceived landmarks would be positioned in the map's reference frame. (E) The position which gives the best overall match between the perceived landmarks and the map is assumed to be the robot's true position (correspondence 1-b in this example).

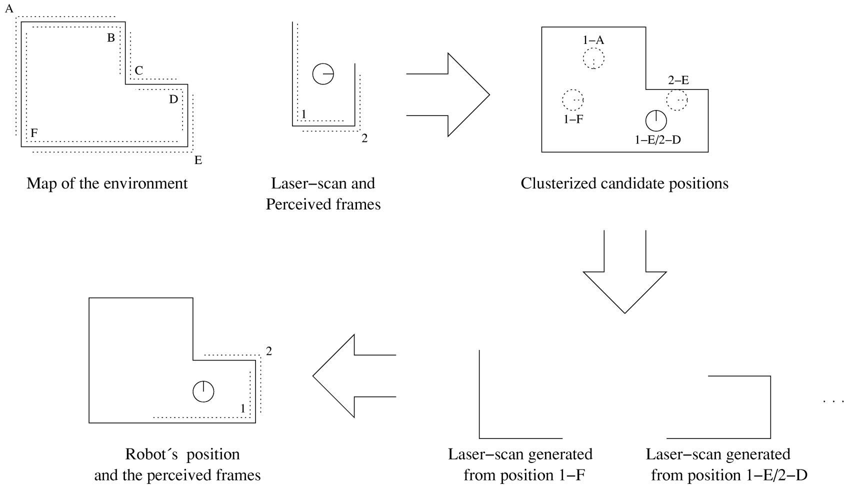

A
B
C
D
E
F
Map of the environment
1
2
Laser−scan and Perceived frames
1−A
2−E
1−F
1−E/2−D
Clusterized candidate positions
2
1
Robot´s position and the perceived frames
Laser−scan generated from position 1−F
Laser−scan generated from position 1−E/2−D
. . .

Fig. 14. In the localization procedure used by Gomes-Mota & Ribeiro (2000), *frames* are used as landmarks (represented by dotted lines). Matching frames from the map (marked by letters) with frames from the current allothetic situation (marked by numbers) gives a set of candidate positions (designated by the names of the two frames 1-A, 1-F . . .). This set is clusterized and yields several possible positions for the robot. A laser scan is then simulated for each of these positions, using the map and supposing that the robot is located in the corresponding position. The candidate position which leads to the best correspondence between the perceived laser scan and the simulated laser scan is assumed to be the robot's actual position (position 1-E/2-D in this example).

calculating the position from which the image was taken using each landmark, by linearly interpolating between the positions from which the most similar landmarks stored in the map have been taken. In order to counter the influence of erroneous landmark detection, the robot's final position is taken as the median of position estimates given by all the landmarks taken from the image.

- A second category of methods making it possible to localize a robot searches out the position that affords the best correspondence between a local map, built using recent perceptions, and the global map of the environment. As the search space is huge and often contains a lot of local-extrema, these *map-matching* methods are mainly used in conjunction with a position estimate that restricts the search space (see Section 4.4.3). However, Olson (2000) proposes a method which is able to globally find the position affording the best correspondence. A similarity measure is used that allows local and global maps to be compared. This measure may be defined for many map representations using either geometric features or occupancy grids. An additional function provides a bound to the similarity measure over an area of the environment, given the similarity value at the center of this area. This function makes it possible to quickly discard areas of the environment where the similarity cannot be higher than the current best similarity. The search strategy is a multi-resolution approach (Fig. 15) that allows the robot's position to be retrieved. A still finer localization estimate may be obtained by computing the similarity in neighboring cells at the finest resolution and hypothesizing a Gaussian distribution of this similarity around the true position. Finding the center of this Gaussian allows *sub-pixel* localization in a way similar to population vector coding.
- Finally, a third category of localization calls upon the computation of *feasible poses*, i.e. given a sensory measurement, the positions in the environment where this measurement is possible are estimated. Brown and Donald (2000) present such a strategy which uses obstacle distances only, without resorting to any landmark identification. This approach calls upon several laser range-finder measurements taken from a single place and allows the robot to uniquely determine its location by gradually eliminating the positions from which the measurements would not have been possible. The corresponding model is presented in an exact mathematical formalization, while a rasterized version—i.e. a version operating in a discretized world—is implemented on a real robot using an occupancy grid as a map. Howell & Donald (2000) present a simplified version of this algorithm which exploits local directions of the surfaces, in addition to their position, in order to enhance the algorithm's robustness.

Guibas et al. (1997) take a similar approach, based on the computation of the regions of the map where the current perception is possible. It is based on a pre-processing of the map that generates *visibility cells* from which the perceptions are similar. In this work, perceptions are obtained by a laser range-finder and lead to a *visibility polygon* that surrounds the robot and encloses the free space visible from the robot's point of view. The authors provide efficient algorithms for finding the robot's possible locations, given a visibility polygon and the visibility-cell decomposition. Karch & Wahl (1999) improve and extend this framework to noisy perceptions, using simulated laser scans.

4.4. Single-hypothesis tracking

The models described in this section acknowledge the perceptual aliasing problem and deal with the fact that allothetic cue processing does not allow all

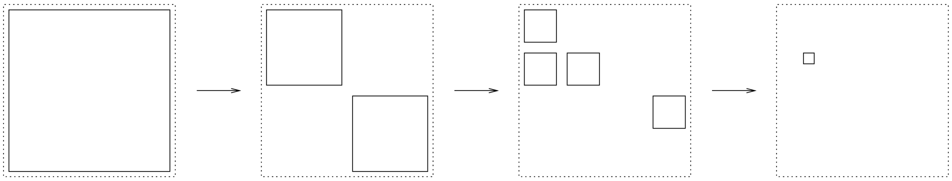


Fig. 15. The multi-resolution search strategy used by Olson (2000). Whenever a cell is found that cannot contain the position affording the best similarity with the current local map, it is removed. Otherwise, it is divided, and the process is repeated at a higher resolution level. The process is stopped when the optimal resolution has been reached.

the positions in the environment to be differentiated between. As a consequence, idiothetic cues and previous position estimates have to be used to resolve ambiguities. The problem is solved by selecting, whenever new allothetic cues become available, the position which is the most coherent with the previous estimate and the movements made since, ignoring any other candidate position.

When a single metric position has to be tracked, many navigation systems make use of Kalman filtering. A Kalman filter (Maybeck, 1979) is a general algorithm used to estimate the state of a system, given a model of this system's evolution and a capacity to measure this system's state approximately. Each of these estimates is supposed to be corrupted by a Gaussian white noise. This filter is an *optimal linear estimator*, i.e. it gives the best estimation of the system state given the available information, provided that the system's evolution is modelled by a linear function.

To localize a robot, its position defines the state of the system (Fig. 16) whose evolution is related to idiothetic cues, while state measurements are afforded by allothetic cues. As shown by examples in Sections 4.4.2 and 4.4.3, there are numerous implementations of this basic scheme, and Kalman filtering turns out to be a convenient way to fuse idiothetic and allothetic information. When the evolution equations are non-linear, which is almost always the case in robotics, an extended version of the filter may be used, in which a local linear approximation of the system is obtained through Taylor expansion. However, this approximation may entail some lack of robustness (see Section 4.4.3).

4.4.1. Topological map/topological position

Single-hypothesis tracking in topological maps usually first entails selecting nodes whose place definition corresponds to the current sensory situation. This may be achieved by procedures similar to those of models described in Section 4.3.1, or by simpler procedures, because tolerating perceptual aliasing allows unreliable node recognition to be coped with. Under such conditions, idiothetic information stored in the links between nodes is then used to select the node which is the most coherent with respect to the previously recognized node (Fig. 17).

The nodes of the maps therefore often encode easy-to-recognize, but non-unique, situations. Corridor angles and junctions like those used by Kunz et al. (1997) and Dedeoglu et al. (1999) are very

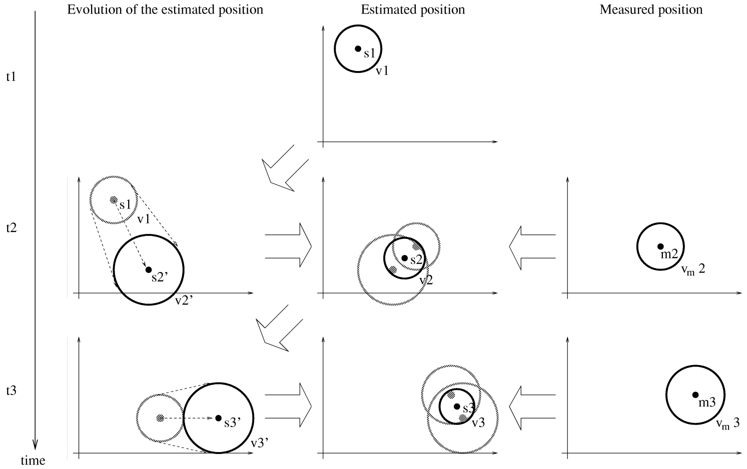


Fig. 16. Within the context of localization in a navigating robot, the state estimated by the Kalman filter corresponds to the Cartesian coordinates of the robot's position. From time $t1$ to $t2$, the estimated state of the robot changes from state $s1$ to state $s2'$, according to the idiothetic cues. The estimated variance of the estimation also evolves from $v1$ to $v2'$ to reflect noise. At time $t2$, a measurement $m2$, with an associated variance $v_m 2$, is made on the robot's position through allothetic cues. The estimated state of the system is accordingly adjusted to a better estimate $s2$, with an associated variance $v2$. This process is then repeated between times $t2$ and $t3$, when a new measurement of the robot's position, $m3$, is made.

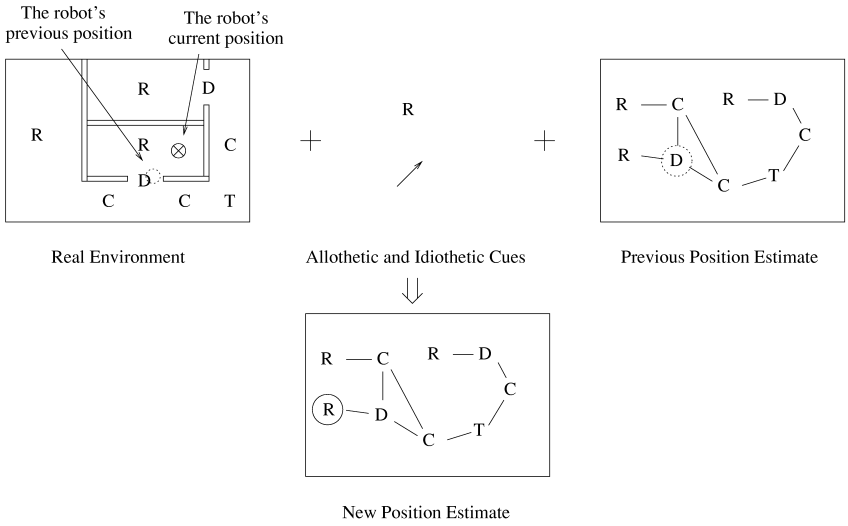

The robot's previous position
The robot's current position
R
R
D
R
C
D
C
C
T
+
R
+
R — C
R — D
C
R — D
C
T
Real Environment
Allothetic and Idiothetic Cues
Previous Position Estimate
⇓
R — C
R — D
C
R — D
C
T
New Position Estimate

Fig. 17. When a topological map contains nodes which encode different positions with the same sensory view (here, for example, the robot is unable to differentiate between the different rooms), the previous position estimate and the allothetic cues have to be used to resolve these ambiguities. In a model which tracks a single hypothesis, this is achieved by selecting the node which is the most coherent with the previous position estimate.

common choices in office environments. Ad-hoc procedures based on sonars or laser range-finders for such place recognitions are numerous. Places may also be defined through an unsupervised classification of the sensory situations. In the model of Nehmzow & Owen (2000), for example, places are automatically defined via an unsupervised clustering algorithm applied to the vector of sonar-sensor values. Mataric (1992) uses a coarser space segmentation with automatically defined large-sized features, such as walls and corridors, as places. Such places are detected using the robot motion, which is constrained by low-level control laws such as wall-following and sonar data.

Having recognized several nodes that may correspond to the current sensory situation, the right position has to be determined using knowledge about the previous position, together with the information that is stored in the links of the map. In particular, this information may concern the control procedure used to move from one place to another, as in the model of Kuipers (2000). In this application, the robot will be considered as localized in a node A in the map if the control procedure linking the node B, which corresponds to its previous localization, to the node A is the same as the procedure that was effectively performed by the robot since the previous localization. Metric information may also be used to disambiguate place recognition. For instance, adjacency and approximate position relative to the previously recognized node are used in the models of Mataric (1992), Kunz et al. (1997), Dedeoglu et al. (1999) and Nehmzow & Owen (2000).

An important characteristic of the above models is that, even if idiothetic cues are used, their cumulative error does not affect the localization quality. In Kuipers' model, this property results from the fact that the closed-loop control strategy which is memorized in the links does not make explicit use of idiothetic cues and from the fact that, when a node is reached, the hill-climbing strategy (described in Section 3.2.1) allows the robot to reach a locally uniquely defined reference point, thus suppressing any idiothetic error. In Kunz et al., Dedeoglu et al. and Nehmzow and Owen's models, idiothetic cues are explicitly used to relate nodes, but this information is only used locally—i.e. between two nodes—and it is reset whenever a new node is recognized. Moreover, in Kunz et al. and in Dedeoglu et al.'s models, the environment includes orthogonal corridors only, which allows the robot to frequently adjust its direction estimation by considering that its direction is the closest to four possible orthogonal reference directions. In Nehmzow and Owen's model, where the directions of links are not assumed orthogonal, direction drift is corrected using a magnetic compass. However, the authors report that this correction is not sufficient for large-scale environments. To solve this problem, they introduce *behaviors* that resemble Kuipers' control strategies (e.g. reaching a corridor center).

Instead of recognizing some nodes using allothetic cues and choosing the right one using idiothetic cues, some models work in the opposite way. Accordingly, the previous position is used to select a set of possible nodes, and the most likely node is chosen according to allothetic cues.

For example, Ulrich & Nourbakhsh (2000) store histograms in several color bands of images taken by an omnidirectional camera at each node of the map. In their system, a node represents a room or a corridor, and the histograms of several images are stored for each location. Place recognition first entails taking an image and comparing its histograms with the nodes neighboring the previous node. The new node is then determined using the results from each color band through a voting scheme. If all the color bands vote for the same location, it is considered recognized. If the vote is ambiguous, the system does not recognize a new node and assumes that it is still at the previous location. This model extends the approach of Radhakrishnan & Nourbakhsh (1999), in which a Bayesian classifier was used on the set of pixels associated with a panoramic color image for node recognition. Von Wichert (1998) also resorts to a similar idea by constraining the recognition to nodes that are situated less than 2 m away from the previously recognized node. In this model, each node stores features extracted from images taken in eight absolute directions.

4.4.2. Topological map/metric position

The nodes in the maps mentioned in this section associate sensory situations with metric positions in a 2D-space. Solving the perceptual aliasing problem can therefore be based on a comparison between the positions of the nodes and of the robot. The latter calls upon a previous estimate that is updated through idiothetic cues. Resolving ambiguities usually entails choosing the node closest to the robot's position estimate (Fig. 18). Once the correct node has been recognized, the robot's position may be corrected using this node position.

Several models of this section exhibit the same structure, with three distinct functional modules (Fig. 19). The first module allows the allothetic cues to be categorized so as to discretize the space in which allothetic cues are represented. In the model of Arleo & Gerstner (2000), allothetic cues are provided by a camera and determine the activity of a set of cells, thus making clue classification possible (see Section 4.3.2). In the model of Balakrishnan et al. (1999), allothetic information is provided by distance and identity of landmarks relative to the robot, together with their direction in an absolute reference frame. This information is also categorized by a set of cells.

The second functional module characterizing the models of this section monitors an estimate of the robot's metric position in a 2D Cartesian space. This module is updated using idiothetic cues and is in strong interaction with the third module, both because the position estimate is used to disambiguate node recognition in the map and because node recognition is used to correct the position estimate. In the model of Arleo & Gerstner (2000) the robot's position is coded by the activity distribution of a set of cells that have an associated position in the 2D space. The activity of each of these cells is a Gaussian of the distance between its position and the robot's estimated position, the latter being periodically reset to the position estimated from the map activity using population vector coding. In the model of Balakrishnan et al. (1999), the robot's position is simply represented by its coordinates, without further encoding. It is continuously updated thanks to positions recognized on the map using Kalman filtering, instead of being updated only from time to time.

Finally, the third functional module contains the map, which associates categories detected in the first part with positions coded in the second part. In the model of Arleo & Gerstner (2000), a set of cells that have connections with the cells of the two previous modules is used, thereby associating a sensory situation with a position in space, because cell activations in the map module depend on the activity of the cells in the two other modules. The result is that only those cells that correspond to the current sensory situation and that are close to the position

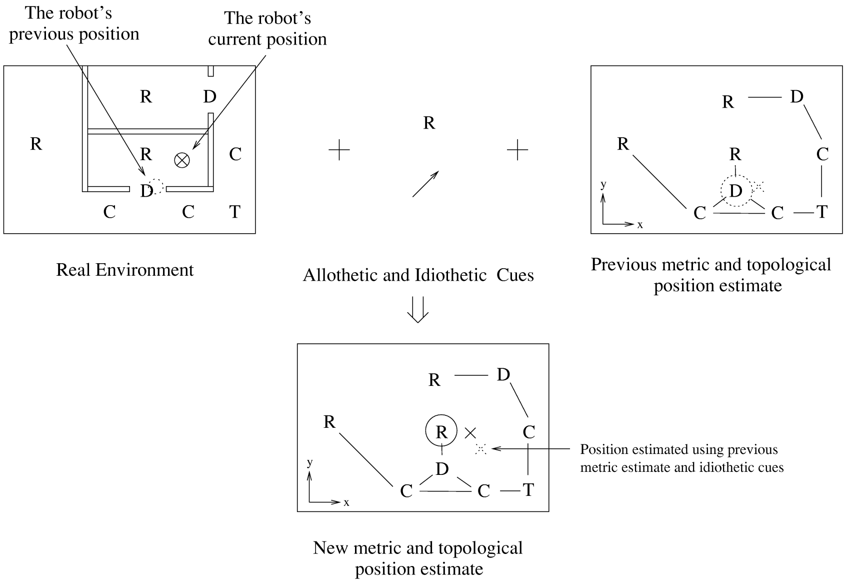

The robot's previous position
The robot's current position
R
D
R
R
C
D
C
C
T
Real Environment
+
R
+
Allothetic and Idiothetic Cues
R
D
R
R
C
D
C
C
T
y
x
Previous metric and topological position estimate
R
D
R
R
C
D
C
C
T
y
x
Position estimated using previous metric estimate and idiothetic cues
New metric and topological position estimate

Fig. 18. When seeking the metric position of a robot with an absolute diktiometric map, idiothetic cues are first used to estimate a new metric position. Ambiguities arising from perceptual aliasing (e.g. being unable to recognize a particular room using allothetic cues only) are then solved by selecting the node which is the most consistent with this position estimate and the idiothetic cues. The position of the selected node then allows this first estimate of the robot's position to be improved.

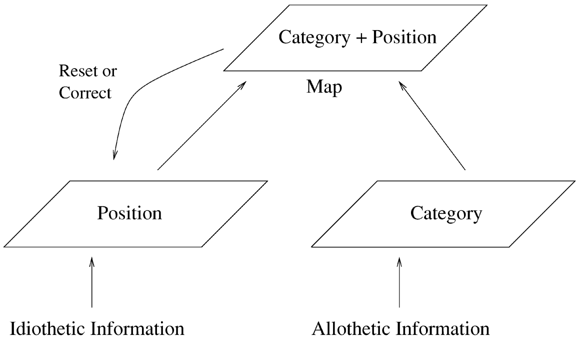


Fig. 19. A model structure that allows position tracking with a topological map and metric position estimates (see text for details).

estimate are activated, thereby unambiguously defining the robot's position. In the model of Balakrishnan et al. (1999), the map is composed of a set of nodes, each of which has an associated position and an associated sensory situation. When several nodes are associated to the same sensory situation, disambiguation is performed by selecting the node whose associated sensory description is the closest to the current position estimate. The model proposed by Kurz (1995) is very similar to those of Balakrishnan et al. but is implemented on a real robot and uses a ring of sonar range-finders as allothetic cues. The simulated model of Touretzky et al. (1994) is another example of a similar, but simpler, structure.

The model of Yamauchi & Beer (1996) has a different structure and is unusual in that it does not store allothetic information in the nodes of the topological map except for the first created node, where a local occupancy grid is stored. Nodes are simply associated with evenly-spaced positions in a 2D space. The current position corresponds to the node that is the closest to the current position estimate. Idiothetic cues drift is recalibrated periodically by navigating to the first created node, where the occupancy grid was initially stored. The robot then creates a new local occupancy grid that is matched with the stored one using hill-climbing. The result of the match (see Section 4.3.3 for details on grid matching) allows to correct the position estimate.

It should be noted that, although the models of this section are only able to track a single-position hypothesis, some are also able to globally localize the robot thanks to dedicated procedures. This is the case, for instance, with the models of Arleo and Gerstner and of Balakrishnan et al. that were previously described. It is also true of the model of Touretzky et al. (1994), which computes a map activity using allothetic cues, and which sets the position estimate at the center of the active nodes. A high uncertainty value is initially associated with this position, i.e. the Gaussian that gives a node's activity as a function of its distance to the robot's position estimate has a large width. Then the robot moves, and the node activity is calculated according to the standard position-tracking procedure described above. At each time step, the robot's position is set to the center of activated nodes, while the position uncertainty is gradually reduced in order to restrict the dispersion of the map activity, thus ultimately generating a coherent position estimate. The advantage of this procedure over the ones of Arleo and Gerstner and of Balakrishnan et al. is that the robot integrates successive sensory situations to continuously update its position, instead of relying on the occasional discovery of an unambiguous place.

4.4.3. *Metric map/metric position*

This category covers a vast number of implementations in classical robotics. In such implementations, idiothetic cues are straightforwardly used to estimate the robot's position based on a previous estimate. Allothetic cues are then used to correct this position estimate by searching in the vicinity the position which best corresponds to the sensory data (Fig. 20). The joint use of a metric map and a sensor model permits the robot's position to be estimated directly from allothetic cues without resorting to the categorization and the recognition of sensory situations—as it is the case for topological maps—thereby potentially allowing a finer position estimate.

The first task is therefore to estimate the robot's position using allothetic cues. This may be done using techniques mentioned in Section 4.3.3, but often the previous position estimate is used in order to constrain the search for the position which corresponds to the current sensory situation.

The most popular approach in this field is based on *occupancy grids*. The processing of allothetic cues entails building a local occupancy grid using

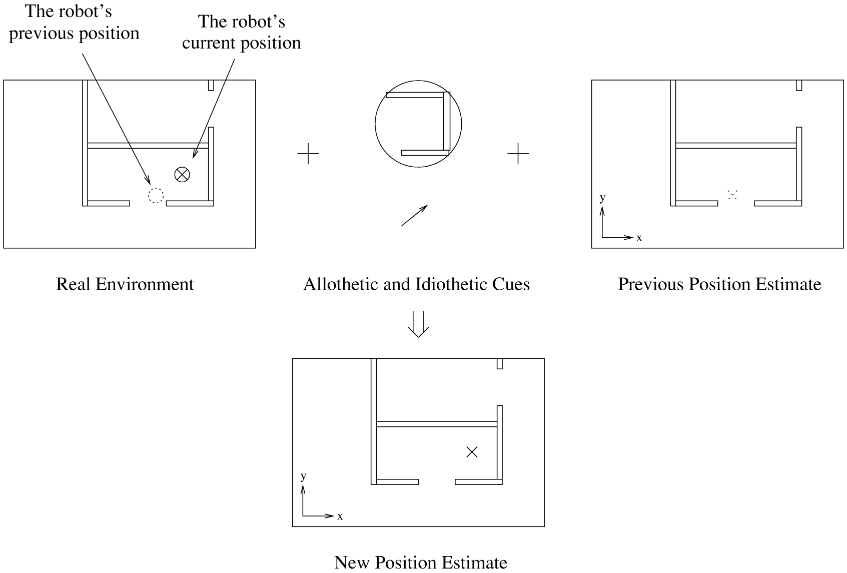

The robot's previous position
The robot's current position
+
+
y
x
Real Environment
Allothetic and Idiothetic Cues
Previous Position Estimate
⇓
y
x
New Position Estimate

Fig. 20. Most models which allow position tracking in a metric framework use idiothetic cues to produce a position estimate. This estimate is used to restrain the search for a position which best corresponds to the current allothetic situation; this is then used to correct the first position estimate.

current or recent data only. The problem then entails finding a position for the robot close to the current estimate which maximizes the correspondence between the local occupancy grid and the global map. This may be performed by various techniques (see Schiele & Crowley, 1994 for a review) of which the following are just examples. Because they are based either on the whole occupancy grid or on segments representing obstacle boundaries extracted from the grids, the corresponding matching procedures may be performed between segments, between grids or between a segment and a grid. The first two procedures proved to be more stable than the last one. Thrun (1999) resorts to grid–grid matching using a differentiable correspondence function that measures the similarity between the grids. The fact that this function is differentiable allows the use of gradient ascent to expedite the search for the best correspondence. Schultz & Adams (1998) also tested two methods for finding the position which maximizes the grid correspondence. The first one is hill-climbing, which entails looking iteratively in the neighborhood of the current position estimation for positions which produce higher correspondences. The second one entails calculating the similarity of the two maps in several positions close to the estimated position. The new position is given by the barycentre of these positions weighted by the matching score. The second method proved to be more reliable than the first.

Other approaches use geometric features such as points, segments or polygons, instead of occupancy grids. With single-point landmarks, a triangulation algorithm similar to those described in Section 4.3.3 can be used to localize the robot. However, in this case, as landmark identification can depend on the robot's position, the system does not require the landmarks to be perceptually unique. Ayache & Faugeras (1989) use 3D segments detected by stereo vision as features. The correspondence between perceived and stored segments depends on their similarity in length and direction, as well as on their proximity in the global space in which the positions of perceived segments are computed using the robot's position estimate (Fig. 21). Castellanos et al. (1999) describe a similar strategy for a model based on segments extracted from laser range-finder data. The approach of Gomes-Mota & Ribeiro (2000), which calls upon direct position inference (see Section 4.3.3), is very similar. Cox (1991) also uses segments as features in the map. However, only points sensed by an infrared range-finder are used as

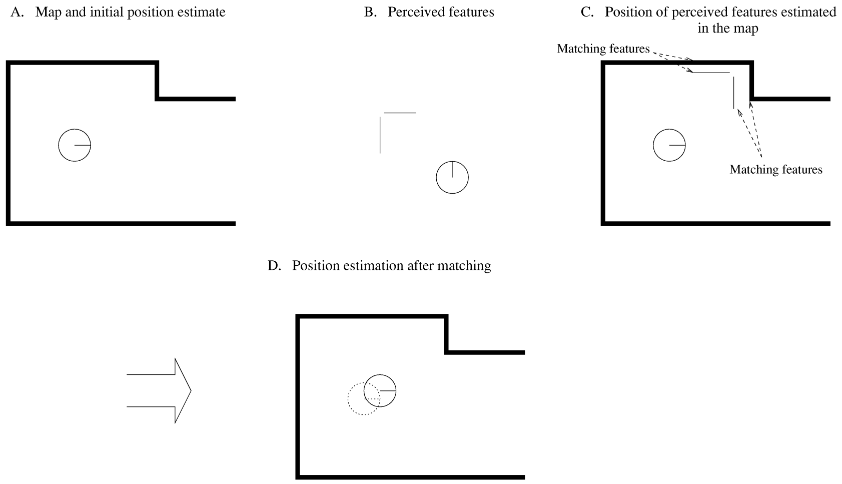

A. Map and initial position estimate
B. Perceived features
C. Position of perceived features estimated in the map
Matching features
Matching features
D. Position estimation after matching

Fig. 21. In the model of Ayache & Faugeras (1989), when an initial coarse position estimate is available for the robot (*A*), feature matching is simplified. To this end, the position of perceived features (*B*) is estimated in the map using the initial position estimate (*C*). Perceived features are then matched to the closest feature in the map, and the position of the robot that superimposes perceived and stored features is calculated (*D*).

allothetic cues. The position of each point is estimated in the map's reference frame using the robot's initial position estimate. Each point is then made to correspond with the closest segment in the map, and the robot's position is the one that gives the smallest summed square distance between the points and their corresponding line.

Likewise, the approach of Leonard et al. (1992) is based on features extracted from consecutive sonar readings during robot movement and tackles the issue of dynamic environments in which map features may appear or disappear, either because of environment modification or because of temporary false sensory measurements. Their solution is to assign a credibility to each feature in the map that depends on how often the feature has been correctly perceived and how often it has been predicted but not perceived. Thus, using only the most credible features to compute the position from perceptions affords additional robustness to the localization process. Dudek & MacKenzie (1993) and Arsenio & Ribeiro (1998b) also make use of a confidence measure for each perceived feature to give more weight to the more reliable ones.

Moutarlier & Chatila (1990) describe a similar method which is an implementation of the *stochastic mapping* scheme presented by Smith et al. (1988). Their model is based on segments extracted from laser scans. However, the authors report a problem of instability in feature correspondence due to the poor modeling of robot movements. They stress that the linear movement model of the robot may lead to a very poor estimate of the robot's position due to wheel slippage, which is highly non-linear and not modeled. This precludes simply matching the closest and most similar segments in the global reference frame. The corresponding solution calls upon a heuristic that compensates for the unmodeled errors in the robot's movements before finding corresponding segments and estimating the robot position.

Besides helping feature matching, odometry can be used to select useful landmarks, for example to avoid landmark configurations that lead to poor triangulation precision. For instance, Greiner & Isukapalli (1996) propose an algorithm which learns to select the landmarks leading to the best position estimate. Likewise, Madsen et al. (1997) present a triangulation algorithm that uses landmarks extracted from a single camera image and that estimates the precision of the result according to the corresponding landmark configuration. Using this algorithm in conjunction with the map and the current position estimate, their system is therefore able to select the camera direction which will lead to the best position update. Such an active localization scheme is also implemented by Arsenio & Ribeiro (1998b), who adapted their visibility cell decomposition algorithm to situations where an initial position estimate is available (see Section 4.3.3). Wijk & Christensen (2000) (see Section 4.3.3) use the current position to discard position estimates that are impossible, thus reducing the computational complexity of their global localization algorithm. Borghi & Brugali (1995) (see Section 4.3.3) also use an initial position estimate in case of ambiguous position recognitions in order to select the most likely one.

Finally, Lu & Milios (1997) propose a different approach, where the map encodes a set of laser scans along with the absolute position of the robot where the scans have been taken. In this system, raw sensor data are stored in the map instead of extracted features. Finding the robot's position using a perceived laser scan is therefore achieved by scan-matching. Among the various procedures that may be used to this end, Gutmann & Schlegel (1996) review three different possibilities. Although each of these has drawbacks, the authors propose an algorithm that selects the most appropriate one each time a new match is performed. This results in increased reliability in the whole positioning process. Einsele (1997) also describes a dynamic-programming method for matching laser scans, based on segments extracted from the laser scans, and reviews some additional matching techniques.

Once a position has been estimated using allothetic cues, it can be directly used as the new robot's position estimate. This, for example, is the case with the models of Einsele (1997), Madsen et al. (1997), Yamauchi et al. (1999), Gomes-Mota & Ribeiro (2000) and Wijk & Christensen (2000). However, this new estimate can be used with the previous one to produce an enhanced position estimate. For this purpose, Kalman filtering and its improvements are techniques frequently called upon, which are used, for example, by Smith et al. (1988), Ayache & Faugeras (1989), Moutarlier & Chatila

(1990), Cox (1991), Leonard et al. (1992), Schiele & Crowley (1994), Betke & Gurvits (1994), Borghi & Brugali (1995), Lu & Milios (1997) and Castellanos et al. (1999). Other techniques may be called on. For example, Thrun (1999) uses the minimization of a cost function which effects a trade-off between the closeness to the current position and the quality of map matching in the position considered. Boley et al. (1996) also suggest using the *recursive total least squares* methodology, instead of a Kalman filter, to estimate the robot's position. The proposed method proved to be more effective than Kalman filtering, particularly when the initial position estimate is poor. Cox & Leonard (1991) choose a different approach to take into account the different positions estimated through allothetic-cues processing. Instead of selecting the most likely position, given the previous position estimate, they use each possible position to generate a new position hypothesis through Kalman filtering. Each of these hypotheses is assigned a probability of being correct that is estimated from the degree of correspondence between the sensed and stored features. The new position estimate is then the weighted sum of all the position hypotheses. This approach is said to afford additional robustness to position tracking.

4.5. Multiple-hypothesis tracking

The models of the previous section deal with the perceptual aliasing problem by selecting the position corresponding to the current allothetic cues that is the closest to the current position estimation. Unlike these, models of this section try to keep track of all the possible positions that could correspond to the sensory situation, in order to update these possibilities in parallel with new idiothetic and allothetic information, and to select the most likely among these possibilities. Such an approach naturally allows global localization because, in this framework, total uncertainty about the initial position estimation is simply represented by an equal probability for each candidate position, and does not represent a special case of the localization procedures.

A general framework to represent multiple position hypotheses and to tackle position uncertainty is that of Markov localization (Thrun, 2000), i.e. an adaptation of the state estimation procedure in Partially Observable Markov Decision Processes (POMDP). Navigation systems based on Markov localization, like those developed by Simmons and Koenig (1995) or by Thrun, Gutmann, Fox, Burgard, & Kuipers (1998), rely on a discretization of the space into a finite set of states S, each state being either a node in a topological map or a cell in a metrical map. The robot's position is represented by a probability distribution $P(s)$ over S. Contrary to Kalman filtering, the probability distribution is not supposed Gaussian, which makes it possible to represent multiple possible positions. $P(s)$ is updated in two ways whenever the robot moves or senses its environment, using probabilistic models of the robot's actions and perceptions (Fig. 22). Like Kalman filtering, Markov localization provides a very natural tool for the fusion of idiothetic and allothetic information in a common reference frame, i.e. the probability distribution of the robot's position over the environment.

The models based on this method are among the most efficient and noise resistant, but they can be difficult to adapt to map-learning (see Meyer & Filliat, 2003).

4.5.1. Topological map/topological position

Whenever a robot perceives allothetic cues that could correspond to several nodes, instead of selecting the node which is the most coherent with the previous estimate as in Section 4.4.1, models of this section keep track of these multiple possible positions by updating estimates of the possibility of currently being in each node. Idiothetic cues may also be used to update these possibilities (Fig. 23).

The simplest approach to this method is to explicitly track multiple hypotheses about the robot's location. For example, the model of Engelson & McDermott (1992) uses a diktiometric map where information is assumed to be uncertain and is represented using intervals. The nodes of the map are associated with world features, such as doors or corners, and store sensory views that characterize them. When perceptual aliasing occurs—i.e. when several nodes correspond to the current sensory situation—the algorithm creates a set of *tracks* that represents each possible location. Each of these tracks is then simply updated as in the single hypothesis case (see Section 4.4.1). Whenever a

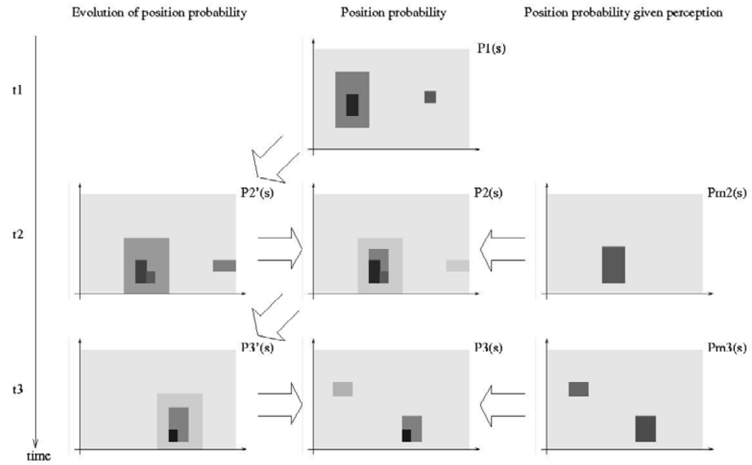

Evolution of position probability
Position probability
Position probability given perception
t1
t2
t3
time
P1(s)
P2'(s)
P2(s)
Pm2(s)
P3'(s)
P3(s)
Pm3(s)

Fig. 22. In the context of Markov localization, the states of the system are the robot's possible positions within the environment. At time $t1$, the robot's position is represented as a probability distribution $P1(s)$ over these states. Using a model of the robot's odometry, these probabilities are updated to a probability distribution $P2'(s)$ at time $t2$. A measurement $Pm2(s)$ is then made of the robot's position which provides a probability distribution over the states, given the allothetic cues perceived by the robot. These two distributions are used to estimate a new probability distribution $P2(s)$ that better estimates the robot's position at time $t2$. This process is then repeated between times $t2$ and $t3$, when a new measurement $Pm3(s)$ is made on the robot's position.

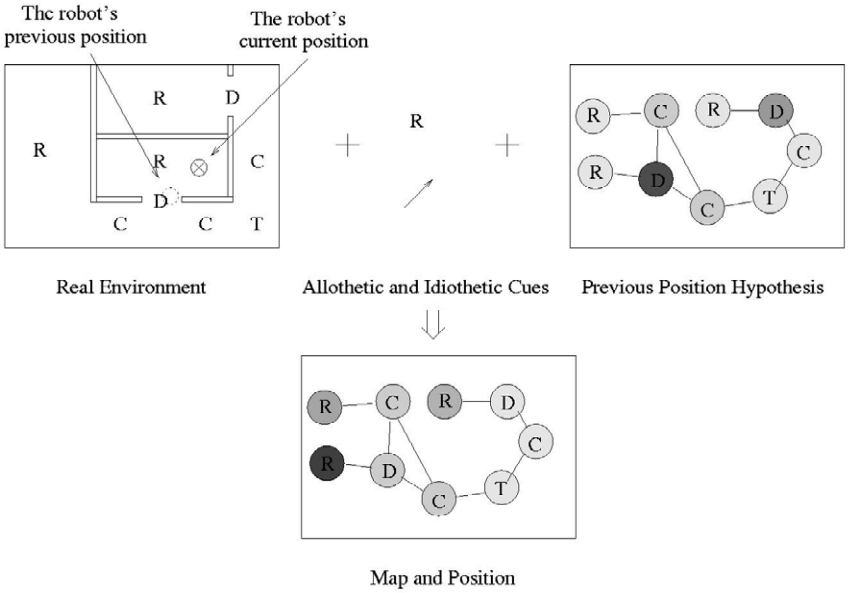

The robot's previous position
The robot's current position
R
D
R
R
C
D
C
C
T
+
R
+
R
C
R
D
C
R
D
C
T
Real Environment
Allothetic and Idiothetic Cues
Previous Position Hypothesis
R
C
R
D
C
R
D
C
T
Map and Position

Fig. 23. Multiple-hypothesis tracking in a topological map entails assessing the probability that the robot is positioned in the place represented by each node. This probability, which is represented by the node's grey level in the figure, is updated using both idiothetic and allothetic cues. The node with the highest probability corresponds to the estimated robot position, but alternative hypotheses are still in competition.

hypothesis does not match any possible location in the map, it is eliminated from the track list. Although this is not mentioned by the authors, this system seems to be able to globally localize the robot if the set of tracks is initialized with all possible positions. However, the estimation of the robot's position is unambiguous when there is a single hypothesis only, as no relative credibility is assigned to different hypotheses.

Other models implicitly perform multiple-hypothesis tracking by representing the position as a probability distribution over the nodes of the map. Most of these models are based on Markov localization. The major strength of these models being their capacity to continuously accumulate any idiothetic or allothetic cues about the robot's position, they do not strongly rely on individual node recognition. They consequently use allothetic cues that are subject to very strong perceptual aliasing, relying on sequences of these data to disambiguate places.

For example, Hertzberg & Kirchner (1996) use the junctions in a sewer system as topological nodes. Node recognition is performed by a neural network that is trained on sonar data to recognize different types of junctions (for example T-junctions or X-junctions). The probability of being at a given node is assigned a priori on an empirical basis, given the recognized situation and its uncertainty.[4] Likewise, Shatkay & Kaelbling (2002) use corridor junctions in an office environment. Node definitions in their model are based on perception vectors that are acquired in the corresponding positions. Each node stores a probability distribution over these vectors that encodes what the robot may perceive at the corresponding place. This distribution is used, given a perception, to evaluate the robot's probability of being at this place. In the models of Cassandra, Kaelbling, & Kurien (1996) and Simmons & Koenig (1995), the states of the underlying POMDP are 1-m wide squares which cover an office environment where the corridors are assumed to be orthogonal. Features such as WALLS, DOORS or OPEN-AREAS are used as perceptions in these models. These features are extracted from local occupancy grids constructed from sonar scans which, according to the authors, allows feature detections that are much more reliable than detections based directly on the sonar scans. According to these approaches, each node stores the probability that the robot detects each feature at the corresponding location. The model of Theocharous, Rohanimanesh, & Mahadevan (2001) also relies on local occupancy grids to define allothetic cues, but only takes into account the presence or absence of an obstacle at a predefined distance in four directions around the robot. Moreover, the map is based on a hierarchical POMDP, a structure that allows low-level states similar to those of the model of Simmons & Koenig (1995) to be grouped into higher-level states that represent concepts such as corridors or junctions. Only low-level states have associated probabilities of making each perception. High-level states simply reflect the robot's probability of presence in one of the low-level states it merges, but are not associated with perceptions. The nodes of the map in the model of Filliat & Meyer (2002) densely cover the whole environment in an irregular fashion with a mean spacing of 50 cm. Two kinds of allothetic cues are memorized in each node: the values of a sonar sensors belt and a grey-level panoramic image of the robot's surroundings subsampled to a 36×1 image. The very low resolution used leads to strong perceptual aliasing. The probability for the robot's being at a given node is estimated according to the similarity of the current perceptions with the perceptions memorized in this node. Finally, the model of Kuipers & Beeson (2002) memorizes in each node several laser range-finder scans of the environment that are acquired by the robot with different orientations in the corresponding place.

In all these models, idiothetic information is used to update the probability of being in each node when the robot moves according to the information stored in the links between nodes. In the models of Shatkay and Kaelbling and of Filliat and Meyer, the relative metric position of the nodes is available. The probability of making a given transition simply depends on the similarity between the length and direction recorded by the odometry between the two nodes, on the one hand, and the length and direction of the link stored in the map, on the other. However, most

[4] For example, the probability of being at the mid-leg of a T-junction, having detected a mid-leg of a T-junction, is 0.8, while the probability of being at the mid-leg of a T-junction, having detected an exit, is only 0.05.

models resort to idiothetic cue discretization in the form of a set of actions that the robot can accomplish. In the above-mentioned work of Hertzberg and Kirchner, because information about sewer lengths is highly unreliable, only actions made at sewer junctions are used. Moreover, as sewer junctions are supposed orthogonal, only a limited set of actions of the type LEFT-TURN or GO-STRAIGHT are used. Between junctions, the robot simply follows the sewer. Actions have probabilistic outcomes that are estimated empirically.[5] In Cassandra et al., Simmons and Koenig and Theocharous et al.'s models, the possible actions are RIGHT-TURN, LEFT-TURN and GO-FORWARD 1 m. The outcomes of these actions are modeled probabilistically.[6] The model of Simmons and Koenig additionally uses several parallel chains of Markov states in order to represent the corridor length uncertainty. In Theocharous et al.'s model, specific probabilities are associated with the transitions between high-level states.

The model of Kuipers & Beeson (2002) is singular in the sense that it makes it possible to switch from multiple hypothesis tracking to single hypothesis tracking or to direct position inference. It makes use of a Markov localization approach during mapping, which allows global localization and affords robustness to mapping errors. When the map is complete and precise, this model can use either direct position inference, if the sensory situation is unique, or single position tracking, when the position is subject to perceptual aliasing.

Very similar approaches characterize other models even though they do not resort to Markov localization. The concept of *state set progression* in Nourbakhsh et al. (1995), for example, shares most of its features with the previous Markov localization-based models. The model of Kortenkamp et al. (1994) also proposes a similar framework for global localization in a topological map. Finally, Hafner (2000) describes a model where the map is represented by a neural network. This network receives inputs from a 16-neuron layer which encodes allothetic information derived from a panoramic image, sub-sampled to a 16×1 pixel picture. Moreover, the topology of the neural network encodes the connectedness of the places, i.e. whether or not the robot may move from one place to another, together with the direction followed to move from one place to another. The activity of each node in the map depends on the sum of three terms that respectively depend upon the current view, the activation at the previous time-step, and the recent idiothetic cues, in a way very similar to Markov localization models.

In all the above models, the probability distribution over states is updated whenever a new action or a new perception is made. In the POMDP-based models, the state representing the robot's position is usually taken as the most likely state, although Cassandra et al. (1996) review other possibilities, such as taking the barycentre of the nodes, which is similar to population vector coding. Such a population vector coding method is also used by Filliat & Meyer (2002). Nourbakhsh et al. and Kortenkamp et al.'s models simply choose the most activated node. However, when this activation exceeds a given threshold, multiple-hypothesis tracking is stopped and a simple-hypothesis tracking algorithm is used instead.

4.5.2. *Topological map/metric position*

When a position is associated to each node of the map, several metric-position hypotheses about the robot's position can be monitored. These positions can be updated in parallel when any new information becomes available. A credibility can also be assigned to each hypothesis, thus making it possible to assess which is the current best position estimate (Fig. 24).

Positions and likelihood updates can be performed using techniques similar to those of Section 4.4.2, as it was done by Duckett & Nehmzow (1998) for example. The model of these authors calls upon a topological map where each node has an associated position in a 2D space and stores a definition of the corresponding place using sonars and infrared sensors. All the hypotheses are updated using the idiothetic cues recorded by the robot. When new allothetic information is received, the nodes that correspond to these cues are sought and the likeli-

[5] For example, when the robot detects that it has performed a RIGHT-TURN, the probability that it has actually performed a RIGHT-TURN is 0.9, while the probability that it has actually performed a U-TURN is 0.04.

[6] For example, going forward 1 m leads to the state ahead of the current state with a probability 0.7, but to the state 2 m away with probability 0.05.

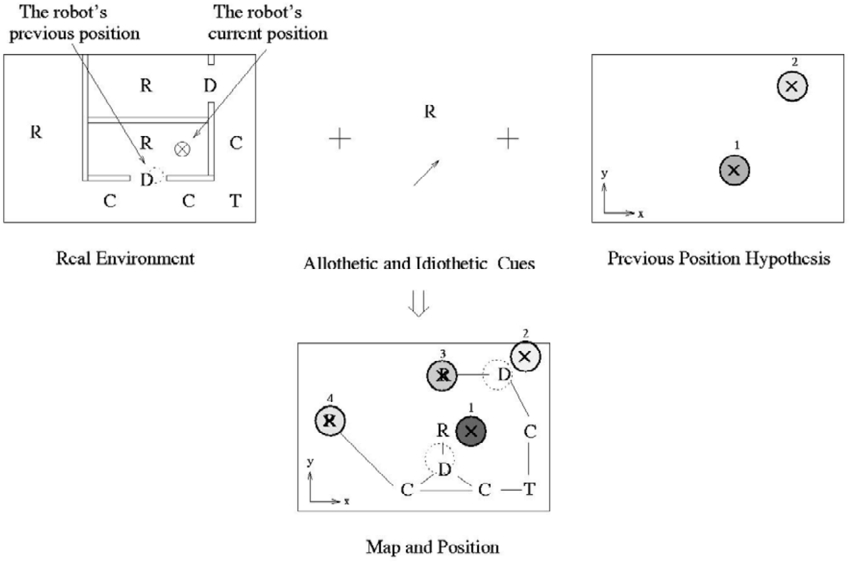

The robot's previous position
The robot's current position
R
R
D
R
C
D
C
C
T
Real Environment
R
Allothetic and Idiothetic Cues
2
1
y
x
Previous Position Hypothesis
2
3
D
4
1
R
C
D
C
C
T
y
x
Map and Position

Fig. 24. A set of possible metric positions is maintained and updated using idiothetic and allothetic cues, with procedures similar to those of Section 4.4.2 (hypotheses 1 and 2). Each hypothesis is assigned a probability (represented by the grey level of the circle surrounding each position) that is also updated using the newly acquired information. Whenever allothetic cues correspond to nodes that do not fit any existing hypothesis, new position hypotheses are generated (hypotheses 3 and 4).

hood of each hypothesis is updated by increasing the likelihood of hypotheses that are close to a recognized node, on the one hand, and by decreasing the likelihood of hypotheses that do not fall within a recognized node, on the other hand. A Kalman filter is used to update each hypothesis' coordinates with the coordinates of the corresponding recognized nodes. If the position of one recognized node does not correspond to any hypothesis in the current set, this position is simply added as a new hypothesis. The position of the robot is finally considered to be the position with the highest likelihood. Note that, concerning a single hypothesis, this process is very similar to the one used by Kurz or Balakrishnan et al. (see Section 4.4.2).

However, completely different frameworks may be used, achieving similar results. Donnart & Meyer (1996), for example, propose a very unusual model which is able to perform multiple-hypothesis tracking. In their approach, the environment is mapped as a set of production rules, organized in a hierarchical structure. The system is designed for an environment containing polygonal obstacles in which a robot heads towards a goal whose position is known in the 2D space. Landmarks are associated with points where the robot encounters or leaves the neighborhood of an obstacle, and where it increases or decreases its proximity to the goal. Consequently, landmarks are not directly associated with the robot's perception, but rather with the variations of an abstract notion of 'satisfaction'. Localization in this context entails managing a list of location hypotheses associated with an error and a confidence level. These values are updated using the correspondence of the encountered landmarks with the landmarks the robot would detect if it were at the hypothesized location. New hypotheses are added whenever a detected landmark corresponds to a landmark stored in the map and does not correspond to any existing hypothesis.

In the model of Oore et al. (1997) a robot's position is represented as a probability distribution over the cells of a grid that covers the whole environment. Using a new allothetic situation and the position of a given cell as input, a neural network is trained to estimate the probability of making this perception from this cell. After learning, the network can be used to estimate, through Bayes' rule, the probability for each cell to be the robot's current position, given the allothetic cues. Idiothetic cues are used to update the position probability distribution, in a manner similar to the Markov localization update cycle. This position estimation technique resembles that of Burgard, Fox, Hennig, & Schmidt (1996) (see Section 4.5.3). However, the probability of being in a cell given an allothetic perception is not estimated using a metric model, but through the neural network. This is why this model may be assigned in this paper amongst the category of models calling upon topological maps, as the use of a sensor model is not mandatory in this application. Nevertheless, the corresponding map is admittedly not a topological one in the strict sense of the term.

4.5.3. Metric map/metric position

The models in this section basically extend the framework of the models in Section 4.4.3 by relaxing the single-hypothesis assumption. This may be achieved by explicitly tracking multiple position hypotheses, via multiple Kalman filters, or by associating the position with a probability distribution other than a mere Gaussian, as is the case when Kalman filtering is used (Fig. 25). The corresponding models usually afford the most precise and efficient localization methods.

Explicitly tracking multiple hypotheses by Kalman filtering is done in the model of Piasecki (1995). His robot uses sonar sensors and moves in a world made of polygonal obstacles. Given a sensor reading, hypotheses are made on the objects which may have generated the current perception. The hypotheses at successive time steps are grouped into scenarios. Within each scenario, a Kalman filter is used to update the robot's position using odometry and perceptions. A credibility is assigned to each scenario by comparing, at each time step, the perceptions predicted given the robot's position estimate with the actual perceptions. Simulation results show a gain in robustness as compared with standard Kalman filtering techniques. Following a similar scheme, Jensfelt & Kristensen (1999) use a laser range-finder, which makes it possible to detect several types of features that are divided into two groups: *creative features*, that allow a robot's precise position to be estimated (such as doors), and *supportive features*, that only allow the robot's position to be partially estimated (such as walls that provide information in their perpendicular direction only). When new features are detected, they are used in conjunction with the map to generate position candidates. These candidates are then matched with the current position hypotheses. A match is considered

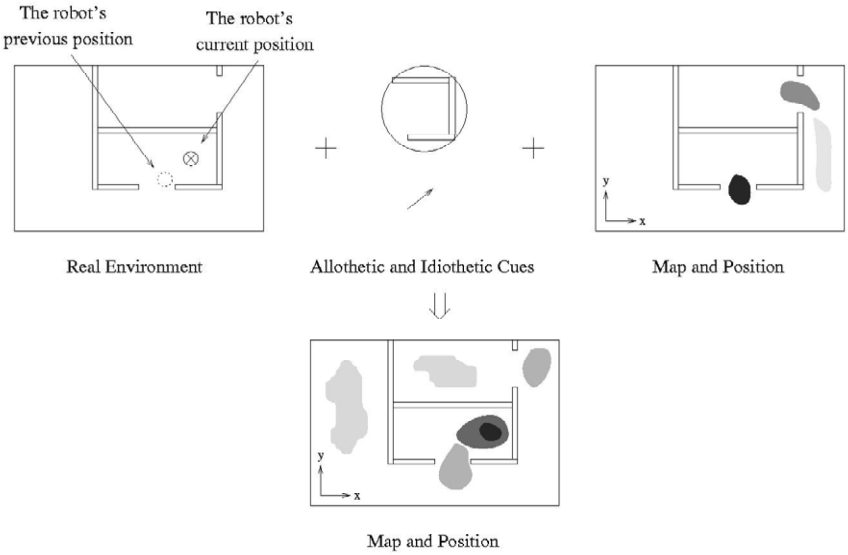

The robot's previous position
The robot's current position
+
+
Real Environment
Allothetic and Idiothetic Cues
Map and Position
y
x
⇓
y
x
Map and Position

Fig. 25. To perform multiple hypotheses tracking, a probability distribution over positions may be maintained and updated using idiothetic and allothetic cues. Unlike models based on Kalman filters, this probability distribution is not assumed to be Gaussian.

successful if the Mahalanobis distance between a candidate and a hypothesis falls below a given threshold and is then used to update the hypothesis estimate. Unmatched candidates provided by creative features are used to create new hypotheses. The likelihood of each hypothesis is then assessed, given the probability of each detected feature to be perceived at the corresponding position. In these two models, as the number of hypotheses could grow exponentially with time, various heuristics are introduced to keep the algorithm practicable.

Other models directly monitor general probability distributions through Markov localization without resorting to Kalman filtering. For example, in the model of Burgard et al. (1996), the position and orientation in the environment are discretized into a fine-grained regular grid, while the map is represented by an *occupancy grid* based on the same discretization. The robot's position is represented as a probability distribution over the elements of this grid. Sonar scans provide allothetic cues and the probability of making a given perception at a given position in the environment is assessed by using a simplified sonar sensor model. The model of the robot's odometry is also simplified and is based on the assumption that a Gaussian noise corrupts the dead-reckoning estimation. These two models are used through the standard cycle of Markov localization to update the probability distribution. Despite the simplified models for sensors and odometry, this approach proved to be reliable in a variety of environments. Extending this approach to highly populated environments, Fox, Burgard, Thrun, & Cremers (1998) found that this basic scheme may lead to poor performance because highly corrupted sensor measurements are used by the localization process. The author's solution is to filter the sensor readings so as to get rid of most of the measurements originating from people around the robot. Only sensor readings that decrease the robot's position uncertainty and that are consistent enough with respect to the current position estimate are used. Extending this approach by using a laser range-finder and a camera, Thrun et al. (1999) were able to reliably localize the robot in a crowded museum over several weeks. Olson (2000) also mentioned that the similarity measurement used in his global localization system (see Section 4.3.3) may be used to estimate the probability distribution over the states of the discretization and may be integrated into a similar localization system.

The Monte-Carlo localization method of Fox, Burgard, Dellaert, & Thrun (1999) represents the probability distribution over the continuous 2D space by a weighted set of samples (i.e. a set of points with an associated *importance factor*). This representation, called *importance sampling*, is a powerful way of representing arbitrary probability distributions that may be updated using sensor models similar to the one used by Burgard et al. This approach proved to be superior to the previous one because it afforded more accurate localization of the robot while using an order of magnitude less memory and computations.

Alternative frameworks that do not use probabilities may be called on, as in the model of Saffiotti & Wesley (1995), in which a fuzzy set is used to represent the likelihood of the robot's position. This fuzzy set yields the *possibility* of the robot's being at any given position in the environment (Fig. 26). The robot's environment is modeled as a set of objects (doors, wall, corridors), each of them having an associated fuzzy set that represents its position. The fuzzy set representing the robot's position is updated using idiothetic cues in order to take the robot's displacements into account. As the robot moves, features are extracted from sonar values and are represented by fuzzy sets relative to the robot. The perceived and memorized objects are then matched to provide a fuzzy set representing the possible robot's position in the map, given its perceptions. This set is then merged with the set representing the position before perceptions were made. Such an update cycle is similar to the Markov localization update cycle. This technique is computationally efficient and may compare favorably with probability-based techniques in case of high uncertainty.

5. Discussion

For robots, animals, and even for men, to be able to use an internal representation of the spatial layout of their environment to position themselves is a very complex task, which raises numerous issues of

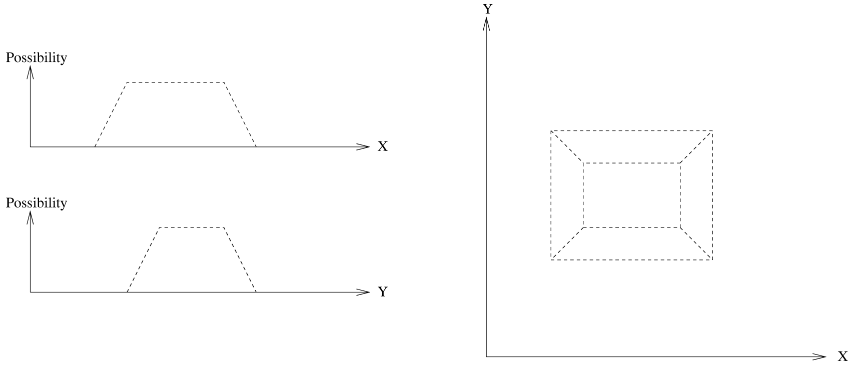


Fig. 26. Example of position representation using fuzzy sets. The left portion of the figure shows the projection of the set on the *x* and *y* axes. The right part shows the position possibility in the *x*, *y* space. The height of the fuzzy set at a given position represents the credibility for this position to be the robot's current position.

perception, categorization and motor control that must all be solved in an integrated manner to promote survival. Thus, among the different ways a robot may categorize its perceptions to build an internal representation of its environment and to localize itself, some are clearly better suited than others, depending upon the characteristics of the environment, the nature and reliability of the allothetic or idiothetic sensors that the robot may call upon, and the possibilities it has to move itself or to perceptually scan the surroundings.

5.1. Robots

In the case of robots, because they are often equipped with powerful distance sensors such as laser range-finders, a lot of localization systems that have been described here made use of metric maps and of metric sensor models. The corresponding approaches capitalize upon highly accurate, robot-independent, maps of the environment. Although such objective representations are not very close to the robot perceptual capacities and although they are more difficult to build than topological maps, they are very meaningful to humans operating the robots, and they facilitate the use of the same map by different robots.

The fact that topological maps may call upon any sensor modality, without the requirement of a sensor model, is an important advantage for localization. This capacity potentially allows the use of richer sensory data than mere distances to obstacles. However, the resulting precision of the estimated position is usually very coarse, and using a metric model for sensor data together with a metric map, usually makes a finer estimation of the position possible. Models that use topological maps and compute a metric position estimate for the robot seem a good compromise in this respect.

A completely autonomous localization system should allow a robot to keep an estimation of its position under any circumstances, and in any environment. Models calling upon direct position inference that have been presented in this review are clearly limited in this respect, as they require an environment where no perceptual aliasing occurs. This is seldom the case and, when the robot is unable to perceptually distinguish between two places, the corresponding localization system is either unable to estimate the robot's position or requires human intervention. Such intervention must modify the robot's perceptual system or must change the environment in order to cancel the perceptual aliasing problem. Models implementing position tracking make a first step toward solving this problem. They, however, rely on an initial position estimate that must be supplied by a human operator, or on an initial position that is not subject to perceptual aliasing. Moreover, if the position estimate accidentally becomes false, the corresponding systems are usually unable to autonomously recover a correct estimation. These problems are solved by systems that call upon multiple-hypothesis tracking, because they are able to estimate the robot's position without any initial estimate. Moreover, at least in cases were Markov localization implementations are used, the corresponding systems are reliable and resistant to noise. Several robots operating in challenging environments, such as crowded museums, successfully demonstrate the advantages of this approach (Buhmann et al., 1995; Thrun et al., 1999). Other advantages are afforded by the approach of Kuipers & Beeson (2002) that makes it possible to use the most appropriate localization method according to the characteristics of the current situation.

However, still from the point of view of autonomy, models with multiple-hypothesis tracking face difficulties when a map of an unknown environment has to be built. Indeed, most of these models cope poorly with partial maps of the environment because they capitalize on the fact that the probability of a given perception is correctly estimated over the whole environment. Consequently, systems that use such a localization scheme often resort to predefined maps provided by a human, or they build their map off-line using data gathered beforehand. Only a few models can use multiple-hypothesis tracking along with map learning. They capitalize either on the use of powerful metric sensors to rapidly acquire a map of the environment that is known to cover the current position (Thrun et al., 2000), or they make use of heuristics to autonomously detect whether the current position is already covered by the map or not (Kuipers & Beeson, 2002; Filliat & Meyer, 2002). In this latter model, a single-hypothesis tracking is temporarily used whenever the current position is detected to be outside the mapped area.

5.2. Animals

Concerning animals, all the models that were described in this review aimed at modeling the anatomy and functionalities of the rat's hippocampus, and they accordingly shared several characteristics.

In particular, most of them used a topological map, which encodes a set of places along with their allothetic definition from the animat's point-of-view, and which is relatively easy to elaborate with models of neuronal mechanisms found in rats. Indeed, such topological maps can be implemented using only simple associative memory mechanisms. Moreover, they do not require metric models for sensors, and they are therefore well adapted to integrating any non-metric cues used by rats, such as odors.

Most models also fall into the category of direct position inference or of position tracking. The underlying assumption is that place cells code for a unimodal representation of the position, and that the animal's hippocampus functionally acts as a Kalman filter. However, recent modeling efforts (Filliat & Meyer, 2002) raised the alternative hypothesis that cell activities could code for an arbitrary probability distribution with respect to the animal's position. According to such hypothesis, the more activated a given cell, the higher the probability that the animal is located in the corresponding place. Functionally, it would afford an animal the possibility of quickly relocalizing itself if it is passively moved from one place to another, or if it gets temporarily lost for whatever reason. It would also keep the animal from having to frequently calibrate its position using a special purpose mechanism, as it would be mandatory if it were using a single-hypothesis tracking strategy. Finally, it would permit an animal to survive in environments not specifically protected against perceptual aliasing difficulties, notably because the advantages of such probability distribution coding might be strengthened by those of preferentially using background cues—which are presumably more stable than foreground ones (Zugaro, Berthoz, & Wiener, 2001)—or by those of using active perception (Terzopoulos & Rabie, 1997; Ballard, 1991; Aloimonos, 1990) and active exploration strategies (Berlyne, 1950; Fehrer, 1956; Zimbardo & Miller, 1958) known to be called on by animals. Be that as it may, it turns out that arguments in favor of such a representation of an arbitrary probability distribution by cellular population coding are starting to be brought forth by neurobiologists (Zemel, Dayan, & Pouget, 1997). In this respect, POMDP-based models—like those that have been described in this review—would be well suited to providing hypotheses about how such a capacity might be implemented in vivo.

5.3. Men

This review definitely does not contribute to a better understanding of localization strategies that men may not share with other animals—because such strategies may call on specific cognitive processes like the faculty of reading written information provided by other men to characterize given places in the environment. This review nevertheless provides some clues as to the lower levels of the hierarchical structure that, according to Piaget, Inhelder, & Szeminska (1960), characterizes human spatial knowledge. Hence, although men are clearly capable of 'reading an internal map from above' and of planning metric detours (Trullier et al., 1997) in their head—a capacity that they may share with some animals like dogs (Chapuis, 1988) or chimpanzees (Menzel, 1973), and that they definitely share with the robots using metric maps and metric sensor models which were described in this review—they may also occasionally navigate and self-localize according to the simpler strategies that were ascribed herein to rats and to robots using topological maps and non-metric sensors. Numerous factors may affect the switch from one strategy to another, notably the possibility of applying the specific cognitive processes just mentioned, the nature and capacities of the sensors currently available, the degree of perceptual aliasing experienced in the environment, and various neural, motivational or emotional contingencies.

6. Conclusion

To build a topological or a metric map of its environment, a robot may call upon both allothetic and idiothetic sensors. Although the former are prone to perceptual aliasing and although the latter may suffer from cumulative errors, there are several ways in which such sensors may be used jointly to elaborate precise and robust internal representations of the environment's spatial layout. The corresponding methods may come under the heading of traditional engineering—e.g. Kalman filtering—or they may draw inspiration from bio-mimetic mechanisms—e.g. place-cell activity. Using such maps and such sensors, robots may self-localize according to three strategies: direct position inference, single-hypothesis tracking, and multiple-hypothesis tracking. However, the successful implementation of any such strategy entails a highly integrated approach, in which the type of sensors, the structure of the map, and the details of the localization algorithm must not only fit together, but also be adapted to the characteristics of the environment and to the robot's mission. A variety of such robotic implementations have been reviewed here, and a few hints about how animals might tackle the same issues of self-localization have been occasionally mentioned.

Acknowledgements

The authors are indebted to anonymous reviewers who greatly helped in improving the article. This work was supported by Robea, an interdisciplinary program of the French Centre National de la Recherche Scientifique.

References

Aloimonos, Y. (1990). Purposive and qualitative active vision. In *Proceedings of the tenth international conference on pattern recognition*. IEEE Press, pp. 346–360.

Arleo, A., & Gerstner, W. (2000). Spatial cognition and neuromimetic navigation: a model of hippocampal place-cell activity. *Biological Cybernetics, Special Issue on Navigation in Biological and Artificial Systems, 83*, 287–299.

Arleo, A., del Millán, J. R., & Floreano, D. (1999). Efficient learning of variable-resolution cognitive maps for autonomous indoor navigation. *IEEE Transactions on Robotics and Automation, 15*(6), 990–1000.

Arsenio, A., & Ribeiro, M. I. (1998a). Absolute localization of mobile robots using natural landmarks. In *Proceedings of the fifth IEEE international conference on electronics, circuits and systems*. IEEE Press.

Arsenio, A., & Ribeiro, M. I. (1998b). Active range sensing for mobile robot localization. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems*. IEEE Press, pp. 1066–1071.

Ayache, N., & Faugeras, O. (1989). Maintaining representations of the environment of a mobile robot. *IEEE Transactions on Robotics and Automation, 5*(6), 804–819.

Bachelder, I. A., & Waxman, A. M. (1994). Mobile robot visual mapping and localization: a view-based neurocomputational architecture that emulates hippocampal place learning. *Neural Networks, 7*(6/7), 1083–1099.

Balakrishnan, K., Bousquet, O., & Honavar, V. (1999). Spatial learning and localization in rodents: a computation model of the hippocampus and its implications for mobile robots. *Adaptive Behavior*, *7*(2), 173–216.

Ballard, D. H. (1991). Animate vision. *Artificial Intelligence*, *48*, 57–86.

Berlyne, D. E. (1950). Novelty and curiosity as determinants of exploratory behavior. *British Journal of Psychology*, *41*, 68–80.

Betke, M., & Gurvits, K. (1994). Mobile robot localization using landmarks. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-94)*, vol. 2. IEEE Press, pp. 135–142.

Boley, D., Steinmetz, E., & Sutherland, K. (1996). Robot localization from landmarks using recursive total least squares. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-96)*, vol. 4. IEEE Press, pp. 1381–1386.

Borghi, G., & Brugali, D. (1995). Autonomous map-learning for a multi-sensor mobile robot using diktiometric representation and negotiation mechanism. In *Proceedings of the international conference on advanced robotics (ICAR-95)*. IEEE Press, pp. 521–528.

Brooks, R. (1991a). Intelligence without reason. In *Proceedings of the 12th international joint conference on artificial intelligence*. Morgan Kaufman, pp. 569–595.

Brooks, R. A. (1991b). Intelligence without representation. *Artificial Intelligence*, *47*, 139–159.

Brown, R. G., & Donald, B. R. (2000). Mobile robot self-localization without explicit landmarks. *Algorithmica*, *26*, 515–559.

Buhmann, J., Burgard, W., Cremers, A. B., Fox, D., Hofmann, T., Schneider, F., Strikos, J., & Thrun, S. (1995). The mobile robot rhino. *AI Magazine*, *16*(1), 31–38.

Burgard, W., Fox, D., Hennig, D., & Schmidt, T. (1996). Estimating the absolute position of a mobile robot using position probability grids. In *Proceedings of the thirteenth national conference on artificial intelligence (AAAI-96)*. MIT Press, pp. 896–901.

Burgess, N., Recce, M., & O'Keefe, J. (1994). A model of hippocampal function. *Neural Networks*, *7*, 1065–1081.

Cassandra, A. R., Kaelbling, L. P., & Kurien, J. A. (1996). Acting under uncertainty: discrete bayesian models for mobile-robot navigation. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems*. IEEE Press, pp. 963–972.

Castellanos, J. A., Montiel, J. M. M., Neira, J., & Tardos, J. D. (1999). The spmap: a probabilistic framework for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*, *15*(5), 948–953.

Chapuis, N. (1988). *Les opérations structurantes dans la connaissance de l'espace chez les mamifères: Détour, raccourci et retour*. Marseille II: Université d'Aix, PhD thesis.

Clark, A. (1999). Where brain, body, and world collide. *Journal of Cognitive Systems Research*, *1*(1), 5–17.

Cliff, D., Husbands, P., Meyer, J. A., & Wilson, S. W. (Eds.), (1994). *From animals to animats 3. Proceedings of the third international conference on simulation of adaptive behavior*. MIT Press/Bradford Books.

Cox, I. J., & Leonard, J. J. (1991). Probabilistic data association for dynamic world modeling: a multiple hypothesis approach. In *Proceedings of the international conference on advanced robotics (ICAR-91)*. IEEE Press.

Cox, I. J. (1991). Blanche – an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, *7*(2), 193–204.

Dedeoglu, G., Mataric, M., & Sukhatme, G. S. (1999). Incremental, online topological map building with a mobile robot. In *Proceedings of mobile robots XIV*. SPIE, Society of Photo Optical, pp. 129–139.

Donnart, J. Y., & Meyer, J. A. (1996). Spatial exploration, map learning, and self-positioning with monalysa. In *From animals to animats 4. Proceedings of the fourth international conference on simulation of adaptive behavior (SAB-96)*. MIT Press, pp. 204–213.

Duckett, T., & Nehmzow, U. (1997). Experiments in evidence based localisation for a mobile robot. In *Proceedings of the AISB 97 workshop on spatial reasoning in animals and robots*. Springer.

Duckett, T., & Nehmzow, U. (1998). Mobile robot self-localization and measurement of performance in middle scale environments. *Robotics and Autonomous Systems*, *24*(1–2), 57–69.

Dudek, G., & MacKenzie, P. (1993). Model-based map construction for robot localization. In *Proceedings of vision interface*, pp. 97–102.

Einsele, T. (1997). Real-time self-localization in unknown indoor environments using a panorama laser range finder. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS-97)*. IEEE Press, pp. 697–703.

Engelson, S. P., & McDermott, D. V. (1992). Error correction in mobile robot map-learning. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-92)*. IEEE Press, pp. 2555–2560.

Feder, H., Leonard, J., & Smith, C. (1999). Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, *18*(7), 650–668.

Fehrer, J. A. (1956). The effects of hunger and familiarity of locale on exploration. *Journal of Comparative and Physiological Psychology*, *49*, 549–552.

Filliat, D., & Meyer, J. A. (2002). Global localization and topological map learning for robot navigation. In *From animals to animats 7. The seventh international conference on simulation of adaptive behavior (SAB02)*.

Fox, D., Burgard, W., Dellaert, F., & Thrun, S. (1999). Monte Carlo localization: efficient position estimation for mobile robots. In *Proceedings of the sixteenth national conference on artificial intelligence (AAAI-99)*. MIT Press, pp. 343–349.

Fox, D., Burgard, W., Thrun, S., & Cremers, A. B. (1998). Position estimation for mobile robots in dynamic environments. In *Proceedings of the fifteenth national conference on artificial intelligence (AAAI-98)*. MIT Press, pp. 983–988.

Franz, M. O., & Mallot, H. A. (2000). Biomimetic robot navigation. *Robotics and Autonomous Systems, Special Issue on Biomimetic Robot Navigation*, *30*, 133–153.

Franz, M., Scholkopf, B., Georg, P., Mallot, H., & Bulthoff, H. (1998). Learning view graphs for robot navigation. *Autonomous Robots*, *5*, 111–125.

Gasós, J., & Martín, A. (1997). Mobile robot localization using fuzzy maps. In Martin, T., & Ralescu, A. (Eds.), *Fuzzy logic in AI – selected papers from the IJCAI '95 workshop, number 1188 in LNCS*. Springer-Verlag, pp. 207–224.

Gaussier, P., Leprêtre, S., Joulain, C., Revel, A., Quoy, M., & Banquet, J. P. (1998). Animal and robot learning: experiments and models about visual navigation. In *Proceedings of the seventh European workshop on learning robots*. Springer Verlag.

Gaussier, P., Joulain, C., Banquet, J. P., Lepretre, S., & Revel, A. (2000). The visual homing problem: an example of robotics/biology cross-fertilisation. *Robotics and Autonomous Systems*, *30*(1–2), 155–180.

Georgopoulos, A. P., Schwartz, A. B., & Kettner, R. E. (1986). Neuronal population coding of movement direction. *Science*, *233*, 1416–1419.

Gomes-Mota, J., & Ribeiro, M. I. (2000). Mobile robot localisation on reconstructed 3d models. *Robotics and Autonomous Systems*, *31*(1–2), 17–30.

Greiner, R., & Isukapalli, R. (1996). Learning to select useful landmarks. *IEEE Transactions on Systems, Man, and Cybernetics-Part B. Special Issue on Learning Autonomous Robots*, *26*(3), 437–449.

Guibas, L. J., Motwani, R., & Raghavan, P. (1997). The robot localization problem. *SIAM Journal on Computing*, *26*(4), 1120–1138.

Guillot, A., & Meyer, J. A. (2001). The animat contribution to cognitive systems research. *Journal of Cognitive Systems Research*, *2*(2), 157–165.

Gutmann, J., & Konolige, K. (2000). Incremental mapping of large cyclic environments. In *Proceedings of the IEEE international symposium on computational intelligence in robotics and automation (CIRA-2000)*. IEEE Press.

Gutmann, J. S., & Schlegel, C. (1996). Amos: comparison of scan matching approaches for self-localization in indoor environments. In *Proceedings of the 1st Euromicro workshop on advanced mobile robots*. IEEE Press, pp. 61–67.

Hafner, V. V. (2000). Learning places in newly explored environments. In *From animals to animats 6. Proceedings of the sixth international conference on simulation of adaptive behavior (SAB2000). Proceedings supplement*. ISAB, pp. 111–120.

Hallam, B., Floreano, D., Hallam, J., Hayes, G., & Meyer, J. A. (Eds.), (2002). *From animals to animats 7. The seventh international conference on simulation of adaptive behavior (SAB02)*. MIT Press.

Hara, F., & Pfeifer, R. (2000). On the relation among morphology, material and control in morpho-functional machines. In *From animals to animats 6. Proceedings of the sixth international conference on simulation of adaptive behavior (SAB2000)*. ISAB, pp. 33–40.

Hébert, P., Betgé-Brezetz, S., & Chatila, R. (1996). Decoupling odometry and exteroceptive perception in building a global world map of a mobile robot: the use of local maps. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-1996)*. IEEE Press, pp. 757–764.

Hertzberg, J., & Kirchner, F. (1996). Landmark-based autonomous navigation in sewerage pipes. In *Proceedings of the first Euromicro workshop on advanced mobile robots*. IEEE Press, pp. 68–73.

Howell, J., & Donald, B. R. (2000). Practical mobile robot self-localization. In *Proceedings of the international conference on robotics and automation*. IEEE Press, pp. 3485–3492.

Jensfelt, P., & Kristensen, S. (1999). Active global localisation for a mobile robot using multiple hypothesis tracking. In *Proceedings of the IJCAI-99 workshop on reasoning with uncertainty in robot navigation*. Morgan Kaufmann, pp. 13–22.

Karch, O., & Wahl, T. (1999). Relocalization—theory and practice. *Discrete Applied Mathematics: Special Issue on Computational Geometry*, *93*, 89–108.

Kortenkamp, D., Huber, M., Koss, F., Belding, W., Lee, J., Wu, A., Bidlack, C., & Rogers, S. (1994). Mobile robot exploration and navigation of indoor spaces using sonar and vision. In *Proceedings of the AIAA/NASA conference on intelligent robots in field, factory, service, and space (CIRFFSS 94)*. AIAA, pp. 509–519.

Kortenkamp, D., & Weymouth, T. (1994). Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the twelfth national conference on artificial intelligence (AAAI-94)*. MIT Press, pp. 979–984.

Kuipers, B., & Beeson, P. (2002). Bootstrap learning for place recognition. In *Proceedings of the 18th national conference on artificial intelligence (AAAI-02)*. AAAI/MIT Press.

Kuipers, B. J. (2000). The spatial semantic hierarchy. *Artificial Intelligence*, *119*, 191–233.

Kuipers, B. J., & Byun, Y. T. (1991). A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*, *8*, 47–63.

Kunz, C., Willeke, T., & Nourbakhsh, I. (1997). Automatic mapping of dynamic office environments. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-97)*, vol. 2. IEEE Press, pp. 1681–1687.

Kurz, A. (1995). Alef: an autonomous vehicle which learns basic skills and construct maps for navigation. *Robotics and Autonomous Systems*, *14*, 172–183.

Lee, W. Y. (1996). *Spatial semantic hierarchy for a physical mobile robot*. University of Texas at Austin, Computer Science Department, PhD thesis.

Leonard, J. J., Durrant-Whyte, H. F., & Cox, I. J. (1992). Dynamic map building for an autonomous mobile robot. *International Journal of Robotics Research*, *11*(4), 89–96.

Levitt, T. S., & Lawton, D. T. (1990). Qualitative navigation for mobile robots. *Artificial Intelligence*, *44*, 305–360.

Lu, F., & Milios, E. (1997). Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, *4*, 333–349.

Madsen, C., Andersen, C., & Rensen, J. (1997). A robustness analysis of triangulation-based robot self-positioning. In *Proceedings of the 5th symposium for intelligent robotics systems*.

Maes, P., Mataric, M., Meyer, J. A., Pollack, J., & Wilson, S. W. (Eds.), (1996). *From animals to animats 4. Proceedings of the fourth international conference on simulation of adaptive behavior*. MIT Press/Bradford Books.

Mataric, M. J. (1992). Integration of representation into goaldriven behaviour-based robots. *IEEE Transactions on Robotics and Automation*, *8*(3), 304–312.

Maybeck, P. S. (1979). *Stochastic models, estimation and control*. Academic Press.

Menzel, E. W. (1973). Chimpanzee spatial memory organization. *Science*, *182*, 943–945.

Meyer, J. A., Berthoz, A., Floreano, D., Roitblat, H., & Wilson, S. W. (Eds.), (2000). *From animals to animats 6. Proceedings of the sixth international conference on simulation of adaptive behavior*. MIT Press.

Meyer, J. A., & Filliat, D. (2003). Map-based navigation in mobile robots: II. A review of map-learning and path-planning strategies. *Cognitive Systems Research*, to appear, Xref: doi:10.1016/S1389-0417(03)00007-X.

Meyer, J. A., Roitblat, H., & Wilson, S. W. (Eds.), (1993). *From animals to animats 2. Proceedings of the second international conference on simulation of adaptive behavior*. MIT Press/Bradford Books.

Meyer, J. A., & Wilson, S. W. (Eds.), (1991). *From animals to animats. Proceedings of the first international conference on simulation of adaptive behavior*. MIT Press/Bradford Books.

Moravec, H., & Elfes, A. (1985). High resolution maps from wide angular sensors. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-85).* IEEE Press, pp. 116–121.

Moutarlier, P., & Chatila, R. (1990). An experimental system for incremental environment modeling by an autonomous mobile robot. In *Experimental robotics 1.* Springer-Verlag, pp. 327–346.

Nehmzow, U., & Owen, C. (2000). Robot navigation in the real world: experiments with manchester's fortytwo in unmodified, large environments. *Robotics and Autonomous Systems*, *33*(4), 223–242.

Nourbakhsh, I., Powers, R., & Birchfield, S. (1995). Dervish, an office navigating robot. *AI Magazine*, *16*(2), 53–60.

O'Keefe, J., & Dostrovsky, J. (1971). The hippocampus as a spatial map, preliminary evidence from unit activity in the freely moving rat. *Experimental Brain Research*, *34*, 171–175.

Olson, C. F. (2000). Probabilistic self-localization for mobile robots. *IEEE Transactions on Robotics and Automation*, *16*(1), 55–66.

Oore, S., Hinton, G., & Dudek, G. (1997). A mobile robot that learns its place. *Neural Computation*, *9*, 683–699.

Pfeifer, R., Blumberg, B., Meyer, J. A., & Wilson, S. W. (Eds.), (1998). *From animals to animats 5. Proceedings of the fifth international conference on simulation of adaptive behavior*. MIT Press/Bradford Books.

Pfeifer, R., & Scheier, C. (1999). *Understanding intelligence*. MIT Press.

Piaget, J., Inhelder, B., & Szeminska, A. (1960). *The child's conception of geometry*. New York: Basic Books.

Piasecki, M. (1995). Global localization for mobile robots by multiple hypothesis tracking. *Robotics and Autonomous Systems*, *16*, 93–104.

Prescott, T. J. (1995). Spatial representation for navigation in animats. *Adaptive Behavior*, *4*(2), 85–123.

Radhakrishnan, D., & Nourbakhsh, I. (1999). Topological localization by training a vision-based transition detector. In *Proceedings of the 1999 IEEE/RSJ international conference on intelligent robots and systems (IROS-99).* IEEE Press, pp. 468–473.

Saffiotti, A., & Wesley, L. P. (1995). Perception-based self-localization using fuzzy locations. In *Reasoning with uncertainty in robotics*, *Lecture notes in computer science*, vol. 1093. Springer-Verlag, pp. 368–385.

Schiele, B., & Crowley, J. (1994). A comparison of position estimation techniques using occupancy grids. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-94).* IEEE Press, pp. 1628–1634.

Schultz, A. C., & Adams, W. (1998). Continuous localization using evidence grids. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-98).* IEEE Press, pp. 2833–2839.

Sharp, P. E. (1991). Computer simulation of hippocampal place cells. *Psychobiology*, *19*(2), 103–115.

Shatkay, H., & Kaelbling, L. P. (2002). Learning geometrically-constrained hidden Markov models for robot navigation: bridging the topological–geometrical gap. *Journal of Artificial Intelligence Research*, *16*, 167–207.

Sim, R., & Dudek, G. (1999). Learning visual landmarks for pose estimation. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-1999).* IEEE Press, pp. 1217–1222.

Simmons, R., & Koenig, S. (1995). Probabilistic navigation in partially observable environments. In *Proceedings of IJCAI'95.* Morgan Kaufman, pp. 1080–1087.

Smith, R., Self, M., & Cheeseman, P. (1988). Estimating uncertain spatial relationships in robotics. In *Uncertainty in artificial intelligence*. Elsevier, pp. 435–461.

Terzopoulos, D., & Rabie, T. F. (1997). Animat vision: active vision in artificial animals. *Videre: Journal of Computer Vision Research*, *1*(1), 2–19.

Theocharous, G., Rohanimanesh, K., & Mahadevan, S. (2001). Learning hierarchical partially observable markov decision processes for robot navigation. In *Proceedings of the IEEE conference on robotics and automation (ICRA-2001).* IEEE Press.

Thrun, S., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., Fox, D., Haehnel, D., Rosenberg, C., Roy, N., Schulte, J., & Schulz, D. (1999). Minerva: a second generation mobile tour-guide robot. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-1999).* IEEE Press.

Thrun, S., Burgard, W., & Fox, D. (2000). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-2000).* IEEE Press, pp. 321–328.

Thrun, S., Gutmann, S., Fox, D., Burgard, W., & Kuipers, B. (1998). Integrating topological and metric maps for mobile robot navigation: a statistical approach. In *Proceedings of the fifteenth national conference on artificial intelligence (AAAI-98).* MIT Press, pp. 989–995.

Thrun, S. (1999). Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, *99*(1), 21–71.

Thrun, S. (2000). Probabilistic algorithms in robotics. *AI Magazine*, *21*(4), 93–109.

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, *55*, 189–208.

Touretzky, D. S., Wan, H. S., & Redish, A. D. (1994). Neural representations of space in rats and robots. In Zurada, J. M., Marks, R. J., & Robinson, C. J. (Eds.), *Computational intelligence: imitating life*. IEEE Press, pp. 57–68.

Trullier, O., & Meyer, J. A. (2000). Animat navigation using a cognitive graph. *Biological Cybernetics*, *83*(3), 271–285.

Trullier, O., Wiener, S., Berthoz, A., & Meyer, J. A. (1997). Biologically-based artificial navigation systems: review and prospects. *Progress in Neurobiology, 51*, 483–544.

Ulrich, I., & Nourbakhsh, I. (2000). Appearance-based place recognition for topological localization. In *Proceedings of the IEEE international conference on robotics and automation (ICRA-2000)*, vol. 2. IEEE Press, pp. 1023–1029.

Varela, F., & Rosch, T. E. (1991). *The embodied mind: cognitive science and human experience*. MIT Press.

Von Wichert, G. (1998). Mobile robot localization using a self-organised visual environment representation. *Robotics and Autonomous Systems*, *25*, 185–194.

Werner, S., Krieg-Bruckner, B., Mallot, H., & Schweizer, K. (1997). Spatial cognition: the role of landmark, route, and survey knowledge in human and robot navigation. In *Informatik '97*. Springer, pp. 41–50.

Wijk, O., & Christensen, H. I. (2000). Localization and navigation of a mobile robot using natural point landmarks extracted from sonar data. *Robotics and Autonomous Systems, 31*(1–2), 31–42.

Yamauchi, B., & Beer, R. (1996). Spatial learning for navigation in dynamic environments. *IEEE Transactions on Systems, Man, and Cybernetics-Part B. Special Issue on Learning Autonomous Robots*, *26*(3), 496–505.

Yamauchi, B., & Langley, P. (1997). Place recognition in dynamic environments. *Journal of Robotic Systems, Special Issue on Mobile Robots*, *14*(2), 107–120.

Yamauchi, B., Schultz, A., & Adams, W. (1999). Integrating exploration and localization for mobile robots. *Adaptive Behavior*, *7*(2), 217–230.

Zemel, R. S., Dayan, P., & Pouget, A. (1997). Probabilistic interpretation of population codes. In *Advances in neural information processing systems*, vol. 9. MIT Press, p. 676.

Zimbardo, P. G., & Miller, N. E. (1958). Facilitation of exploration by hunger in rats. *Journal of Comparative and Physiological Psychology*, *51,* 43–46.

Zugaro, M. B., Berthoz, A., & Wiener, S. (2001). Background, but not foreground, spatial cues are taken as references for head direction responses by rat antedorsal thalamus neurons. *Journal of Neuroscience*, *21*(RC154), 1–5.