Appl. Math. and Comp. Sci., 1996, vol.6, No.3, 581–594

SELF-ORGANIZING NEURO-FUZZY CONTROL OF COMPLEX SYSTEMS

TOĂO FABRO*, FERNANDO GOMIDE*

An autonomous system control creates a need for classes of control systems whose behaviour should emerge as a consequence of its interaction with the environment. Autonomous systems must be able to adapt continuously to new and unpredictable situations and to be successful in accomplishing their tasks. In this paper, a self-organizing, neuro-fuzzy control architecture for complex systems is presented. The emphasis is on an autonomous-vehicle navigation problem that has been recognized to be of considerable challenge. The aim is to find target positions without colliding with obstacles within an unknown environment. The architecture combines neural networks and fuzzy systems with the theory of neuronal group selection to acquire navigation skills. Neuro-fuzzy sensor information builds up adaptive fields whose intensity triggers fuzzy control actions in response to the environment characteristics. The control system develops emergent, adaptive behaviour from the interactions between the vehicle, environment, and learning strategies. Simulation results show that the control system is able to learn efficiently navigation strategies, to re-adapt in different environments and to perform better than alternative schemes.

1. Introduction

The aim of this work is to develop a self-organizing, neuro-fuzzy control approach to complex systems, with emphasis on the autonomous-vehicle navigation in unknown environments.

Autonomous navigation is essentially a trajectory control problem. In general terms, the control system must execute a given task, such as reaching a target, while avoiding obstacles. The main difficulties lie in the multiplicity of distinct relative positions of the vehicle. To establish appropriate decisions in any situation, the control system must either recognize each distinct position, which may rapidly lead to memory exhaustion, or be able to generalize.

The intrinsic difficulties of the autonomous navigation problem have captivated many artificial-intelligence researchers who have found it to be of considerable challenge. Navigation control of autonomous mobile vehicles is a research area that can be roughly divided into two main approaches: path planning, and sensor-based navigation.

Path planning is based on environment knowledge, and many approaches, ranging from mathematical analysis and path calculations (Lozan-pérez and Wesley, 1979), to

* Unicamp/Fee/Dca C.P. 6101, CEP 13.083-970 Campinas – SP, Brazil,
e-mail: gomide@dca.fee.unicamp.br

symbol manipulation on a knowledge base about the environment (Fikes *et al.*, 1972), are available. These methods can solve the path planning problems for completely known environments, and with off-line simulations. When facing real-time situations and unknown environments, or dynamically changing environments, these methods cannot be used. To overcome these difficulties, methods considering real-time environment information from sensors must be considered. Based on sensor readings, the mobile vehicle should be able to perform local path planning and to take appropriate control actions. Borenstein and Koern (1989) introduced the *virtual force field method* to solve this problem. However, their method has problems in finding force coefficients in cluttered environments which cannot be described by a mathematical model (Borenstein and Koren, 1991). Brooks (1986) presented a behaviour-based approach, called the *subsumption architecture*, which is based on pre-specified behaviour encoded in task-achieving modules. This architecture has succeeded in navigating in unknown environments, but it depends highly on the pre-defined knowledge structures implemented by each module. The success of this approach depends on how completely the behaviour can be described beforehand.

Many other approaches have recently been developed, mainly by using fuzzy sets and neural networks. The fuzzy set approach has the advantage of treating uncertainty and imprecision through simple rule bases (Ishikawa, 1991). The knowledge must also be provided in the form of fuzzy *if-then* rules. However, even after rule definition and refinement, it is generally difficult to treat all possible cases with specific rules. To overcome these difficulties, neural networks have been used. The main advantage of the neural network approach is that there is no need for knowledge programming. For instance, by using error back-propagation neural networks and a set of training patterns (Kozakiewicz and Ejiri, 1991), it is possible to *train* a vehicle to navigate in several environments (Sekiguchi *et al.*, 1990). However, when there are contradictory situations, training is difficult. The system is not well-prepared for certain changes in the environment conditions.

In an attempt to unify the best of path planning, sensor based navigation, fuzzy logic and neural networks, Beom and Cho (1995) have recently introduced a control system based on a reinforcement learning scheme to tune a fuzzy rule base, and to obtain adaptive behaviour during interaction with the environment. There is no knowledge pre-programming of actions in this approach, and reinforcement training should be performed to tune the control system properly.

A method based on neural models, developed by Verschure *et al.* (1992) and called distributed adaptive control (DAC), introduces new capabilities to adapt in unknown environments. The DAC control architecture can learn to navigate in an environment through interaction, with no need for pre-programming or definition. This architecture is based on a learning-by-interaction scheme, and produces emergent behaviour during the learning process. The main DAC problems are related to its performance in cluttered environments and in difficult situations found in real-life navigation problems.

In this paper, a self-organizing, neuro-fuzzy control architecture for complex systems, based on the DAC model introduced by Verschure *et al.* (1992), is proposed. The concept of neuro-fuzzy sensors (Gomide and Rocha, 1992) and a set of basic motor actions (Edelman, 1987) are used here to assemble a control architecture which is able to learn control strategies on a learning-by-interaction basis (Fabro, 1996). Neuro-fuzzy sensor information builds up adaptive fields whose intensity triggers fuzzy control actions derived from basic motor behaviour and environment characteristics. Simulation experiments have shown that the control system performs better when compared with alternative schemes, see e.g. (Oliveira *et al.*, 1995; Verschure *et al.*, 1992). Here the results provided by the proposed architecture are compared with the DAC scheme presented in (Verschure *et al.*, 1992).

The paper is organized as follows. After this brief introduction, the next section states the autonomous navigation problem and the characteristics of the vehicle. Section 3 describes the details of the system architecture including the learning procedure, fuzzy sensors, and fuzzy control rules which are assigned to the basic behaviours. In Section 4, computational results are presented and a comparison is made between the proposed system behaviour and the DAC approach presented by Verschure *et al.* Finally, the conclusions and future work to be pursued are addressed.

2. Navigation Control for Mobile Vehicles

Typically, in control problems of autonomous vehicles in unknown environments, the goal is to reach specified environment (target) positions, without colliding with obstacles or walls.

The vehicle model addressed below is similar to that presented in (Verschure *et al.*, 1992), except for the positioning of the target sensors. It interacts with the environment through three kinds of sensors: collision, target position and distance-to-obstacles ones. The sensors are distributed on the vehicle in the following manner: the collision and the distance sensors cover the region between $-90°$ and $+90°$ from its centre. Sensors positioned from $0°$ (in front of the vehicle) to $33°$ are placed every $3°$. This spacing is increased proportionally to the *vision angle* of the sensor, as can be seen in Fig. 1. There are collision and distance sensors at each of 39 positions; the distance sensors provide readings proportional to the distance from the sensor to the nearest obstacle or wall; there are two target sensors, positioned at $-90°$ and $+90°$ respectively. Each of these sensors furnishes readings proportional to its distance to the target. Based on these readings, the control system can estimate the position of the target, either on the left or on the right side.

The basic control actions are: move a determined number of steps ahead; turn a determined angle to the right or left; stop and move back a determined number of steps. Combining the basic control actions, the vehicle must be able to find target positions while avoiding any obstacle in its path.

3. System Architecture

The distributed adaptive control (DAC) structure is based on the theory of adaptive fields and has been introduced by Verschure *et al.* (1991). This theory assumes that an organism is capable of perceiving a set of stimuli from the environment, and to

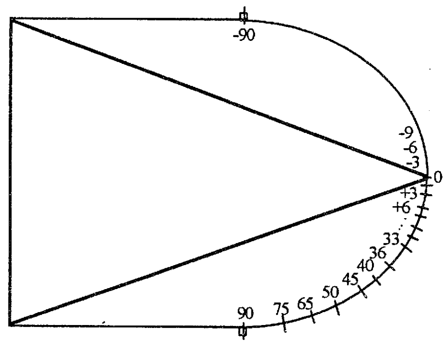


Fig. 1. Collision and distance sensor positioning.

modify its own behaviour to adjust to the environment characteristics. In this case, the system objective is to learn how to find the target positions without colliding with any obstacle. In the beginning, the control system has no information about the environment. The only information that can be used by the control system is that provided by the collision and target sensors. Therefore, initially there is no way to avoid a collision with the obstacle unless the collision possibility is detected in advance and the vehicle moves a step back and turns. During this process of interaction with the environment, the system acquires information and uses it to learn when to turn and avoid a collision, or when to turn and reach a target. This learning process is based on neural group interaction.

3.1. Neural Groups

There are three neural groups (Edelman, 1987) that enter into the structure of the control system. Each neural group is a set of independent neurons, each connected to a sensor. The three groups are respectively called the Distance Detection Group, Collision Detection Group and Target Detection Group. Connections are established between each output of the distance group and each input of the collision group. This provides a fully-connected two-layer neural network, whose connection weights can be modified. In the same manner, connections are set between the target and the distance group. The interaction between the target and the collision groups is managed by an Inhibitor Group. This group receives its input from both the target and collision group, and performs a composition of the corresponding activation levels to evaluate the action to be taken.

The DAC system (Verschure *et al.*, 1992) has the advantages of learning and adapting to any environment with which it interacts. However, in complex interactions with the environment, and with random positioned target positions, the DAC does not present high accuracy in avoiding obstacles and has learning difficulties. On the other hand, when new collision and target sensors, or fuzzy sensors in brief, and a new set of basic control actions using fuzzy set theory techniques are introduced, the advantages of learning and adaptation to any environment are retained with higher accuracy and better learning capabilities. The schematic diagram of the proposed control system architecture is illustrated in Fig. 2.

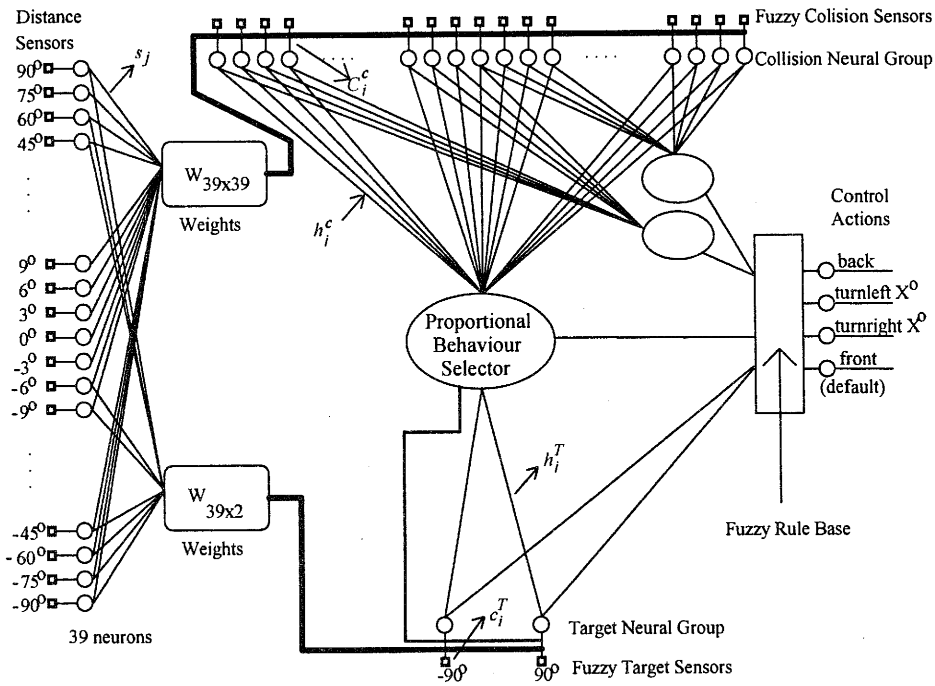


Fig. 2. Control system architecture.

The interaction between the neural groups, the control blocks, and the environment induces the emergent, adaptive behaviour of the mobile vehicle. By continuously changing the inter-group connection weights, the system evolves and adapts to the situations found in the environment.

3.2. Learning Mechanisms

At each vehicle/environment interaction step, all sensors' readings are updated, and a learning step takes place. Each learning step consists in changing the connection weights between the neural groups.

For the collision and target neural groups, the output h_i of each neuron is given in the following form:

$$h_i^\lambda = c_i^\lambda + \sum_{j=1}^{N} K_{ij}^\lambda s_j \tag{1}$$

where λ denotes the neural group, which can be C for the collision group or T for the target group. The quantity c_i^λ denotes the sensor input to which the neuron is directly connected and N is the number on neurons of the neural group. Moreover, K_{ij}^λ denotes the connection weight between the output of neuron j of the distance group (s_j) and the input of the neuron i of the neural group λ. The weights are updated according to the rule

$$\Delta K_{ij}^{\lambda} = \frac{1}{N}\left[\eta^{\lambda} s_i^{\lambda} s_j - \varepsilon s^{-\lambda} K_{ij}^{\lambda}\right] \tag{2}$$

where η^{λ} denotes the learning rate, ε is the decay rate, and $s^{-\lambda}$ stands for the average activation of the group λ. This parameter introduces an *active decay* that takes place just when there is something to learn, i.e. the neural group is active. The active decay provides the continuous learning when interacting with the environment. Learning takes place only when interaction reveals relevant information, e.g. the vehicle collides or a collision is predicted. Without this decay mechanism, the system would not be able to adapt in the case of changes in the environment (Verschure *et al.*, 1992).

3.3. Distance Detection Sensors

The distance sensors must provide, on each move of the vehicle, distance measures to the obstacle in the direction of the sensor reading. These measures are inputs to the distance detection neural group. The output (s_j) of each neuron in this group is given by the inverse of the corresponding distance sensor (r_j). Therefore, these neuron outputs code a measure for the *time to contact* (Lee, 1976). The transduction function is

$$s_j = 1/e^{x(r_j - d_{\max})} \tag{3}$$

where $d_{\max}$ is the distance at which the sensor will give a maximal response, and x is a scaling factor.

3.4. Fuzzy Collision Sensors

Usually, the collision sensors are bumpers. This type of sensor can produce only binary outputs which are active when collided, and inactive otherwise. To improve the system performance in avoiding collisions, we introduce a new kind of sensor, using the fuzzy set theory (Gomide and Rocha, 1992). The fuzzy collision sensor substitutes the on/off output of the usual collision sensors to provide a fuzzy value which represents a *measure of time to collide*. These sensors can be implemented using the fuzzy sets shown in Fig. 3.

Therefore, each fuzzy collision sensor provides a *collision degree*, according to the obstacle proximity. The obstacle proximity measure is provided by the distance sensors. The collision sensor outputs are used by the control system to update the weights between the collision and distance-to-obstacles neural groups, and by the rule base to trigger control actions concerning the situation.

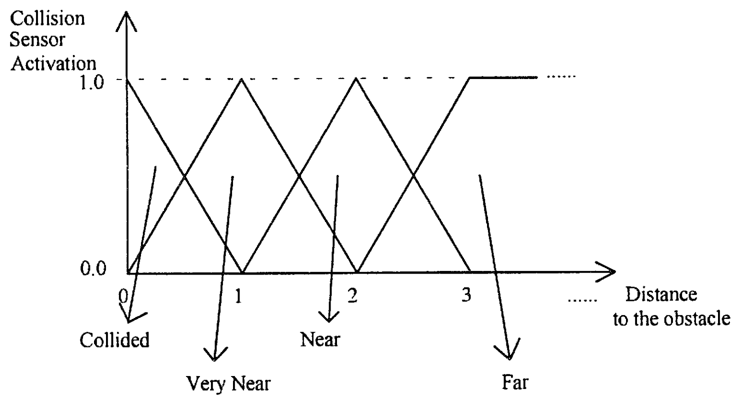


Fig. 3. Fuzzy sets for collision sensors.

3.5. Fuzzy Target Sensors

There are just two target sensors placed at -90° and $+90^{\circ}$ from the vehicle centre. The sensor output gives an estimate of the distance to a target. From the two measurements, the control system can infer the side to turn and reach the target. Fuzzy target sensors are used to provide information about the target proximity. Therefore, if the target is on the left, but is far, then the vehicle turns just a little in that direction. On the other hand, if the target is near, then the control action must be stronger, and the vehicle must turn quickly to reach it. Target information is also used by the fuzzy control rules. The fuzzy sets are as shown in Fig. 4.

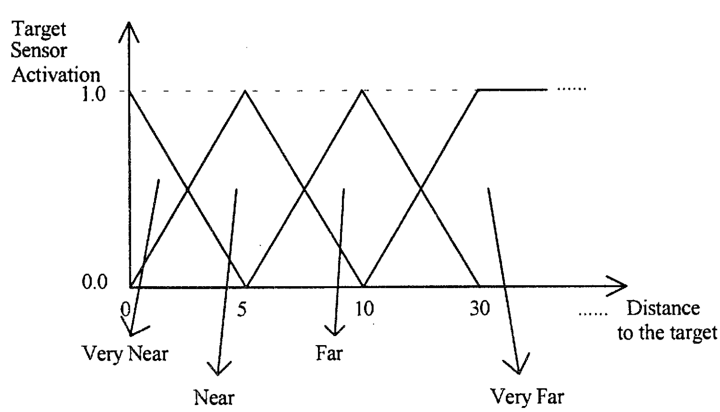


Fig. 4. Fuzzy sets for target sensors.

3.6. Proportional Behaviour Selection

To ensure the global performance of the system in finding target positions without colliding, even if targets are very close to the obstacles, a *proportional behaviour selector* or a *proportional cross-inhibition* is proposed. This selector takes its decisions based on measurements of *activation* of the target and collision neural groups as follows: when the collision group is more active, and the target detection group is inactive, the system must avoid obstacles, independently of target position. This situation occurs when the vehicle is near to the obstacles, but far from the target. When the target detection group is more active, and the collision group is not active, the system must go towards the target, independently of obstacles. This situation occurs when the vehicle is closer to the target than to any obstacle; but when both the target and collision groups are active, the system must find a *compromise solution* between the actions to be taken, based on the activation levels of each neural group. This situation occurs, e.g., when the target position is very close to an obstacle or a wall.

The formulae to compute the activation levels of each group are as follows:

- The collision group activation level (α_c):

$$\alpha_c = \frac{1}{m_c} \sum_{n=1}^{m_c} s_n \tag{4}$$

where m_c is the number of neurons of the collision detection group, and s_n are the activation (output) levels of each neuron.

- The target detection group activation level (α_t):

$$\alpha_t = \mu \Big/ \sum_{n=1}^{m_t} \mathrm{dist}_n \tag{5}$$

where dist_n is a function that evaluates the distance between each target sensor and the target, and m_t is the number of neurons of the target detection group. The factor μ is used to scale the activation levels between collision and target groups. In the simulation experiment μ was set to 10, m_c to 39 and m_t to 2.

3.7. Fuzzy Control Rules

The original DAC formulation uses a pre-wired neural network to relate output patterns from the collision and target detection groups, with very simple crisp control actions. These actions are: go ahead (the default action), go back, and turn to the left or to the right by a constant angle (typically 9°).

In the approach proposed here, the basic control actions are specified by a set of fuzzy control rules (Pedrycz, 1993). The fuzzy rules are based on information given by the fuzzy sensors, and on information provided by the neural groups outputs. The fuzzy collision and target sensors furnish the information about the environment. From the collision neural group, the system gets the *position of the obstacle.* This information is obtained by finding the most active neuron. The neurons are associated with obstacle positions, and are characterized by the fuzzy sets shown in Fig. 5.

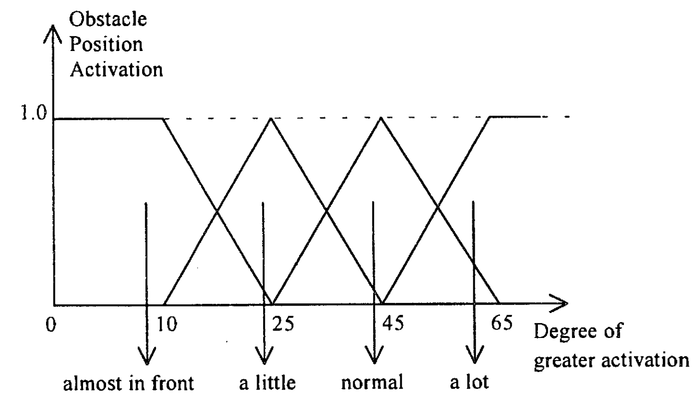


Fig. 5. Fuzzy sets for obstacle positioning.

With this information, the control rules for obstacle avoidance become very simple, e.g.

- If the obstacle is almost on the front left, then turn right a lot.
- If the obstacle is much to the right, then turn left a little.

Additional fuzzy control rules are used for collision avoidance, as illustrated by the following examples, where *collision sensor* and *obstacle* are fuzzy variables:

- If the collision sensor is collided and the obstacle is on the right, then move back and make a left turn.
- If the collision sensor is collided and the obstacle is on the left, then move back and make a right turn.
- If the collision sensor is very near and the obstacle is on the right, then turn left a lot.

Target reaching rules are as shown by the following examples, where *target* and *distance to target* are fuzzy variables:

- If the target is right and the distance to the target is near, then turn right a lot.
- If the target is left and the distance to the target is near, then turn left a lot.

Note that the target seeking and obstacle avoidance rules may be in conflict, and that is why the proportional behaviour selection scheme is necessary. The fuzzy controller uses the sup-mim compositional rule of inference, with centre of area deffuzification (Driankov *et al.*, 1993).

Let u_c and u_t be the deffuzified control actions derived from the fuzzy control rules for collision avoidance and target reaching, respectively. Thus the actual control action u to be adopted depends on the collision group and target detection group activation levels as found in Section 3.6. It is determined by

$$u = \frac{\alpha_c}{\alpha_c + \alpha_t} u_c + \frac{\alpha_t}{\alpha_c + \alpha_t} u_t \tag{6}$$

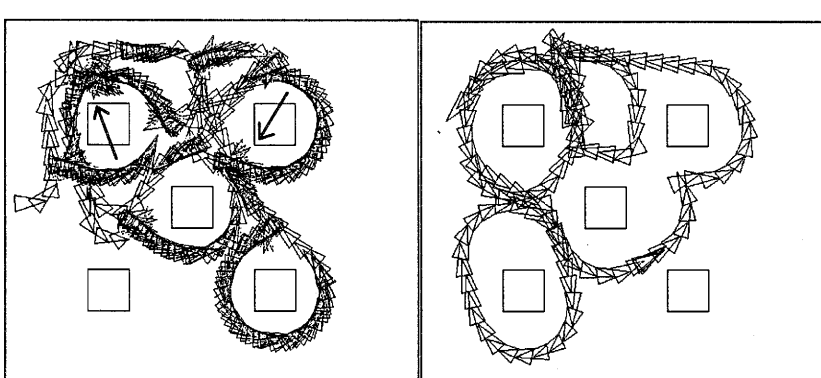
4. Computational Results

In this section, simulation results are presented to illustrate the performance of the proposed control architecture. It was assumed that the sensors could ideally detect the nearest obstacles within their ranges, and the target distances in any direction.

To run the experiments on, obstacles were positioned in the environment. Target positions were randomly generated. The vehicle had a prescribed number of steps to reach the target (300 steps), otherwise, the target was re-positioned. This random target positioning introduces a higher level of complexity during interactions because the system is always faced with various situations.

In the simulation experiments, the learning rate η^λ was set to 0.2, the decay rate ε to 0.8, $d_{\max}$ to 15, and x to 0.1.

Figure 6 shows the original DAC system performance (a) and the proposed neuro-fuzzy control performance (b). Each simulation took 10000 steps, and the trajectories shown were taken after an initial learning period (5000 steps). In Fig. 6(a) one can see the points (indicated by arrows) where the DAC system was not able to avoid a collision. The DAC system cannot generalize what was learned to all the situations found during the movement. In Fig. 6(b) a typical trajectory generated by the proposed system is shown. No collision occurred at all. The fuzzy techniques used in sensors and control actions provide higher accuracy and better learning capabilities.



(a) (b)

Fig. 6. Simulation results for the DAC (a) and neuro-fuzzy (b) control systems.

Figure 7 presents a typical performance of the DAC scheme during one complete simulation run. Observe the collisions that occur while the control system adapts to the environment. In Fig. 8, a complete simulation run of the proposed control system is presented. There are no collisions, even at the beginning of the iteration.

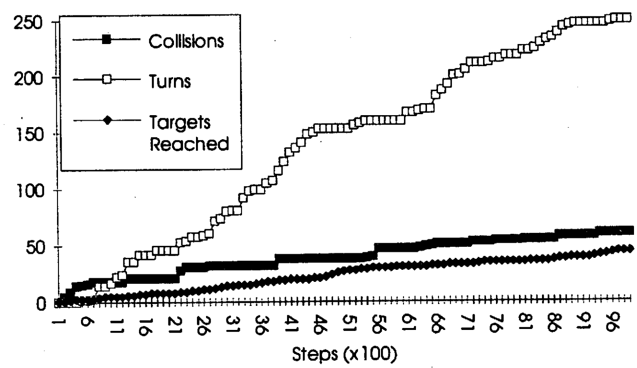


Fig. 7. Performance of the DAC system.

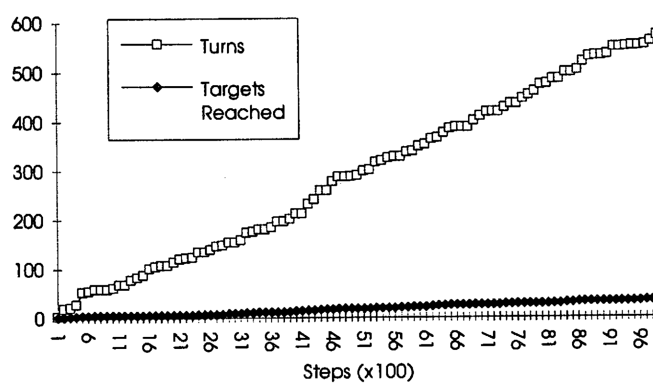
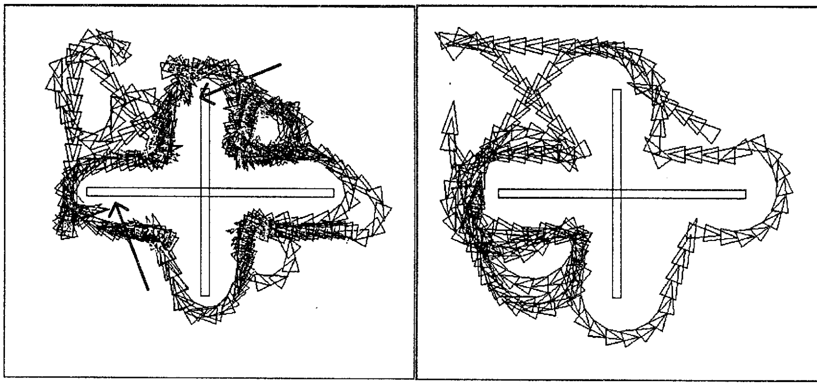


Fig. 8. Performance of the proposed neuro-fuzzy system.

After a number of additional simulations (with 10000 steps in each of them), the DAC got an average number of 30 targets, but was colliding 50 times on average. For the neuro-fuzzy approach, the target-reaching average was also 30, but the system did not collide at all. At the beginning of each experiment, the fuzzy sensors provide information rich enough to avoid collisions. As the neural groups learn through their interaction with the environment, the performance of the system in avoiding obstacles and finding targets increases further.

Immediately after the simulation experiments described above were performed, the vehicles were put into a completely different environment, as shown in Fig. 9. The behaviour of the DAC (a) and of the proposed neuro-fuzzy (b) control systems are



(a) (b)

Fig. 9. Simulation results.

also shown. Both systems performed well when re-adapting to the new environment, but the DAC system (see Fig. 10) did allow new collisions to occur to keep its adaptive behaviour during this process. As shown in Fig. 11, the proposed system still avoids collision during the re-adaptation. Therefore it becomes clear that the generalizing capabilities of the system developed herein are much better than those provided by the DAC approach.

5. Conclusions

Autonomous system control possesses considerable challenges for control system designers because its behaviour, in contrast to traditional set-point or disturbance rejection control, cannot be fully specified in advance. Autonomous systems must be able to adapt continuously to new, unpredictable situations to achieve their tasks. This requirement creates a need for a class of control systems whose behaviour, given the objectives to be fulfilled, emerges from its interaction with the environment.

In this work, a self-organizing, neuro-fuzzy control architecture was developed, aiming at solving autonomous control problems, with the focus of attention on navigation of autonomous vehicles in unknown environments. The task of the control system was to reach randomly generated target positions while avoiding obstacles. The main paradigms behind the proposed architecture are the adaptive-field concept of the theory of neuronal group selection, coupled with the theory of fuzzy sets. By introducing fuzzy sensors and basic fuzzy-control actions, the navigation problem was solved efficiently. In addition, the architecture developed allows the adaptive behaviour to emerge. An instance of emergent behaviour relates to the case of avoiding collisions with obstacles, even if the vehicle is put in different environments. In this case, the system was able to make collision avoidance decisions, as a result of training. Clearly, target seeking behaviour was also developed through adaptation. Simulation results show that the control architecture proposed herein performs better when compared with the DAC architecture. Due to its structure and adaptation characteristics, it is quite reasonable to expect that the proposed approach may also

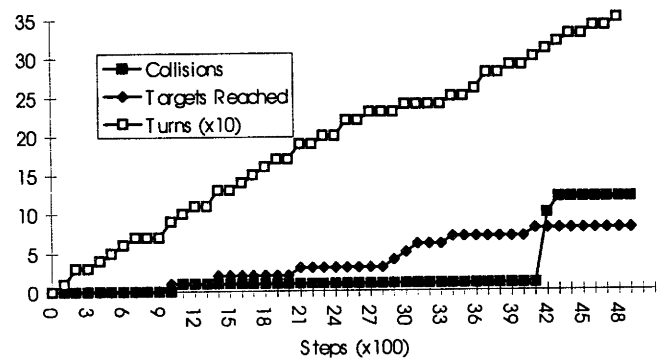


Fig. 10. DAC performance after changing the environment.

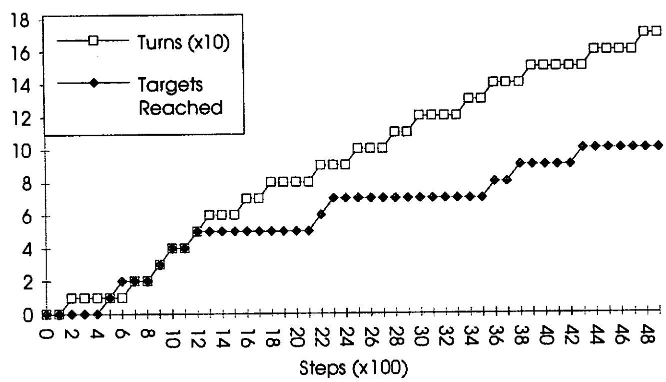


Fig. 11. Proposed system performance after changing the environment.

have a competing performance when compared with alternative control architectures (see e.g. Beom and Cho, 1995; Brooks, 1986; Figueiredo and Gomide, 1995; Oliveira *et al.*, 1995). Further work concentrates on actual implementation of the architecture to verify its robustness in real situations.

Acknowledgments

The first author acknowledges CAPES-PET for his fellowship, and the other the CNPq for grant # 300729/86-3. The support of ESPRIT-ECLA005 is also acknowledged.

References

Beom H.R. and Cho H.S. (1995): *A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning.* — IEEE Trans. Sys. Man Cyber., v.25, No.3, pp.464–477.

Borenstein J. and Koren Y. (1989): *Real-time obstacle avoidance for fast mobile robot.* — IEEE Trans. Sys. Man Cyber., v.19, No.5, pp.1179–1187.

Borenstein J. and Koren Y. (1991): *Potential field method and their inherent limitations for mobile robot navigation.* — Proc. IEEE Int. Conf. *Robotics and Automation*, Sacramento, CA., Apr.9–11, pp.818–823.

Brooks R.A. (1986): *A robust layered control system for a mobile robot.* — IEEE J. Robotics Automat., v.RA-2, No.1, pp.14–23.

Driankov D., Hellendoorn H. and Reinfrank M. (1993): *An Introduction to Fuzzy Control.* — Berlin: Springer-Verlag.

Edelman G.M. (1987): *Neural Darwinism: The Theory of Neuronal Group Selection.* — New York: Basic Books.

Fabro J.A. (1996): *Neural Groups and Fuzzy Systems — Applications to Autonomous Navigation.* — M.Sc. Thesis, UNICAMP/FEE/DCA, (in Portuguese).

Figueiredo M. and Gomide F. (1995): *Evolving neurofuzzy networks for basic behaviours and a recategorization approach for their coordination*, (to be published).

Fikes R.E., Hart P.E. and Nilson N.J. (1972): *Learning and executing generalized robot plans.* — Artificial Intelligence, v.3, pp.251–288.

Gomide F. and Rocha A. (1992): *Neurofuzzy components based on threshold.* — Prep. IFAC Symp. *Intelligent Components and Instruments for Control Applications*, Malaga, Spain, pp.425–430.

Ishikawa S. (1991): *A method of indoor mobile robot navigation by fuzzy control.* — Proc. Int. Conf. *Intelligent Robots and Systems*, Osaka, Japan, pp.1013–1018.

Kozakiewicz C. and Ejiri M. (1991): *Neural network approach to path planning for two dimensional robot motion.* — Proc. Int. Conf. *Intelligent Robots and Systems*, Osaka, Japan, pp.818–823.

Lee D.N. (1976): *A theory of visual control of braking based on information about time to collision.* — Perception, v.5, pp.437–459.

Lozan-pérez T. and Wesley M.A. (1979): *An algorithm for planning collision-free paths among the polyhedral obstacles.* — Communications of the ACM, v.22, No.10, pp.560–570.

Oliveira M.A., Figueiredo M., Gomide F. and Romero L. (1995): *Neurofuzzy navigation control and neuronal group selection.* — Proc. 6th IFSA World Congress, São Paulo, Brazil, pp.22–28.

Pedrycz W. (1993): *Fuzzy Control and Fuzzy Systems*, (2nd Edition). — New York: RSP Ltd.

Sekiguchi M., Nagata S. and Asakawa K. (1990): *Mobile robot control by a structured hierarchical neural network.* — IEEE Control Systems Mag., v.10, No.3, pp.69–76.

Verschure P.F.M.J. and Colen A.C. (1991): *Adaptive fields: distributed representations of classically conditioned associations.* — Network-Computation in Neural System, v.2, pp.189–206.

Verschure P.F.M.J., Pfeifer R. and Kröse B.J.A. (1992): *Distributed adaptive control: The self-organization of structured behaviour.* — Robots and Autonomous Agents, v.9, pp.181–196.