A Reactive Type-2 Fuzzy Logic Control Architecture for Mobile Robot Navigation

Mouloud Ider
Electrical Engineering Department, LTII Laboratory,
A/Mira University, Targa Ouzemour Street,
06000, Bejaia, Algeria
mouloudider@yahoo.fr

Boubekeur Mendil
Electrical Engineering Department, LTII Laboratory,
A/Mira University, Targa Ouzemour Street,
06000, Bejaia, Algeria
bmendil@yahoo.fr

Abstract—This paper presents a type-2 fuzzy reactive architecture for mobile robot navigation in cluttered environments. The proposed control scheme allows to the robot to avoid obstacles and to reach the target. Robot control actions are generated by different behaviors: attraction to the target, obstacle avoidance and fusion block.

Since the robot evolve in unstructured and unknown environments and they need to cope with large amounts of uncertainties, fuzzy logic, especially type-2 fuzzy logic, seems to be the most convenient solution to design different parts of robot control system. Simulation results show the effectiveness and the robustness of the proposed architecture.

Keywords—Behavioral control architecture; mobile robot navigation; type-2 fuzzy logic; uncertainty and modeling

I. INTRODUCTION

Nowadays, mobile robots are used in many industrial situations such as transport, security or cleaning tasks. However, their design requires many engineering and science disciplines, from mechanical, electrical and electronics engineering to artificial vision, computer and cognitive.

The control of mobile robot navigation in cluttered environments is a fundamental problem that has been receiving a large amount of attention. The main issue in this field is how to obtain accurate, flexible and reliable navigation? One part of the literature in this domain considers that the robot is fully actuated with no control bound and focuses the attention on path planning. Voronoi diagrams and visibility graphs [1] or navigation functions [2] are among these roadmap-based methods. However, the other part of the literature considers that to control a robot with safety, flexibility and reliability, it is essential to accurately take into account: robot’s structural constraints (e.g., nonholonomy); avoid command discontinuities and set-point jerk, etc. Nevertheless, even in this method, there are two schools of thought, one uses the notion of planning and re-planning to reach the target, e.g., [3] and [4] and the other more reactive (without planning) like in [5], [6] or [7].

Our proposed control architecture is linked to this last approach. Therefore, where the stability of robot control is rigorously demonstrated and the overall robot behavior is constructed with modular and bottom-up approach [8]. To guarantee multi-objective criteria, control architectures can be elaborated in a modular and bottom-up way as introduced in [9] and so-called behavioral architectures [8]. These techniques are based on the concept that a robot can achieve a complex global task while using only the coordination of several elementary behaviors. In fact, to tackle this complexity, behavioral control architecture decompose the global controller into a set of elementary behavior/controller (e.g., attraction to the objective, obstacle avoidance, trajectory following, etc.) to master better the overall robot behavior. In this kind of control, it exists two major principles for behavior coordination: action selection and fusion of actions which lead respectively to competitive and cooperative architectures of control. In competitive architectures (action selection), the set-points sent to the robot actuators at each sample time are given by a unique behavior which has been selected among a set of active behaviors. The principle of competition can be defined by a set of fixed priorities like in the subsumption architecture [9] where a hierarchy is defined between the behaviors. The action selection can also be dynamic without any hierarchy between behaviors [10], [11]. In cooperative architectures (fusion of actions), the set-points sent to the robot actuators are the result of a compromise or a fusion between controls generated by different active behaviors. These mechanisms include fuzzy control [12] via the process of defuzzification, or the multi-objective techniques to merge the controls [13].

Our work deals with the problem of robots evolving in unstructured and unknown environments that need to cope with large amounts of uncertainties. Many stochastic techniques, such as Kalman filtering [14], are used to get the best estimate of the measured variables, especially, in the presence of sensor noise. However, in this work, we use the powerful of fuzzy logic to design the different parts of the robot navigation system.

Fuzzy logic is an adequate methodology for designing robust controllers that are able to deliver satisfactory
performance in the presence of disturbances and uncertainties. The general framework of fuzzy reasoning allows handling much of the uncertainty, using fuzzy sets characterized by type-1 membership functions. However, in many situations, the designer has no information about the adequate membership function shapes. Thus, the use of type-2 fuzzy sets becomes natural [15], [16].

Type-2 fuzzy logic is a more general formulation using fuzzy membership functions with additional dimension [17]. This provides additional degree of freedom to handle uncertainties and the lack of information.

In this work, fuzzy logic is used to design the controllers for the basic tasks (behaviors) for robot navigation: attraction to the target and obstacles avoidance. However, the coordination and the fusion of the elementary behaviors are more difficult, since we have no valuable online information to safely avoid obstacles and to guarantee the optimal convergence to the target. Hence, type-2 fuzzy logic is used to design the fusion controller, to compensate our lack of information. This is the core stone of our design. The results are compared to those of type-1 fuzzy logic controller.

The remainder of the paper is organized as follows. In the next section (II), the proposed control architecture is given. Section III details the design of the different control parts (attraction to the target, obstacle avoidance and fusion controller). Section IV discusses simulations results. Finally, section V concludes this paper.

II. BEHAVIORAL CONTROL ARCHITECTURE

A. NAVIGATION IN PRESENCE OF OBSTACLES

The objective of the navigation, in an unstructured environment, is to lead the robot towards the specified target, while avoiding static and dynamic obstacles. It is assumed that the obstacles and the robot are circular with respectively $R_O$ and $R_R$ radii [18], [19]. The target to reach is also characterized by a circle of $R_T$ radius. Several perceptions are also necessary for safe robot navigation (cf. Figure 1).

- $D_{RO}$ distance between the robot and the obstacle “$j$”.
- $D_{RT}$ distance between the robot and the target.

For each detected obstacle, we define a circle of influence (cf. Figure 1) with a radius of

$$R_I = R_R + R_O + \text{Margin}$$

Where: Margin corresponds to a safety tolerance which includes perception uncertainty, control reliability and so on. Hence, we can now consider robot as a point in the configuration space with obstacles of augmented radii $R_I$.

THE PROPOSED CONTROL ARCHITECTURE

The proposed control architecture (cf. Figure 2) is dedicated for mobile robot navigation in presence of obstacles. It permits to manage the interactions between different elementary controllers. The robot can therefore have very smooth trajectories while guaranteeing safe obstacle avoidance. The specific blocks (controllers): attraction to the target, obstacle avoidance and a fusion block composing this control architecture are detailed below.

III. FUZZY CONTROL DESIGN

This section details the design of the different control parts. Fuzzy logic is used to design the controllers for the basic tasks (behaviors) for mobile robot navigation: attraction to the target and obstacles avoidance. However, type-2 fuzzy logic is used to design the fusion controller and its performance is compared to that of type-1 fuzzy controller.

A. ATTRACTION TO THE TARGET CONTROLLER

Consider a robot with $(x, y, \theta)$ its configuration in the absolute frame (cf. Figure 3). This robot has to reach the target $T(x_T, y_T)$

The kinematic model of the unicycle mobile robot is given by

$$\begin{align*}
\dot{x} &= v \cos(\theta) \\
\dot{y} &= v \sin(\theta) \\
\dot{\theta} &= \omega
\end{align*}$$

Where $\theta$, $v$ and $\omega$ are respectively the robot orientation, linear and angular velocities.

From figure (3), we define the position errors as


\[
\begin{align*}
e'_x = (x_T - x) &= D_{RT} \cos(\theta_{RT}) \\
e'_y = (y_T - y) &= D_{RT} \sin(\theta_{RT})
\end{align*}
\] (2)

Where \( D_{RT} \) corresponds to the current distance between the robot and the target \( T \), which is expressed by

\[
D_{RT} = \sqrt{e'_x^2 + e'_y^2}
\] (3)

Similarly, the current angle of the robot according to the target, noted \( \theta_{RT} \), is computed as

\[
\theta_{RT} = a \tan 2(e'_y, e'_x)
\] (4)

The angle error is given by

\[
\theta_e = (\theta_{RT} - \theta)
\] (5)

The control objective is to design a fuzzy logic controller to drive the robot to the desired configuration (i.e., \( D_{RT}(t) = 0 \), \( \theta_e(t) = 0 \)).

Basically the attraction to the target FLC has:

- Two input variables: the distance and the angle errors, between the robot and the target \( T \), given by (3) and (5),
- Two output variables: linear and angular velocities denoted by \( v_o \) and \( \omega_o \), respectively (cf. Figure 2).

Triangle and trapezoidal membership functions (MFs) are used for both input and output variables. Their shapes are given in Figure 4. The values of the input, \( D_{RT} \), and the output, \( v_o \), are indicated by the linguistic symbols Z, M, and G which correspond respectively to the linguistic values: Zero, Middle, and Great. The values of the second input, \( \theta_e \), and the output, \( \omega_o \), are indicated by the symbols (NB, N, Z, P, PB) which correspond respectively to: Negative Big, Negative, Zero, Positive, Positive Big.

Table 1 illustrates the decision table of the FLC. The rules are of the form:

Rule, \( r \): \( \text{IF } D_{RT} \text{ is } G_i \text{ and } \theta_e \text{ is } G_j \text{ THEN } v_o \text{ is } G_3 \text{ and } \omega_o \text{ is } G_4 \)

Where \( r = 1 \ldots 15 \), and \( (G_1, G_2) \) are the antecedents, and \( (G_3, G_4) \) are the consequences. The center of gravity defuzzification method is used [19].
C. Fusion controller

We introduce type-2 FLC which can handle rule and numerical uncertainties [15], [20]. Its implementation involves the operations of fuzzification, inference and output processing. Output processing is performed in two stages: type reduction and defuzzification. Type-reduction schemes correspond to extended versions of type-1 defuzzification methods. Type reduction captures more information about rule uncertainties than does the defuzzified value (a crisp number), but it is computationally intensive [15], [16], [20]. A type-2 FLC is again characterized by IF-THEN rules, but its antecedents or consequents are now type-2 fuzzy sets. The uncertainty effect that comes from the instrumentation elements (amplifier, sensors perception, digital to analog and analog to digital converters, etc.) is simulated by adding noise to the measured FLC input vector.

The control signals of the robots \( (v_r, \omega_r) \) are obtained by linearly combining (fusion) the outputs of the obstacle avoidance controller and that of attraction to the target controller

\[
\begin{align*}
v_r &= v_o + v_a \cdot (1 - \beta) \\
\omega_r &= \omega_a + \omega_a \cdot (1 - \beta)
\end{align*}
\]

(6)

Where \( \beta \) corresponds to an adaptive weighting gain \( \epsilon [0 1] \) generated by the fusion controller. The rules set used for both type-1 and type-2 fusion FLCs are given in Table 3. Like the obstacle avoidance controller, fusion FLCs have 2 input variables corresponding to the distance and the angle between the robot and the obstacle, and one output variable (the gain, \( g \)).

<table>
<thead>
<tr>
<th>( D_{RO_j} ) ( \theta_{RO_j} )</th>
<th>N</th>
<th>Z</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>M</td>
<td>G</td>
<td>M</td>
</tr>
<tr>
<td>G</td>
<td>P</td>
<td>P</td>
<td>P</td>
</tr>
</tbody>
</table>

Table III. Fuzzy rule set of the Type-1 and Type-2 fusion block based FLC

IV. SIMULATION RESULTS

To test the relevance of the proposed control architecture (cf. Figure 2) which must permit to robot to avoid obstacles and to reach the target, the proposed control architecture is embedded on robot.

A. Attraction to the target with obstacles avoidance

From the results (cf. Figure 8), it can be seen that the robot join the closest target. However, hindering obstacles prevent the robot from directly reaching their targets. It is observed that the robot avoids the obstacles and reach the closest target. A large number of simulations in different cluttered and unstructured environments was done. All simulations permit to the robot to reach the target in finite time. These simulations prove also the gain in time given when type-2 FLC is used instead of the type-1 one. The trajectories given by the proposed type-2 are also smoother than those with type-1 FLC (cf. Figure 9).

The distance and the angle errors between the robot and the target are given in figure 10 (a) and (b) respectively. It can be seen that the robot reach the target (the distance and the angle error tend to 0).

Figure 6. Membership function shapes of the behavior fusion type-1 FLC

Figure 7. Membership function shapes of the behavior fusion type-2 FLC

Figure 8. Trajectory of the robot reaching a target after avoiding obstacles
Where the $CT$ denotes the Collision Time which can be defined as the dwell time outside of the configuration space (i.e., in the influence region of the obstacles), due to the presence of different uncertainties. Mathematically, $CT$ is a variable which is incremented, each sampling time, when at least one robot is in the undesired region, i.e., $D_{RO} < R_{ij}$ (cf. Figure 1). In this test, we ran simulations for different values of the standard deviation (std) of the noise ranging from std=0 to std=0.5.

For each std value, the simulation is done using Gaussian and Triangular/Trapezoidal MFs. The evolution of the collision time ($CT$) of the robots is given in figure 12. It can be seen that better collision time are obtained using type-2 FLC.

**B. Avoiding trajectory oscillations**

Figure 11 shows the efficiency of the proposed type-2 FLC to avoid the trajectory oscillations when the robot skirts the obstacle. The adaptive weighting gain $g \in [0, 1]$ generated by the fusion controller permits to the robot to do not oscillate between the position where $D_{RO} \leq R_{ij}$ and $D_{RO} \geq R_{ij}$.

**C. Performance test using collision time**

For evaluating the performance of the type-2 and type-1 FLCs, we define the objective function

$$\text{fitness} = \min (CT)$$

(7)
It is clearly observed in these two simulations that the robot succeeded to converge and follow accurately the dynamical target while avoiding efficiently the cumbersome obstacles.

V. CONCLUSIONS

In this work, fuzzy reactive control architecture is proposed and applied to the navigation of mobile robot in unstructured environments. Fuzzy logic is used to design the controllers for the basic tasks (behaviors): attraction to the target and obstacles avoidance. The coordination and the fusion of the elementary behaviors is done using type-2 fuzzy logic controller. The aim was to handle more uncertainties and to compensate our lack of knowledge, since we have no valuable online information to safely avoid obstacles and guarantee the optimal convergence to the target.

Combining behaviors using a type-2 FLC provides better performances in comparison to the type-1 FLC. The type-2 FLC makes a robot overall control architecture more robust in the presence of the uncertainty and the resulting collision time is more optimized (smooth trajectory). The efficiency and robustness of this architecture have been checked by a series of evaluations in a simulated environment.

Dynamical obstacles avoidance was not studied in this paper. It will be the subject of our future work. Another significant aspect is the possibility of creating a type-2 FLC, for which the shape and the position of member functions are continually adjusted, while the control architecture is in operation. This can increase its potential to handle very uncertain and dynamic environments.

REFERENCES