Intelligent autonomous navigation for mobile robots: spatial concept acquisition and object discrimination — Source link ↗

Eric Aislan Antonelo, M. Figueiredo, Albert-Jan Baerveldt, Rodrigo Calvo

Institutions: Halmstad University, University of São Paulo

Related papers:

- A neural-network based autonomous navigation system using mobile robots

- Applying topological maps to accelerate reinforcement learning in mobile robot navigation

- A Hierarchical Self-Organizing Controller for Navigation of Mobile Robots

- Convolution Neural Network Based Autonomous Navigation System for Mobile Robots

- Hierarchical Reinforcement Learning for Robot Navigation using the Intelligent Space Concept

Intelligent Autonomous Navigation for Mobile Robots: Spatial Concept Acquisition and Object Discrimination[*]

Eric Aislan Antonelo

School of Information Science, Computer and Electrical Engineering Halmstad University

Halmstad, Sweden

d404eran@stud.hh.se

Mauricio Figueiredo

Department of Computer Science

State University of Maringá 87020-900, Maringá – PR, Brazil

mauricio@din.uem.br

Albert-Jan Baerveldt

School of Information Science, Computer and Electrical Engineering Halmstad University

Halmstad, Sweden

albert@ide.hh.se

Rodrigo Calvo

Department of Computer Science and Statistics

University of São Paulo

São Carlos, Brazil

rcalvo@icmc.usp.br

Abstract - **An autonomous system able to construct its own navigation strategy for mobile robots is proposed. The navigation strategy is molded from navigation experiences (succeeding as the robot navigates) according to a classical reinforcement learning procedure. The autonomous system is based on modular hierarchical neural networks. Initially the navigation performance is poor (many collisions occur). Computer simulations show that after a period of learning the autonomous system generates efficient obstacle avoidance and target seeking behaviors. Experiments also offer support for concluding that the autonomous system develops a variety of object discrimination capability and of spatial concepts.**

Index Terms – intelligent autonomous navigation, neural networks, reinforcement learning, mobile robots, biologically inspired models.

I. INTRODUCTION

Usually it is possible to understand mobile robot navigation as a problem of establishing trajectories, so that tasks (goals) are accomplished with acceptable performance.

Research about navigation systems has been done in several ways, depending on the characteristics of the environment, the robot model, the type of the task, and the performance criteria [1 and 11]. A class of navigation systems, the autonomous systems, has captivated the scientific community not only because of the challenge involved but also because of the strategic importance.

Such systems determine the robot trajectory in an unknown environment without external help. In this context, navigation systems learn their own navigation strategy from the environmental interactions (based on their own experiences) [1].

The autonomous robot navigation imposes strict difficulties to the traditional control engineering techniques due to the lack of a suitable environmental model. Computational intelligence based strategies have been disseminated as a viable alternative, e.g. neural networks and evolutionary systems. Important results have been achieved and impelled the research of the perspective of intelligent autonomous systems [2, 4, 5 and 7].

It is intrinsic to neural networks the potential for learning, becoming them very attractive to autonomous navigation systems projects [2 to 6]. Proposals in [4] and [5]

present the behavior-based approach associated with the (classical) reinforcement learning. Two repertoires reproduce instinctive behaviors of target seeking and obstacle avoidance (basic behaviors). At first these repertories work independently, generating conflicting behaviors, e.g. if an obstacle is between the robot and the target. A third repertory is assigned for coordination of the innate behaviors. Initially the performance frustrates the expectations, since the autonomous system does not have an expert knowledge about navigation. As the navigation proceeds, environment interactions provide the bases for a reinforcement learning strategy. In the end the navigation system guides efficiently the robot to the target.

In a different way, the system proposed in [7] does not have a behavior-based architecture, that is, there are not instinctive behaviors a priori. The classic reinforcement learning is integrated to a learning classifier system to conceive an evolutionary navigation system. The evolutionary system learns simultaneously obstacle avoidance and target seeking behaviors, and it learns to coordinate them as well.

Such as the aforementioned references, most research reports in the literature model the autonomous navigation problems considering robots are equipped with target and obstacle sensors, and learn to generate obstacle avoidance and target seeking behaviors. Here the problem model presents, at least, two innovative and curious aspects: a) there are different classes of objects in the environment, each of which associated with a respective color and; b) the sensors, to detect the colors of the objects as well as the distance from the objects to the robot, are not specific either for targets or obstacles. The autonomous system consists of a hierarchical and modular neural network with also innovations on the characteristics of neuron units and neuron connection arrangements. Exploiting navigation experiences, the autonomous system learns, becoming able to distinguish objects (generating specific behaviors for each one of the different classes of objects). Furthermore it learns to take navigation decisions to move the robot to the closer attractive objects (targets). Simulation results confirm the autonomous system capabilities.

This paper is organized as follows. Section 2 describes the robot model and the navigation environment. The

proposed autonomous navigation system is described in Section 3. Simulation results are analyzed in Section 4. Section 5 concludes this paper with a brief discussion about the characteristics and potentialities of the proposed system.

II. ROBOT MODEL AND ENVIRONMENT

Repulsive and attractive objects compose the navigation environment. Each object is of a particular color, coherently with its respective class (for instance, repulsive and attractive objects may exhibit blue and green colors, respectively). Obstacles also belong to the repulsive class.

The robot model is shown in Fig. 1. The robot interacts with the environment by distance, color and contact sensors; and by one actuator system that controls the movement inclusive the direction. The sensors are organized in 67 predefined positions in front of the robot (distributed at the interval of –90° a +90°). Thus, there are three distinct sensors for each one of the 67 predetermined positions. Each sensor provides specific information related to the closest object to the robot on the direction of the sensor (see Fig. 1). A distance sensor detects the distance between the robot and the closest object and emits values in the interval [0,1]. Color sensors also emit values in the interval [0,1] – the conversion is made by normalization of the "hue" component of the HSV system (Hue, Saturation, Value). Binary contact sensors detect the touches of the robot with any object in the environment.

The task of the robot is to reach attractive objects establishing a trajectory free of collision with obstacles. When the robot reaches an attractive object (a possible target), this object is removed from the environment (simulating a collector robot).

The velocity of the robot is constant during the navigation (0.35 distance units per iteration). At each iteration, the robot is able to execute no direction adjustment greater than 15°.

III. AUTONOMOUS NAVIGATION SYSTEM

A. Introduction

Biologic neuronal systems hold innate (instinctive) mechanisms (e.g., sucking, grasping; and hunger and fear sensations) to support the emergence of specific behaviors acquired from the interactions with the environment [8 and 9]. Innate behaviours are essential to the integrity and development of the animal, although many can be evoked only in a short period of time after birth. They are not associated to particular life experiences established with the environment, but are a priori incorporated in the nervous system, during the epigenesis.

Considering biologic systems as a reference, innate (pre-incorporated) behaviors hold a function equally preponderant in intelligent autonomous systems [10]. Under this interpretation, the design of the proposed system is conducted, either referring to its learning scheme or to its architecture.

B. Architecture

The intelligent autonomous system corresponds to a neural network arranged in three layers (Fig. 2). At the first layer, there are two neural repertoires: Proximity Identifier repertoire (PI) and Colour Identifier repertoire (CI). Both

repertoires receive stimuli from contact sensors. PI and CI repertoires receive stimuli from the two other sensor fields, distance sensors and colour sensors, respectively. The second layer comprehends two neural repertoires: Attraction repertoire (AR) and Repulsion repertoire (RR). Each one establishes connections with both networks at the first layer, as well as with contact sensors. The actuator network, connected to AR and RR repertoires, determines the direction adjustments.
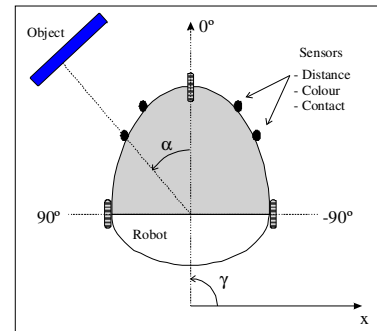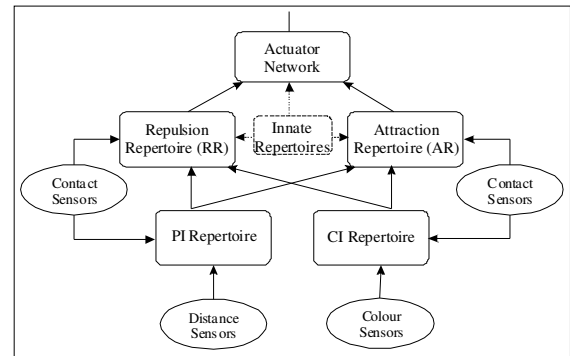

Fig. 1 Robot model.

Fig. 2 Autonomous system architecture.

PI and CI repertoires: The architecture of the neural network that composes each repertoire is presented in Fig. 3. Each column of neurons is fixed, topological, and one-dimensional structured. There is no spatial influence among neurons of different columns. Each neuron of a column establishes connections with a group of sensors defined according to a specific probability distribution (Fig. 3).

AR and RR Repertoires: A unique layer of neurons forms AR and RR neural networks. The size of each network corresponds to the number of columns in IP or IC networks. Each neuron in this layer connects to every neuron in the corresponding column of IP and IC networks, respectively (Fig. 4).

Actuator Network: There are two layers in the actuators network. Each neuron in the first layer is associated to a fixed and predefined direction adjustment value. Each neuron receives stimuli from neurons in RA and RR networks and it is also connected to the unique neuron in the output layer (Fig. 5).

Innate Repertoire: They are connected to the contact sensorial field (not shown in Fig. 1 to raise clarity), and to the AR, RR, and actuator networks.
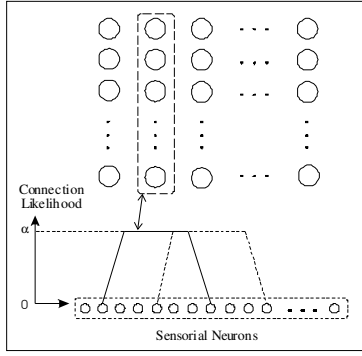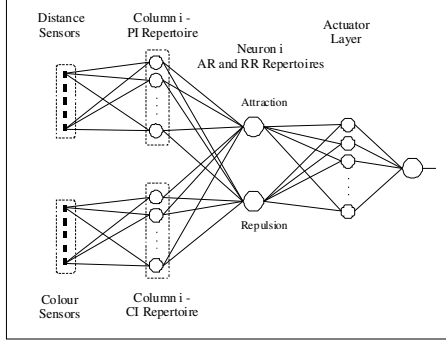
Fig. 3 PI and CI architectures.

Fig. 4 Architecture overview showing only one PI and CI column as well as the corresponding AR and RR neuron.
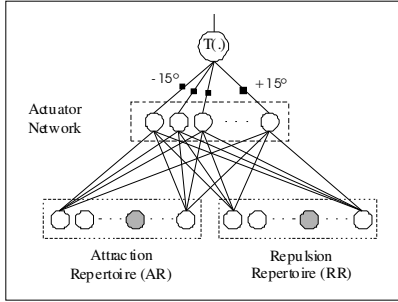

Fig. 5 Actuator Network.

C. Reasoning

Different neuron models are adopted to design the intelligent autonomous system. The particular characteristics determine which neural network they compose.

Repertoires PI and CI: The output of the jth neuron at the kth column of PI repertoire, at iteration n, is defined as in (1):

$$y_j^k(n) = \begin{cases} u_j^k(n) & \text{se } j = i^k(\mathbf{x}^k(n)) \\ 0 & \text{se } j \neq i^k(\mathbf{x}^k(n)), \end{cases} \quad (1)$$

where: $i^k(\mathbf{x}^k(n))$ is the winner neuron; $\mathbf{x}^k(n) = [x_1, x_2, ..., x_m]^T$ is the input vector (distance to objects); $u_j^k(n)$ is the activation potential (defined in Section *D. Learning*); and $k = 1,2,...q$ (q: numbers of columns). The winner neuron $i^k(\mathbf{x}^k(n))$ is defined by:

$$i^k(\mathbf{x}^k(n)) = \arg\min_j \left\| \mathbf{x}^k(n) - \mathbf{w}_j^k(n) \right\|, \ j = 1,2,...,l, \quad (2)$$

where: $\mathbf{w}_j^k(n)$ is the vector of synaptic weights and l is the number of neurons in each column.

Besides $u_j^k(n)$, each neuron exhibits another inner parameter: $e_j^k(n)$ (degree of activity), defined as:

$$e_j^k(n+1) = \begin{cases} e_j^k(n) + \varphi e_j^k(n) & \text{se } j = i^k(\mathbf{x}^k(n)) \\ e_j^k(n) - \sigma e_j^k(n) & \text{se } j \neq i^k(\mathbf{x}^k(n)), \end{cases} \quad (3)$$

where: φ is the gain factor and σ is the loss factor.

Equation (4) defines the output of the jth neuron at the kth column of the CI repertoire:

$$y_j^k(n) = \begin{cases} 1 & \text{se } j = i^k(\mathbf{x}^k(n)) \\ 0 & \text{se } j \neq i^k(\mathbf{x}^k(n)), \end{cases} \quad (4)$$

where $i^k(\mathbf{x}^k(n))$ is defined in the same manner as in (2).

AR and RR Repertoires: The output of the jth neuron of both AR and RR repertoires is given by:

$$y_j(n) = \begin{cases} 0 & \text{se } s_j <= \phi \\ \alpha(s_j - \phi) & \text{se } \phi < s_j < (\phi + 1/\alpha) \\ 1 & \text{se } s_j >= (\phi + 1/\alpha), \end{cases} \quad (5)$$

where: $s_j = \mathbf{x}_j(n) \cdot \mathbf{w}_j(n)$ is the inner product between the input vector and the synaptic weights vector; α and ϕ are constants; and $j = 1, 2, ..., q$ (q: size of the neural network).

Actuator Network: The output of the jth neuron at the first layer of the actuator network is given by:

$$y_j(n) = g\left(\sum_{i=1}^{2q} x_{ji}(n) w_{ji} \right), \quad (6)$$

where: $g(.)$ is the hyperbolic function. Observe that there are $2q$ inputs corresponding to the total of $2q$ neurons in AR and RR networks.

The output of the actuator network has the following expression:

$$y = \sum_{j=1}^{r} w_j \bar{x}_j, \ \bar{x}_j = x_j / \sum_{p=1}^{r} x_p, \text{ and } w_j = 15(2j - r)/r, \quad (7)$$

where: r is the number of neurons at the first layer (w_j is constant and equal to some value in the interval [-15, 15]; and this interval corresponds to the possible adjustments on the movement direction).

D. Learning

The learning procedure develops according to the classic reinforcement learning theory and is based on the activity of innate repertoires. The main concepts involved are inspired by the biological counterpart [9]. Once a contact with objects is detected, the innate repertoire generates one of the innate behaviors (attraction or repulsion) and it starts the learning process. The learning mechanisms, which deal with adjustments on the synaptic weights, are specific for each neural repertoire and are described next.

PI and CI Repertoires: Each time a contact occurs (between the robot and objects) only the sensor (contact sensorial field) closest to the point of the contact detects it. The corresponding stimulus defines a unique column in CI and in PI networks (consider the kth column). The adjustment occurs on the synaptic weights from the selected column (there is a spatial correspondence between positions of contact sensors and columns in CI, as well as in PI networks).

The following procedures model the synaptic adjustment mechanisms on the PI repertoire:

1- Competition by Similarity: calculate the winner neuron $i(n)$ given by:

$$i(n) = \arg\min_{j}\left\|\mathbf{x}(n) - \mathbf{w}_j(n)\right\|,\ j = 1,2,...,l. \tag{8}$$

2- Adjustment:
i) Adjust the synaptic weight vectors of all neurons by applying (9):

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta h(i(n),j)(\mathbf{x}(n) - \mathbf{w}_j(n)), \tag{9}$$

where: η is the learning rate, and $h(i(n),j) = \exp\left(-(i(n)-j)^2/l\right)$ is the neighbourhood function;
ii) Adjust the following parameters for $j = 1,2,...,l$:

$$u_j(n+1) = u_j(n) + \eta h(i(n),j)(1 - u_j(n))e_j(n)/3$$
$$e_j(n+1) = 1. \tag{10}$$

Note that $u_j(n)$ is the activation potential in the neuron reasoning model and, $e_j(n)$ is also adjusted during reasoning, at each iteration n (see Section *C. Reasoning*).

The learning algorithm for the CI repertoire is described next:
1- Competition by Similarity: calculate the winner neuron $i(n)$ given by:

$$i(n) = \arg\min_{j}\left\|\mathbf{x}(n) - \mathbf{w}_j(n)\right\|,$$
$$j \in \left\{k / k \in \{1,2,...,l\}\, e\, \left\|\mathbf{x}(n) - \mathbf{w}_k(n)\right\| < m_k(n)\right\}. \tag{11}$$

where $m_k(n)$ is the acceptance parameter of the neuron k.

2- Adjustment: Adjust the synaptic weight vector and the acceptance parameter:

$$\mathbf{w}_{i(n)}(n+1) = \mathbf{w}_{i(n)}(n) + \eta(\mathbf{x}(n) - \mathbf{w}_{i(n)}(n))$$
$$m_{i(n)}(n) = \left\|\mathbf{x}(n) - \mathbf{w}_{i(n)}\right\|, \tag{12}$$

where η is the learning rate.

AR and RR Repertoires: The synaptic weights of the jth neuron (AR or RR) correlated to the inputs connected to the PI repertoire are all unitary (constant values). Differently, the synaptic weights associated with the inputs connected to the CI repertoire are adjustable.

Consider that neurons at the kth column of the CI repertoire have adjusted their synaptic weights. Then, the learning mechanism acts on the kth neuron of the AR network if the innate behavior generated is of attraction type. Otherwise, it acts on the kth neuron of the RR repertoire.

The synaptic weights adjustment is given by (13):

$$\mathbf{w}_k(n+1) = \mathbf{w}_k(n) + \eta\mathbf{x}(n)(1 - \mathbf{w}_k(n)), \tag{13}$$

where η is the learning rate.

Actuator Network: Only the synaptic weights of the neurons at the first layer of the actuator network are adjusted. The innate repertoire reproduces some of the innate behaviours (attraction or repulsion) when a contact between robot and some object occurs. To accomplish that, it coherently activates neurons (in the first layer of actuator network) to generate the respective behavior.

The adjustment of the synaptic weights of the jth neuron at the first layer of the actuator network is expressed in (14):

$$w_{ji}(n+1) = w_{ji}(n) + \eta y_j x_i(n)(1 - w_{ji}(n)), \tag{14}$$

where $y_j(n)$ is the output of the jth neuron, $x_i(n)$ is the ith input and η is the learning rate.

IV. SIMULATION RESULTS

This section shows simulation results that enable analysis on characteristics and potentialities of the proposed controller. The experiments are executed on a computational environment provided with suitable simulation tools developed specifically to test the proposed controller. Concerning simulation, PI and CI networks configurations are: structured in 25 columns with 12 neurons each; $e_j(n) \in$ [0, 2]; $\varphi = 0.028$ e $\sigma = 0.013$ and; $e_j(0) = 0.06$, $u_j(0) = 1$ e $m_j(0) = 50$. Regarding RR and AR networks: $\alpha = 1.3$ e $\phi = 1.1$. Every synaptic adjustment uses learning rate $\eta = 0.4$. Weights in PI and CI are initialised with random values. In the case of AR, RR and actuator network (first layer), the random values are restricted within the interval [0, 0.07]. The slope parameter for the hyperbolic function in (6) is 2.5.

The following figures show a global view of the navigation environment at some specific iterations and, in particular, the robot trajectories (a black line). The robot is represented by a semicircle. Repulsive and attractive objects are represented by dark and clear polygons, respectively (due to edition limitations it can be impossible to exhibit colour objects). Repulsive contacts (or collisions) are marked with clear circles. Attractive contacts are marked with black circles.

Before any environmental interaction, the robot does not have any particular behavior and the environment is unknown. The trajectories presented in Fig. 6 (a), (b) and (c) show progressive stages of the navigation strategy regarding repulsive objects. After several collisions (Fig. 6 (a)), the robot develops a more efficient trajectory (Fig. 6 (b)) where it drives suitably the vehicle inside large corridors (generalization of the acquired knowledge). In narrower corridors there are still some collisions since it is necessary better abilities to guide the robot. Fig. 6(c)) shows a trajectory with only two collisions. After interacting with different environments, the robot is placed in the environment at Fig. 6 (d), where there are attractive objects (targets). The robot needs touching two or three times each target until it reactively seeks them (each touching generates an attractive behavior and triggers a coherent learning process).

The subsequent experiments offer support for analysing the importance of PI and CI repertoires joint actuation in the autonomous system.

The environment at Fig. 7 (a) contains a target close to the robot and another far from it (on the left and the right, respectively). The trajectory exhibited at Fig. 7 (b) indicates that the robot behavior prioritises the closest object on its left. With the purpose to confirm this ability, the targets are re-positioned according to the situation at Fig. 7 (c). It is possible to observe in Fig. 7 (d) that the trajectory of the robot indicates the preference for the closest object (the closest objects appear neither in Fig. 7(b) nor in Fig. 7(d) because the robot captures them.
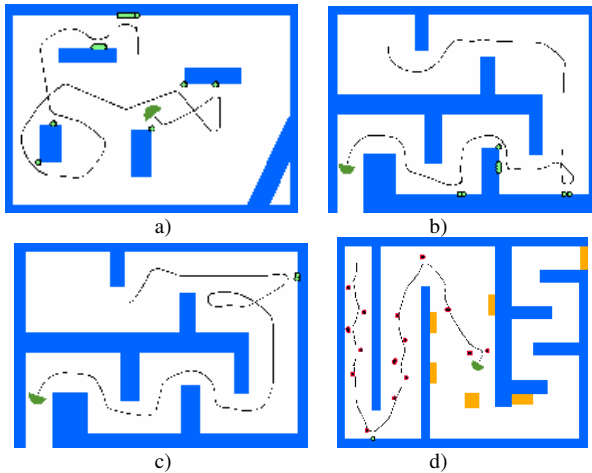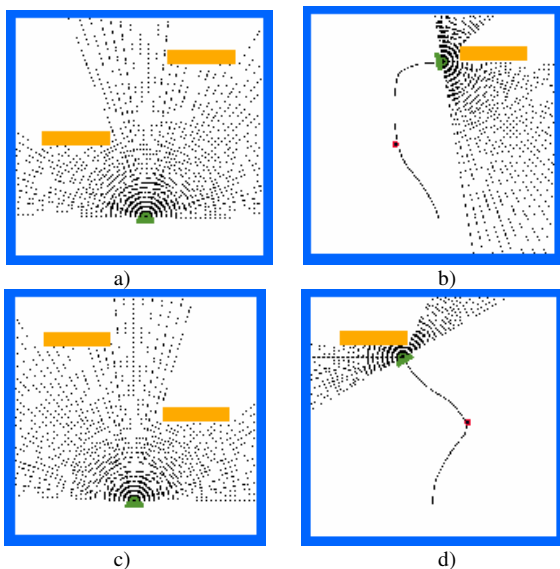

Fig. 6 Obstacle deviation and target seeking learning.

Fig. 7 Spatial discrimination.

V. CONCLUSION

Navigation systems with high degree of autonomy represent a field of great interest to the scientific community. This work proposes an autonomous navigation system that constructs its navigation strategy while it interacts with the environment. A neural network that functions in accordance with classical reinforcement learning theory composes the system. The neural network design tries to keep the biologic plausibility according to arguments asserted in [10], including neurophysiologic and psychological aspects [9].

The proposed system innovates in two aspects: it learns to discriminate distinct classes of objects (attractive and repulsive), exploiting the occult knowledge diffused in the stimuli coming from the environment; neuron models with inner dynamics provide the learning of spatial proximity concept, which makes possible the construction of a navigation strategy capable of prioritising the attractive closest objects.

Initially, the system is unable to distinguish attractive objects from repulsive ones as well as it is not able of prioritising the closest attractive objects. As navigation proceeds, the interactions with the environment (contacts with objects) occur, providing necessary conditions to the autonomous system continuously learns efficient attractive and repulsive behaviors. Simulation experiments consider general environments with different objects (attractive and repulsive) and show results that confirm the aforementioned system potentiality, as well as evidence the generalization capacity for navigation in distinct environments.

REFERENCES

[1] Figueiredo, M. (1999) "Autonomous Robot Navigation", Proceedings of VII SBC School for Informatics, Novo Hamburgo (in portuguese).
[2] Calvo, R., Figueiredo M., Antonelo, E.A. (2005) "Architecture control in constructive neural networks via classifier systems in autonomous navigation systems", 6th IEEE International Symposium on Computational Intelligence in Robotics and Automation, Helsinki, Finland, in press.
[3] Vershure, P. and Voegtlin, T. (1998) "A bottom up approach towards the acquisition and expression of sequential representations applied to behaving real-world device: distributed adaptive control III", Neural Networks, Vol. 11, pp. 1531 – 1549, 1998.
[4] Crestani, P.R., Figueiredo, M., and Von Zuben, F. (2002) "A hierarchical neuro-fuzzy approach to autonomous navigation", Proceedings of 2002 International Joint Conference on Neural Networks, (cd-rom), Honolulu, USA.
[5] Calvo, R. and Figueiredo M. (2003) "Reinforcement learning for hierarchical and modular neural network in autonomous robot navigation", Proceedings of 2003 International Joint Conference on Neural Networks – IJCNN, Oregon, USA.
[6] Millan, J. (1996) "Rapid, safe, and incremental learning of navigation strategies"; IEEE Transactions on Systems, Man, and Cybernetics, Vol. 26, no 3, pp.408-420, 1996.
[7] Cazangi, R. and Figueiredo, M. (2002) "Simultaneous emergence of conflicting basic behaviors and their coordination in an evolutionary autonomous navigation system", Proceedings of 2002 IEEE Congress on Evolutionary Computation, (cd-rom), Honolulu, EUA.
[8] Kandell, E.; Schwartz, J. and Jessel, T. (1991) "Principles of neural science", Elsevier, New York.
[9] Donahoe, J. W. & Palmer, D. C. (1994) "Learning and complex behavior", Allyn and Bacon, Needham Heights, Massachusetts, EUA.
[10] Edelman, G., (1987) "Neural darwinism: the theory of neuronal group selection", Basic books, New York, USA.
[11] Arkin, R.C. (1998) "Behavior-Based Robotics", The MIT Press.