Reactive Control Using Behavior Modelling of a Mobile Robot

Imen Ayari, Abderrazak Chatti

Abstract: This paper deals with the reactive control of an autonomous robot which should move safely in a crowded unknown environment to reach a goal. A behavior based approach is used to realize obstacle avoidance within a neural model conceived from a set of examples of perception/action relations; supervised learning is used for the aim; while goal-reaching task is realized using a fuzzy rule-based system. A task activation module is used to generate the overall command, resulting from the fuzzy controller and the neural model. Real time simulation examples of generated path with proposed techniques are presented.

Keywords: reactive control, mobile robots, neural networks, learning,fuzzy control.

1 Introduction

One of the major challenges in the development of intelligent mobile robotic systems is endowing them with an ability to plan motions and to navigate autonomously in a crowded environment avoiding any type of obstacles. Different kinds of the path planning problem can be illustrated; the simplest one is to find a continuous path from a starting position to a goal position given the exact description of the environment. Different global approaches were applied for the purpose such as decomposition, road-map, and retraction methods.

When dealing with unknown environment, much fewer approaches are used; obstacles in that case are either detected locally during the robot movement and dynamically incorporated into the path generation process, or approximated in the workspace (Miura, Uozumi, and Shirai [9]; Yu and Su [10]; Bennewitz, Burgard and Thrun [11]). [12]
A way to deal with the navigation problem is within behavior based navigation approaches. The main idea is to subdivide the navigation task into small subtasks. Several behavior-based control schemes have been achieved inspired by the subsumption architecture of Brooks[13][14]such us reactive behaviors defined by Arkin using motor schemes (Arkin [15][16][17]); DAMN architecture presented by Rosenblatt et al.[18][19][12] in which a centralized module of votes provided by independent behaviors combines into a voted output and fuzzy logic approach to manage behaviors used by Saffiotti [13], Seraji et al. [14][15] and others [16][17].[18]

This framework provides an overview on the contribution of soft computing to the field of behavior based robot control, a hybrid approach is adopted using both neural and fuzzy logic methods to realize the specified task of reaching a goal with obstacles avoiding in a crowded unknown environment. A neural model is conceived to synthesize avoiding obstacle behavior from a set of examples representing the perception/action relation using a supervised learning strategy. Goal-reaching task is realized using a fuzzy rule-based system. A task activation module is used to generate the overall command, resulting from the fuzzy controller and the neural model. Simulation examples of generated path with proposed techniques are presented.

2 Neural networks in robotics

Historically, robots for industrial purposes involved little or no learning. Recently, a growing interest in unstructured environments has encouraged learning intensive design methodologies. The emerging class of robots must be able to interact responsively with people and other robots providing assistance and service that will increasingly affect everyday life. Robots may learn by adjusting parameters, building

environmental models such as maps, exploiting patterns, evolving rule sets, generating entire behaviors, devising new strategies, predicting environmental changes, recognizing the strategies of opponents or exchanging knowledge with other robots.

Essentially, neural networks deal with cognitive tasks such as learning, adaptation, generalization and optimization. Indeed, recognition, learning, decision-making and action constitute the principal navigation problems. To solve these problems fuzzy logic and neural networks are used. They improve the learning and adaptation capabilities related to variations in the environment where information is qualitative, inaccurate, uncertain or incomplete. Artificial neural networks (ANNs) are algorithms based very loosely on the neural phenomenon of spreading activation. ANNs can encode knowledge and skill implicitly as associative connections between nodes. Stimulation introduced to the input nodes of a neural network travels between layers of the network to produce some output on the other end. This output is evaluated by a trainer who applies supervised or unsupervised learning to alter the weights of synaptic connections and thereby change the way the network will respond. In this way, ANNs allow human knowledge and guidance to orchestrate the learning process. Such techniques, where learning is applied by a knowledgeable teacher, are often referred to as robot shaping.

Usually robot shaping involves symbolic interaction between a human and robot and may even involve a high-level tasking language. Such an interface allows humans to supply high-level assistance and allows the robot to accomplish the low-level learning necessary to achieve the goal. [19]

3   Robot description

The robot used in this work has been conceived in our laboratory [20][21] around a hardware architecture that facilitates the communication with a remote computer in different ways. It was based on different modules as shown in Figure (1). Every module has processing capabilities that allows updating the type of the low level current controller for the motor driver (2), test different dead reckoning approximations for the wheel encoders that determines the position (3) or compute different averaging methods for the infrared sensors (4). The robot can be totally free with an embedded master controller while transmitting wireless information to a computer, or the computer can be its master taking inputs and giving orders.

A mechanical structure handling this architecture was based on two DC motors controlling through gears two differential wheels. Wheels are provided with encoders that allows after processing determining an approximate relative position of the robot. Eight Infrared sensors are placed on a circle every 45°, they sense for obstacles by measuring the reflected light.
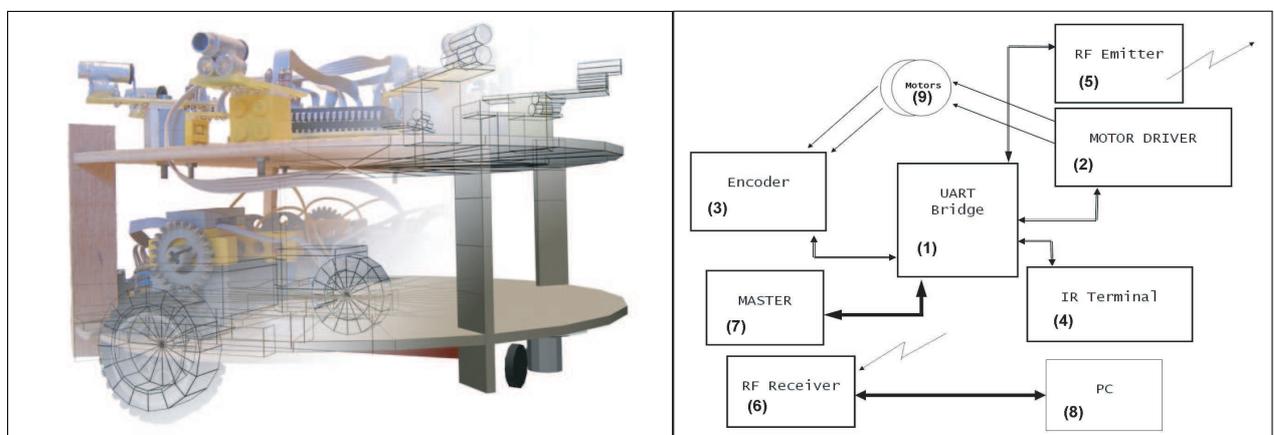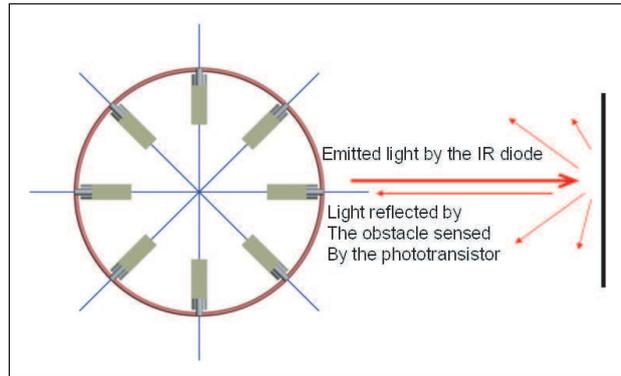

Figure 1: Robot Architecture

Figure 2: IR sensors (IR diode + Phototransistor)
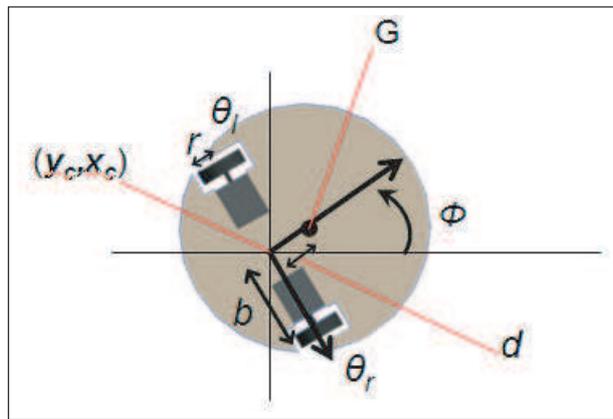

Figure 3: Robot parameters

The figure (3) shows the robot parameters where C is the center of the wheels and G is the gravity center, d separate these two centers as the in the general case d is always different than zero.

The movement vector q containing variables of the robot has the Cartesian coordinates of the center, the orientation and angle of each of left and right wheels:

$$q = (x_c, y_c, \phi, \theta_r, \theta_l) \tag{1}$$

Given the kinematics equations of this model:

$$\dot{y}_c \cos\phi - \dot{x}_c \sin\phi - d\dot{\phi} = 0$$

$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi - b\dot{\phi} = r\dot{\theta}_l$$

$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi + b\dot{\phi} = r\dot{\theta}_r$$

We can obtain the Pfaffian kinematics constraint:
$A(q)\dot{q} = 0$ With

$$A(q) = \begin{bmatrix} -\sin\phi & \cos\phi & -d & 0 & 0 \\ -\cos\phi & -\sin\phi & -b & r & 0 \\ -\cos\phi & -\sin\phi & b & 0 & r \end{bmatrix}$$

Deriving the Kinetic energy equation and considering the system with the Pfaffian constraint, we obtain the matrix differential system:

$$M(q)\ddot{q} + V(q,\dot{q}) = E(q)\tau - A^T(q)\lambda$$

$$A(q)S(q) = 0$$

$$S^T(MS\dot{v}(t) + M\dot{S}v(t) + V) = \tau$$

The resulting system is based on non-stationary matrices that will be re-computed every simulation step:

$$\dot{x} = \begin{bmatrix} Sv \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u$$

Thus we have an approximate model of the robot dynamics. Matrices of this differential equation have parameters measured on the real robot and some others are variables computed on the run with the simulation like position and speed.

The input of this equation is the torque applied on every wheel. Resolving theses equations provide the speed and position of the robot.
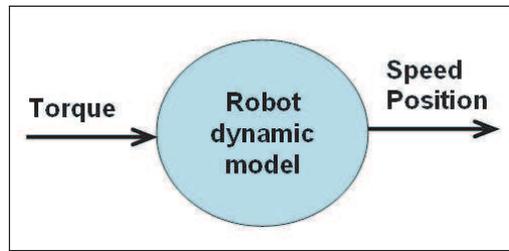

Figure 4: Robot dynamic model input output

4   Proposed approach

A hybrid approach is adopted using both neural and fuzzy logic methods to realize the specified task of reaching a goal point while avoiding obstacles. The control structure is made of two tasks as seen in the figure (5), a task for reaching the goal and a second one for avoiding obstacles; the input of the control system is sensors data and the outputs are the actuators command.
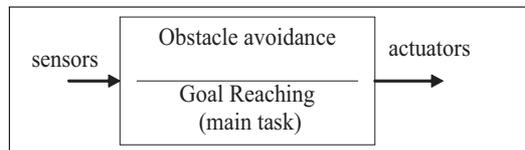

Figure 5: Control structure

5   Goal reaching task

The Goal Reaching task is expected to align the robot's heading with the direction of the goal; goal is the orientation difference between the robot axe and the goal.

A fuzzy controller is used to accomplish the task as shows the figure (7), with FC-lw and FC-rw the fuzzy command respectively of the left wheel and the right wheel.
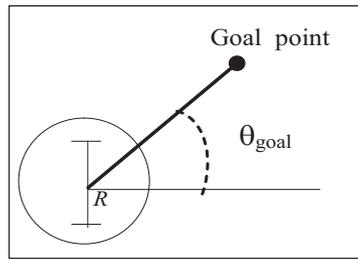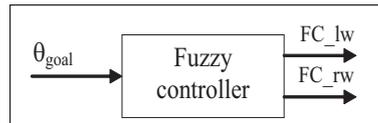
Figure 6: Goal orientation resolving

Figure 7: Goal reaching controller

Fuzzy sets

The variable ' goal ' has 8 fuzzy sets: (left, front left, front, front right , right, right back, back, left back)

The variables FC-lw and FC-rw have 6 fuzzy sets: (go ahead, go more on left, go more on right, quick turn left, quick turn right, go back)

Fuzzy inference system

The inference system id defined by the following rules:

• If goal is on front then go ahead

• If goal is on left then go more on left

• If goal is right then go more on right

• If goal is back then go back

• If goal is on front left then quick turn left

• If goal is left back then go back and go more on left

• If goal is on right back then go back and Go more on right
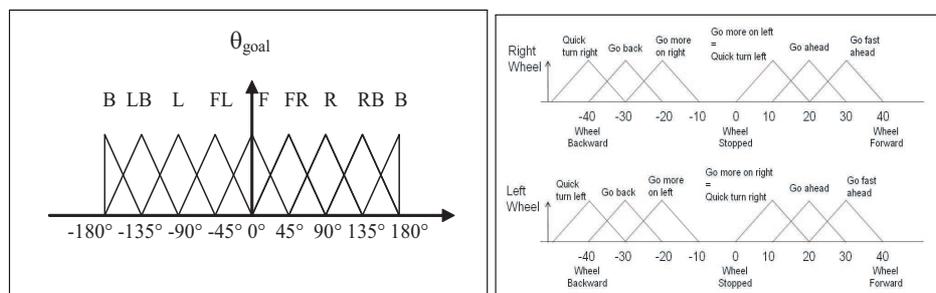
• If goal is on front right then quick turn right

Figure 8: Inputs and Outputs membership functions

6 Obstacle avoidance task using neural robot behavior modelling

For the task of avoiding obstacles, a neural model is utilized, reproducing the robot behavior in the used environment in order to avoid obstacles. It is a way of learning the environment with obstacle existence by affecting command values to each vector data from sensors. In our application the model inputs are the 8 infrared sensors data and the outputs are the commands of each wheel motor.
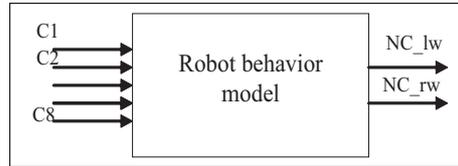

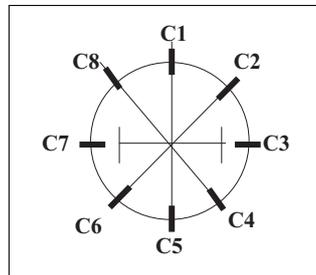Figure 9: Obstacle avoidance strategy

Figure 10: Infrared sensors position

The corresponding neural network model is a normalized multilayer perceptron (MLP) composed of 5 hidden neurons with sigmoidal activation functions and two output linear neuron representing wheels command. The training algorithm used is the backpropagation algorithm with simple gradient. The simple gradient version consists in modifying the weights w according to the following formula:

$w(k) = w(k-1) + \Delta w(k)$ with :

$$\Delta w(k) = -\mu \frac{\partial J}{\partial w}$$
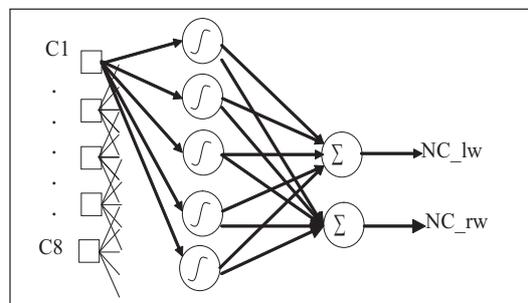
where

$$\mu$$

is the training step.

Figure 11: Neural network architecture

The network training is made using a conceived indoor environment with many obstacles of different shapes. The training data was collected using a joystick to guide the robot navigation in the environment

while avoiding obstacles. A sequence of the 8 sensors data and the corresponding wheels command is registered in a file used for the training and test of the achieved model.

The model validation is made by computing the prediction error and by testing the robot behavior in the same environment after its training. The training examples describing the obstacle avoiding behavior should be chosen carefully in order to be representative and can be generalized for other environments. A first training was made in a learning environment endowed with simple obstacles shape; we use a more type of the learning environment. The training principle is to learn the robot how to dodge obstacles; the robot goes forward, its front should be free otherwise it turns and follows the obstacle border as seen in figures (12) and (13). As a basic environment was used the one shown in the figure1.a; the test result was satisfactory in the same environment but collisions were detected when adding particular shapes of obstacles like corners or U traps. A second training is then achieved to learn avoiding this particular type of obstacles as seen in figure (13); the navigation test was satisfactory for corners but not for U traps; the robot begins to oscillate between the right and left side without having an outlet, in fact the same command is alternately applied for left and right wheel which causes the oscillation behavior. This situation was treated in the task activation module that gives the overall command for reaching goal and avoiding obstacles tasks.
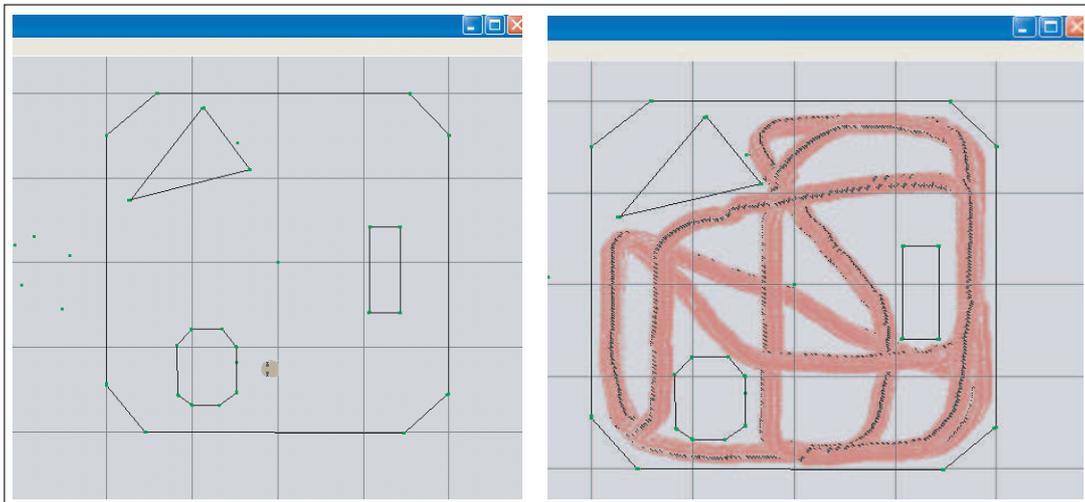

Figure 12: Behavior robot learning in simple environment

7 Task activation

The overall command, resulting from the fuzzy controller for reaching goal and the neural model for avoiding obstacles, is computed by a task activation module as shows the figure (14). A third unit is used at the input of the task activation module; it determines the situation (context) concerning the robot localization. Three situations are considered according to sensors data giving information about obstacles existence.

S1: IF C1 is far AND C2 is farĚAND C8 is far THEN obstacle is far

S2: IF C1 is near OR C2 is nearĚOR C8 is near THEN obstacle is near

S3: IF C1 is near AND C2 is near AND C4 is near AND C6 is near AND C7 is near AND C8 is near THEN trap situation
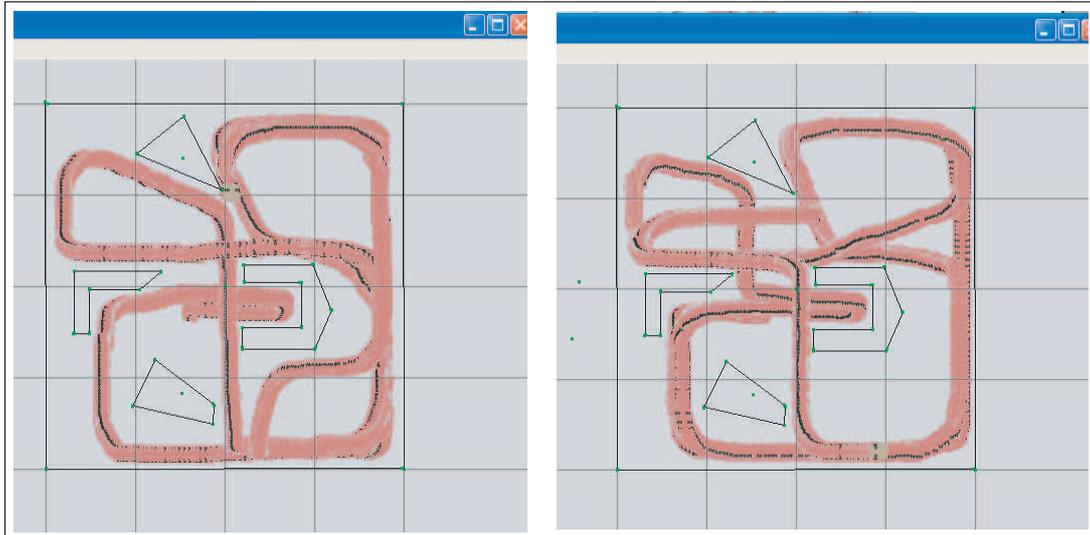
Figure 13: Behavior robot learning in complex environment (particular shaped obstacles)
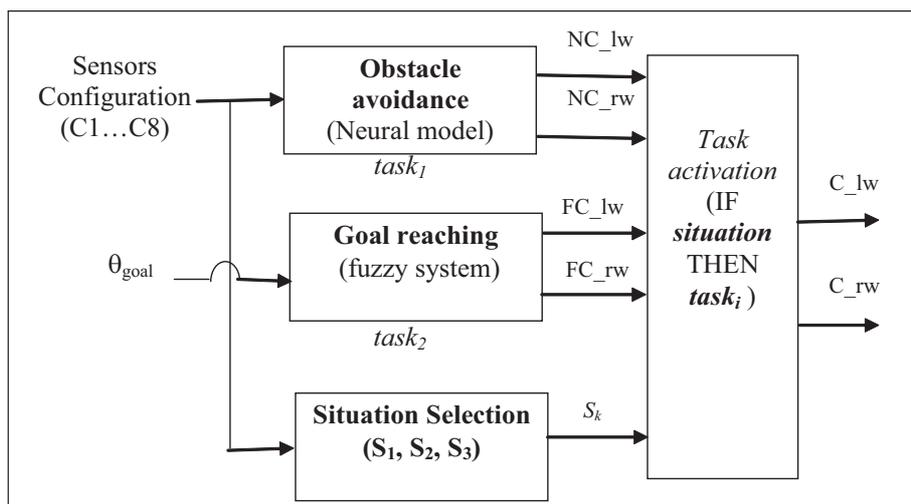

Figure 14: Task activation module

The input fuzzy sets describing the distances of each sensor to the nearest obstacle is defined in the figure (15).
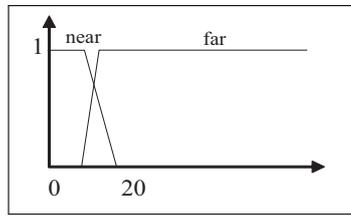

Figure 15: Membership functions for distance to obstacle

Trap situation is activated when the robot is trapped in a narrow corridor; the neural controller gives in this case a bad behavior; the robot begins to oscillate between the right and left side without having an outlet. The figure (16) shows an example of a trap situation.
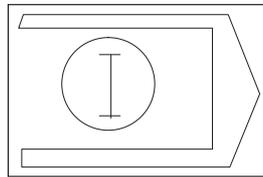

Figure 16: Example of a trap situation

The task activation module is made up of a fuzzy rule base:
IF situation THEN task i Three rules are considered as follows:

- IF obstacle is far THEN Goal reaching

- IF obstacle is near THEN obstacle avoidance

- IF trap situation AND 0 < goal < pi THEN turn left ELSE turn right

8   Simulation results

To verify the validity of the proposed approach, some cases were illustrated using a real time simulation of the upper presented robot; it has to move from a given current position to a desired goal position in an unknown environment endowed with different obstacle shapes: simple shapes as seen in figure (17) and particular ones depicted by figure (18.2). The learned environment and a new one were both tested for the robot navigation depicted respectively by figures (18) and (17). The robot is any case, able to reach the goal placed behind obstacles while avoiding obstacles successfully.

9   Summary and Conclusions

This framework presented a behavior based approach making use of the neural network and fuzzy control to realize the task of reaching goal in a crowded environment. A neural network modelling the robot behavior when encountering obstacles is conceived using supervised learning. Different shapes of obstacles were learned; the neural model should associate a particular sensor data to a given action. The goal reaching task was achieved using a fuzzy rule-based system; coordination of the two tasks was addressed using a task activation module giving the overall action depending on the context. The simulation examples of the generation of the collision free path with goal reaching show that designed strategy is acceptable as solution of this problem.
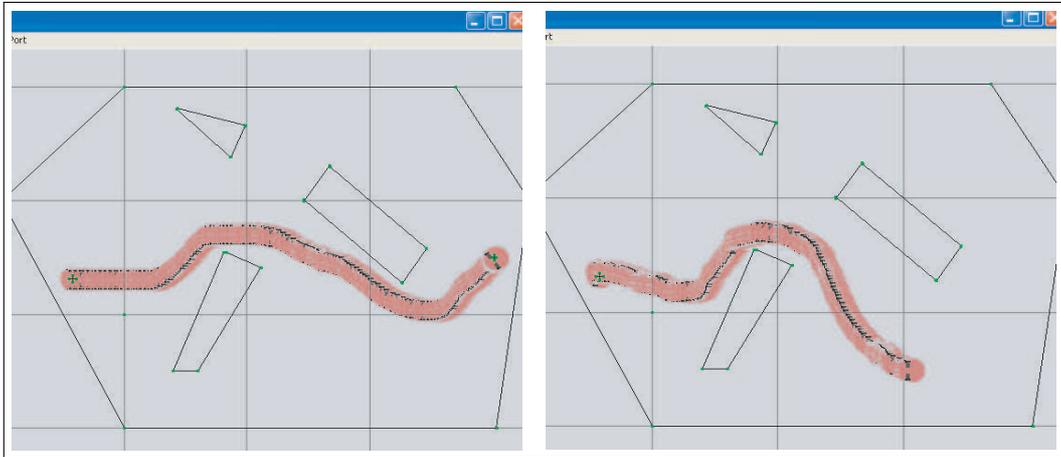
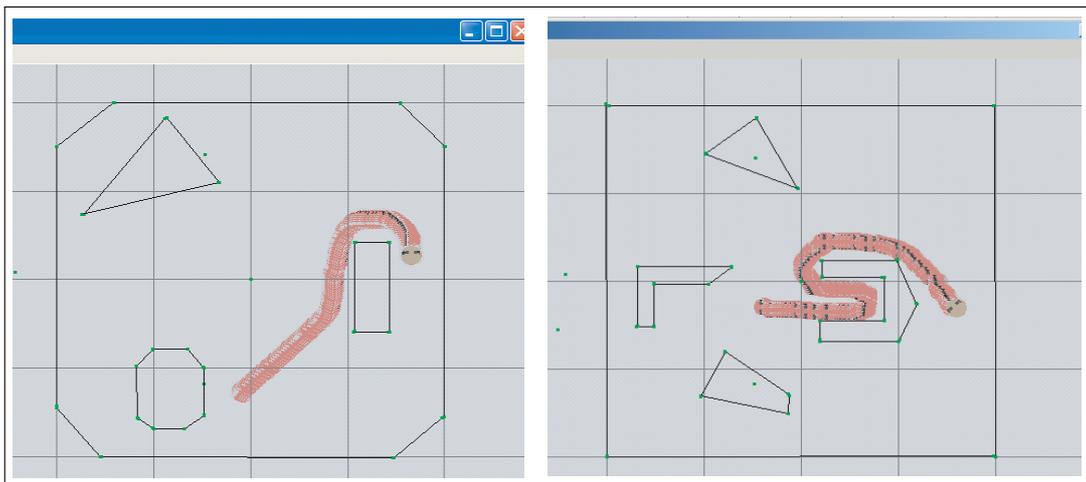Figure 17: Collision free goal reaching in an unlearned environment

Figure 18: Collision free goal reaching in a learned environment

Bibliography

[1] J.Miura, H.Uozumi and Y.Shirai, Mobile robot motion planning considering the motion uncertainty of moving obstacles, *In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp.692-697, 1999.

[2] H.Yu and T. Su, A destination driven navigator with dynamic obstacle motion prediction, *In Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2692-2697, Seoul, Korea: IEEE, 2001.

[3] M.Bennewitz, W.Burgard and S. Thrun, Adapting navigation strategies using motions patterns of people, *In Proceedings of IEEE International Conference on Robotics and Automation*, pp.2000-2005, Taipei, Taiwan: IEEE, 2003.

[4] Dmitry V. Lebedev;1, Jochen J. Steil, Helge J. Ritter,The dynamic wave expansion neural network model for robot motion planning in time-varying environments, University of Bielefeld, Faculty of Technology, Neuroinformatics Group, 2005

[5] R. C. Arkin, *Towards Cosmopolitan Robots: Intelligent Navigation in Extended Man-made Environments,* PhD Thesis, University of Massachusetts, Department of Computer and Information Science, 1987.

[6] R. C. Arkin, Motor schema-based mobile robot navigation, *Int. J. of Robotic Research,* Vol 8, pp. 92-112, 1989.

[7] R. C. Arkin and Tucker Balch, Principles and Practice in Review , *Journal of Experimental and Theoretical Artificial Intelligence (JETAI),* Vol. 9, No. 2/3, pp. 175-188, April, 1997

[8] R. A. Brooks, A Robust Layered Control System for a Mobile Robot, *IEEE Journal of Robotics and Automation,* Vol. 2, No. 1, March 1986, pp. 14-23; also MIT AI Memo 864, September 1985.

[9] R. A. Brooks, A Robot that Walks; Emergent Behavior from a Carefully Evolved Network, *Neural Computation,* 1:2, pp. 253-262, 1989,. Also in *IEEE International Conference on Robotics and Automation,* Scottsdale, AZ, pp. 292-296, May 1989.

[10] J. Rosenblatt, *DAMN: A Distributed Architecture for Mobile Navigation,* Ph.D. dissertation, Carnegie Mellon University Robotics Institute Technical Report CMU-RI-TR-97-01, Pittsburgh, PA, 1995.

[11] J. Rosenblatt and D.W. Payton, A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control, *Proceedings of the IEEE/INNS International Joint Conference on Neural Networks*, Washington DC, Vol. 2, pp. 317-324, June 1989.

[12] D. Langer, J.K. Rosenblatt, and M. Hebert, A Behavior-Based System For Off-Road Navigation, *IEEE Journal of Robotics and Automation,* Vol. 10, No. 6, pp. 776-782, 1994.

[13] Saffiotti, The uses of fuzzy logic for autonomous robot navigation: a catalogue raisonn'e, *Soft Computing Research journal*, Vol. 1, No. 4, pp. 180-197, 1997.

[14] H. Seraji and A. Howard, Behavior-based robot navigation on challenging terrain: A fuzzy logic approach, *IEEE Trans. Rob. Autom.,* Vol. 18, No. 3, pp. 308-321, 2002.

[15] Simon X. Yang, Hao Li, , Max Q.-H Meng, and Peter X Liu, An Embedded Fuzzy Controller for a Behavior-Based Mobile Robot with Guaranteed Performance, *IEEE Transactions on Fuzzy Systems,* Vol. 12, No. 4, pp.436-446,August 2004.

[16] X. Yang, M. Moallem, and R. V. Patel, A Layered Goal-Oriented Fuzzy Motion Planning Strategy for Mobile Robot Navigation, *IEEE transactions on systems, man, and cybernetics,* Vol. 35, no. 6, 1214-1224, December 2005.

[17] Majura F. Selekwa, Damion D. Dunlap, and Emmanuel G. Collins, Jr., Implementation of Multi-valued Fuzzy Behavior Control for Robot Navigation in Cluttered Environments, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation Barcelona,* Spain, pp.3699-3706, April 2005.

[18] Fatmi, A. Al Yahmadi, L. Khriji, and N. Masmoudi, A Fuzzy Logic Based Navigation of a Mobile Robot, *Transactions on engineering, computing and technology,* Vol. 15, ISSN 1305-5313, 2006

[19] D. Janglová, Neural Networks in Mobile Robot Motion, *Inernational Journal of Advanced Robotic Systems,* Vol.1 N.1, pp. 15-22,2004.

[20] W. Filali, I. Ayari, A.Chatti, Modelling, Real time simulation and Fuzzy control of a differential wheels mobile robot, *Congrès International en Sciences et Techniques de l'Automatique STA,* Hammamet, 2006.

[21] W. Filali, I. Ayari, A. Chatti, Research Platform on Mobile Robotics With Modular Architecture, *Tunisian-Japanese seminar on Culture, Science and Technology TJCST,* Sousse, 2005.

[22] I. Rivals, L. Personnaz, G. Dreyfus and D. Canas, Real-time control of an autonomous vehicle: A neural network approach to the path following problem, *In Neuro-Nimes,* pp 219-229, Nimes, 1994.

Imen Ayari, Abderrazak Chatti
Institut National des Sciences Appliquées et de Technologie (INSAT)
Centre Urbain Nord, BP. 676, 1080 Tunis Cedex, Tunisie
E-mail : ayari.imen@gmail.com, abderrazak.chatti@insat.rnu.tn

Imen Ayari (born on July 10, 1979) received the Engineer Diploma degree in 2003 and the master degree in 2005, both in Automatic and Industrial Computing from the "Institut National des Sciences Appliquées et de Technologie", Tunisia. She is currently preparing the Ph.D. degree in automatic and computer science within the framework of LAGIS-EC-Lille and LARA-ENIT (Tunisia) cooperation. Her research is related to the appliance of soft computing methods especially neural networks for mobile robots control.

Abderrazak Chatti received the Master of Science degree in 1988 from Polytechnical School of Saint-Pétersbourg from Russia and the PhD degree in Technical Sciences in 1994 from Polytechnical University of Saint-Pétersbourg, Russia, both in Automatics and Industrial Computing. Presently he is associate professor in the "Institut National des Sciences Appliquées et de Technologie", Tunisia. His research interest includes topics such as modelling and control of non linear dynamic processes, robust and intelligent control.