



A NEURAL FUZZY INFERENCE BASED ADAPTIVE CONTROLLER FOR NONHOLONOMIC ROBOTS

Ting Wang, Fabien Gautero, Christophe Sabourin, Kurosh Madani

Images, Signals, and Intelligent Systems Laboratory (LISSI / EA 3956),
University PARIS-EST Creteil (UPEC),
Senart Institute of Technology, Avenue Pierre Point,
77127 Lieusaint, France,
{sabourin, madani}@univ-paris12.fr, <http://lissi.univ-paris12.fr>

Abstract: *In this paper, we propose a control strategy for a nonholonomic robot which is based on an Adaptive Neural Fuzzy Inference System. The neuro-controller makes it possible the robot track a desired reference trajectory. After a short reminder about Adaptive Neural Fuzzy Inference System, we describe the control strategy which is used on our virtual nonholonomic robot. And finally, we give the simulations' results where the robot have to pass into a narrow path as well as the first validation results concerning the implementation of the proposed concepts on real robot.*

Keywords: *Neuro-Fuzzy inference system, Nonholonomic robot, learning, Neuro-controller, Artificial neural network, Software and hardware implementation.*

1. INTRODUCTION

Research about the multi-robot systems have started in the late 1980s, like for instance the project CEBOT [1]. Indeed, the multi-robot systems offer many advantages in comparison with systems using only one robot ([2] and [3])

- In first, cooperation between a group of several robots can carried out more complex tasks,
- Secondly, the use of several robots for a given task allows to increase robustness,
- And finally, the design and the use of several simple robots can be cheaper and more flexible.

Today, and in the future, many applications can benefit of advantages of multiple robot systems like, for instance, in the warehouse management, for the industrial assembling, in military applications, or for daily tasks, so on. But generally, the design of one control strategy for systems with several robots requires cooperation and coordination between all robots. This means that robots can communicate between them and self-organize in the group. With the new recent technologies like wireless communication, one robot can easily send information to another robot. Consequently, in the future works, the main challenge will focus on the design of control strategies allowing to a group of robots to self-organize with, if possible, emergent

behaviors. In this context, the goal of our laboratory is to design control strategies for multi-robot systems. But one major problem about the control of a multi-robot system is coordination and formation control, and namely the design of control strategy making it possible for a wheeled robot to track a desired trajectory. And generally, the wheeled robots are nonholonomic robots increasing the difficulty to design the control strategy.

Most of the control approaches are based on asymptotic stabilization with the feedback controls. Different methods have been used to reduce or to transform the nonlinear kinematics issued equations into a linear approximation system. The approach proposed by Samson in 1995 transforms the nonlinear system into a chained system with the feedback control to solve the path-following problem [4]. Several authors have addressed the problem of tracking admissible trajectory by applying dynamic feedback linearization techniques ([5] to [8]).

In [9] Morin and Samon are certainly the firsts to address the problem of tracking arbitrary trajectories (i.e., not necessarily for the controlled robot) based on the conception of transverse functions. And in [10], Barfoot and Clark propose a feedback control law inheriting strong robustness properties associated with stable linear systems. However, the above-mentioned approach yields slow convergence, making it quite inappropriate for real-time

applications.

This short overview of control strategies for nonholonomic robots emphasizes the fact that all proposed approaches are based on a kinematical modeling and most of them have a slow convergence. The main drawback of the above-referenced approaches is related to the fact that they have been designed for specific cases (specific usage) and could not always be generalized for the general cases. An alternative solution to the kinematical modeling is to use artificial neural networks' based approaches.

In this paper, we propose a new approach to control nonholonomic robot based on Adaptive Neural Fuzzy Inference System (ANFIS). This approach may be decomposed in two parts: the first one allows rotting an arbitrary (e.g. given) path into several trajectories (characterized as "desired trajectories"), and the second is composed of two neuro-controllers (one controlling the position and the other the orientation), allowing to track the desired trajectories. In fact, the suggested ANFIS based control approach doesn't depend on kinematics issued equations, and although in the present work it is applied for nonholonomic robot's control, the concept may be generalized as a global control strategy to another kind of wheeled robots.

The paper is organized in six sections. In the next section, we introduce the Adaptive Neural Fuzzy Inference System. In the third section, we give the kinematical model of the nonholonomic robot in order to state the control parameters, describing how we can control the wheeled robot with ANFIS. In section 4, we present the suggested control strategy. Simulations' and validation's results are shown in the fifth section. We give the simulations' results where the robot have to pass into a narrow path as well as the first validation results concerning the implementation of the proposed concepts on real robot. At last, we get some brief conclusions.

2. ADAPTIVE NEURAL FUZZY INFERENCE SYSTEM

The main advantage of a Fuzzy Inference Systems is that they allow dealing with linguistic rules, making such systems appropriate to design control strategy where it is difficult to use the mathematical modeling. The control of nonholonomic systems (and thus, nonholonomic robots) is among those classes of problems where the mathematical modeling is difficult to be used because of strong non-linearity of such systems. However, the main disadvantage of a fuzzy inference system is that it needs knowledge of an expert. It also needs a relatively long time to get the accurate membership functions.

Neural network based approaches, or more generally adaptive systems based on learning process (i.e. Q-learning, genetic algorithm, so on), can overcome this disadvantage improving the basic fuzzy inference system. ANFIS, combining neural networks and fuzzy inference systems, is a class of such adaptive fuzzy inference systems. In this section, we briefly remind the ANFIS architecture initially proposed by Jang in 1995 [11].

Let us assume a control system with m inputs X_1, X_2, \dots, X_m and one output Y ; let suppose n linguistic rules, where each rule R_i , where $i \in \{1, 2, \dots, n\}$ is the index of the rule, can be expressed as:

If X_1 is A_{i1} and X_2 is A_{i2} and ... and X_m is A_{im} Then Y is w_i

A_{ij} is a fuzzy set for i -th rule and j -th input and w_i is a real number that represents a consequent part. In the present case, the membership function is defined as a Gaussian function, given by equation (1). The output of the issued neural-fuzzy network is given by the equation (2), where u_i is obtained from the equation (3).

$$\mu_{ij} = e^{-\frac{(x_j - a_{ij})^2}{2b_{ij}^2}} \quad (1)$$

$$Y = \frac{\sum_{i=1}^n u_i w_i}{\sum_{i=1}^n u_i} \quad (2)$$

$$u_i = \mu_{i1} \mu_{i2} \dots \mu_{ij} \dots \mu_{im} \quad (3)$$

Now, let us define the set of parameters z , given by equations (4) and (5), supposed to be adapted during the neural based minimization process as well as the function $F(z, t)$, defined by equation (6), to be minimized during the aforementioned process, where t is the current time, $Y(t)$ is the output of the neural fuzzy network and $Y^d(t)$ is the desired output.

$$z = [z_a, z_b, z_w] \quad (4)$$

$$\begin{aligned} z_a &= (a_{i1}, \dots, a_{ij}, \dots, a_{nm}) \\ z_b &= (b_{i1}, \dots, b_{ij}, \dots, b_{nm}) \\ z_w &= (w_{i1}, \dots, w_{ij}, \dots, w_{nm}) \end{aligned} \quad (5)$$

$$F(z, t) = \frac{1}{2} (Y(t) - Y^d(t))^2 \quad (6)$$

In 1995, Godjevac show that if $F(z, t)$ is defined as it has been stated-above, then it is possible to use an iterative procedure to update the above-defined parameters in order to minimize the function $F(z, t)$ [12]. According to Godjevac, the three kinds of above-defined parameters a_{ij} , b_{ij} and w_i may be updated by equations (7), (8) and (9), respectively. Γ_a , Γ_b and Γ_w are predefined constants.

$$a_{ij}(t+1) = a_{ij}(t) - \Gamma_a \frac{u_i}{\sum_{k=1}^n u_k} (Y - Y^d) \quad (7)$$

$$(w_i - Y) \frac{(x_i - a_{ij}(t))}{b_{ij}(t)^2}$$

$$b_{ij}(t+1) = b_{ij}(t) - \Gamma_b \frac{u_i}{\sum_{k=1}^n u_k} (Y - Y^d) \quad (8)$$

$$(w_i - Y) \frac{(x_i - a_{ij}(t))}{b_{ij}(t)^3}$$

$$w_i(t+1) = w_i(t) - \Gamma_w \frac{u_i}{\sum_{k=1}^n u_k} (Y - Y^d) \quad (9)$$

3. NONHOLONOMIC ROBOT'S DYNAMICS

Generally, the control of wheeled robots is performed by a follow of reference path and supposes to measure both the position and orientation with respect to a fixed frame. Let us consider a given trajectory C in the reference frame, and a point P attached to the robot chassis, at the mid-distance of the wheels, as illustrated in Fig.1. The state of the robot can be described by a triplet as $P(x, y, \theta)$, in which x and y are the robot's current coordinates, measured in the reference frame. θ is the angle formed by the robot's motion direction and the x-axis.

The kinematical modeling of this wheeled robot (i.e. unicycle-type mobile robot) may be represented by the set of equations (10) and (11) (see [11] and [12]), where V_x and V_y represent the instantaneous horizontal and vertical velocities of the point P located at mid-distance of the actuated wheels, respectively. V represents the intensity of the longitudinal velocity and Ω the angular velocity of the robot. Ω^{left} and Ω^{right} are the angular velocity of the "left" and "right" wheels, respectively, r is the radius of the wheels and l is the distance between the two wheels.

$$\begin{aligned} V_x &= V \cos(\theta) \\ V_y &= V \sin(\theta) \\ \dot{\theta} &= \Omega \end{aligned} \quad (10)$$

$$\begin{aligned} V &= \frac{r}{2} (\Omega^{right} + \Omega^{left}) \\ \Omega &= \frac{r}{2l} (\Omega^{right} - \Omega^{left}) \end{aligned} \quad (11)$$

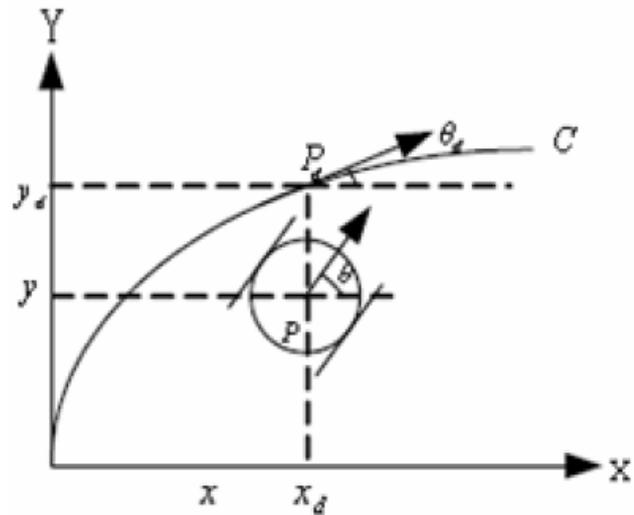


Fig. 1 – Robot's coordinates described by a triplet as $P(x, y, \theta)$

For an unicycle-type mobile robot, the goal of the control strategy is to compute the velocities of each wheel in order to make the robot follow a given (e.g. desired) path. The given trajectory can be expressed as a time-dependent function $P(x^d(t), y^d(t), \theta^d(t))$, where $\theta^d(t)$ represents of the trajectory's curvature at each step time t , $x^d(t)$ is the desired x-axis coordinate and $y^d(t)$ is the desired y-axis coordinate of the robot.

However, it is pertinent to note that in the case of nonholonomic robots complying with the kinematical model, represented by equations (10) and (11), the achievement of the aforementioned goal is not a trivial problem. The difficulties are inherent to the nonlinear nature of the robot's dynamics which is sensitive as well to the given trajectory's shape as to the environment (e.g. the nature of ground, etc...).

4. SUGGESTED CONTROL STRATEGY

As it has been mentioned in the introductory section, we propose a new approach based on neural fuzzy networks, taking advantage from combining artificial neural network's learning skills and Fuzzy

Inference System's linguistic rules. The proposed approach includes two neuro-controllers (one controlling the position and the other the orientation), allowing to track the desired trajectories. The goal of the two neural networks is to control the velocity of each wheel in order to minimize the position error, defined as the error between robot's current position and the desired position $(x(t) - x^d(t), y(t) - y^d(t))$, and the orientation error, defined as the error between the robot's current orientation and the desired orientation $(\theta(t) - \theta^d(t))$.

4.1. ORIENTATION CONTROL

The orientation control allows the robot to rotate in order to follow the target angle. Consequently, the related ANFIS based controller requires one input $(X_y(t))$ which is the aforementioned orientation error (see equation (12)), and one output $(Y_\theta(t))$ which is defined as an angular velocities difference (see equation (13)). Ω_θ^{right} and Ω_θ^{left} represent right wheel's and left wheel's current angular velocities, respectively.

$$X_\theta(t) = \theta(t) - \theta^d(t) \quad (12)$$

$$Y_\theta(t) = \Delta\Omega(t) = \Omega_\theta^{right}(t) - \Omega_\theta^{left}(t) \quad (13)$$

Thus, according to ANFIS described in Section 2, the ANFIS based controller's output (defined by the equation (13)) could be expressed as equation (14).

$$Y_\theta(t) = \Delta\Omega(t) = \frac{\sum_{i=1}^n u_i^\theta w_i^\theta}{\sum_{i=1}^n u_i^\theta} \quad (14)$$

In the same way, according to ANFIS described in Section 2, at each time step (e.g. iteration), the parameters $w_i^\theta(t)$ are update in order to minimize the function $F_\theta(t)$ defined as expressed by equation (15).

$$F_\theta(t) = (\theta(t) - \theta^d(t))^2 \quad (15)$$

4.2. POSITION CONTROL

As for the orientation, the position control allows the robot to move in order to track the target point

$(X^d(t), Y^d(t))$ according to a given (e.g. some desired) path. Consequently, in this case, the related ANFIS based controller requires two inputs $X_{Px}(t)$ and $Y_{Py}(t)$ which are defined on the basis of the aforementioned position error (see equations (16) and (17)), and one output $Y_P(t)$ which is also defined as an angular velocity (see equation (18)), where $\Omega_\theta^{right}(t)$ and $\Omega_\theta^{left}(t)$ represent the above-stated right wheel's and left wheel's current angular velocities.

$$X_{Px}(t) = x(t) - x^d(t) \quad (16)$$

$$X_{Py}(t) = y(t) - y^d(t) \quad (17)$$

$$Y_P(t) = \Omega_P^{right}(t) = \Omega_P^{left}(t) \quad (18)$$

$x^d(t)$ and $y^d(t)$ correspond to robot's current coordinates and $x^d(t)$ and $y^d(t)$ to the desired coordinates (according to the desired robot's trajectory). $\Omega_P^{right}(t)$ and $\Omega_P^{left}(t)$ represent, respectively, right wheel's and left wheel's current angular velocities, regarding the position. Here also, according to what has been stated in Section 2, the ANFIS based controller's output (defined by the equation (18)) could be expressed as equation (19).

$$Y_P(t) = \frac{\sum_{i=1}^n u_i^P w_i^P}{\sum_{i=1}^n u_i^P} \quad (19)$$

In the same way, at each time step (e.g. iteration), the parameters w_i^P are update in order to minimize the function $F_P(t)$ defined as expressed by equation (20).

$$F_P(t) = \sqrt{(x(t) - x^d(t))^2 + (y(t) - y^d(t))^2} \quad (20)$$

4.3. TRAJECTORY CONTROL

The trajectory's control strategy combines the above-described orientation's and position's ANFIS based controllers. Fig.2 gives the bloc-diagram of the suggested control strategy, where the two involved neuron-controllers correct the robot's orientation and its position in order to make the robot track some desired trajectory.

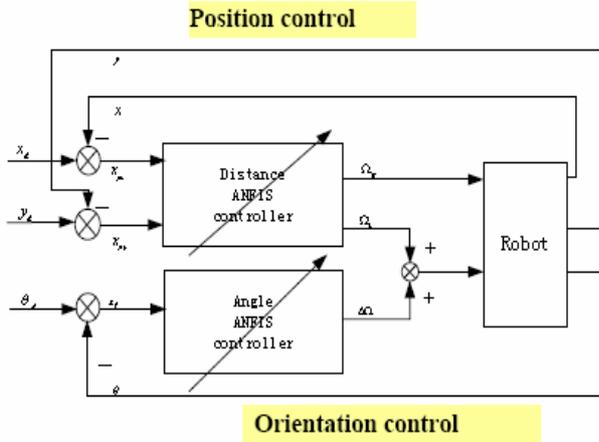


Fig. 2 – Bloc-diagram of the trajectory’s ANFIS based control strategy, including an orientation’s dedicated neuron controller and a position’s devoted neuron-controller

In this case, the angular velocity of two wheels ($\Omega^{right}(t)$ and $\Omega^{left}(t)$) are given by the set of two equation (21). $\Omega_p^{right}(t)$ and $\Omega_p^{left}(t)$ are obtained from the ANFIS based position controller, and $\Delta\Omega$ is obtained from the ANFIS based orientation controller [14].

$$\begin{aligned} \Omega^{right}(t) &= \Omega_p^{right}(t) \\ \Omega^{left}(t) &= \Omega_p^{left}(t) + \Delta\Omega \end{aligned} \quad (21)$$

In the most of proposed neuron-controllers, the control task is performed in “generalization” phase (e.g. after accomplishing the training phase) It is pertinent to note that accordingly to what has been mentioned in Section 2, the trajectory’s control (orientation and position control), in the proposed approach, is performed by adjusting (updating) the set of parameters z (defined in Section 2) minimizing the function $F(z, t)$. In other words, our control strategy uses the “learning” process (e.g. training phase) to control the robot’s wheels. This way of doing may be interpreted as some kind of “unconscious” artificial cognitive mechanism (by opposition to “conscious” cognitive mechanism – see [15]) where the “knowledge” (e.g. learning) based process operates in a “reflexive” way (by opposition to “intentional” regulation) regulating the robot’s rolling organs (wheels).

5. VALIDATION RESULTS

In this section we present two sets of results, validating the suggested ANFIS based control

strategy. The first one, issued from simulation, presents a practical example where the robot is supposed to move from an initial position to a destination (e.g. goal) position by passing a narrow path. The second set gives results obtained from implementation of the suggested concept on real robot (a Khepera III robot) performing a simple but a nonlinear trajectory.

5.1. SIMULATION BASED CASE STUDY

The case study concerns the control of a wheeled robot moving from an initial position to a destination (e.g. goal) position by passing a narrow path, according to a predefined trajectory. Fig.3 shows the case study’s frame where, starting from the point “A” (initial position), the robot is supposed to reach the point “C” (e.g. final position) accordingly to the indicated trajectory.

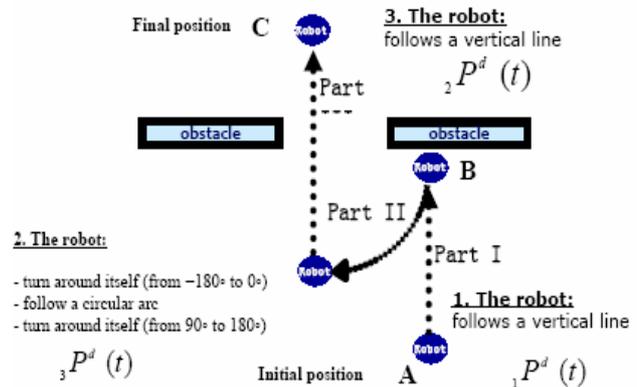


Fig. 3 – Description of the “case study” frame and the desired (e.g. planned trajectory)

As shows Fig.3, the proposed example may be decomposed into three parts: firstly the robot moves from the point A toward the obstacles, secondly the robot follows a circle trajectory, and finally the robot goes towards the final position.

- During the above-mentioned three parts (of the whole trajectory), the desired fractional trajectories $P(x^d(t), y^d(t), \theta^d(t))$ are computed as follow:
- During the first part, the robot moves from initial position A to the obstacle with position’s control only (the orientation remains unchanged). In this part, robot follow the vertical line $x^d(t) = 0.3$ without the orientation control. Equation (22) gives the path definition, where $x^d(t)$ and $y^d(t)$

represent the robot's current coordinates within the reference frame. $\theta^d(t) = 0^\circ$ is the orientation of the robot. Δy is chosen according to both length L and duration T of the path's accomplishment (execution).

$$P^d = \begin{cases} x^d(t) = 0.3 \\ y^d(t) = y^d(t-1) + \Delta y \\ \theta^d = 0 \end{cases} \quad (22)$$

- During the second part, firstly the robot turn around itself from $\theta^d(t) = 0^\circ$ to $\theta^d(t) = -180^\circ$ by using the orientation control, and secondly the robot use trajectory control to follow a circular arc (according to the set of equations (23)). Finally, the robot rotates from $\theta^d(t) = -270^\circ$ to $\theta^d(t) = 0^\circ$.

$$P^d = \begin{cases} x^d(t) = 0.3 \cdot \cos(\theta(t)) \\ y^d(t) = -0.3 \cdot \cos(\theta(t)) \\ \theta^d(t) = \theta^d(t-1) + \Delta\theta \end{cases} \quad (23)$$

- During the last part, the robot follows a vertical line $x^d(t) = 0.0$ and passes the narrow path in order to reach the final position (e.g. the point "C").

Simulation have been performed by using software Webots (www.cyberbotics.com) simulating a virtual robot KheperaIII within the above described case study's faram. The controller have been designed under MatLab software platform (www.mathworks.com). Fig.4 and Fig.5 show the trajectory and orientation of the robot during the simulation, respectively. On both two figures, the red line represents the desired items (trajectories orientation) and the blue dot line the accomplished items (position or orientation) of the robot.

Fig.6, (6-a, 6-b, 6-c and 6-d) shows a snapshot of this simulation. The path of the robot can be interpreted as follow:

- From $t = 0$ (figure 6-a) to $t = 200$ (figure 6-b), the robot follow a vertical line and moves from the point $(x = 0.3, y = -0.8)$ to point $(x = 0.3, y = 0)$.
- At $t = 200$ (figure 6-b), the robot turns on itself during 100 step time. During this stage, the robot stay at the point $(x = 0.3, y = 0)$ but turns from 0°

to -180° .

- From $t = 300$ to $t = 500$ (figure 6-c), the robot follows the above-defined desired circular trajectory and moves progressively from the point $(x = 0.3, y = 0, \theta = -180^\circ)$ to $(x = 0, y = -0.3, \theta = -270^\circ)$
- At $t = 500$ the robot turns on itself during 100 step time. During this stage, the robot stay at the point $(x = 0, y = -0.3)$ but turns from -270° to 0° .
- Finally, from $t = 600$ to $t = 800$ (figure 6-d), the robot follow a vertical line and moves to goal position $(x = 0, y = 0.6)$.

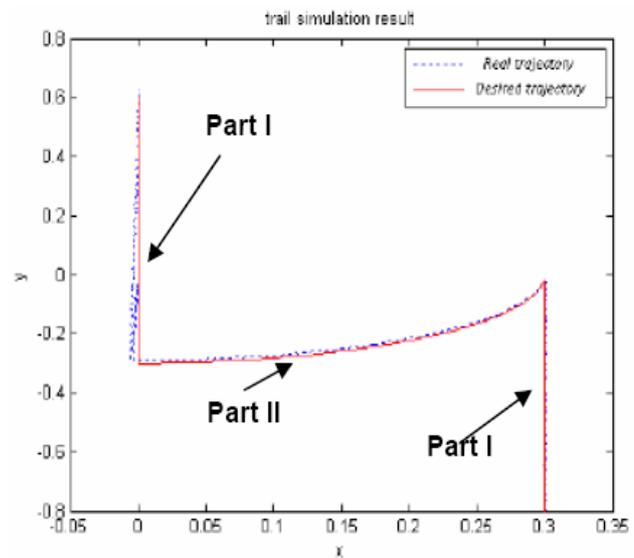


Fig. 4 – Accomplished and desired positions relative to the three sub-trajectories defining the case study example

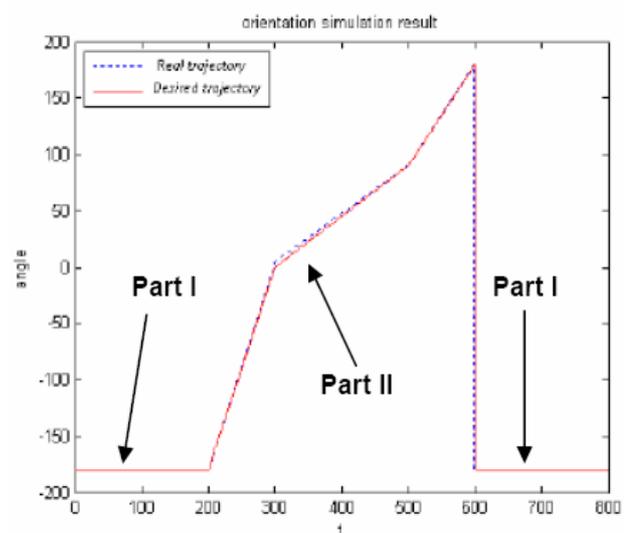


Fig. 5 – Accomplished and desired orientations relative to the three sub-trajectories defining the case study example

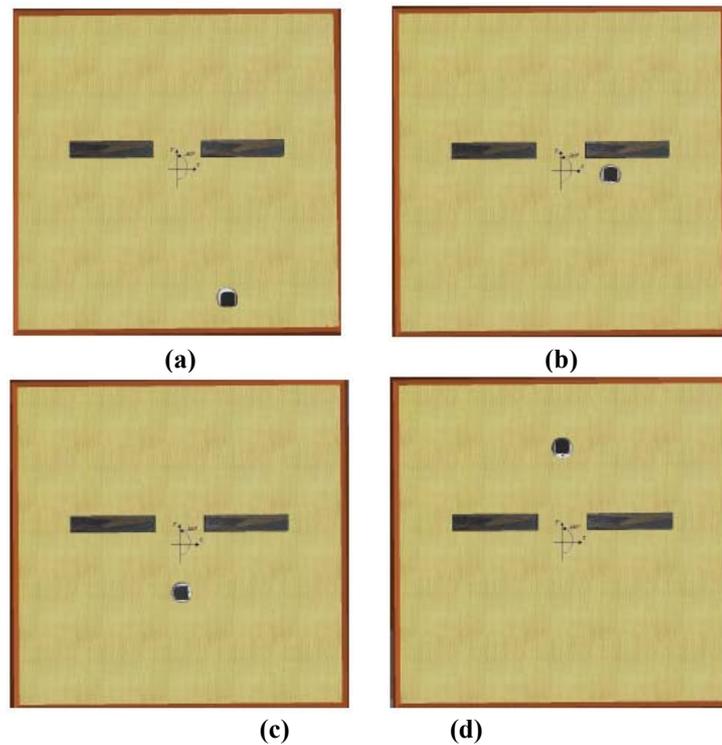


Fig. 6 – Simulation results showing virtual Khepera III robot executing the Case study example

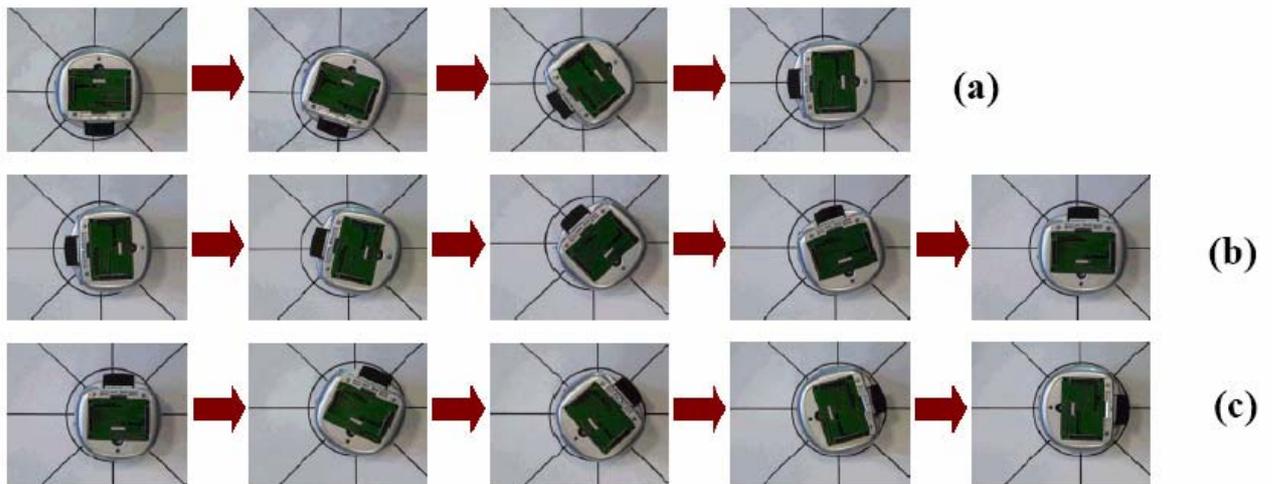


Fig. 7 – Snapshot of the experimental validation concerning the real kheperaIII robot's orientation's control: rotation of -90° (a), -180° (b) and -270° (c)

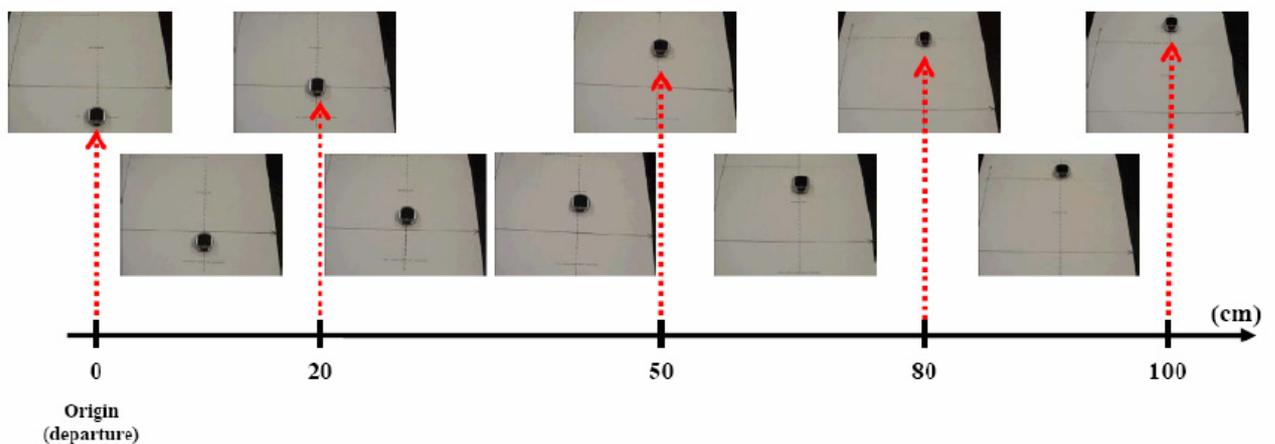


Fig. 8 – Snapshot of the experimental validation concerning the real kheperaIII robot's position's control. From left to right: robot moves away 20, 50, 80 and 100 centimeters from its starting position

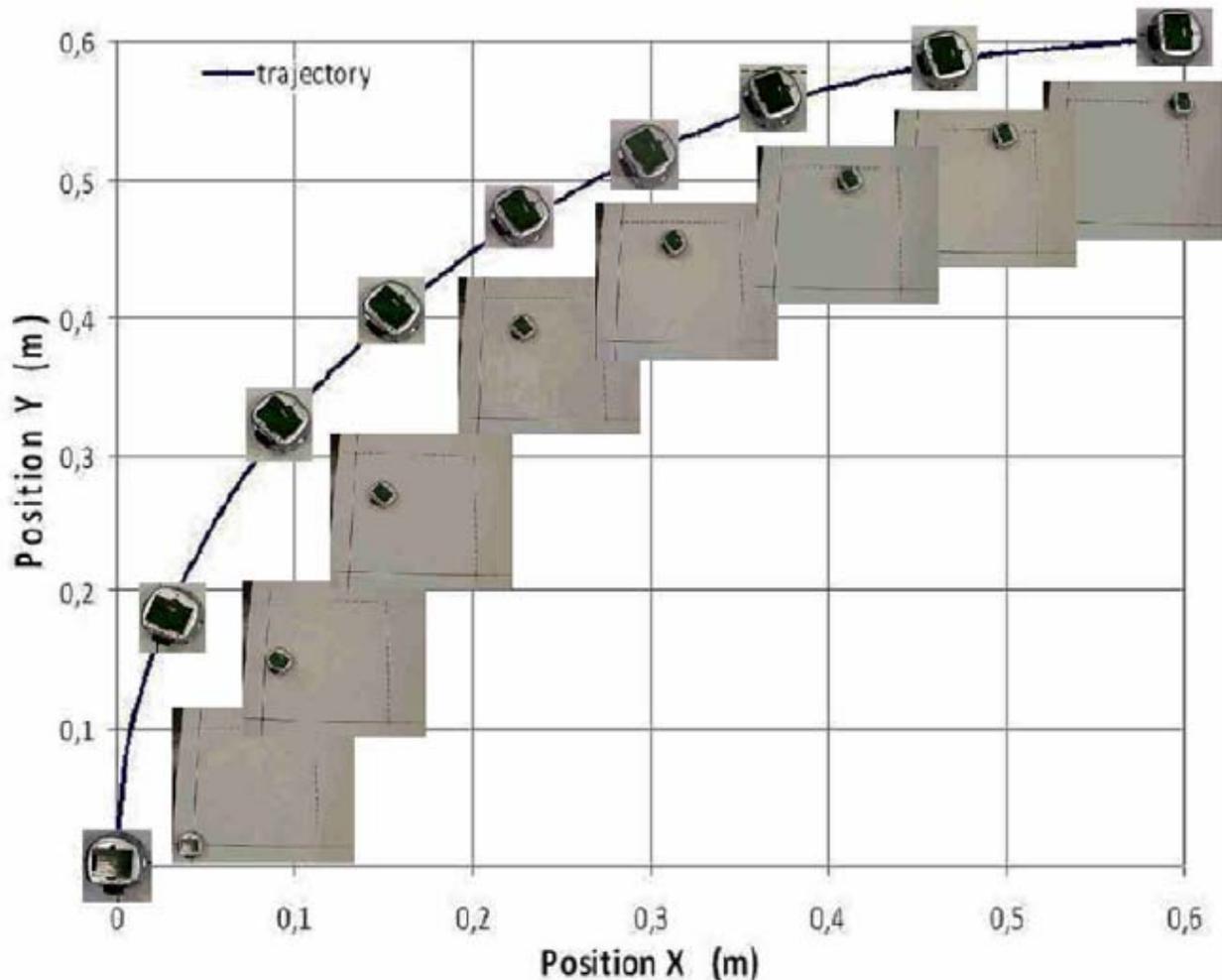


Fig. 9 – Snapshot of the experimental validation concerning the real kheperaIII robot's trajectory's control. The robot performs a curvilinear trajectory

5.2. EXPERIMENTAL VALIDATION USING REAL ROBOT

The used robot is the robot kheperaIII with the korebotLE module (see <http://www.k-team.com>). The robot Khepera III is equipped with two motors associated with incremental encoders, 9 infrared sensors and five ultrasonic sensors. A “dsPIC 30F5011” microprocessor allows to manage all devices of the robot through a I2C communication. In addition, this robot offers the possibility to connect a KoreBot board allowing increasing the computational abilities. The main component of the KoreBot board is an Intel “PXA255” XScale processor running at 400 MHz with 60 MB RAM and 32 MB flash memory. When the KoreBot board is mounted on the Khepera III robot, the dsPIC microcontroller runs the communication protocol and switches to the I2C slave mode. To control the robot, we used the “Khepera III Toolbox”. This is a set of scripts, programs and code modules for the Khepera III robot (see http://en.wikibooks.org/wiki/Khepera_III_Toolbox) allowing to control the robot. The previously described cognitive controllers

which are based on ANFIS have been designed with c language and implemented on the korebot. Both orientation (e.g. rotation) and position of the robot are computed by using an odometer based process. Fig.7 and Fig.8 show the first results relative to the experimental validation. Fig.7 shows the ANFIS based controller's validity as well as its performance on controlling the real kheperaIII robot's orientation. In this experimental validation, the robot is supposed to perform three successive rotations according to the following protocol: first, starting from its initial orientation (shown by the first picture of the figure 7-a), the robot performs a rotation of -90° (e.g. “turning-right” operation) maintaining its position. Then, starting from its new orientation, the robot repeats two times the above-mentioned operation (e.g. turns-right) attaining successively the -180° and -270° (see figures 7-b and 7-c).

Fig.8 shows the experimental validation relative to the ANFIS based controller's performance on controlling the real kheperaIII robot's position. In this second experimental validation, the robot is supposed to move respecting a straight line (e.g.

without changing its initial orientation shown in the leftist picture of the figure) attaining four successive new positions: 20cm, 50cm, 80cm and 100cm from its starting position, respectively. Fig.9 gives results of experimental validation on a real Khepera III robot considering a curvilinear desired trajectory. The predefined trajectory is shaped in a 60-b-60 cm 2-D frame. The starting point is the location characterized by $(x = 0, y = 0, \theta = 0^\circ)$ and the final destination is located at $(x = 60, y = 60)$ with an orientation of $\theta = +90^\circ$. So, as well the robot's position as its orientation change between departure location and final destination in a nonlinear way. In fact, as shows this figure through the 9 sampled positions (and the associated photographs) the robot follows correctly the planned (desired) trajectory.

It is pertinent to remind that such control mechanism could be generalized to different kind of wheeled robots independently from the concerned robots' dynamics (because it operated on the basis of a "learning" process).

6. CONCLUSION

In this paper, we have proposed a control strategy for nonholonomic robot based on Adaptive Neural Fuzzy Inference System. This neuro-controller makes it possible the robot track a given reference trajectories. We have presented results relative to the control of a robot aiming to avoid an obstacle. We also presented the experimental validation results relative to the implementation of the proposed cognitive controller using a real kheperaIII robot. The obtained results show the viability of the proposed machine-learning based approach in controlling as well the robot's position as its orientation.

The first interest of our approach is that it is independent from the robot's kinematical model. Consequently, it is possible to extend our control strategy for another kind of robot as cart-like model. The second interest is that it offers the possibility to design multi-level control, where the path planning and the trajectory computing are separated.

Finally, it is pertinent to remind that contrary to other neuron-controller (where the control task is performed after accomplishing the training phase), the proposed ANFIS based approach uses the training process (continuously) to perform the control. This way of doing may be interpreted as kind of "unconscious" artificial cognitive mechanism where the "knowledge" (e.g. learning) based process operates in a "reflexive" way regulating the robot's rolling organs (wheels).

Further works will focus, on the one hand, the generalization of such multi-level control strategy to the control of a robot's formation (e.g. a group of

several Khepera III robots), and on the other hand, the experimental validation on a real group of kheperaIII robots including several robots.

ACKNOWLEDGMENTS

This work was supported by a doctoral fellowship from the China Scholarship Council (CSC). Authors wish to express their gratitude to CSC.

7. REFERENCES

- [1] Fukuda T., Nakagawa S., Kawauchi Y. and Buss M. Self organizing robots based on cell structures CEBOT. *Proc IEEE Int. Workshop Intell. Robot. Syst.*, (1998) pp. 145-150.
- [2] Parker L.E. Multiple Mobile Robot Systems. *Springer Handbook of Robotics*, 2008, pp. 921-941.
- [3] Cao Y.U., Fukunaga A.S., Kahng A.B. and Meng F. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, (Vol. 4) (1997) pp. 1-23.
- [4] Samson C. Control of chained systems application to the path following and time varying point-stabilization. *IEEE Transactions on Automatic Control*, (Vol. 40.1) (1995) pp. 64-77.
- [5] Kolmanovsky I. and McClamroch N.H. Developments in nonholonomic control problems. *IEEE Control Systems Magazine*, (Vol. 15) (1995) pp. 20-36.
- [6] D'Andrea-Novel B., Campion G. and Bastin G. Control of nonholonomic wheeled mobile robots by state feedback linearization. *The International Journal of Robotics Research*, (Vol. 14) (No. 6) (1995) pp. 543-559.
- [7] De Luca A. and Di Benedetto M.D. Control of nonholonomic systems via dynamic compensation. *Workshop on system structure and control*. (No.2) (Vol. 29) (No 6) (1993) pp. 593-608.
- [8] Fliess M., Levine J. and Martin P. Flatness and defect of non-linear systems: introductory theory and examples. *International Journal of Control*, (Vol. 61), (No. 6) (1995) pp. 1327-1361.
- [9] Morin P. and Samon C. Practical stabilization of drift-less systems on Lie groups: the transverse function approach. *IEEE transactions on automatic control*, (vol. 48), (no 9) (2003) pp. 1496-1508.
- [10] Barfoot T.D. and Clark C.M. Motion Planning for Formations of Mobile Robots. *Robotics and Autonomous Systems*, (Vol. 46.2) (2004) pp. 65-78.
- [11] Jang J.-S.R. and Sun C.T. Neuro-fuzzy

modeling and control. *Proceedings of the IEEE*, (1995) pp. 378-406.

- [12] J. Godjevac. A Learning Procedure for a Fuzzy System: Application to Obstacle Avoidance. In *Proceedings of the International Symposium on Fuzzy Logic*, (1995) pp. 142-148.
- [13] Pascal M. and Claude S. Motion Control of Wheeled Mobile Robots. *Handbook of Robotics*, S. Bruno, K.S Oussama (Eds), Springer-Verlag Berlin Heidelberg, (2008) pp. 799-825.
- [14] T. Wang, C. Sabourin and K. Madani. ANFIS controller for nonholonomic robots. *Proceedings of International Conference on Neural Networks and Artificial Intelligence (ICNNAI 2010)*, Brest, Byelorussia, June 1-4, 2010, pp. 320-325. ISBN: 978-985-6329-79-4.
- [15] K. Madani, C. Sabourin. Multi-level cognitive machine-learning based concept for human-like "artificial" walking: application to autonomous stroll of humanoid robots. *Neurocomputing*, ISSN 0925-2312, Elsevier Publisher, Special Issue on Linking of phenomenological data and cognition. In Press (DOI N°10.1016/j.neucom.2010.07.021).



Dr. Christophe Sabourin graduated in Electrical Engineering in 1992 from University of Poitiers, France, he received his MSc. in Automation and Computer Science from this same University in 1993. He received his Ph.D. in Robotics and Control from University of Orleans, France in November

2004. In September 2005, he joined Senart-FB Institute of Technology of University PARIS-EST / PARIS 12 where he works as Assistant professor in Electrical Engineering department of this institute. Since 2005, he is researcher and a staff-member of SCTIC research division, one of the two research components of Images, Signals and Intelligent Systems Laboratory (LISSI / EA 3956) of University PARIS-EST Creteil. His current interests relate areas of complex and bio-inspired intelligent artificial systems, humanoid robotics, collective and social robotics.



Prof. Kurosh Madani received his Ph.D. degree in Electrical Engineering and Computer Sciences from University PARIS XI, Orsay, France, in 1990. From 1989 to 1990, he worked as assistant professor at Institute of Fundamental Electronics of PARIS XI University, Orsay,

France. In 1990, he joined Creteil-Senart Institute of Technology of University PARIS-EST Creteil (UPEC), Lieusaint, France, where he worked from 1990 to 1998 as assistant professor. In 1995, he received the DHDR Doctor Habilitate degree (senior research Dr. Hab. degree) from UPEC. Since 1998 he works as Chair Professor in Electrical Engineering of Senart Institute of Technology of UPEC. From 1992 to 2004 he has been head of Intelligence in Instrumentation and Systems Laboratory (I2S / JE 2353). Co-initiator in 2005 of Images, Signals and Intelligent Systems Laboratory (LISSI / EA 3956), he is head of one of the two research groups of LISSI. He has worked on both digital and analog implementation of processors arrays for image processing, electrooptical random number generation, and both analog and digital ANN implementation. His current research interests include large ANN structures modeling and implementation, hybrid neural based information processing systems and their software and hardware implementations, design and implementation of real-time neurocontrol and neural based fault detection and diagnosis systems. Since 1996 he is a permanent member (elected Academician) of International Informatization Academy. In 1997, he was also elected as Academician of International Academy of Technological Cybernetics.



Ting Wang received her Master Degree of Mechanic and Electrical Engineering in 2006 from Northwestern Polytechnical University (NWPU), Xian, China. She integrated the LISSI Laboratory – EA 3956 of University PARIS-EST Creteil (UPEC), in 2008 as PhD student.

Since October 2008, she continue her doctoral studies under Prof. Kurosh Madani's supervision. Her doctoral research topic deals with collective intelligence and collective robotics.



Fabien Gautero received his DUT. Degree in Electrical Engineering and Computer Science in 2010 from Senart-Fontainebleau Institute of Technology, of University PARIS-EST Creteil (UPEC). He has integrated the Prof. Kurosh Madani's research team at LISSI Laboratory – EA 3956 of UPEC for

an internship from April to July 2010, where he worked on implementation of Bio-inspired Control strategies on real Khepera III robots under supervision of Dr. Christophe Sabourin and Prof. Kurosh Madani.