Neuronal Architecture for Reactive and Adaptive Navigation of a Mobile Robot

Francisco García-Córdova[1], Antonio Guerrero-González[1],
and Fulgencio Marín-García[2]

[1] Department of System Engineering and Automation,
[2] Department of Electrical Engineering
Polytechnic University of Cartagena (UPCT),
Campus Muralla del Mar, 30202, Cartagena, Murcia, Spain.
{francisco.garcia, antonio.guerrero, pentxo.marin}@upct.es

Abstract. A neural architecture that makes possible the integration of a kinematic adaptive neuro-controller for trajectory tracking and an obstacle avoidance adaptive neuro-controller is proposed for nonholonomic mobile robots. The kinematic adaptive neuro-controller is a real-time, unsupervised neural network that learns to control a nonholonomic mobile robot in a nonstationary environment, which is termed Self-Organization Direction Mapping Network (SODMN), and combines associative learning and Vector Associative Map (VAM) learning to generate transformations between spatial and velocity coordinates. The transformations are learned in an unsupervised training phase, during which the robot moves as a result of randomly selected wheel velocities. The robot learns the relationship between these velocities and the resulting incremental movements. The obstacle avoidance adaptive neuro-controller is a neural network that learns to control avoidance behaviors in a mobile robot based on a form of animal learning known as operant conditioning. Learning, which requires no supervision, takes place as the robot moves around an cluttered environment with obstacles. The neural network requires no knowledge of the geometry of the robot or of the quality, number, or configuration of the robot's sensors. The efficacy of the proposed neural architecture is tested experimentally by a differentially driven mobile robot.

1 Introduction

Several heuristic approaches based on neural networks (NNs) have been proposed for identification and adaptive control of nonlinear dynamic systems [1].

In wheeled mobile robots (WMR), the trajectory-tracking problem with exponential convergence has been solved theoretically using time-varying state feedback based on the backstepping technique in [2]. The kinematic model of mobile robots can be described in polar coordinates and stabilization achieved using the backstepping technique [3].

The study of autonomous behavior has become an active research area in the field of robotics. Even the simplest organisms are capable of behavioral feats unimaginable for the most sophisticated machines. When an animal has to operate in an unknown environment it must somehow learn to predict the consequences of its own actions. By learning the causality of environmental events, it becomes possible for an animal to predict future and new events. A related form of learning is known as operant conditioning [4]. Chang and Gaudiano [5] introduce a neural network for obstacle avoidance that is based on a model of classical and operant conditioning.

In the classical conditioning paradigm, learning occurs by repeated association of a Conditioned Stimulus (CS), which normally has no particular significance for an animal, with an Unconditioned Stimulus (UCS), which has significance for an animal and always gives rise to an Unconditioned Response (UCR). The response that comes to be elicited by the CS after classical conditioning is known as the Conditioned Response (CR) [4]. In this case an animal learns the consequences of its actions. In the field of neural networks research, it is often suggested that neural networks based on associative learning laws can model the mechanisms of classical conditioning, while neural networks based on reinforcement learning laws can model the mechanisms of operant conditioning [6].

In this paper, a neural architecture that makes possible the integration of a kinematic adaptive neuro-controller for trajectory tracking and an obstacle avoidance adaptive neuro-controller is proposed for nonholonomic mobile robots. The kinematic adaptive neuro-controller is a real-time, unsupervised neural network, which is termed Self-Organization Direction Mapping Network (SODMN), uses an associative learning to generate transformations between spatial and velocity coordinates. The obstacle avoidance adaptive neuro-controller is a neural network that learns to control avoidance behaviors based on a form of animal learning known as operant conditioning. The efficacy of the proposed neural architecture is tested experimentally by a differentially driven mobile robot.

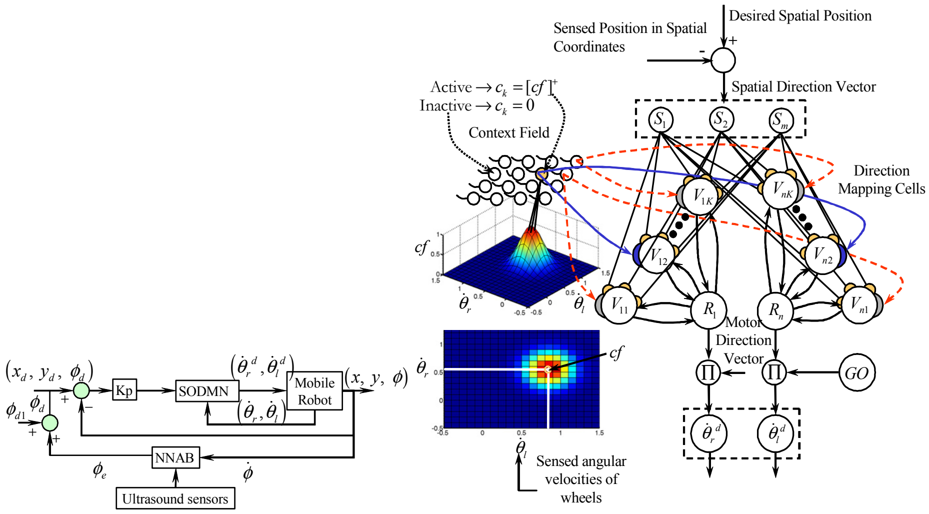
This paper is organized as follows. We first describe the architecture of the neural control system to mobile robot avoidance and approach behaviors in Section II. Experimental results with the proposed scheme for control of avoidance and approach behavior on a mobile robot are addressed in Section III. Finally, in Section IV, conclusions based on experimental results are given.

2 Architecture of the Neural Control System

Figure 1(a) illustrates our proposed neural architecture. The trajectory tracking control without obstacles is implemented by the SODMN and the avoidance behavior of obstacles is implemented by a neural network of biological behavior.

2.1 Self-Organization Direction Mapping Network (SODMN)

At a given set of angular velocities the differential relationship between mobile robot motions in spatial coordinates and angular velocities of wheels is expressed like a linear mapping. The transformation of spatial directions to wheels angular velocities is shown in Fig. 1(b). The spatial error is computed to get a spatial direction vector (DVs). The DVs is transformed by the *direction mapping network* elements V_{ik} to corresponding motor direction vector (DVm). On the other hand, a set of tonically active inhibitory cells which receive broad-based inputs that determine the context of a motor action was implemented as a context field. The context field selects the V_{ik} elements based on the wheels angular velocities configuration.



(a) (b)

Fig. 1. a) Neural architecture for reactive and adaptive navigation of a mobile robot. b) Self-organization direction mapping network for the trajectory tracking of a mobile robot.

A speed-control GO signal acts as a nonspecific multiplicative gate and control the movement's overall speed. The GO signal is a input from a decision center in the brain, and starts at zero before movement and then grows smoothly to a positive value as the movement develops. During the learning, sensed angular velocities of wheels are fed into the DVm and the GO signal is inactive.

Activities of cells of the DVs are represented in the neural network by quantities $(S_1, S_2, ..., S_m)$, while activities of the cells of the motor direction vector (DVm) are represented by quantities $(R_1, R_2, ..., R_n)$. The direction mapping is formed with a field of cells with activities V_{ik}. Each V_{ik} cell receives the complete set of spatial inputs S_j, $j = 1, ..., m$, but connects to only one R_i cell (see Figure 1(b)). The mechanism that is used to ensure weights converge to the correct linear mapping is similar to the VAM learning construction [7]. The direction mapping cells ($\boldsymbol{V} \in \mathbb{R}^{n \times k}$) compute a difference of activity between the spatial and motor direction vectors via feedback from DVm. During learning, this difference drives the adjustment of the weights. During performance, the difference drives DVm activity to the value encoded in the learned mapping.

A context field cell pauses when it recognizes a particular velocity state (i.e., a velocity configuration) on its inputs, and thereby disinhibits its target cells. The target cells (direction mapping cells) are completely shut off when their context cells are active. This is shown in Fig. 1(b). Each context field cell projects to a set of direction mapping cells, one for each velocity vector component. Each velocity vector component has a set of direction mapping cells associated with it, one for each context. A cell is “off” for a compact region of the velocity space. It is assumed for simplicity that only one context field cell turns “off” at a time. In Figure 1(b), inactive cells are shown as white disks. The center context field cell is “off” when the angular velocities are in the center region of the velocity space. The “off” context cell enables a subset of direction mapping cells through the inhibition variable c_k, while “on” context cells disable to the other subsets.

The learning is obtained by decreasing weights in proportion to the product of the presynaptic and postsynaptic activities [7]. The training is done by generating random movements, and by using the resulting angular velocities and observed spatial velocities of the mobile robot as training vectors to the direction mapping network.

2.2 Neural Network for the Avoidance Behavior (NNAB)

Grossberg proposed a model of classical and operant conditioning, which was designed to account for a variety of behavioral data on learning in vertebrates [4], [8]. This model was used to explain a number of phenomena from classical conditioning. Our implementation is based in the Grossberg’s conditioning circuit, which follows closely that of Grossberg & Levine [8] and Chang & Gaudiano [5], and is shown in Figure 2.

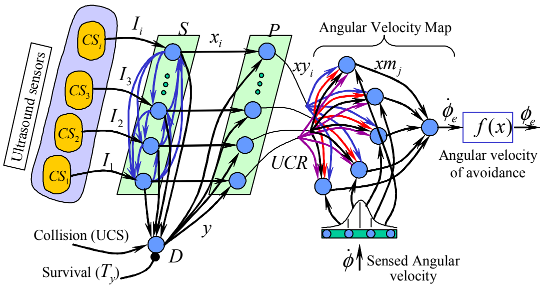


Fig. 2. Neural Network for the avoidance behavior.

In this model the sensory cues (both CSs and UCS) are stored in Short Term Memory (STM) within the population labeled S, which includes competitive interactions to ensure that the most salient cues are contrast enhanced and stored in STM while less salient cues are suppressed. The population S is modeled as a recurrent competitive field in simplified discrete-time version, which removes the inherent noise, efficiently normalizes and contrast-enhances from the ultrasound sensors activations. In the present model the CS nodes correspond to activation from the robot’s ultrasound sensors. In the network I_i represents a sensor value which codes proximal objects with large values and distal objects with small values. The network requires no knowledge of the geometry of the mobile robot or the quality, number, or distribution of sensors over the robot's body.

The drive node D corresponds to the Reward/Punishment component of operant conditioning (an animal/robot learns the consequences of its own actions). Learning can only occur when the drive node is active. Activation of drive node D is determined by the weighted sum of all the CS inputs, plus the UCS input, which is presumed to have a large, fixed connection strength. The drive node is active when the robot collides with an obstacle, which could be detected through a collision sensor, or when any one of the proximity sensors indicates that an obstacle is closer than the sensor's minimum range. Then the unconditioned stimulus (USC) in this case corresponds to a collision detected by the mobile robot. The activation of the drive node and of the sensory nodes converges upon the population of polyvalent cells P. Polyvalent cells require the convergence of two types of inputs in order to become active. In particular each polyvalent cell receives input from only one sensory node, and all polyvalent cells also receive input from the drive node D.

Finally, the neurons (xm_i) represent the response conditioned or unconditioned and are thus connected to the motor system. The motor population consists of nodes (i.e., neurons) encoding desired angular velocities of avoidance, i.e., the activity of a given node corresponds to a particular desired angular velocity for the mobile robot. When driving the robot, activation is distributed as a Gaussian centered on the desired angular velocity of avoidance. The use of a Gaussian leads to smooth transitions in angular velocity even with few nodes.

The output of the angular velocity population is decomposed by SODMN into left and right wheel angular velocities. A gain term can be used to specify the maximum possible velocity. In NNAB the proximity sensors initially do not propagate activity to the motor population because the initial weights are small or zero. The robot is trained by allowing it to make random movements in a cluttered environment. Specifically, we systematically activate each node in the angular velocity map for a short time, causing the robot to cover a certain distance and rotate through a certain angle depending on which node is activated. Whenever the robot collides with an obstacle during one of these movements (or comes very close to it), the nodes corresponding to the largest (closest) proximity sensor measurements just prior to the collision will be active. Activation of the drive node D allows two different kinds of learning to take place: the learning that couples sensory nodes (infrared or ultrasounds) with the drive node (the collision), and the learning of the angular velocity pattern that existed just before the collision.

The first type of learning follows an associative learning law with decay. This learning enables the most active sensory cues to accrue strength in their connection to the drive node, so that eventually they will be able to generate their own drive signal, and thus activate the polyvalent cells P, and ultimately a motor response. The primary purpose of this learning scheme is to ensure that learning occurs only for those CS nodes that were active within some time window prior to the collision (UCS). The second type of learning, which is also of an associative type but inhibitory in nature, is used to map the sensor activations to the angular velocity map. By using an inhibitory learning law, the polyvalent cell corresponding to each sensory node learns to generate a pattern of inhibition that matches the activity profile active at the time of collision. For instance, if the robot was turning right and collided with an obstacle, the proximity sensor neuron most active shortly before the collision will learn to generate an inhibitory Gaussian centered upon the right-turn node in the angular velocity population.

Once learning has occurred, the activation of the angular velocity map is given by two components (see Figure 3). An excitatory component, which is generated directly by the sensory system, reflects the angular velocity required to reach a given target in the absence of obstacles. The second, inhibitory component, generated by the conditioning model in response to sensed obstacles, moves the robot away from the obstacles as a result of the activation of sensory signals in the conditioning circuit.

One reason for using a Gaussian distribution of activity (see Figure 3) is because when an excitatory Gaussian is combined with an inhibitory Gaussian at a slightly shifted position, the resulting net pattern of activity exhibits a maximum peak that is shifted in a direction away from the peak of the inhibitory Gaussian. Hence, the presence of an obstacle to the right causes that the mobile robot to shift to the left, and viceversa.

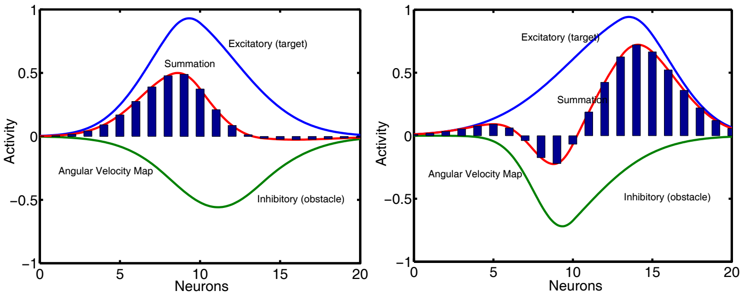


Fig. 3. Positive Gaussian distribution represents the angular velocity without obstacle and negative distribution represents activation from the conditioning circuit. The summation represents the angular velocity that will be used to drive the mobile robot.

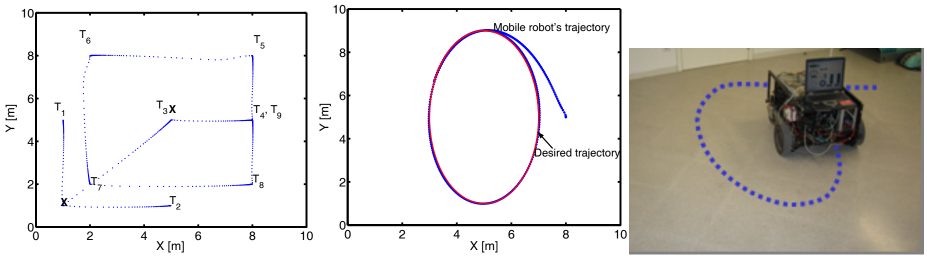
3 Experimental Results

The proposed control algorithm is implemented on a mobile robot from the Polytechnic University of Cartagena (UPCT) named "CHAMAN". The platform has two driving wheels (in the rear) mounted on the same axis and two passive supporting wheels (in front) of free orientation. The two driving wheels are independently driven by two DC-motors to achieve the motion and orientation. The wheels have a radius $r = 18$ cm and are mounted on an axle of length $2R = 22$ cm. The aluminum chassis of the robot measures 102.25×68×44 cm

($L \times W \times h$) and contains, transmission elements, 12-VDC battery, two CCD cameras, and 12 ultrasound sensors. Each motor is equipped with incremental encoder counting 600 pulses/turn and a gear which reduces the speed to 1.25 m/s.

High-level control algorithms (SODMN and NNAB) are written in VC++ and run with a sampling time of 10 ms on a remote server (a Pentium IV processor). The PC communicates through a serial port with the microcontroller on the robot. Wheel PWM duty cycle commands are sent to the robot and encoder measures are received for odometric computation. The lower level control layer is in charge of the execution of the high-level velocity commands. It consists of a Texas Instruments TMS320C6701 Digital Signal Processor (DSP). The microcontroller performs three basis tasks: 1) to communicate with the higher-level controller through RS 232; 2) reading encoder counts interrupt driven; and 3) generation of PWM duty cycle.

Figure 4 shows approach behaviors and the tracking of a trajectory by the mobile robot with respect to the reference trajectory.



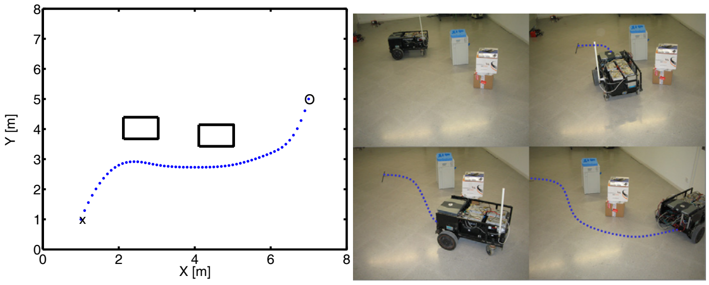
(a) (b) (c)

Fig. 4. Adaptive control by the SODMN. a) Approach behaviors. The symbol X indicates the start of the mobile robot and T_i indicates the desired reach. b) Tracking control of a desired trajectory. c) Real-time tracking performance.

Figure 5 illustrates the mobile robot's performance in the presence of several obstacles. The mobile robot starts from the initial position labeled X and reaches a desired position. During the movements, whenever the mobile robot is approaching an obstacle, the inhibitory profile from the conditioning circuit (NNAB) changes the selected angular velocity and makes the mobile robot turn away from the obstacle. The presence of multiple obstacles at different positions in the mobile robot's sensory field causes a complex pattern of activation that steers the mobile robot between obstacles.

4 Conclusions

In this paper, we have implemented a neural architecture for trajectory tracking and avoidance behaviors of mobile robot. A biologically inspired neural network for the spatial reaching tracking has been developed. This neural network



(a) (b)

Fig. 5. Trajectory followed by the mobile robot in presence of obstacles using the NNAB.

is implemented as a kinematic adaptive neuro-controller. The SODMN uses a context field for learning the direction mapping between spatial and angular velocity coordinates. The transformations are learned during an unsupervised training phase, during which the mobile robot moves as result of randomly selected angular velocities of wheels. The performance of this neural network has been successfully demonstrated in experimental results with the trajectory tracking and reaching of a mobile robot. The avoidance behaviors of obstacles were implemented by a neural network that is based on a form of animal learning known as operant conditioning. The efficacy of the proposed neural network for avoidance behaviors was tested experimentally by a differentially driven mobile robot. In future works, the NNAB can be extended to avoid moving obstacles and the SODMN can be extended and implemented as a kinematic adaptive neuro-controller for robotic manipulators.

References

1. Fierro, R., Lewis, F.L.: Control of a nonholonomic mobile robot using neural networks. IEEE Trans. Neural Netw. **9** (1998) 589–600
2. Ping, Z., Nijmeijer, H.: Tracking control of mobile robots: A case study in backstepping. Automatica **33** (1997) 1393–1399
3. Das, T., Kar, I.N.: Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots. IEEE Transactions on Control SystemsTechnology **14** (2006) 501–510
4. Grossberg, S.: On the dynamics of operant conditioning. Journal of Theorical Biology **33** (1971) 225–255
5. Chang, C., Gaudiano, P.: Application of biological learning theories to mobile robot avoidance and approach behaviors. J. Complex Systems **1** (1998) 79–114
6. Sutton, R.S., Barto, A.G.: Toward a modern theory of adaptive networks: Expectation and prediction. Psychological Review **88** (1981) 135–170
7. Gaudiano, P., Grossberg, S.: Vector associative maps: Unsupervised real-time error-based learning and control of movement trajectories. Neural Networks **4** (1991) 147–183
8. Grossberg, S., Levine, D.: Neural dynamics of attentionally moduled Pavlovian conditioning: Blocking, interstimulus interval, and secondary reinforcement. Applied Optics **26** (1987) 5015–5030