Robotica (2000) volume 18, pp. 227–234. Printed in the United Kingdom © 2000 Cambridge University Press

Navigation of mobile robots: open questions

M.A. Salichs and L. Moreno

Ingenieria de Sistemas y Automática, Universidad Carlos III de Madrid, Av. Universidad 30, Leganés, Madrid, (Spain)
E-mail: salichs@ing.uc3m.es and moreno@ing.uc3m.es

SUMMARY
This survey paper is focused on the navigation of mobile robots. The most important associated problems and the solutions given by different researchers are discussed. Several aspects are not yet satisfactorily resolved and some promising research approaches are addressed.

KEYWORDS: Navigation; Mobile robots; Intelligent autonomous vehicles; Localization; Planning.

1. INTRODUCTION

A basic capability of mobile robotics is navigation.[1] In the most ambitious expression of this capability, a mobile robot must be able to pass the following test:

The robot is placed in an environment that is unknown, large, complex and dynamic. After a time needed by the robot to explore the environment, the robot must be able to go to any selected place, trying to minimize a cost function (e.g. time, energy, etc).

We will call this test the *Maximum Navigation Test* (*MNT*). The robot must solve several problems to successfully perform the *MNT*. First, the robot must be able to move fast enough, under control and safely, avoiding static and moving obstacles (Motion Control problem). While moving, the robot must be able to collect knowledge of the environment (World Modeling problem) and be aware of its location (Localization problem). To move optimally in a large area, some planning is also necessary (Planning problem). These problems are not independent (e.g. world models are necessary to produce plans or to localize the robot), and they must be considered only as sub-problems of a more general one. The general problem is not just the sum of the sub-problems. It is also necessary, using a holistic point of view, to determine and manage the different sub-modules that appear in the solution of the sub-problems and the interactions between them (Architecture problem). In Sections 2 to 5, a brief state-of-the-art of the basic problems previously related can be found, highlighting some unresolved questions and new research areas.

After three decades of research in mobile robotics, the *MNT* can still be considered as unresolved, although many advances have been made. Due to the difficulty of working simultaneously on all the previous sub-problems, most of the hitherto research has been focused on trying to solve only some of them. Another common approach has been to simplify the requirements of the *MNT*, mainly those that concern the environment. For instance, it is frequently supposed that some knowledge of the environment is *a priori* known (e.g. landmarks for relocalization), or that tests are performed in a narrow, static or very structured area. As a consequence of an incorrect way of approaching the navigation problem in the mid 80s, the results obtained were very disappointing. Then, some researchers not only proposed a shift in the way of solving the problem (e.g. by eliminating symbolic models of the environment), but they also proposed a change in the problem to be solved. This historic evolution will be further analyzed in Section 2.

To perform the *MNT*, not only excellent algorithms are necessary. A real robot is quite different from a computer simulation. A real robot also needs sensory systems, locomotion systems, energy supply, computing devices, etc. Although they are not of this paper's scope, it is essential to always keep in mind their importance and the way they influence and constrain the design of the robot's navigation modules.

2. ARCHITECTURE

Early architectures were based on the Sense-Model-Plan-Act approach. Sensed information of the environment was incorporated into maps, which were used to plan trajectories to be followed by the robot. The results obtained with this approach were very disappointing. The reasons are easy to understand. Sensors only provide partial information of the real world, with errors in the measurements. This incomplete and corrupted information was then used to build maps where the complexity of the environment was further reduced to simplify the modeling process (e.g. the real 3D word modeled as 2D polygons). In addition, models only represented static environments (e.g. without people moving near the robot). The conclusion is a general rule of modeling: models (i.e. maps and the localization of the robot in the maps) are only approximations of the real world. In the early architectures, motion decisions were based only on the information of the maps. Results were foreseeable: models were inaccurate, hence motion decisions were also inaccurate. Robots using this architecture would move very slowly to avoid collisions. They had to stop frequently when an obstacle was detected in a previously planned path in order to replan a new one.

Different researchers noticed that the source of the robots' malfunction was related to the inaccuracy of the maps and, to solve it, they suggested a radical solution: not to use world models. Instead of maps, they proposed to use direct information from sensors. Brooks' works are the best-known departure from early architectures.[2,3] A general principle of the new systems was: "The world is its own best model". Motion in these systems was the consequence of a reaction to the data from sensors, instead of a planned action. The result was a radical shift from plan-based actions to sensor-based actions. Different control systems produced different behaviours of the robot. Simple behaviours could be fused to produce complex behaviours. With the new methodology, robots began to move fast and safely in complex environments, even with mobile obstacles. These manifest results produced an immediate shift of many researchers to the new architectures. It is interesting to notice that reactive approaches were new, even revolutionary, in mobile robotics or *AI*. But they were the result of adapting very well known techniques in other areas such as control theory.

Renouncing the use of maps, pure reactive systems were also giving up performing navigation as defined in the *MNT.* Global goals (e.g. moving to a distant place) could not be optimally reached without global information (e.g. maps for navigation). The shift was too radical. Early mobile robots only used maps and pure reactive robots exclusively used sensory instantaneous information. Each approach had both benefits and disadvantages, but none was able to solve the navigation problem.

It was not necessary to wait too long before some researchers began to search for a compromise between both extremes in defining hybrid architectures. They tried to use the best of maps and instantaneous sensory data, without renouncing any of them. Planners were in charge of long-term decisions and reactive behaviours were used for motion execution. But the coexistence of planners and reactive behaviours is not easy. Classic planners produce rigid orders, while the response of reactive executors is unpredictable. A classic dilemma is authority vs. freedom. A solution found by most researchers is to produce sketchy plans, allowing a reactive executor enough freedom to control the robot at run-time. An intermediate module is also frequently introduced between the planner and the executor. Three level architectures, with a deliberative level, an intermediate level and a reactive level are the most common hybrid architectures. The intermediate level is usually a discrete-event controller.[4] Examples of hybrid architectures are ATLANTIS,[5] 3T[6] and the architecture proposed in reference 1.

2.1 Some open problems and new research areas

Architectures for learning. Most mobile robots are based on pre-programmed procedures. This approach does not make easy the scaling of the robot capabilities. Small changes in the tasks demanded from the robot, in the environment or in the sensory or locomotion systems, usually require complex re-engineering work. One of the solutions found in Nature is the incremental evolution of capabilities by learning. Up to now, learning in mobile robotics has usually been focused only on specific domains (e.g. behaviour fusion,[7] walking,[8] etc.). The challenge is to find an architecture that allows a robot to increment substantially all its capabilities, without making necessary code modifications by a programmer.

Emotions. There is evidence that humans use emotions to make long-term decisions,[9] and emotions are essentially reactive, i.e. reactions can be used not only to produce fast sensory-motor skills, but also to make fast high-level decisions. This is a new research field and a new way of integrating reaction and deliberation.

Scalable architectures. It is not the objective of this paper to discuss what the meaning of *autonomy* and *intelligence* is, but what is clear is that these attributes might be enhanced in future robots. Furthermore, navigation should be a mandatory skill of an intelligent autonomous mobile robot, but that is not all. A useful mobile robot must also have task-oriented capabilities (e.g. manipulation, cleaning, etc.). New architectures must be scalable and compatible with an increment of the mobile robot capabilities.

3. MOTION CONTROL

Reactive behaviours are a *de facto* standard for motion control of mobile robots (for those interested in a complete study of behaviour-based robotics, see reference [10]). Behaviours are usually developed incrementally. There is a set of controllers, each of which is defined for a basic behaviour (e.g. path following, obstacle avoidance, contour following, etc.). The design and implementation of the controllers are made with any methodology of the many available in control theory, from classic PIDs to non-linear controllers. Basic behaviours are fused to produce emergent behaviours. The fusion can be made in two ways: by combining behaviour and by sequencing behaviour. In the first case, the outputs of the emergent behaviour are a combination of the outputs of the basic behaviours, which are activated simultaneously. It is very common to obtain the outputs of the emergent behaviour by adding, with different weights, the outputs of the basic behaviours.[11] In the second case the basic behaviours are activated in sequence.

The fusion of behaviours must be dynamically adapted to the internal and external states of the robot. If the emergent behaviour is the result of adding outputs of the basic behaviours, an adaptation can be made by dynamically adjusting their weights. An example of this fusion methodology is AFREB.[12] In case of sequencing behaviours, events are necessary to trigger the activation and deactivation of the basic behaviours; in both cases, adaptation is based on perception processes. These perception processes are as important as the basic behaviours to develop emergent behaviours. To refer to both elements in a homogeneous way, some authors prefer the denomination skill, and hence skills are classified as perception skills and action skills.[13]

In three layer architectures, reactive basic controllers (i.e. the lower layer) are co-ordinated and supervised by an intermediate layer. Usually, basic skills are fused by using the sequencing method, and the intermediate level is in charge of controlling the sequencing process. In this case, the intermediate layer acts as a discrete event controller. Firby has proposed the use of reactive actions packages, RAPs, for sequencing of the modular motion skills.[14] RAPs are emergent behaviours obtained by sequencing primitive skills. The RAP system takes sketchy plans and breaks them down into steps that can be accomplished by activating a set of skills. This system has been used in several architectures, such as 3T and first versions of ATLANTIS. Gat, later, developed a new language, ESL,[4] that has been incorporated into ATLANTIS. Other examples of conditional sequencing systems are PRS[15] and REX/GAPPS.[16] Instead of defining specific languages, some researchers have used traditional methods of modeling discrete event systems, such as Petri nets.[17]

Pure reactive controllers produce motion commands using instantaneous information from sensors without foreseeing future actions. Future actions are usually only foreseen in planners working on maps. But the relationships sensors/reaction and maps/planning are not mandatory. Why not make short-term plans using direct sensor information? Sensor-based planners are an alternative option to design behaviour. Sensor information reflects the current state of the environment into the robot's planning process as a realistic way of considering autonomous robots operating in a dynamic and complex environment. Among these works, we can find roadmap methods like Canny and Lin's Opportunistic Path Planner,[18] adapted by Rimon[19] for sensor-based planning use.

3.1 Some open problems and new research areas

Perception skills. Up to now, there have been great efforts to develop action skills, but as indicated before, perception skills are equally important to develop complex emergent behaviours, e.g. sequencing systems need discrete events produced by perception skills. Human beings frequently use their skill in recognizing objects to produce these events. Unfortunately, in this area robot skills are very far removed from human skills. The skill in recognizing objects is fundamental to solve the *MNT*, not only for motion control, but also for world modeling and localization, using topological maps. Advances in this research area, which is not specific to mobile robotics, will allow one to significantly increase the capabilities of mobile robots.

Learning. The automatic generation of skills is an area that has not been explored enough. Research should be focused not only on basic behaviours, but also on the process of developing complex emergent behaviours by learning. The concepts of basic and emergent behaviours (or skills) are relative, because emergent behaviours can be fused recursively with other behaviours to produce new generations of emergent behaviours. It is necessary to develop architectures which are able to cope with this recursive evolution process.

Motion dynamics. The design of the motion control systems of many mobile robots is based only on kinematics models. To increase the speed and power efficiency of mobile robots, it is also necessary to take into account dynamic effects.

4. LOCALIZATION AND ENVIRONMENT MODELING

When a mobile robot moves through the environment and observes it, there are some aspects that introduce difficulties to model the environment or to localize the robot precisely: (1) The introduction of a certain degree of uncertainty by the displacement, especially if this displacement is estimated from odometry; (2) The existence of different local maps observed from different locations, and (3) The existence of unexpected obstacles and hidden objects.

Due to the necessity of modeling the environment and localizing the robot starting from the data coming from different information sources and different instant times, a method capable of integrating all this information is required. This method is usually called sensor fusion. The sensor fusion process tries to use all this information in order to improve problems like: the reduction of uncertainty in parameter values about the displacement and the geometrical primitives, the smoothness of the parameter estimates, the reduction of the dependence about non-valid hypothesis or *a priori* considerations. Different sensor fusion methods are used, among them the Bayes rule, Kalman filters and more recently the Hidden Markov models.

The uncertainty introduced by the robot displacement and the uncertainty in the environment detection are tightly coupled. In fact, environment modeling and localization can be considered as dual problems, because in order to localize a mobile robot many localization systems require a world model to match observed environment characteristic with the modeled ones, and also because, in order to build a world model, most systems require a precise localization of the robot. The methods used for sensor fusion in both problems are on many occasions similar. This duality is so important that if there exists a precise localization system, it is possible to reasonably model the environment and if there exists a good environment model, we can localize the robot precisely. Today, even for 2D indoor environments, it is a difficult task to build an environment model and to localize the robot simultaneously.

4.1 Environment modeling

Many different environment representations can be used according to the type of task to perform, the kind of environment and the type of sensor used. The most significant types of representations are: cell decomposition models, geometrical models and topological models.

The representation of an environment can be structured in different levels. A frequent structure considers three levels: A first level of modeling, that can be called *instantaneous model* or *snapshot*, which is constructed from a data acquisition and can be, on occasions, the data provided directly by the sensors; a second level of modeling can be called *local model* which contains different information about the same area of the environment detected during mobile robot operation; and a third level of modeling called *global model* which contains information about the whole operation environment.

Different kinds of model may be used at different levels of modeling, and occasionally, different kind of models coexists at a same level. Depending on the architecture used in the robot, different numbers of representation levels are required.

Cell decomposition modeling. The main idea of this kind of method consists of representing the environment by means of a decomposition in cells. This technique is not able to represent an object to be modeled exactly, but

https://doi.org/10.1017/S0263574799002210 Published online by Cambridge University Press

despite that fact, it is strongly recommended and, even though the object itself can be difficult to represent, the cell decomposition of any object is a very simple representation. The most frequent approaches use cubic cells for 3D environment models and square cells in 2D models.

The maximum resolution obtained depends on the selected size of the cell and can be different at different functional levels. This group of techniques does not require *a priori* maps about the environment, because the modeling is very simple.

Depending on the information associated with each cell, we can find different kinds of maps: occupancy probability maps,[20] accumulation maps[21] and elevation maps.[22,23]

The most usual way of representing the uncertainty in a cell decomposition modeling is to associate a probability to the cell which indicates the uncertainty about the cell occupation. This modeling technique typically uses Bayes' rule to update the occupancy probability of the cell.

Geometrical models. There are different possibilities to geometrically represent objects in the environment, but in mobile robotics the most frequent are *image models* where the geometry is formulated as an array of simple components without any functional description. These models let us have rich statistical information about the uncertainty of the geometric characteristics. One way of modeling a *geometric object* with a level of uncertainty consists of describing its characteristics, position and occasionally relationships by means of a vector function, which describes a particular kind of geometric object, a parameter vector which specifies a particular instance of the class of geometric objects modeled, and a probability distribution characterized by a probability density function which describes the probability of observing a particular instance of a geometric object given a certain robot localization (typically a Gaussian distribution).

To build a geometrical representation of the environment, the mobile robot has to extract geometrical information about the environment. This geometrical information can be extracted directly from sensor information or from other kinds of representations, like cell decomposition maps.

A large number of researchers have used this approach based on different sensors like ultrasound,[24–26] laser scanners[27] and artificial vision.[28]

The fusion process is based on the use of the Extended Kalman Filter to integrate all this information.

Topological models. These models try to model *relationships* between different types of entities. Depending on the type and significance of the model entities, the topological model is named differently and can be structured at different levels.

The main advantage of the topological approach is the qualitative aspects used to describe the environment instead of metric aspects, which are closer to human behaviour than another type of environment modeling. Besides, topological models can be easily structured in a hierarchical model even with metric-based models.[29,30]

In a topological representation, the environment is described by a connected graph where the adjacency property, the order property, or other relationships are modeled.

There are different possible strategies to build a topological model of the environment:

(i) One possibility is to build a geometrical model and to extract from it the topological properties required to build the topological model.[27,31]
(ii) A second possibility is to build the topological map directly from the observed qualitative properties of the environment.[30,32]

These models are easy to manipulate, and there exist powerful and efficient graph methods to solve an important number of problems; it is possible to work at different resolution levels, the integration of the local topological models into a global topological model is simple and does not require any special integration methods due to the uncertain nature of most relationships. Topological models are also compatible with other techniques of environment modeling. The main difficulty in this technique deals with the extraction of topological features from the environment information.

4.2 Localization

When we try to answer the question: "Where is the robot?", it is possible to give different answers. A first one can be (x, y, θ), called *geometric localization*. This way of expressing the localization of the vehicle does not depend on any characteristic of the environment, and gives us precise geometric information (with some additional parameters to express the uncertainty). This form of localization can be used in all kinds of environments but this is not the way a human operates.

Another possibility is to express the localization of the robot by means of relationships with the environment, for instance: the robot is *in room A, in front of the door*. This kind of localization is called *topological*. This method is closer to the way humans operate, but it requires a kind of environment with rich topological information in order to reach a reasonable level of precision. For instance, it is not suitable in a desert or on the sea.

Humans probably use a combination of both methods, having some knowledge of the relationship with the environment and also having rough estimations of the displacements in order to know their location at a given moment, e.g. the robot is *in room A*, *close to the door.*

There are different solutions to the localization problem that can be roughly grouped in three classes: dead reckoning systems, integrated localization systems and perception-based localization systems. Due to the lack of a single and general good method, mobile robots usually combine methods, one or more from each kind.

Dead reckoning systems are incremental positioning methods which do not require external sensor information to estimate the position and the orientation, and are always capable of providing an estimation of the increment in the robot position. These methods require periodical updating to correct the error accumulated in the robot's position. Among them, we can distinguish two groups: odometry and inertial navigation systems.

Integrated localization systems are absolute positioning methods which require external absolute references in order

https://doi.org/10.1017/S0263574799002210 Published online by Cambridge University Press

to estimate the position and the orientation. This technique estimates the absolute position and orientation of the robot by measuring the direction of incidence or the distances to several signals transmitted or reflected by artificial beacons. The locations of the beacons in the environment have to be known *a priori*. These methods have been used for ship navigation for many years as a very reliable localization method. Today, the Global Positioning System (GPS) is becoming very usual in outdoor vehicles, while laser-based systems are preferred in indoor applications. The only problem of these methods is that beacons have to be built and installed on the appropriate sites of the environment where the robot can perceive them.

Perception-based localization systems. These systems operate at two levels. At the first level, the perception system detects a feature in the environment and estimates the relative position of this feature with respect to the robot.

The second level is different depending on the kind of localization being used. In case we try to localize the robot geometrically, the detected characteristics are matched with those stored in a model of the environment, and by using a sensor fusion method, a geometrical location is estimated. Many researchers have been working on this system.[26,33–36]

A different possibility consists of directly extracting topological relationships from the perceived features to localize the robot topologically. The detection of distinctive landmarks located in the environment can be easy in case of artificial landmarks (designed and located especially for this purpose) and not so easy in case of natural landmarks (selected from the most distinctive characteristics of the environment). Artificial landmark detection methods are well developed and have proved to be reliable, but natural landmark detection methods are not yet sufficiently developed. A difficult point is to develop a system capable of detecting different natural landmarks in different environments. Ambient conditions can make the detection of the landmarks difficult, other objects with similar conditions can be mistaken for a proper landmark or feature, or on occasions may not be recognized. Koenig[37] and Kaelbling[38] have developed topological localization systems which operate in sample environment forms of T and L-shaped junctions and corridors. Other researchers, like Thrun,[39] use systems with a combination of geometrical and topological properties to localize the vehicle.

4.3. Open problems and new research areas

3D environment models and outdoor models. 3D models are required in open field operation environments to move the robot safely. This kind of environment requires more sophisticated models in order to determine the crossability characteristics of the environment. The segmentation in different areas, according to some crossability criteria, is a difficult task and an active research area. Outdoor urban environments where lights, signals and roads are present, introduce additional difficulties and are also unresolved problems in modeling.

Dynamic objects. In order to cope with real situations, dynamic objects should be detected, modeled and avoided in order to have reliable displacements. Despite the work done on this topic, the problem is still unresolved.

Initial point determination. Kalman filter-based techniques have proved to be robust and accurate for local position tracking of the robot's position, which means that once a robot has been localized on the map, it is possible to keep track of that position over time. Another different question is the global position estimation problem, that can be defined as the ability to estimate the initial robot position in an *a priori* known or previously learned map.

Feature and landmark extraction. Perception-based localization methods using vision are very active research areas, especially in topics related with the identification of objects and the pose estimation of the identified objects.

Topological modeling and localization. Most of the traditional localization methods try to determine geometrically the position and the orientation of the robot. Recent approaches look for methods to build topological models once features and landmarks are detected and for the later topological estimation of the robot's situation.

5. PLANNING

The planning problem is to determine a collision-free motion plan to reach a goal position from an initial position. This problem is, in most systems, simplified by considering uniquely the planning of the mobile robot movements, and by assuming that the mobile robot is a rigid solid and it is the only moving object within the environment (perfectly known). Dynamics and interaction problems are ignored in many systems. With all these assumptions, the movement problem is reduced to a geometrical path planning problem.

This is the most classic approach used in path planning problems, and it is still the most widely used. But there are some important assumptions which make frequent replanning necessary. One of them is the assumption of a known and precisely located environment, and another one is the assumption of a precise knowledge of the robot location.

The first assumption, about the precise knowledge of the obstacle's positions in the workspace, implies that, on many occasions, the planned path is not feasible because the obstacles are not located at expected locations or there are unmodeled obstacles. The second assumption, about the precise knowledge of the robot location, implies that a possible path to go from an initial location to a goal location avoiding obstacles is not feasible from a localization point of view because of the absence of information to locate the robot precisely.

In spite of the variety of methods, it is possible to group them into a few classes if we look at the approach to the problem. A first group of methods tries to determine a path between the initial configuration until the final configuration (path planning methods). It is possible to distinguish the following classes: roadmap methods, cell decomposition methods and reaction simulation methods.

5.1 Roadmap methods

The main idea of this method consists of capturing the connectivity of the free space into a network of unidimensional curves called *roadmap, R.* Once the roadmap is built,

https://doi.org/10.1017/S0263574799002210 Published online by Cambridge University Press

it is used as a set of standardized paths, and the path planning is reduced to finding a path between an initial and a final point on the *roadmap*. Different methods have been proposed. In mobile robotics, two are frequently used: *visibility graphs*[40,41] and *Voronoi diagrams*.[42]

The *visibility graph* is constructed considering as nodes the initial configuration, the goal configuration and all the vertices of configuration space obstacles, and edges of the visibility graph, the straight line segments that connect two nodes without penetrating inside the configuration space obstacles. To find a path on the visibility graph between the initial and final configurations, different techniques can be used. A classic method is the A^* algorithm and others can be used to obtain the shortest path in the graph.

The retraction method approach consists of defining a continuous mapping of the robot's free space onto a one-dimensional network of curves R lying in the free space. This mapping has to conserve the connectivity of the robot's free space in the roadmap R obtained through the retraction. This property reduced the motion planning problem in a free space to a motion planning into R. The method developed by O'Dúnlaip and Yap[43] retracts the free space onto the *generalized Voronoi diagram*. The Voronoi diagram is then the set of points with a minimal distance to two or more objects. This diagram has the interesting property of maximizing the separation between the robot path and the obstacles.

5.2 Cell decomposition methods

In this group of techniques, the robot's free space is decomposed into non-overlapping regions called *cells*. There are two types of cell decomposition methods: exact and approximate. The exact cell decomposition decomposes the free space into cells whose union is the free space. Two of these methods are: the *trapezoidal decomposition* and the *generalized Delaunay triangulation*. In approximate cell decomposition methods,[41,44] the free space is decomposed into cells having predefined shapes and the union of the cells is a subset of the entire free space.

Once the free space is decomposed, a *connectivity graph* is constructed based on the adjacency between cells where nodes represent the cells and edges correspond to adjacent cells. Motion planning is performed by searching a path on this connectivity graph.

There exist two good reasons which support the cell standardization: one is the simplification of the decomposition algorithm and the other is the possibility to obtain directly from sensor environment models with this kind of regular decomposition (probability maps, accumulation maps).

These approximate methods are easier to implement than exact methods and have been widely used.

5.3 Reaction simulation methods

The previous methods intend to capture the global connectivity of the robot's free space into a graph in order to search a path on this graph. In the reaction simulation methods, the most suitable motion address is determined by simulating the response of the robot to the compound influence of the obstacles located in the environment, the current position of the robot and the goal destination. The path generation is obtained by giving an increment along this direction and by repeating the process. The influence of the environment on the robot can be modeled in different ways like: potential fields or other behaviour responses.

In the potential field approach,[45,46] the robot is modeled as a particle acting under the influence of a potential function U whose local variations show the structure of the free space, encoding information about the position of the obstacles in the surrounding environment and the goal location. Typically, these potential functions have two components: an *attractive potential* guiding the robot towards the goal configuration and a repulsive potential steering the robot away from obstacles. Path planning is done iteratively. In each iteration, the force induced by the global potential function, $\vec{F}(\mathbf{q}) = -\vec{\nabla}(\mathbf{q})$, at this point is considered as the most suitable motion address and the path generation is obtained by giving an increment along this address and by repeating the process.

Other behaviour response approaches generate a linear and angular speed at each point (or some kind of force) as a combination of the response outputs of different behaviours to the obstacles, the robot position and the goal destination (the behaviours responses can be defined by means of potential fields, PID algorithms, heuristic rules, among others). The path is obtained by simulating the advance of the robot to obtain a new position and by repeating the process. The most important characteristic of these approaches is the efficiency of the algorithm which facilitates their real time operation.

The second group of methods tries to obtain a behaviour plan that can lead the robot from the initial position to the final one. Among them we find the behaviour planning methods.

5.4 Behaviour planning methods

These methods do not define a movement path as a sequence of configurations. Instead, the plan is determined as a sequence of behaviours. Behaviours are related with sensorial information, such as following a wall, following a road, moving to the door; or related with geometric positions, like moving to x, y (Bouilly[47]). Behaviours are activated and deactivated by using events. Typically, events are related to time, perception or position of the robot. Different researchers have worked with this approach.[48,50] In most cases, plans are based on plan units (e.g. RAPs[14] or ACTS[49]). Plan units are manually pre-programmed and each one serves to obtain a specific goal.

5.5 Open problems and new research areas

Perception planning. To overcome the uncertainty in robot position, some new approaches tend to determine motion plans which include the relocalization requirements of the robot together with the path plan.[51] Another important question seldom treated deals with determining what motion plan has to be followed in case the robot gets lost (or the uncertainty becomes too big).

Behaviour planning. Further research must be devoted to develop planners that work directly with a library of skills (motor and perception skills), instead of working with pre-

https://doi.org/10.1017/S0263574799002210 Published online by Cambridge University Press

programmed plan units. The library of skills should be dynamic and must allow the incorporation of new learned skills.

Planning for world modeling. Important too, and still open is the problem of determining a plan for the robot in order to build a model of the environment which permits a safe motion planning and a good localization of the robot into the operations area.

6. CONCLUSIONS

Even if the mobile robotics has experimented a substantial number of theoretical and practical advances during the last decades, the *Maximum Navigation Test (MNT)* remains unsolved. New research lines are very promising in order to resolve the different open problems which are not satisfactorily solved with current techniques.

Hybrid architectures appear to be the most robust and flexible solution to provide a suitable degree of deliberation and reactivity in a mobile robot.

During the last few years a great number of advances have been made to develop reactive behaviour-based systems for motion control. In spite of that, big efforts should be carried out to develop perception skills and also to generate automatically skills through learning.

Topological localization and world modeling are effective solutions to cope with uncertainties in position and information about the environment but a substantial effort to improve perception capabilities to recognize environment objects is necessary.

Planning activities tend to be less dependent from an *a priori* knowledge of the environment, to be more capable of dealing with inaccuracies in sensor and world models, and also to integrate an increasing number of perception activities into the planning process.

References

1. R. Chatila, "Deliberation and Reactivity in Autonomous Mobile-Robots", *Robots and Autonomous Systems* **16**, 197–211 (1995).
2. R. Brooks, "Elephants Don't Play Chess", *Robotics and Autonomous Systems* **6**, 3–15 (1990).
3. R. Brooks, "A Robust Layered Control System for a Mobile Robot", *IEEE Journal on Robotics and Automation* **RA-2**(1), 14–23 (March, 1986).
4. E. Gat, *Three-Layer Architectures, Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*, (AAAI Press/The MIT Press, Menlo-Park, CA, 1998) pp. 195–210.
5. E. Gat, "Integrated Planning and Reacting in a Heterogeneous Asynchronous Architecture for Controlling Real-World Mobile Robots", *Proceedings of the AAAI-92* (AAAI-Press/MIT Press, Menlo-Park, CA (1992) pp. 809–815.
6. R.P. Bonasso et al., "Experiences with an Architecture for Intelligent, Reactive Agents", *J. Experimental and Theoretical Artificial Intelligence* **9**, Nos. 2–3, 237–256 (1997).
7. L. Moreno, M.A. Salichs, D. Gachet and J.R. Pimentel, *Neural Networks for Mobile Robot Piloting Control, Applications of Artificial Neural Networks in Robotic Control* (Ellis Horwood, 1995) pp. 137–161.
8. P. Maes and R.A. Brooks, "Learning to Coordinate Behaviors", *Proceedings of AAAI-90* (AAAI-Press/MIT Press, Menlo-Park, CA, pp. 796–802.
9. A.R. Damasio, *Descartes' Error: Emotion, Reason and the Human Brain* (G.P. Putnam's Sons, New York, USA, 1994).
10. R.C. Arkin, *Behaviour-Based Robotics* (The MIT Press, Cambridge, MA, 1998).
11. R.C. Arkin, "Motor Schema-based Mobile Robot Navigation", *Int. J. Robotics Research* **8**, No. 4, 92–111 (August, 1989).
12. M.A. Salichs, E.A. Puente, D. Gachet and J.R. Pimentel, "Learning Behavioural Control by Reinforcement for an Autonomous Mobile Robot", *IEEE International Conference on Industrial Electronics, Control and Instrumentation*, Hawaii, USA (1993) pp. 1436–1441.
13. R.J. Firby, N. Prokopowicz and M.J. Swain, *The Animate Agent Architecture, Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems* (AAAI Press/The MIT press, Menlo-Park, CA, 1998) pp. 243–275.
14. R.J. Firby, "The RAP Language Manual", *Animate Agent Project Working Note AAP-6 Version 1*, (University of Chicago. Ill., 1995).
15. M. Georgeff and A. Lansky, "Reactive Reasoning and Planning", *Proceedings of the AAAI-87* (AAAI-Press/MIT Press, Menlo-Park, CA., 1987) pp. 677–682.
16. L. Kaelbling, "REX: A Symbolic Language for the Design and Parallel Implementation of Embedded Systems", *Proceedings of the AIAA Conference on Computer in Aerospace* VI (1987) Vol. 6, pp. 255–260.
17. A. Pascoal et al., "MARIUS: An Autonomous Underwater Vehicle for Coastal Oceanography", *IEEE Robotics and Automation Magazine* 46–59 (Dec., 1997).
18. J.F. Canny and M.C. Lin, "An Oportunistic Global Path Planner", *Algorithmica* **10**, 102–120 (1993).
19. E. Rimon and J.F. Canny, "Construction of C-space Roadmaps Using Local Sensory Data — What Should the Sensors Look For?", *IEEE Int. Conf. on Robotics and Automation*, San Diego, USA (1994) pp. 117–124.
20. H.P. Moravec and A. Elfes, "High Resolution Maps from Wide Angle Sonar", *IEEE Int. Conf. on Robotics and Automation*, St Louis, USA (1985) pp. 116–121.
21. J. Borenstein and Y. Koren, "Histogramic In-Motion Mapping for Mobile Robot Obstacle Avoidance", *IEEE Transactions on Robotics and Automation* **7**, No. 4, 535–539 (August, 1991).
22. M. Hebert, C. Caillas, E. Krotkov, I.S. Kweon and T. Kanade, "Terrain Mapping for a Roving Planetary Explorer", *IEEE Int. Conf. on Robotics and Automation*, Scottsdale, USA (1989) pp. 997–1002.
23. I.S. Kweon and T. Kanade, "High-Resolution Terrain Map from Multiple Sensor Data", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**, No. 2, 278–292 (February, 1992).
24. J.J. Leonard, H.F. Durrant-Whyte and I.J. Cox, "Dynamic Map Building for an Autonomous Mobile Robot", *IEEE Int. Workshop on Intelligent Robots and Systems (IROS'90)*, Tsuchiura, Japan (1990) pp. 89–96.
25. H.F. Durrant-Whyte, "Sensor Models and Multisensor Integration", *Int. J. Robotics Research* **7**, No. 6, 97–113 (Dec., 1998).
26. J.L. Crowley, "World Modelling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging", *IEEE International Conference on Robotics and Automation*, Scottsdale, USA (1989) pp. 674–681.
27. R. Chatila, "Position referencing and consistent world modeling for Mobile Robots", *IEEE Robotics and Automation*, St. Louis, USA (1985) pp. 138–145.
28. N. Ayache, *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception* (The MIT Press, Cambridge, MA, 1991).
29. Y.T. Byun and B. Kuipers, "A Qualitative Approach to Robot Exploration and Map-learning", *Procceedings of the Workshop on Spatial Reasoning and Multi-sensor Fusion*, Morgan Kauffman, Los Altos, CA, USA (1987) pp. 390–404.
30. B. Kuipers and Y.T. Byun, "A Robot, Exploration and Mapping Strategy based on Semantic Hierarchy of Spatial Representations", *J. Robotics and Autonomous Systems* **8**, Nos. 1–2, 47–63 (November, 1991).

31. R. Brooks, "Visual Map Making for a Mobile Robot", *IEEE Robotics and Automation*, St Louis, USA (1985) pp. 824–829.
32. M. Mataric, "Environment Learning Using a Distributed Representation", *IEEE International Conference on Robotics and Automation*, Cincinnati, USA (1990) **Vol. 1**, pp. 402–406.
33. J.J. Leonard, H.F. Durrant-Whyte, "Mobile Robot Localization by Tracking Geometric Beacons", *IEEE Trans. on Robotics and Automation*, **7**(3), 376–382 (1991).
34. A. Kosaka and A. Kak, "Fast Visio-Guided Mobile Robot Navigation using Model-Based Reasoning and Prediction of Uncertainties", *CVGIP: Image Understanding* **56**, No. 3, pp. 271–329 (November, 1992).
35. I.J. Cox, "Blanche, An Experiment in Guidance and Navigation of an Autonomous Mobile Robot", *IEEE Trans. on Robotics and Automation* **7**(3), 193–204 (1991).
36. W.B. Thompson et al., "Vision Based Localization" *Proc. Image Understanding Workshop 93* (1993) pp. 491–498.
37. S. Koenig and R.G. Simmons, "Passive distance for robot navigation", *Proceedings of the 13th Int. Conference on Machine Learning*, Morgan Kauffman Publishers, San Francisco, CA, (Morgan Kauffman Publ., USA, 1996) pp. 266–274.
38. L. Kaelbling, M.L. Littman and A.R. Cassandra, "Planning and Acting in Partially Observable Stochastic Domains", *Artificial Intelligence* **101**, No. 1–2, 99–134 (May, 1998).
39. S. Thrun, "Finding Landmarks for Mobile Robot Navigation", *Proc. of the IEEE Int. COnf. on Robotics and Automation*, Leuven, Belgium (May, 1998) pp. 958–963.
40. N.J. Nilsson, "A Mobile Automation: An Application of Artificial Intelligence Techniques", *Proceedings of the 1st Int. Joint Conference on Artificial Intelligence*, Washington D.C. (1969) pp. 509–520.
41. T. Lozano-Pérez, "Automatic Planning of Manipulator Transfer Movements", *IEEE Trans. on Systems, Man and Cybernetics* **SMC-11**(10), 681–698 (1981).
42. H. Choset, "Toward Exact Localization without Explicit Localization with the Generalized Voronoi Graph", *IEEE Robotics and Automation*, Leuven, Belgium (1998) pp. 342–348.
43. C. O'Dúnlaip and C.K. Yap, "A Retraction Method for Planning the Motion of a Disc", *J. Algorithms* **6**, 104–111 (1982).
44. R. Brooks and T. Lozano-Perez, "A Subdivision Algorithm in Configuration Space for Finding Path with Rotation", *Proceedings of the 8th International Conference on Artificial Intelligence*, Karlsruhe, FRG (1983) pp. 799–806.
45. O. Khatib, "Commande Dynamique dans l'Espace Opérationel des Robots Manipulateurs en Présence d'Obstacles", *Thesis Dissertation* (Ecole National Supérieure de l'Aéronautique et de l'Espace, Toulouse, 1980).
46. J.C. Latombe, *Robot Motion Planning* (Kluwer Academic Publishers, Norwell, Massachusetts, USA, 1991).
47. B. Bouilly, T. Simeon and R. Alami, "A Numerical Technique for Planning Motion Strategies of a Mobile Robot in Presence of Uncertainty", *IEEE Int. Conf. on Robotics and Automation*, Nagoya, Japan (1995) pp. 1327–1332.
48. C. Elsaesser and M.G. Slack, "Integrating Deliberative Planning in a Robot Architecture", *Proc. of the AIAA/NASA Conference on Intelligent Robots in Field, Factory, Services and Space (CIRFFSS'94)* (1994) **Vol. 2**, pp. 782–787.
49. D.E. Wilkins and K.L. Myers, "A Common Knowledge Representation for Plan Generation and Reactive Execution", *Logic and Computation* **5**, No. 6, 731–761 (December, 1995).
50. D.E. Wilkins, K.L. Myers, J.D. Lawrence and L.P. Wesley, "Planning and Reacting in Uncertain and Dynamic Environments", *J. Experimental and Theoretical AI* **7**, No. 1, 197–227 (1995).
51. J.M. Armingol, L. Moreno, A. Escalera and M.A. Salichs, "Landmark Perception Planning for Mobile Robot Localization", *IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium (May, 1998) **Vol. 4**, pp. 3425–3430.