Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 561451, 10 pages
http://dx.doi.org/10.1155/2013/561451

Research Article

Behavior-Based Fuzzy Control for Mobile Robot Navigation

Hongwei Mo,[1] Qirong Tang,[2] and Longlong Meng[1]

[1] *Automation College, Harbin Engineering University, No. 145 Nantong Street, Nangang District, Harbin 150001, China*
[2] *University of Stuttgart, Pfaffenwaldring 9, 70569 Stuttgart, Germany*

Correspondence should be addressed to Hongwei Mo; honwei2004@126.com

Received 10 July 2013; Accepted 13 August 2013

Academic Editor: William Guo

A new behavior-based fuzzy control method for mobile robot navigation is presented. It is based on behavioral architecture which can deal with uncertainties in unknown environments and has the ability to accommodate different behaviors. Basic behaviors are controlled by specific fuzzy logic controllers, respectively. The proposed approach qualifies for driving a robot to reach a target while avoiding obstacles in the environment. Simulation and experiments are performed to verify the correctness and feasibility of the proposed method.

1. Introduction

One way to accomplish robot navigation is using behavior arbitration and behavior control [1, 2]. Since the behavior control architecture was proposed in 1986, it has been adopted to solve the robot navigation problem frequently. The behavior control is a special form of decentralized switching control in which each behavior is autonomous and can drive the robot on its own without dependencies on other behaviors. Under the standard behavior paradigm, each behavior triggers its own specific control command according to the behavior. The behavior control architecture can handle the navigation problem in an online manner and it does not require environment model; thus, it has been widely used for mobile robot navigation [3–5]. Fuzzy logic control (FLC) [6] is the most important method for robot behavior-based control and has been investigated for mobile robot navigation and obstacle avoidance by many researchers, see [7–14].

In [9], Selekwa et al. presented the design of a preference-based fuzzy behavior system for navigating robotic vehicles using the multivalued logic framework. The proposed method allows the robot to smoothly and effectively navigate through cluttered environments. In [11], a new fuzzy logic algorithm is developed for mobile robot navigation in local environments. A robot perceives its environment through sensors, while the fuzzy logic algorithm performs the main tasks of obstacle avoidance and target seeking. In [13], Wang et al. proposed behavior-based hierarchical fuzzy control method for mobile robot navigation in dynamic environment.

Fuzzy logic controllers can be used for driving mobile robot to perform specific motions with good robustness. However, a fuzzy mobile robot controller should be self-adaptive since it must deal with very different control situations such as following a wall or a corridor, or avoiding an obstacle. Rules must be designed so that they can handle these different situations. A way of solving this problem is to create a fundamental or basic controller which can be incrementally updated and optimized by tuning both labels and rules adequately. Another way of solving this problem is to decompose the complex behavior into several subbehaviors or motions which are controlled by separate schema. These control schemas can be merged by combining the corresponding actions. Within each control schema, it is possible to define the most appropriate semantics for the linguistic variables which are manipulated in the rules, and thus each behavior can be tuned independently to be more effective in its own context. Complex behaviors can be obtained based upon the composition of simpler behaviors.

This paper will address the mobile robot navigation issue by combining the ideas of behavior-based control and fuzzy logic control. The rest of this paper is organized as follows. Section 2 presents the framework of behavior-based robot navigation. Section 3 gives the robot kinematic model. Section 4 presents the detailed structure of the proposed control method which contains three elementary behaviors. Simulation results that demonstrate the performance of the proposed approach are given in Section 5. Experiments are carried out in Section 6 while concluding remarks are given in Section 7 to close this paper.

2. Framework of Behavior-Based Robot Navigation

This section proposes a framework for a behavior-based navigation strategy for a mobile robot in an unknown environment. The general structure of a fuzzy behavior system consists of some independent fuzzy behaviors and a component of command fusion, see Figure 1. The framework totally includes four components, which are preprocessing, goal determination, behavior arbitration, and command fusion. In each robot control cycle, the robot reasoning system provides a set of motor control commands via an inference process **I**. This inference process can be considered as a relationship between the input **U** and the output **Y**. The input **U** is represented by a multidimensional vector corresponding to a particular set of input data, for example, distance to obstacle, direction to the goal. Similarly, the output **Y** corresponds to motor speed and steering angle, and so on. Thus, the relation between **Y** and **U** can be expressed by

$$\mathbf{I} : \mathbf{U}(U_1, U_2, \dots, U_n) \longrightarrow \mathbf{Y}(Y_1, Y_2, \dots, Y_m). \tag{1}$$

2.1. Preprocessing. Input data from the robot sensors is performed with a simple preprocessing to reduce noises. Specifically, the sonar and infrared sensor measurements are preprocessed. Meanwhile, in the preprocessing module, if the input **U** is too large, the computational complexity also will be reduced. The dimensions of input **U** can be reduced by introducing a limited number of intermediate variables. These variables classify different perceptual situations which are relevant to the robot's current behavior and status. Some intermediate variables are statements such as front-obstacle distance, right-obstacle distance, or left-obstacle distance. These variables are also used for afterwards behavior design.

2.2. Goal Determination. In this investigation, we classify the goal determination to two types which are a determined goal and a undetermined goal. When the goal is determined, its exact coordinates are given. The robot must move to a given target while autonomously avoids the blocking obstacles on the robot path. When the goal is undetermined, it does not have the exact coordinates of the goal.

2.3. Behaviors and Fuzzy Logic Controllers. In general, in our case there are two types of robot behaviors, which are goal seeking (GS) and obstacle avoidance (OA). In order to realize the GS behavior, a controller is designed to generate motor commands to reach the goal as soon as possible. The OA behavior is actually sensor-based behavior, which implements a control strategy based on external sensing.

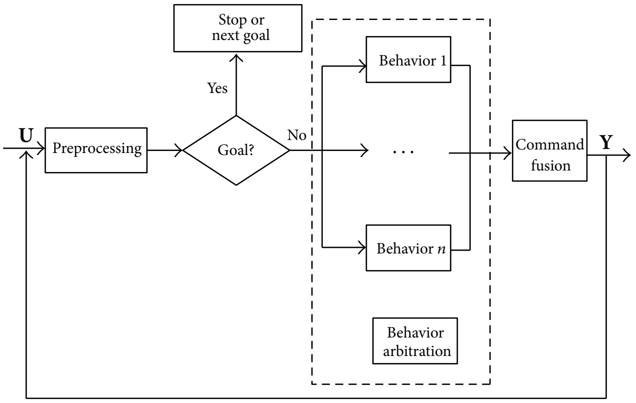


Figure 1: Framework of behavior-based robot navigation.

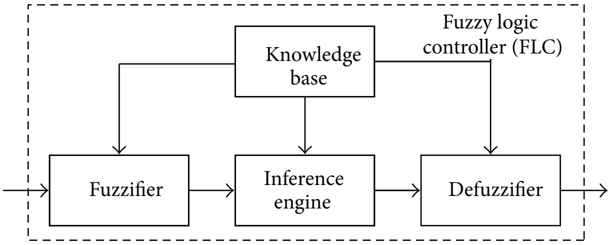


Figure 2: Architecture of a basic fuzzy logic controller.

As we designed, each behavior will be controlled by a specific fuzzy controller. In those controllers, reasoning is contained in the rules operating on linguistic inputs and outputs, such as

$$\begin{aligned} &\text{If } u_1 \text{ is } A_1 \text{ and } u_2 \text{ is } A_2 \text{ and} \cdots \text{and } u_n \text{ is } A_n, \\ &\text{Then } y_1 \text{ is } B_1 \text{ and } y_2 \text{ is } B_2 \text{ and} \cdots \text{and } y_m \text{ is } B_m, \end{aligned} \tag{2}$$

where u_i is the input linguistic variable taking linguistic value A_i. Each linguistic value is defined by a membership function $\mu_{A_i}(u_i)$. The y_i is the output linguistic variable taking linguistic value B_i. Each linguistic value of output is defined by a membership function $\mu_{B_i}(y_i)$.

Given two linguistic values A and B defined on the same universe of discourse, the AND and OR operations are defined by (3) and (4), respectively, which are

$$\mu_{A \cap B} = \min_{u \in U} (\mu_A(u), \mu_B(u)), \tag{3}$$

$$\mu_{A \cup B} = \max_{u \in U} (\mu_A(u), \mu_B(u)). \tag{4}$$

Figure 2 shows the principle of a basic fuzzy logic controller. In this investigation, we use a Takagi-Sugeno-Kang model as the fuzzy inference engine and the Centroid method for defuzzification.

The fuzzy inference engine is defined as in Box 1.

2.4. Behavior Arbitration (Coordination). Behavior arbitration can be taken as two conceptually different problems. The first one is the behaviors of OA and GS. In our case,

```
Input: $u_i$—crisp numerical value.
Output: $y_i$—crisp numerical value.
BEGIN:
Step 1. Fuzzification of the input $u_i$,
Step 2. Application of the fuzzy operators (AND or OR)
in the antecedent of the rules according to (3) and (4),
Step 3. Implication from the antecedent to the consequent
using of the AND operation according to (3),
Step 4. Aggregation of the consequents across
the rules using the OR operation according to (4),
Step 5. Defuzzification to output variable $y_i$.
END
```

Box 1

they are activated during the whole motion process. The second problem is combining the results from different behaviors into one command sent to the robot's actuators. These problems are expressed in Figure 3.

One strategy for behavior arbitration is context-dependent blending (CDB) in which fuzzy logic is applied so that a decision between behaviors can be made in a prevailing situation. Our behavior arbitration strategy is similar to the CDB method. It uses fuzzy context rules to express the behavior arbitration strategy. When the obstacle is close, both OA and GS behaviors are activated and each behavior is assigned a weighting factor. These factors are adjusted dynamically according to the fuzzy weight rules. The weighting factors determine the degree of influence of each behavior on the final motion command. The weight rules continuously update the behavior weighting factors during robot motions.

3. Kinematic Model of Mobile Robot

In this investigation, a differentially driven mobile robot is used, see its kinematic illustration in Figure 4.

The robot has two drive wheels mounted on the same axis. Kinematic equations of this two-wheeled mobile robot are

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos\theta_m & 0 \\ \sin\theta_m & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{5}$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2}r & \frac{1}{2}r \\ \frac{r}{D} & \frac{-r}{D} \end{bmatrix} \cdot \begin{bmatrix} V_R \\ V_L \end{bmatrix}, \tag{6}$$

where x and y are coordinates of the mass center of the mobile robot, θ_m is the angle that represents the current orientation of the robot, v and ω are linear and angular velocities of the robot, V_R and V_L are angular velocities of right and left wheels, r is the wheels radius, and D is the distance between the two wheels centers.

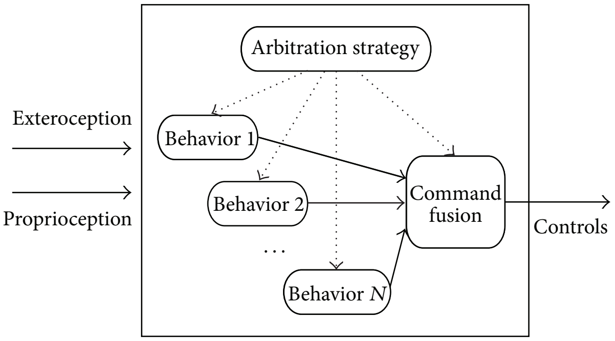


Figure 3: Behavior arbitration.

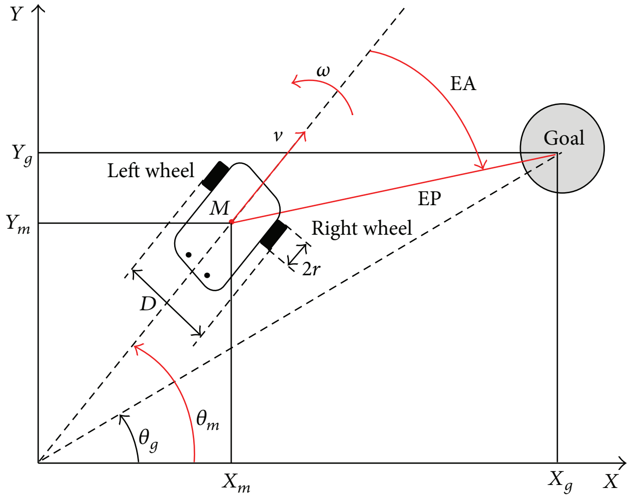


Figure 4: The kinematic illustration of the used mobile robot.

Combining (5) and (6) yields

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \frac{1}{2}r\cos\theta_m & \frac{1}{2}r\cos\theta_m \\ \frac{1}{2}r\sin\theta_m & \frac{1}{2}r\sin\theta_m \\ \frac{r}{D} & \frac{-r}{D} \end{bmatrix} \cdot \begin{bmatrix} V_R \\ V_L \end{bmatrix}. \tag{7}$$

Equation (7) is the kinematic model of the used robot for both simulation and experiments. Right wheel angular velocity V_R and left wheel angular velocity V_L are used as motion commands to the motors for realizing different behaviors.

4. Behavior-Based Fuzzy Control for Mobile Robot Navigation

We decompose the task of robot navigation into three elementary behaviors: goal seeking (GS), obstacle avoidance (OA), and behavior fusion (BF).

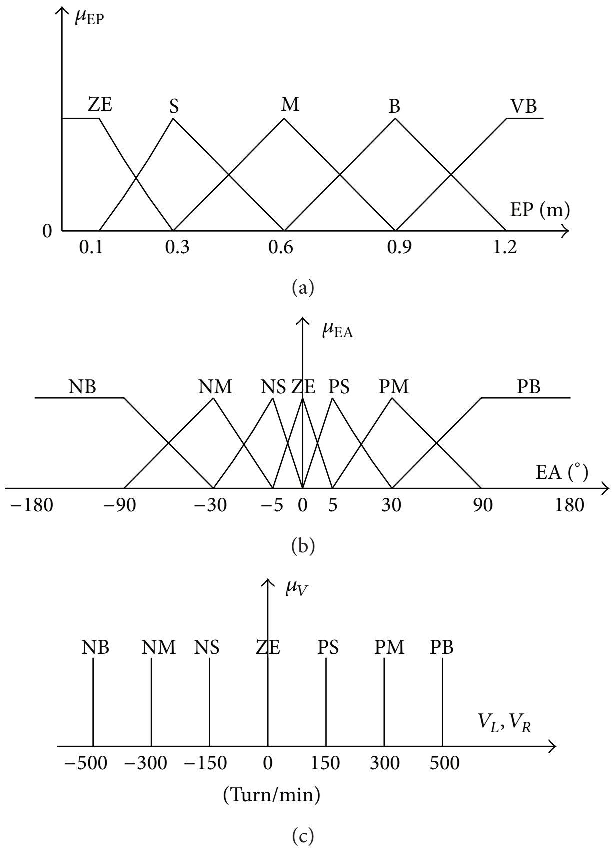
4.1. Global Goal-Seeking Behavior. The GS behavior discussed here is a kind of global behavior which does not rely on external sensed data but seeks for the global and exact goal. The inputs of the fuzzy logic controller include distance from the robot to the goal (EP) and the angular between robot orientation and goal orientation (EA). They are shown in Figure 4. The expressions of these two inputs are

$$\begin{aligned} \mathrm{EP} &= \sqrt{D_x^2 + D_y^2}, \\ \mathrm{EA} &= \theta_m - \theta, \\ D_x &= x_g - x, \\ D_y &= y_g - y, \\ \theta &= \arctan\left(D_y, D_x\right), \end{aligned} \tag{8}$$

where (x, y, θ_m) is the robot position and orientation and (x_g, y_g, θ_g) is the position and orientation of the goal.

The position deviation is represented by a five-variable linguistic fuzzy set {ZE, S, M, B, VB} with its entries for distance of zero, small, medium, big, and very big, respectively. Their corresponding membership functions are shown in Figure 5(a). Similarly, the angular error is represented by {NB, NM, NS, ZE, PS, PM, PB} with linguistic members to indicate negative big, negative medium, negative small, zero, positive small, positive medium, and positive big, respectively, see their membership functions in Figure 5(b). The positive and negative imply that the robot turns to the left and right, respectively.

The motion control variables of the mobile robot are the angular velocities of the right and left wheels. Similarly, the two velocities are represented by a seven linguistic variables' fuzzy set {NB, NM, NS, ZE, PS, PM, PB} with their membership functions shown in Figure 5(c).



(a)

(b)

(c)

Figure 5: Membership functions (a) for position deviation (b) for angular deviation (c) for angular velocities of right and left wheels.

The rule base of the GS behavior is summarized in Table 1. For instance, the (1, 1) entry in Table 1 can be written as

$$\text{If EP is ZE and EA is NB, then } V_L \text{ is NM and } V_R \text{ is PM.} \tag{9}$$

We can use this behavior-based fuzzy controller alone in an environment without obstacles. However, usually the environment contains obstacles.

Table 1: Rule base of the GS behavior.

	Angular error (EA)						
	NB	NM	NS	ZE	PS	PM	PB
Position error (EP)							
ZE							
V_L	NM	NS	ZE	ZE	ZE	PS	PM
V_R	PM	PS	ZE	ZE	ZE	NS	NM
S							
V_L	NB	NB	NM	PS	PS	PM	PM
V_R	PM	PM	PS	PS	NM	NB	NB
M							
V_L	NM	NM	NB	PM	PM	PS	PS
V_R	PS	PS	PM	PM	NB	NM	NM
B							
V_L	PS	PS	PM	PB	PB	PM	PB
V_R	PB	PM	PB	PB	PM	PS	PS
VB							
V_L	PS	PS	PM	PB	PB	PM	PM
V_R	PM	PM	PB	PB	PM	PS	PS

4.2. The Obstacle Avoidance Behavior. Obstacle avoidance is actually a sensor-based behavior which implements a control strategy based on external sensing. We reduce the dimensions of inputs by grouping the robot's sonar reading into three options which are left, front, and right. For example, our robot has 24 ultrasonic sonars that produce a set of obstacle distances by the following equations:

$$\begin{aligned} d_{\text{right}} &= \min\left(d_2, d_3, d_4, d_5\right), \\ d_{\text{front}} &= \min\left(d_0, d_1, d_{21}, d_{22}, d_{23}\right), \\ d_{\text{left}} &= \min\left(d_{17}, d_{18}, d_{19}, d_{20}\right). \end{aligned} \tag{10}$$

This is due to the layout of the ultrasonic sensors. The obstacle distance of each option is represented by three-member fuzzy set {VERYNEAR, NEAR, FAR} with their membership functions shown in Figure 6(a) while behavior weight member functions are shown in Figure 6(b).

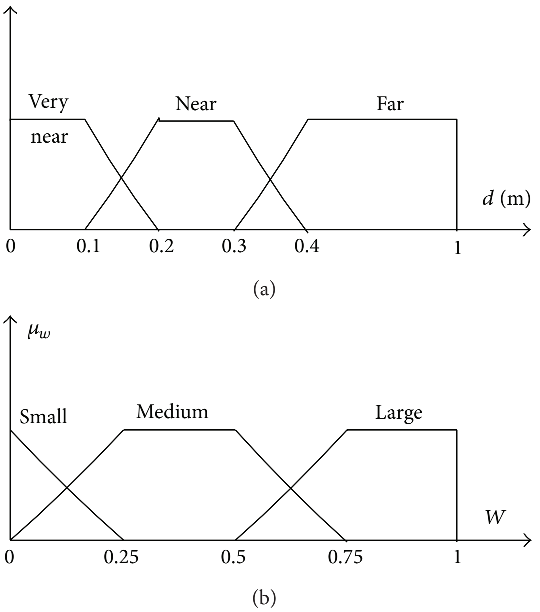


Figure 6: Membership functions (a) for distance and (b) for behavior weight.

The velocity variables V_R and V_L (here for obstacle avoidance) are represented by fuzzy set {NF, NM, PM, PF} which means negative fast, negative medium, positive medium, and positive fast. The velocity rules of left and right wheels for the OA behavior are summarized in Figures 7 and 8, respectively. The rules exhibit a behavior characteristic that, if the obstacle distance in any situation is very near, the robot should turn away to find a safer direction.

For instance, the two (3, 3) elements of the top layers in Figures 7 and 8 have the rules as

$$\begin{aligned}&\text{IF } d\_\text{front is very near AND } d\_\text{left is far AND}\\&d\_\text{right is far, THEN } V_L \text{ is NM;}\\&\text{IF } d\_\text{front is very near AND } d\_\text{left is far AND}\\&d\_\text{right is far, THEN } V_R \text{ is PM,}\end{aligned} \tag{11}$$

respectively.

When the three parts of the robot body corresponding to (10) have similar obstacle distances as shown in the (1, 1) element of the two top layers in Figures 7 and 8, the robot has to escape from its current embarrassed situation, so V_L is NF and V_R is PF which let the robot make a large left turn angle and decrease its speed since

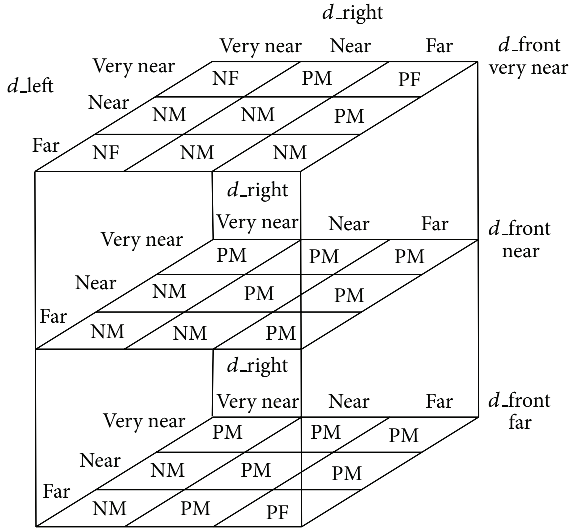
$$\omega = \frac{r}{D}\left(V_R - V_L\right), \qquad v = \frac{r}{2}\left(V_R + V_L\right). \tag{12}$$

4.3. Behavior Fusion. The behavior fusion is based on the weight assigned for each of the behaviors. The weight of OA behavior τ is represented by three-member linguistic fuzzy set {SMALL, MEDIUM, LARGE} with the membership functions shown in Figure 6(b) and rules in Figure 9. At first, we use a behavior arbitration module to calculate the defuzzified weight factors of all behaviors and then carry out command fusion by using these weight factors via

$$V_R = \frac{\sum w_i V_{R_i}}{\sum w_i}, \qquad V_L = \frac{\sum w_i V_{L_i}}{\sum w_i}, \tag{13}$$

where V_R and V_L are angular velocities of right and left wheels as motion commands, while V_{R_i} and V_{L_i} are angular velocities of right and left wheels preference values suggested by each specific behavior.

Here, w_i is the defuzzified weight factor. The implementation of the behavior fusion is depicted in Figure 10.



d_front very near

d_left \ d_right	Very near	Near	Far
Very near	NF	PM	PF
Near	NM	NM	PM
Far	NF	NM	NM

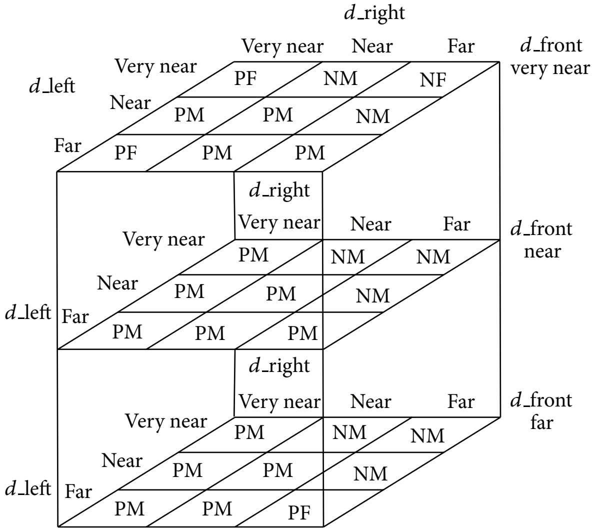
d_front near

d_left \ d_right	Very near	Near	Far
Very near	PM	PM	PM
Near	NM	PM	PM
Far	NM	NM	PM

d_front far

d_left \ d_right	Very near	Near	Far
Very near	PM	PM	PM
Near	NM	PM	PM
Far	NM	PM	PF

Figure 7: The V_L rule base for OA behavior.



d_front very near

d_left \ d_right	Very near	Near	Far
Very near	PF	NM	NF
Near	PM	PM	NM
Far	PF	PM	PM

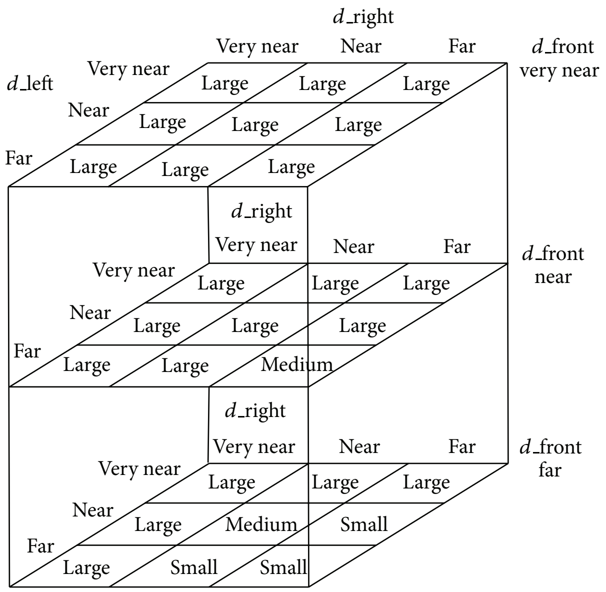
d_front near

d_left \ d_right	Very near	Near	Far
Very near	PM	NM	NM
Near	PM	PM	NM
Far	PM	PM	PM

d_front far

d_left \ d_right	Very near	Near	Far
Very near	PM	NM	NM
Near	PM	PM	NM
Far	PM	PM	PF

Figure 8: The V_R rule base for OA behavior.



d_front very near: d_left \ d_right	Very near	Near	Far
Very near	Large	Large	Large
Near	Large	Large	Large
Far	Large	Large	Large

d_front near: d_left \ d_right	Very near	Near	Far
Very near	Large	Large	Large
Near	Large	Large	Large
Far	Large	Large	Medium

d_front far: d_left \ d_right	Very near	Near	Far
Very near	Large	Large	Large
Near	Large	Medium	Small
Far	Large	Small	Small

Figure 9: Weight rules for OA behavior.

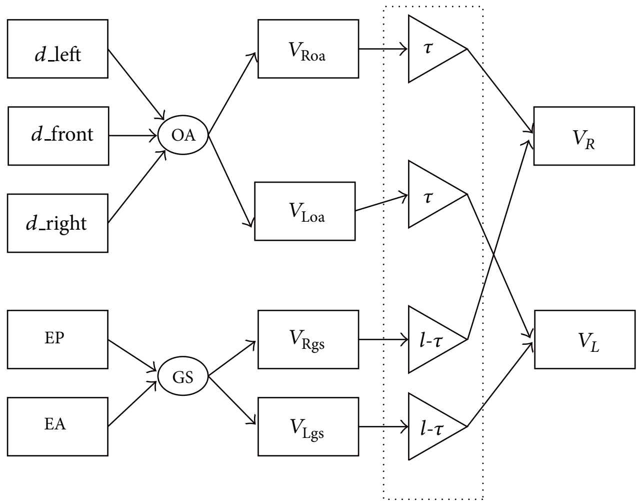


Figure 10: Diagram of behavior fusion.

As shown in the figure, the motion control variables (angular velocities of right and left wheels) are inferred by the GS and OA behaviors. And they are weighted by $1 - \tau$ and τ, respectively. The results can be given by

$$V_R = (1 - \tau)\, V_{\mathrm{Rgs}} + \tau\, V_{\mathrm{Roa}},$$
$$V_L = (1 - \tau)\, V_{\mathrm{Lgs}} + \tau\, V_{\mathrm{Loa}}. \tag{14}$$

For instance, the rule of (3, 3) element of the top layer in Figure 9 can be written as

IF d_front is very near AND d_left is far AND d_right is far, THEN τ is large. (15)

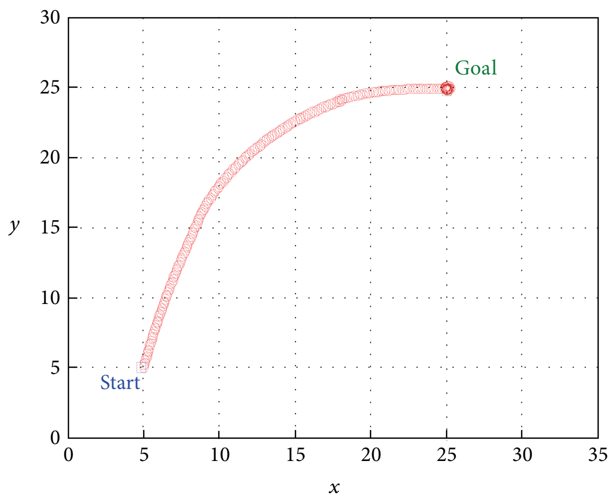


Figure 11: Robot's path in the environment without obstacles.

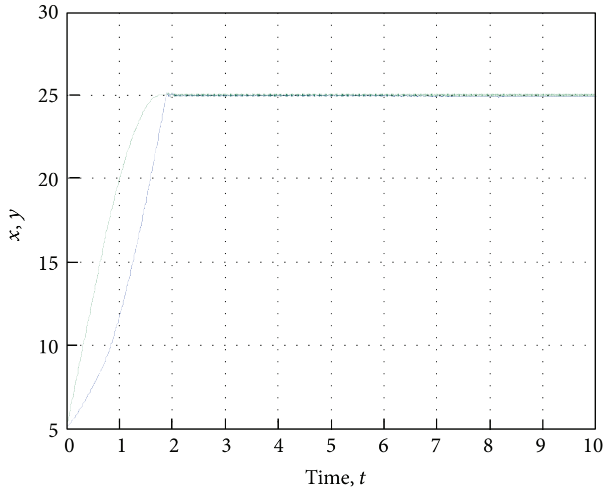


Figure 12: Robot's separated trajectories in the environment without obstacles (blue solid—x direction, green dashed—y direction).

5. Simulation

In order to verify the proposed method, simulation has been carried out within MATLAB. We designed the different fuzzy logic controllers based on different behaviors via fuzzy inference system (FIS).

5.1. Simulation in the Environment without Obstacles. In this case, the robot is supposed to move from the start point $(x_m, y_m) = (5, 5)$ to the goal $(x_g, y_g) = (25, 25)$. Note that the initial orientation of the robot is $\theta_m = \pi/2$ and there are no obstacles in the environment.

As illustrated in Figure 11, the robot reaches the goal. The trajectory is indicated by the chain of red circles. The robot's x and y directions separated trajectories are shown in Figure 12.

The weight of each behavior represents its extent of influence on the final motion commands. In this simulation, the proposed control system uses the GS behavior, because

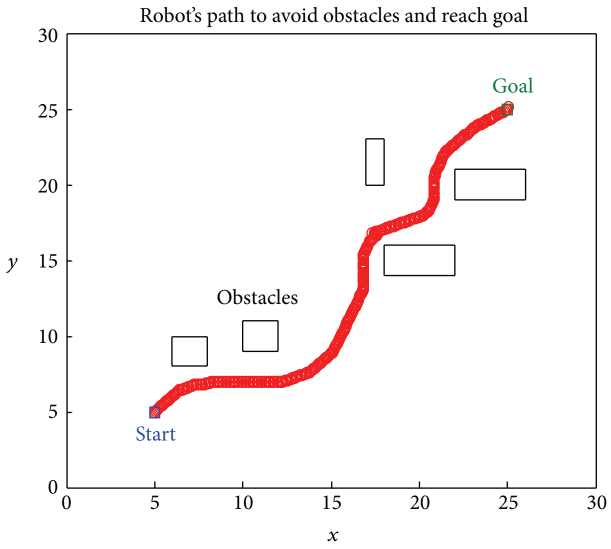


Figure 13: Robot's path in the environment with obstacles.

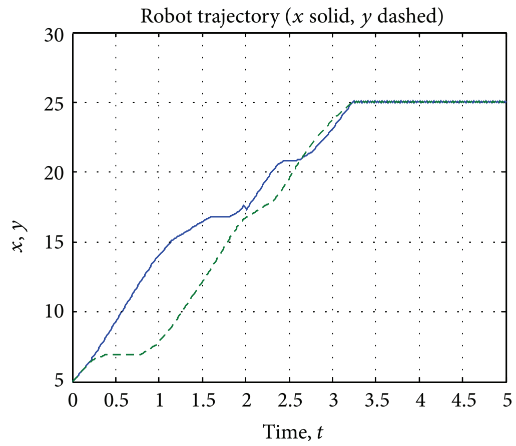


Figure 14: Robot's separated trajectories in the environment with obstacles (blue solid—x direction, green dashed—y direction).

there is no any obstacle so the GS behavior weight is 1 and OA weight is 0.

5.2. Simulation in the Environment with Obstacles. The robot now is moving in the environment with obstacles. The initial position is also $(x_m, y_m) = (5, 5)$ and the final position is $(25, 25)$.

When the robot is close to obstacles, it must decrease its speed and turn for safety. So, the weight of OA behavior is high when the robot is close to the obstacles. Otherwise, the weight is low when the obstacles are far away. The robot is set at a particular start point and the goal is defined in Cartesian coordinates with respect to the start position. The simulation results are shown in Figure 13. It demonstrates that the fuzzy behavior controller has a good performance. From Figure 14, we also can see that the robot reaches the goal after about 3.2 seconds.

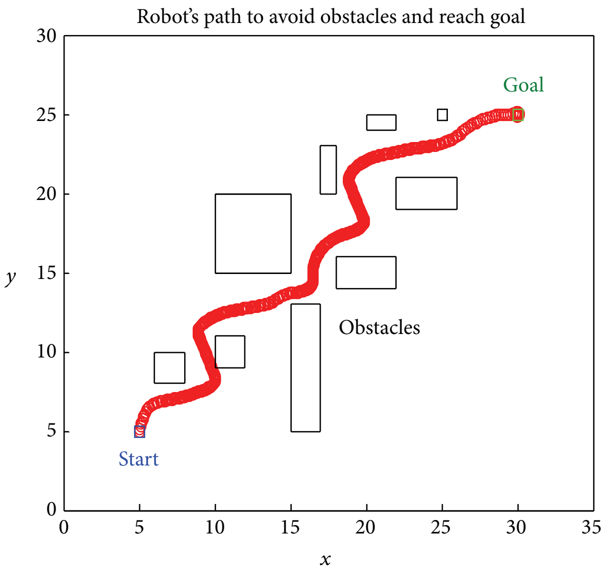


Figure 15: Robot's path in a clustered environment.

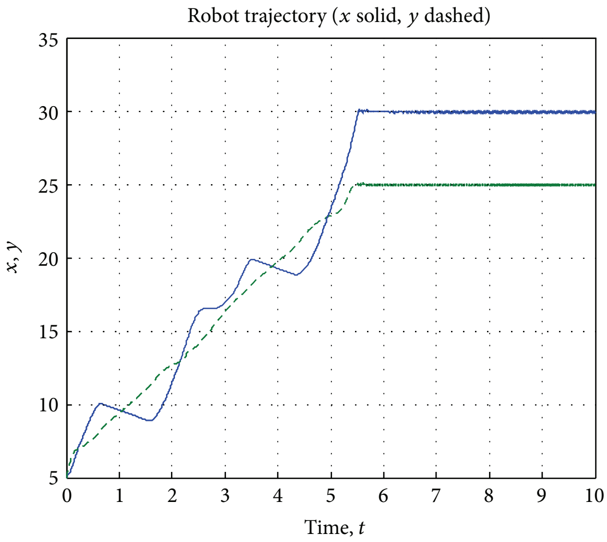


Figure 16: Robot's separated trajectories in a clustered environment (blue solid—x direction, green dashed—y direction).

5.3. Simulation in a Clustered Environment. Figure 15 shows the results in a more complex situation (a clustered environment). It illustrates the method's ability to navigate the robot in very small gaps and even turning away from the goal when it is necessary to avoid obstacles. Here, the initial position is set at $(x_m, y_m) = (5, 5)$ and the final position is set at $(x_g, y_g) = (30, 25)$.

The robot reaches the goal after about 5.5 seconds as shown in Figure 16. Since there are many obstacles, the robot takes much time to reach this goal compared to the simulations in Sections 5.1 and 5.2.

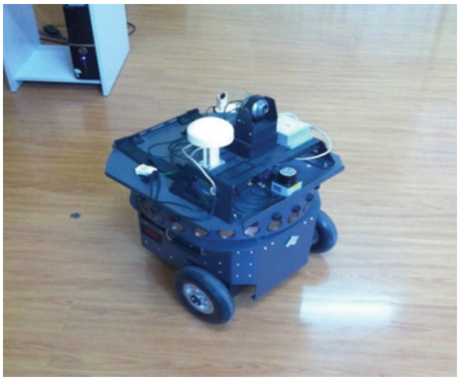


Figure 17: Voyager 2 robot in a laboratory environment: a representative illustration.

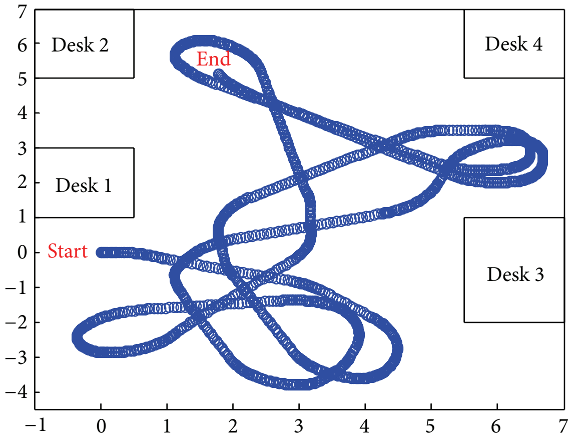


Figure 18: Robot roaming path in the laboratory environment.

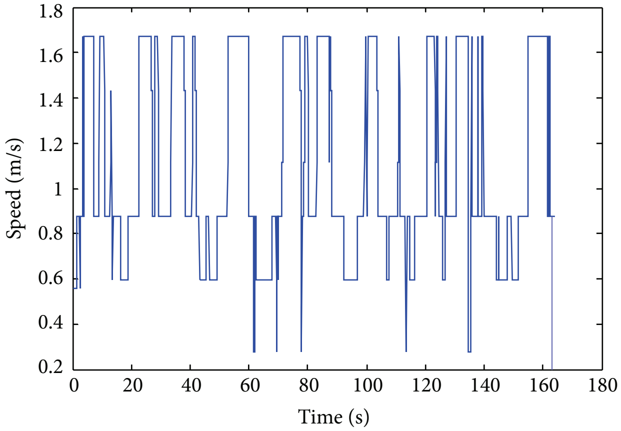


Figure 19: The speed of the robot during wandering.

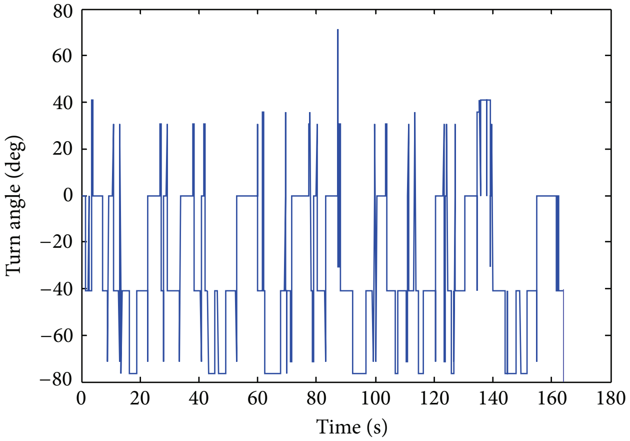


Figure 20: The turn angle during wandering.

6. Experiment

Our used real robot is a mobile robot endowed with a ring of 24 sonars, see the real robot in Figure 17. This robot platform is configured with two drive wheels and two swivel casters for balance. Each wheel is driven independently by a motor with 10 : 1 gear ratio which enables the robot to drive at a maximum speed of 3 m/s and climb a 15% grade [15]. The wheel diameter is 217 mm and the mobile robot base width is $D = 446$ mm.

6.1. Experiment for Robot Roaming. This experiment allows the robot to wander within a small area provided with static obstacles, for example, desks and walls, and dynamic obstacles, for example, moving humans. The weight of OA behavior is very high and the GS behavior is very low. The experiment is stopped after a given time or distance of wandering. Figure 18 shows the robot path in the laboratory environment. The trajectory is indicated by the chain of blue circles. The program draws the circle once every 0.02 second. A concentration of circles indicates that the robot is moving more slowly at that moment. Figures 19 and 20 show the variations of robot speed and turn angle during the wandering.

6.2. Experiment of Robot Avoiding Obstacles and Reaching a Goal. Now the robot's task is using the proposed navigation method to avoid obstacles and reach a goal. For this experiment, the robot is set at the start point $(x_m, y_m) = (0, 0)$ and the goal is defined in Cartesian coordinates with the position of $(x_g, y_g) = (3, 5)$. The obstacles configurations were mapped and the localization data of the robot are recorded and plotted in Figure 21. We can see clearly that the robot reaches the goal and avoids the obstacles.

Figures 22 and 23 show the variations of robot's speed and turn angle, respectively. The positive degree implies turning left while negative indicates right. The robot reaches the goal in almost 20 seconds.

The results presented in the previous sections show the performance of the proposed behavior-based control (navigation) method. Mobile robot usually has uncertain and incomplete information about the environment, so fuzzy rules provide an attractive means for mapping sensor data to appropriate control actions.

The main difference between our approach and the existing ones is that they have different reasoning and different motion commands, while in our approach the final motor

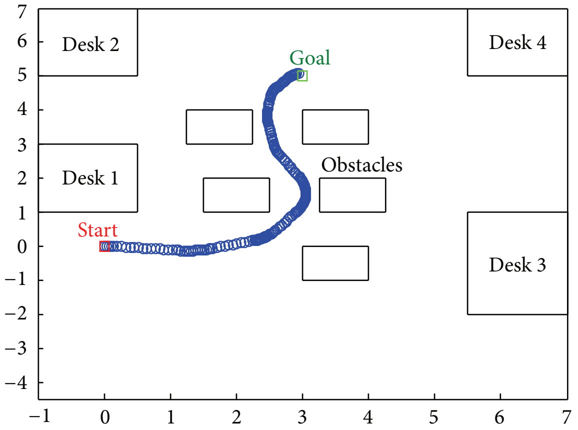


Figure 21: Robot avoids obstacles and reaches the goal.

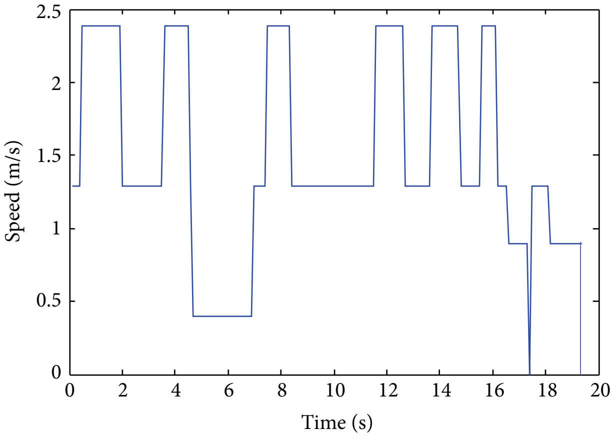


Figure 22: The speed of the robot during the task period.

command is generated by fusing different behavior-based fuzzy logic controllers into a uniform representation.

7. Conclusions

This paper presented a new behavior-based fuzzy control method for mobile robot navigation. This method takes angular velocities of driving wheels as outputs of different behaviors. Fuzzy logic is used to implement the specific behaviors. In order to reduce the number of input variables, we introduced a limited number of intermediate variables to guarantee the consistency and completeness of the fuzzy rule bases.

To verify the correctness and effectiveness of the proposed approach, simulation and experiments were performed. A Voyager II robot equipped with a ring of ultrasonic sensors was involved in the real experiments. The promising results demonstrated that our method is feasible and reasonable for navigating mobile robot. This method could be extended for any behavioral system by introducing more flexible behavior arbitration strategies.

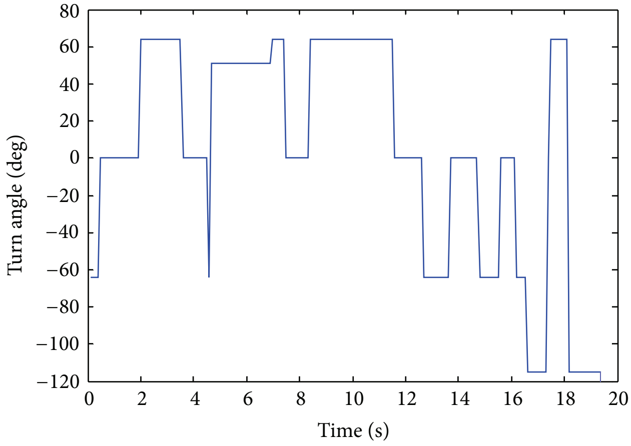


Figure 23: The turn angle during the task period.

Acknowledgments

This work is supported by National Natural Science Foundation of China under Grant no. 61075113, the Excellent Youth Foundation of Heilongjiang Province of China under Grant no. JC2012012, and the Fundamental Research Funds for the Central Universities, under Grant no. HEUCFZ1209.

References

[1] A. Saffiotti, “Fuzzy logic in autonomous robotics: Behavior coordination,” in *Proceedings of the 6th IEEE International Conference on Fuzzy Systems*, pp. 573–578, July 1997.

[2] E. Tunstel, Y. Lippincott, and M. Jamshidi, “Behaviour hierarchy for autonomous mobile robots: fuzzy-behaviour modulation and evolution,” *International Journal of Intelligent Automation and Soft Computing*, vol. 3, no. 1, pp. 37–49, 1997.

[3] K.-T. Song and J.-Y. Lin, “Behavior fusion of robot navigation using a fuzzy neural network,” in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 4910–4915, October 2006.

[4] B. Lei and W. Li, “A fuzzy behaviours fusion algorithm for mobile robot real-time path planning in unknown environment,” in *Proceedings of the IEEE International Conference on Integration Technology (ICIT ’07)*, pp. 173–178, March 2007.

[5] Q.-Y. Bao, S.-M. Li, W.-Y. Shang, and M.-J. An, “A fuzzy behavior-based architecture for mobile robot navigation in unknown environments,” in *Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence (AICI ’09)*, pp. 257–261, November 2009.

[6] L. A. Zadeh, “Fuzzy sets as a basis for a theory of possibility,” *Fuzzy Sets and Systems*, vol. 1, no. 1, pp. 3–28, 1978.

[7] F. Abdessemed, K. Benmahammed, and E. Monacelli, “A fuzzy-based reactive controller for a non-holonomic mobile robot,” *Robotics and Autonomous Systems*, vol. 47, no. 1, pp. 31–46, 2004.

[8] K. Samsudin, F. A. Ahmad, and S. Mashohor, “A highly interpretable fuzzy rule base using ordinal structure for obstacle avoidance of mobile robot,” *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1631–1637, 2011.

[9] M. F. Selekwa, D. D. Dunlap, D. Shi, and E. G. Collins Jr., “Robot navigation in very cluttered environments by preference-based

fuzzy behaviors," *Robotics and Autonomous Systems*, vol. 56, no. 3, pp. 231–246, 2008.

[10] M. Wang and J. N. K. Liu, "Fuzzy logic-based real-time robot navigation in unknown environment with dead ends," *Robotics and Autonomous Systems*, vol. 56, no. 7, pp. 625–643, 2008.

[11] O. R. E. Motlagh, T. S. Hong, and N. Ismail, "Development of a new minimum avoidance system for a behavior-based mobile robot," *Fuzzy Sets and Systems*, vol. 160, no. 13, pp. 1929–1946, 2009.

[12] M. Norouzi, A. Karambakhsh, M. Namazifar, and B. Savkovic, "Object based navigation of mobile robot with obstacle avoidance using fuzzy controller," in *Proceedings of the IEEE International Conference on Control and Automation (ICCA '09)*, pp. 169–174, December 2009.

[13] D. Wang, Y. Zhang, and W. Si, "Behavior-based hierarchical fuzzy control for mobile robot navigation in dynamic environment," in *Proceedings of the Chinese Control and Decision Conference (CCDC '11)*, pp. 2419–2424, May 2011.

[14] R. El Harabi, S. B. A. Naoui, and M. N. Abdelkrim, "Fuzzy control of a mobile robot with two trailers," in *Proceedings of the International Conference on Renewable Energies and Vehicular Technology*, pp. 256–262, 2012.

[15] Beijing UPTECH Robotics, "UP-Voyager IIA, Autonomous Mobile Robot. Experimental Guide Book," http://www.up-tech.com/eng/.