Path Planning Knowledge Modeling for a Generic Autonomous Robot: A Case Study

Rafael Guirado, Clara Marcela Miranda, and José Fernando Bienvenido

Departamento de Lenguajes y Computación, University of Almeria,
Ctra. Sacramento, S/N, E-04120, Almería, Spain
{rguirado,cmiranda,fbienven}@ual.es

Abstract. This paper presents the initial steps followed in order to build an ontology about robot navigation (including specifically alternative navigation algorithms). Tackling the problem from general to specific, we start analyzing the desired behavior for generic mobile robots, in order to get common tasks and methods. Then, we fix our attention into the agricultural spraying robot developed in the University of Almeria by a multidisciplinary team. Because the field of robot navigation is consolidated, there are many algorithms (methods) to perform same activities (tasks). Our goal is to build an ontology including all this alternative methods, applying the dynamic selection of methods to make decisions in real-time depending on the environment conditions. Here we show the task-method diagrams with the parameterized description of some of alternative methods, using Fitorobot as testing case.

Keywords: Knowledge management, Ontology, Dynamic selection of methods, Path planning, Spraying robots.

1 Introduction

For a long time, research centers and universities have been looking how to facilitate human work (releasing humans of the most dangerous tasks), applying robotic techniques. In this sense, and given that most of the tasks require human displacements, it is required to design robots that incorporate navigation algorithms. One of the main income sources of the Province of Almeria is agriculture. In this context, a multidisciplinary research group of the University of Almeria is working in the design of a mobile robot (called *Fitorobot*) that permits movement between lines of crop, and the performance of several greenhouse tasks such as spraying, pruning, and crop transport (see Fig. 1).

Initially, in order to control the robot navigation we considered two different approaches: a map-based deliberative technique and a pseudo-reactive technique. These approaches could be integrated as a two levels decision tree. At the first level it is fixed the task to be performed in order to reach the final objective (when it is required this decision). In the second level, it is selected the best method in order to achieve the objectives of the task (different methods suppose alternative ways to reach the task goal). The general idea is to assemble a navigation model of the mobile robot as

J.D. Velásquez et al. (Eds.): KES 2009, Part II, LNAI 5712, pp. 74–81, 2009.

Fig. 1. Lateral view and back view of the mobile robot *Fitorobot* into a greenhouse

general as possible, deciding between the different navigation alternatives on the fly. In order to do this, we must build a general ontology of navigation algorithms (and their components), that including all the tasks and methods, would constitute a battery of parameterized navigation elements as general and complete as possible.

2 System Description

Our testing robot, *Fitorobot,* has a differential-drive mechanism of locomotion. The system is composed of two rubber-tracks, which provide a larger contact surface with the soft ground of the greenhouses, making it more robust and stable. This robot has a mass of 756 kg (with the spray tank full), and it has appropriate dimensions for the typical corridors of greenhouses in south-eastern Spain. It is driven by a 20-hp gasoline engine. It also has a low-cost sensor system, including low-distance ultrasonic sensors, middle-distance ultrasonic sensors, magnetic compasses, incremental encoders, radars, and security sensors (see Fig. 2a). Furthermore, for spraying control, a pressure sensor has been installed.

About the navigation techniques, it starts evaluating if it is accessible a map of the greenhouse; in this case it applies a deliberative method. On the other hand, when there is no map, a pseudo-reactive algorithm is used. Moreover, a sensorial map is built along the path, to be employed by the deliberative method in later runs. The two previous approaches utilize a security layer to avoid collisions; this layer uses on/off sensors. Finally, it has a low-level (servo) control layer composed of two PID controllers that regulate the speed of the tracks. Fig. 2b shows the navigation strategy schema for the mobile robot proposed in [1] by R. González et al.

Greenhouses are structured environments where the distribution of plants is at least partially known. The main obstacle to the movement of mobile robotics in greenhouses is related to the fact that navigation algorithms should take into account unexpected events (humans working in the greenhouse). Furthermore, appropriate filters for the sensor readings, and robust navigation strategies should be examined.

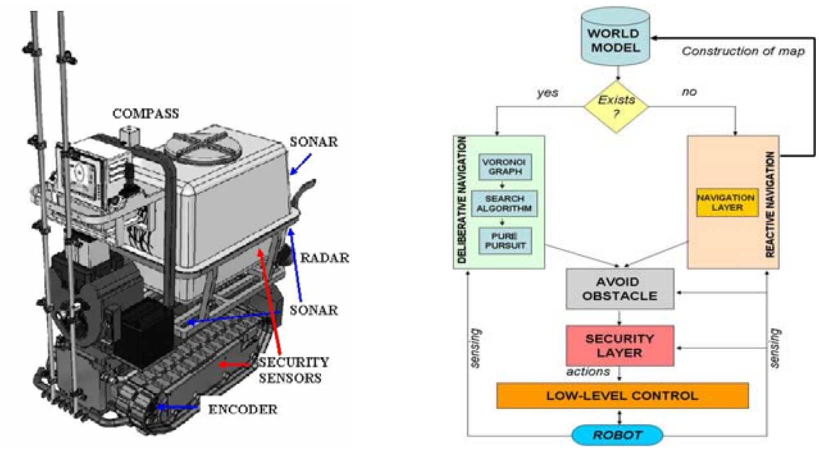

Fig. 2. *Fitorobot*: a) Sensorial system and b) Navigation strategy implementation (after the integration of the different methods, including lower level security)

3 Methodology

The knowledge model, about the navigation of mobile robots described in this works, was assembled using some elements of the CommonKADS methodology and the dynamic selection of methods (DSM). Now, we are going to introduce those techniques and a short resume of the navigation algorithms included in the system. This knowledge model, despite it has been developed for Fitorobot, is general.

3.1 Path-Planning Algorithms (Methods)

From the highest point of view, planning the movements of a mobile robot can be done using two main approaches: global (deliberative) techniques and local (reactive) techniques. First group requires the knowledge of the robot environment, second

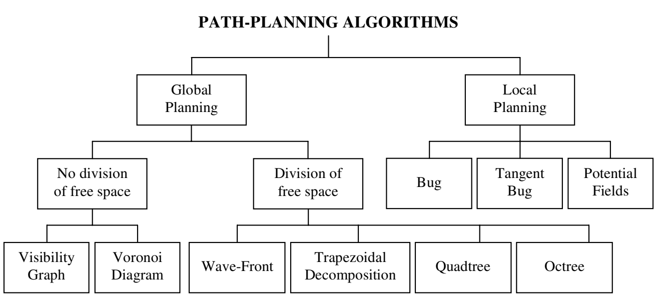

Fig. 3. Common algorithms for path planning (as described in [2])

group do not. In our work case, as the robot is moving into a given environment, a greenhouse, it would be possible to apply the global approach; but, some alterations (as a box in the path) force the application of a combination of map based algorithms with reactive ones.

Fig. 3 shows most usual movement planning algorithms, classified following the previously given dichotomy: global and local planning algorithms. [2] offers a short description of these algorithms with a tool that let us to observe their behavior in different situations in a fully interactive way. This way it has been possible to evaluate the suitability of the different methods for the specific situations to be used in the process of dynamic selection of planning methods.

3.2 CommonKADS Methodology

The CommonKADS methodology was consolidated as a knowledge engineering technique to develop knowledge based systems (KBS) at the beginning of the 90's [3]. It includes a kernel set of models, which are summarized in Fig. 4a. In our case, we have worked in the knowledge model and specifically in the assembling of tasks, methods, inferences and domain knowledge elements, as presented in Fig. 4b.

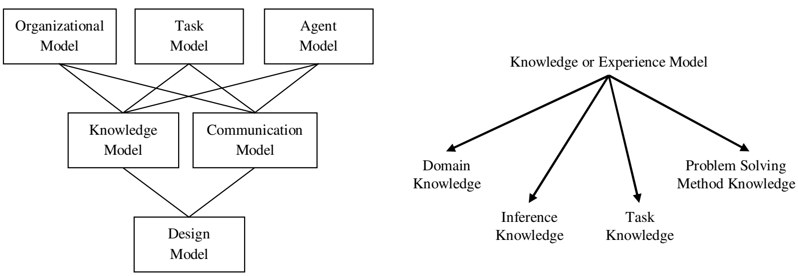

Fig. 4. CommonKADS models: a) Set of models, b) Knowledge model components

The most representative tools to model the problem solving mechanisms are the Task-Method Diagrams (TMD), that present the relation between one task to be performed and the methods that are suitable to achieve this task, followed by the decomposition of these methods in subtasks, transfer functions and inferences (final implemented algorithms), as shown in a general way in Fig. 5. The main problem to be solved is represented by the highest level task [4].

3.3 Dynamic Selection of Methods

A given task can be achieved by more than one method, and these can be applied only in specific environmental conditions. This way, it would be required that the robot selects one of the suitable methods on the fly (using data received from its sensors).

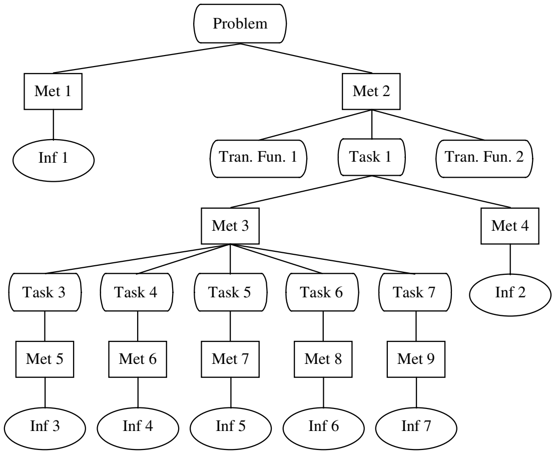

Fig. 5. Simple TMD

Table 1. Suitability criteria of 4 alternative methods (for path-planning)

Algorithm	Minimum distance	Computing time	Obstacle proximity
Wave-Front	1	1	3
Voronoi Diagram	2	2	1
Cell Decomposition (Quadtree)	2	2	4
Irregular Cell Decomposition	1	3	4

We proposed to assemble a general decision module that, taking account of the suitability criteria defined for each alternative method and actual data, would activate the most adequate method. These suitability criteria are assigned weights whose values can be update both manually and automatically [5]. Table 1 shows the main suitability criteria for the methods that compute the most adequate path for the spraying task in a partially known greenhouse. The cost function considers the three criteria, using a higher weight for the third one (it is the most relevant); when there are near obstacles are preferable last two methods, if we have enough computing time. This technique was used previously in configurational design of greenhouses and aeronautic conformation pieces as described in [5].

4 Modeling the Navigation System

Modeling the knowledge, we try to explicit all the knowledge implicit in the texts written by field experts (in our case, the navigation of mobile robots). Assembling these models required of an intensive bibliographical evaluation.

We propose a generic knowledge model about the navigation of robots, based in the CommonKADS methodology (mainly some modeling tools) and the dynamic selection of methods, to be used by the navigation subsystems of mobile robots. From a general point of view, within this area, we can find a huge casuistic, given different environment characteristics (as the different obstacles) and the kinematic, dynamic and sensorial characteristics of the robot. As an extreme example, we could find different types of obstacles less o more "dangerous" for the robot (e.g. a less stable objects could easily fall on the robot when contacting it, causing different damages). For this reason, the knowledge model must include a very detailed specification of each one of the elements involved in the navigation tasks, in order to offer the robot the most appropriate alternative for each particular situation.

This model of knowledge starts from a global problem, that it is navigating in its specific environment in order to achieve a specific objective (this is therefore the main task of our model). Achieving this general objective, under the specific conditions of the system (in this level the availability of a map of the environment, that means that it is known or not), can be done activating two initial alternatives: global or local methods. First group supposes calculating an efficient path using a specific criterion (minimum distance, maximum distance to obstacles,...), rising new tasks (second order tasks). The second group of methods may require the application of an exploration task (recognition of the environment to generate a map) or the application of a task that starts the navigation of the robot detecting and avoiding obstacles on the road. If it is explored the environment, generating a map, the global method previously described would be applicable. As the description of this model is huge, Fig. 6 shows the higher level elements of the associated TMD.

The "Traverse-along-a-Path-Segment" task repeats iteratively the two subtasks that compose it, the "Planning-of-Path-Segment" and "Tracing-Calculated-Path-Segment" tasks, strictly in this order until reaching the goal position or finding that it is unattainable. This iterative execution of a task is represented by an * on the top right of the task symbol.

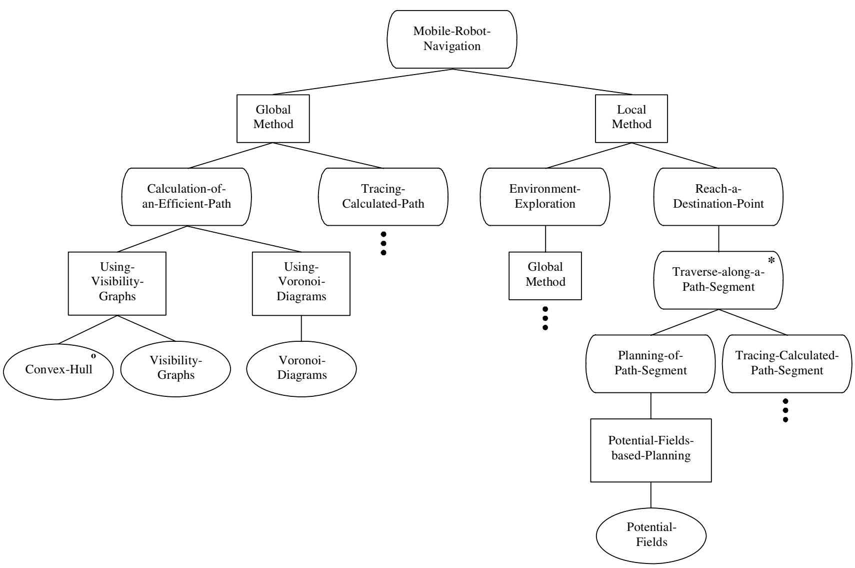

Fig. 6. Partial representation of a TMD for mobile robot navigation

The Convex-Hull inference evaluates the convex boundary for an obstacle or set of obstacles. It is used when the system detects an obstacle or set of obstacles near the robot with any concave vertex in its boundary. As shown in the CML description of this inference, this algorithm is not executed when the robot is located or must access an area inside a concave region of the obstacle (in order to avoid the inclusion of the trajectory inside the virtual object.

Dynamic selection of methods is applied, for example, in the task of calculation of the efficient path. In this case, we show only two alternatives (from the wide set of alternatives found) for the achievement of the task goal; these alternatives have been assigned a set of suitability criteria and associated weights. These suitability criteria and weights let the system to decide which method must be activated.

As proposed in CommonKADS, the different elements (tasks, methods and inferences) of the previously defined TMD are modelled using CML schemas. These schemas formalize all the knowledge associated to each one of these elements. Next, we show some relevant parts of an inference used in the diagram. This is used to model one process (algorithm) that let the robot to reach some partial goal.

A simplified description of the Convex-Hull inference would be:

```
INFERENCE Convex-Hull;
 OPERATION-TYPE: calculate;
 ROLES:
  INPUT:
   visibility-graph:
    "formal description of a graph, that represents all the
    vertices of the obstacles and the initial and final
    configurations; these vertices are matched according to
    their visibility, ie whether the line that connects them
    do not intersect with any other obstacle";
  OUTPUT:
   convex-visibility-graph:
    "formal description of the visibility graph, representing
    the obstacles as convex polygons; if the initial or final
    configuration is included into a concave area of an
    obstacle, this inference is not applicable";
 SPECIFICATION:
    "1. Look for the vertices with higher and lower values of
    x and y (4 vertices).
    2. Order all the vertices by increasing value of x.
    3. Select the vertex with the lower value of x (x_min). Put
    it in a stack and look for the extremes between vertices
    x_min and y_min.
       a. Compute the slope of all the lines that start from
       the vertex which is at the top of the stack and go to
       all the vertices that are at the left of y_min and right
       and below of that located on top of stack.
       b. Select the vertex associated to the line with the
       more negative slope, and put it in the stack.
    4. Repeat step 3 with the new vertex on the stack until
    the vertex corresponding to y_min would be put on the stack.
    5. Repeat steps 3 and 4 drawing the lines between the y_min
    and x_max vertices, looking for the vertices on the right of
    y_min and on the left and down of x_max with minimum positive
    slope.
```

```
     6. Repeat steps 3 and 4 drawing the lines between the x_max
     and y_max vertices, looking for the vertices over x_max and
     right and down of y_max one, with maximum negative slope.
     7. Repeat steps 3 and 4 drawing the lines between the y_max
     and x_min vertices, looking for vertices down of y_max and on
     the right and over the x_min one, with minimum positive
     slope"
END INFERENCE Convex-Hull;
```

This way, the CML schemas of all the elements (task, methods, inferences and transfer functions), that configure the TMD for robot navigation, are assembled.

5 Conclusions and Future Works

Main objective of this work was to present a dynamic mechanism to order the different/alternative algorithms for robot navigation. Main advantages were to facilitate further addition of new algorithm that could be developed in the future, and the capacity of deciding on the fly the most adequate to be used in specific conditions (in a general way). We present a general planning system that would decide the most adequate algorithm to be used, selecting one method from our repository of well parameterized methods.

In order to evaluate the proposed mechanisms of dynamic selection of methods in robotics, actually we are preparing the use of these techniques in the field of social (or sociable) robots, where there are much more application opportunities with wider and more complex alternatives.

References

1. González, R., Rodríguez, F., Sánchez-Hermosilla, J., Donaire, J.G.: Navigation Techniques for Mobile Robots in Greenhouses. Applied Engineering in Agriculture 25(2), 153–165 (2009)
2. Guzmán, J.L., López, O., Berenguel, M., Rodríguez, F., Dormido, S., Piguet, Y.: MRIT: Mobile Robotics Interactive Tool. In: Internet Based Control Education IBCE 2004, Grenoble (2004)
3. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W., Wielinga, B.: Knowledge Engineering and Management. The CommonKADS Methodology. MIT Press, Cambridge (1999)
4. Bienvenido, J.F.: Selección dinámica de métodos y reutilización de elementos de conocimiento: Una extensión a CommonKADS aplicada a problemas de diseño. PhD, University of Murcia (1999)
5. Bienvenido, J.F., Flores-Parra, I., Guirado, R., Marín, R.: Knowledge Based Modeling of the Design Process as a Base of Design Tools. In: Moreno-Díaz Jr., R., Buchberger, B., Freire, J.-L. (eds.) EUROCAST 2001. LNCS, vol. 2178, pp. 209–222. Springer, Heidelberg (2001)