STDP-based behavior learning on TriBot robot

P. Arena[a], S. De Fiore[a], L. Patané[a], M. Pollino[a] and C. Ventura[a].

[a] Dipartimento di Ingegneria Elettrica, Elettronica e dei Sistemi, Universitá degli Studi di Catania, Viale A. Doria 6, 95125 Catania, Italy.

ABSTRACT

This paper describes a correlation-based navigation algorithm, based on an unsupervised learning paradigm for spiking neural networks, called Spike Timing Dependent Plasticity (STDP). This algorithm was implemented on a new bio-inspired hybrid mini-robot called TriBot to learn and increase its behavioral capabilities. In fact correlation based algorithms have been found to explain many basic behaviors in simple animals. The main interesting consequence of STDP is that the system is able to learn high-level sensor features, based on a set of basic reflexes, depending on some low-level sensor inputs. TriBot is composed of 3 modules, the first two being identical and inspired by the Whegs hybrid robot. The peculiar characteristics of the robot consists in the innovative shape of the three-spoke appendages that allow to increase stability of the structure. The last module is composed of two standard legs with 3 degrees of freedom each. Thanks to the cooperation among these modules, TriBot is able to face with irregular terrains overcoming potential deadlock situations, to climb high obstacles compared to its size and to manipulate objects. Robot experiments will be reported to demonstrate the potentiality and the effectiveness of the approach.

Keywords: STDP, hybrid robot, visual cue-based navigation, spiking neurons.

1. INTRODUCTION

The main purpose of this paper is to develop a bio-inspired spiking network used to improve the behavioral capabilities of a new bio-inspired hybrid mini-robot called TriBot. The control architecture exploits correlation-based navigation algorithms, based on an unsupervised learning paradigm for spiking neural networks, called Spike Timing Dependent Plasticity (STDP). This algorithm is used in the proposed framework to allow the emergence of causal relations among events recorded at the sensor level. This type of learning has a solid biological background: in fact, the STDP learning rule has been introduced to explain synaptic plasticity.[1] According to this theory, synaptic connections are reinforced if there is a causal correlation between the spiking times of the two connected neurons. STDP is here used to find correlations among different kinds of sensors, with the aim to predict and anticipate the outcome of a sensor by using another one. The main interesting consequence of STDP is that the system is able to learn high-level sensor features, called conditioned stimuli (CS), based on a set of basic reflexes, depending on some low-level sensor inputs, called unconditioned stimuli (US). The spiking neural networks developed to host the STDP algorithm was inspired by a structure used to model the phonotaxis reflex in crickets, and suitably modified to cope with different kinds of input signals.[2]

Another remark concerns the biological plausibility of large time delays in the STDP rule. STDP is a well-demonstrated mechanism accounting for neuronal plasticity which acts at typical time scales of milliseconds. On the other hand, behavioral sequences occur over arbitrary time scales, for example classical conditioning often operates with time intervals between CS and US which are at the time scale of seconds. A solution to this problem was proposed by Izhikevich[3] that introduced a futher modulation based on dopomine. This solution introduces a slow variable with characteristic time constants of the order of seconds compatible with our experiments.

Moreover there are many works in literature which take into account models reproducing the main features of hippocampal cells, neural structure involved in the storing of spatial-temporal characteristics of the environment, to implement navigation control.[4,5] These models, often based on experimental evidence in rats[4] identify a key role of the hippocampus for goal-directed spatial navigation.

Further author information:
Send correspondence to Luca Patané, E-mail: lpatane@diees.unict.it

The general conclusion which can be derived from all these models is that in any case a slow mechanism should be used. Our simplified network was designed starting from simple reactive controllers (such as those discussed in[2]) and extending their functions to implement more complex behaviors, keeping in mind the suitability of an easy robot implementation.[6,7]

The robot TriBot is a new autonomous robotic structure, conceived for exploratory tasks in heavily unstructured environments. It is composed of 3 modules, the first two being identical and inspired by the Whegs hybrid robot.[8] Their peculiarity consists in the innovative shape of the three-spoke appendages that increase the stability of the structure and the adaptability to different terrains. The front module is composed of two standard legs with 3 degrees of freedom each, that allow the robot to attain a high level of dexterity in the front part of the structure, needed to cope with different kinds of situations. Therefore this hybrid structure is also able to manipulate objects. Thanks to the cooperation among the three modules, TriBot is able to navigate on irregular terrains overcoming potential deadlock situations and to climb high obstacles compared to its size. The implementation of an unsupervised learning structure, like STDP, allows this structure to autonomously learn how to manage different situations, represented by different objects, in order to choose suitable strategies to interact with the environment. The robot will mainly use contact, distance and visual sensor information as input for the neural architecture. Robot experiments are reported in section 5 to demonstrate the potentiality and the effectiveness of the approach. The principles of spiking networks and STDP learning rule are introduced in chapter 2. More details about the TriBot robot are reported in section 3 and the neural architecture proposed here is discussed in section 4. Finally, some conclusions and future works are given in section 6.

2. SPIKING NEURONS AND STDP RULE

In this section the mathematical model of the spiking neurons and the STDP rule applied to learn basic behaviour skills in a real hybrid robot are introduced. The navigation task investigated in this paper includes the exploration of an environment that contains randomly placed obstacles that can be taken, climbed or avoided. By using fixed reactions to contact and distance sensors, the robot is able to learn the appropriate actions in response to conditioned stimuli (i.e. visual sensor). The spiking network used for this navigation task is described in more details in the next section.

Each neuron is modeled by the following equations proposed by Izhikevich[9]:

$$\begin{aligned} \dot{v} &= 0.04v^2 + 5v + 140 - u + I \\ \dot{u} &= a(bv - u) \end{aligned} \tag{1}$$

with the spike-resetting

$$\texttt{if } v \geq 0.03\texttt{, then} \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \tag{2}$$

where v, u and I are dimensionless variables representing the neuron membrane potential, the recovery variable and the input current, respectively, while a, b, c and d are system parameters. The time unit is ms.

According to the parameters chosen,[10] this model could reproduce the main firing patterns and neural dynamical properties of biological neurons, such as spiking behaviors (tonic, phasic and chaotic spiking) and bursting behavior.

Among the possible behaviors, we select the class I excitable neurons. The main characteristic of these neurons is that the spiking rate is proportional to the amplitude of the stimulus.[10] Such property is really important as a way to encode any measured quantity by means of the firing frequency. It also represents a suitable way to fuse sensory data at the network input level. Neuron parameters are chosen as $a = 0.02$, $b = -0.1$, $c = -55$, $d = 6$ (class I excitable neurons[10]), whereas the input I accounts for both external stimuli (e.g. sensorial stimuli) and synaptic inputs. The same model was adopted for all the neurons of the network.

As concerns the model of the synapse, let us consider a neuron j which has synaptic connections with n neurons, and let us indicate with t_s the instant in which a generic neuron i, connected to neuron j, emits a spike. The synaptic input to neuron j is given by the following equation:

$$I_j(t) = \sum w_{ij}\varepsilon(t - t_s) \tag{3}$$

where w_{ij} represents the weight of the synapse from neuron i to neuron j and the function $\varepsilon(t)$ is expressed by the following formula:

$$\varepsilon(t) = \begin{cases} \frac{t}{\tau}e^{1-\frac{t}{\tau}} & \texttt{if } t \geq 0 \\ 0 & \texttt{if } t < 0 \end{cases} \tag{4}$$

Equation (4) describes the contribution of a spike, from a presynaptic neuron emitted at $t = 0$. In our simulations τ has been fixed to $\tau = 5ms$.

To include adaptive capabilities in our model, Hebbian learning was considered. Recent results[1] indicate STDP as a model of experimentally observed biological synaptic plasticity. The synaptic weights of our network are thus allowed to be modifiable according to the STDP rule discussed in[1] and here briefly reported. Let us indicate with w the synaptic weight. A presynaptic spike and a post synaptic spike modify the synaptic weight w by $w \rightarrow w + \Delta w$, where, according to the STDP rule, Δw depends on the timing of pre-synaptic and post-synaptic spikes. The following rule holds[1]:

$$\Delta w = \begin{cases} A_+ e^{\frac{\Delta t}{\tau_+}} & \texttt{if } \Delta t < 0 \\ A_- e^{\frac{-\Delta t}{\tau_-}} & \texttt{if } \Delta t \geq 0 \end{cases} \tag{5}$$

where $\Delta t = t_{pre} - t_{post}$ is the difference between the spiking time of the pre-synaptic neuron (t_{pre}) and that of the post-synaptic one (t_{post}). If $\Delta t < 0$, the post-synaptic spike occurs after the pre-synaptic spike, thus the synapsis should be reinforced. Otherwise if $\Delta t \geq 0$, (the post-synaptic spike occurs before the pre-synaptic spike), the synaptic weight is decreased by the quantity Δw. The choice of the other parameters (A_+, A_-, τ_+ and τ_-) of the learning algorithm will be discussed below. Equation (5) is a rather standard assumption to model STDP. The term A_+ (A_-) represents the maximum Δw which is obtained for almost equal pre- and post-spiking times in the case of potentiation (depression).

The use of the synaptic rule described by equation (4) may lead to an unrealistic growth of the synaptic weights. For this reason, often upper limits for the weight values are fixed.[3, 11] Furthermore, some authors (see for instance[12, 13]) introduce a decay rate in the weight update rule. This solution avoids that the weights of the network increase steadily with training and allows a continuous learning to be implemented. In the simulations, the decay rate has been fixed to the 10% of the weight value and is performed when multiple collisions with an obstacle occurs. Thus, the weight values of all plastic synapses are updated according to the following equation:

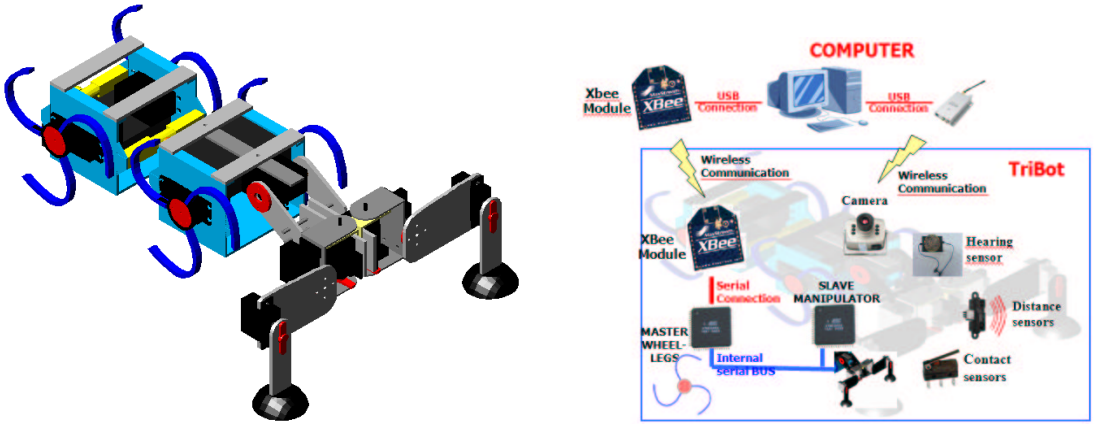
$$w(t+1) = 0.90w(t) + \Delta w \tag{6}$$

where Δw is given by eq. (5).

In the following, the upper limit of the learnable synaptic weight was fixed to 10 (excitatory synapses) or -10 (inhibitory synapses). The initial values of synapses subject to STDP are either 0.05 (excitatory synapses) or -0.05 (inhibitory synapses).

3. TRIBOT ROBOT

In this section, we discuss about the TriBot, an autonomous mobile hybrid robot used during the experiments. The mechanical design of the robotic structure is shown in Fig.1.a. The robot shows a modular structure, in particular it consists of two wheel-legs modules and a two-arms manipulator. The two wheel-legs modules are interconnected by a passive joint, with a spring that allows only the pitching movement, whereas an actuated joint connects these modules with the manipulator, that consists of two legs with three degrees of freedom. The whole mechanical structure is realized in aluminium and plexiglass; both materials have been selected for their



(a) (b)

Figure 1. Mechanical design (a) and hardware structure (b) of the TriBot Robot.

characteristics of cheapness and lightness. The robot dimensions are $36x23x13$ cm (length x height x width) and the maximum speed is 1.30 body length per second.

The hardware architecture of the robot TriBot follows the modularity of the structure. A block diagram is shown in Fig.1.b. The hardware structure of TriBot is managed by two microcontrollers that handle motors and sensors distributed on the structure. Furthermore through a computer, using a wireless connection, it is possible to acquire data and send commands generated by a high level control algorithm.

The robot is controlled by a system of two microcontroller-based boards connected through a master/slave bus.

The computer supervises and controls the robot through a RF wireless XBee module, that uses the ZigBee standard. The master control board is positioned in the central wheel-legs module. Its main role is to control the servomotors that actuate the four wheel-legs and is also used to host the bridge between the PC and the other board mounted on the manipulator. For these experiments, a win-based dual core at 2 GHz PC has been used. It is obvious that it represents a crucial knot for the communication and the general management of the robot. It uses an ATmega64, a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture.[14]

The front manipulator is controlled by a similar board configured as slave; also in this case an ATmega64 microcontroller is used to generate the PWM signals for the six servomotors that actuate the manipulator and to the servomotor that actuates the joint connecting the manipulator with the robot body. This board is also used to read data from the distributed sensory system embedded in the manipulator. In particular, on the manipulator, four distance sensors have been distributed for obstacle detection and a series of micro-switches are used to detect collisions and to grasp objects.

The sensory system is needed for autonomous navigation and to use the robot as test bed for perceptual algorithms. For this reason to implement targeting and homing strategies, a hearing circuit inspired by phonotaxis in crickets has been also included. The aim is to give the robot the ability to approach sound sources, reproducing the behaviour shown by the female cricket to follow a particular sound chirp emitted by a male cricket.[2]

Furthermore, one of the most useful and rich senses that guides animal's actions is the visual system. Therefore the robot is equipped with a wireless camera that can be used for landmark identification, objects following and other higher level tasks.

Finally, Fig. 2 shows some configurations that the TriBot can assume thanks the particular structure and sensory-motor system. The actuated joint, that connects the manipulator to the robot body, allows to reach different configurations: therefore it can be useful to improve locomotion capabilities when it is moved down (i.e.

(a) (b) (c) (d)

Figure 2. Different TriBot configurations. Thanks to the actuated joint, the manipulator can be used in different scenarios: to increase the locomotion or climbing capabilities (a)(c), to manipulate the environment (b), to detect an obstacles during navigation (d).

used as legs to perform small movements such as fig. 2.a), while when it is moved up it can grasp objects (fig. 2.b), improve climbing capabilities (fig. 2.c) and detect obstacles (fig. 2.d).

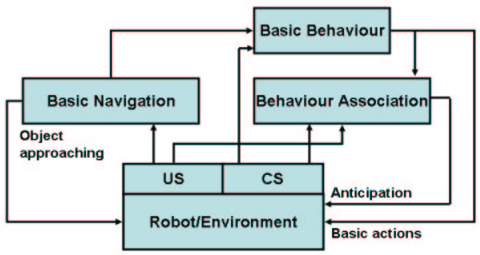
4. THE SPIKING NETWORK

The navigation architecture here proposed can be divided into three main blocks (see Fig. 3.a). The unconditioned and conditioned stimuli (CS and US), coming from the robot, represent the inputs for the neural networks. The first network, called "Basic Navigation", is composed of two layers constituted by sensory neurons and motor neurons as depicted in Fig. 3.b. The connections and the fixed-synapses values allow the approaching behaviour toward the object placed in the scene. The second block, called "Behaviours Association" network, is the core of the proposed architecture: it is able, after a learning phase, to choose the right behaviour depending on the visual sense feature extraction (see Fig. 3.c). Finally the last network (called "Basic Behaviours") is used to drive the learning phase generating a hierarchical sequence of the different behaviours that the robot can perform (see Fig. 3.d).

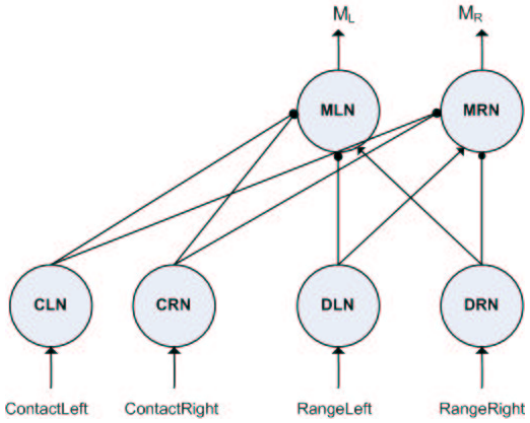
The Basic Navigation network represents an inherited behaviour and consists of a series of spiking neurons connected with fixed synapses. The sensory input connected to the system includes contact (CLN, CRN) and distance (DLN, DRN) sensors and the outputs, generated by the motor neurons, are used to control the velocity of the wheels on the left (MLN) and right (MRN) side of the robot. Therefore the network is devoted to control the robot movement during the exploration phase. The distance sensors, placed in the robot manipulator, point at the ground with a given angle; the robot shows an attractive behaviour towards the objects found in the environment. When the detected distance goes under a given threshold a contact is triggered and the robot stops just in front of the object. At this point the robot changes its posture to use the visual sensor. Simple processing functions extract features from the acquired images: red (R) and yellow (Y) circles can be distinguished, or if not present, the robot labels the scene as "other" (O). During the visual processing phase, the robot acquires the centroid of the colored circle (if present) and performs some centering manoeuvres. The visual stimuli are then used as CS inputs for the other two networks (see Fig. 3.c,d).

The Basic Behaviour network activates a sequence of behaviours performed by the robot that tries to take the object (TL), to climb it (CL) and finally to avoid (AL). Practically, the network activates the basic behaviors in sequence until the unconditioned stimulus that identifies the end of action is reached. This stimulus occurs either after a maximum execution time or when the action has been successfully completed. The event (Et, Ec and Ea for taking, climbing and avoiding behaviors respectively) is provided by the robot sensory system.

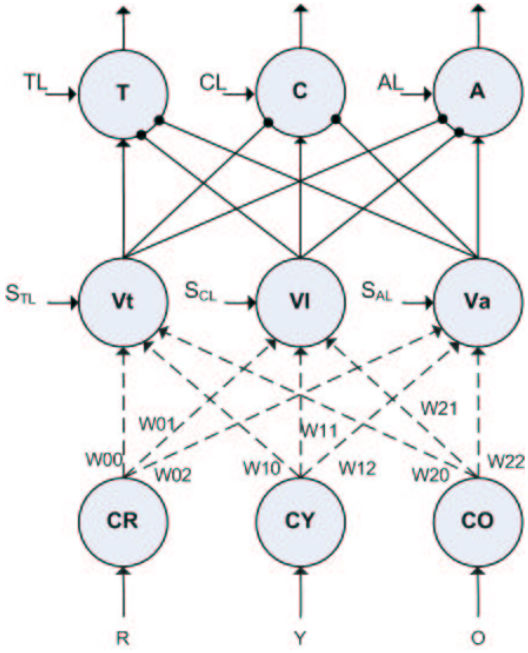
In this case, the unconditioned stimuli are given by: contact sensors that detect success in taking an object S_{TL}; an inclinometer that detects if an obstacle has been overcome S_{CL}; distance sensors are used to determinate if an object has been correctly avoided S_{AL}. For practical reasons, in the following experiments, the US for climbing has been substituted with a visual routine that generates the climbing US after a successful climbing action.



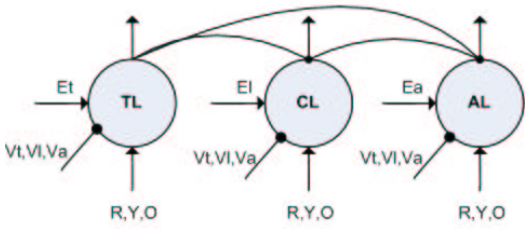
(a)



(b)



(c)



(d)

Figure 3. The whole neural architecture (a) is composed of three subnetworks: (b) object approaching, (c) object behaviour association (d) basic behaviour.

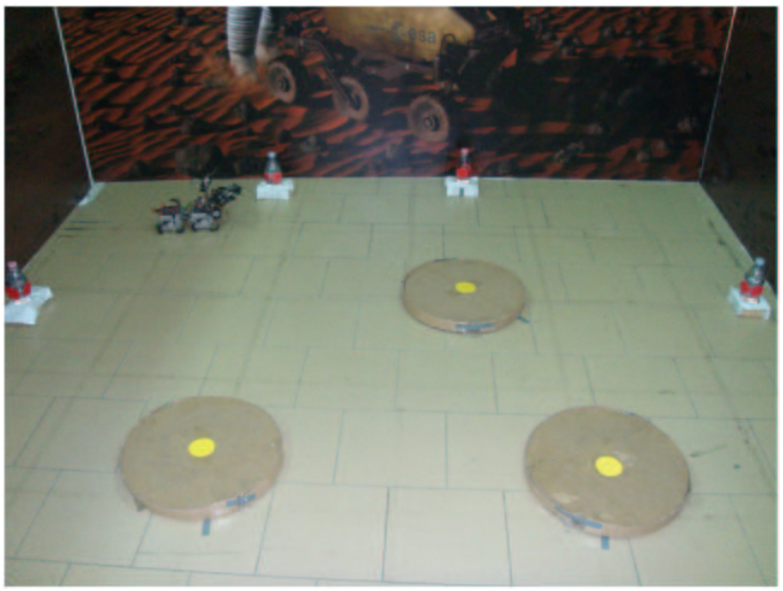


Figure 4. Environment used during the experiments.

The robot, while performing its basic behaviours, tries to find correlations among visual stimuli (CS) and the low level sensors output (US) here represented by S_{TL}, S_{CL} and S_{AL} for taking, climbing and avoiding behaviors respectively. Therefore the STDP learning rule is used to update the synaptic weights of the plastic synapses (dashed lines) in the Behaviour Association network (see Fig. 3.c). When a basic behaviour is successfully completed, a low level sensor S_{TL}, S_{CL} and S_{AL} is activated in order to train the network parameters and to correlate the conditioned stimuli, starting from a priori known responses, to unconditioned stimuli.

Due to the modular structure used to design the networks, the architecture can be easly extended to include new sensors and basic behaviours increasing the robot capabilities.[16]

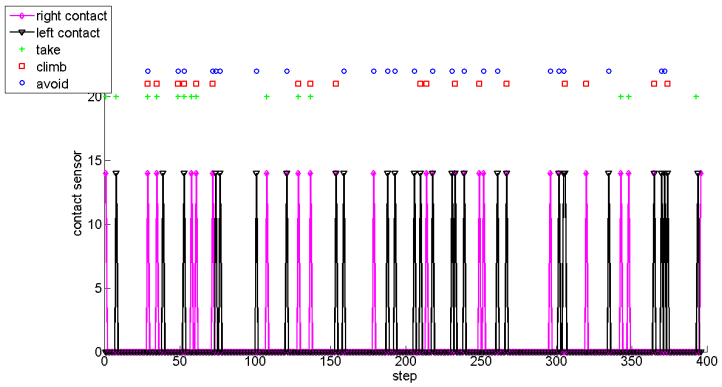
5. EXPERIMENTAL RESULTS

The arena used for the robot experiments is shown in Fig. 4 with dimension of $10x10$ *body length* randomly filled with objects that can be taken (bottles signed with a red circle), climbed (circular objects signed with a yellow circle) or avoided (walls).

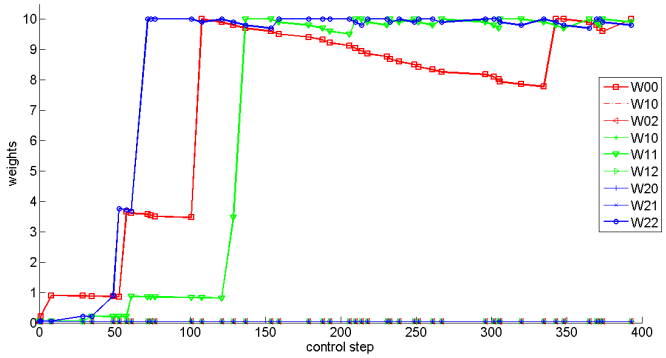
During the learning phase the robot navigates in the environment showing an attractive behaviour. When an object is detected the robot tries firstly to grab it. If the action does not succeed the robot tries to climb the obstacle or eventually to avoid it. At the end of the procedure the synapses subject to the STDP learning rule are updated. Fig. 5 shows the robot actions sequence and the learnable synapses behaviours. At the beginning of the experiment more than one action can be performed on each identified object due to the absence of association between object features and corresponding behaviours. The learning parameters chosen are $A_+ = A_- = 0.05$, $\tau_+ = \tau_- = 20$. Fig. 5.b shows that the robot is able to learn correlation between sensory stimuli and actions exploiting the dynamic of the synapses improving the robot basic capabilities after few trials.

It is important to notice that the forgetting factor applied to the synaptic weights guaranties high flexibility to the learning: in fact the robot is also able to change the association maps created between visual cues and behaviours if the characteristics of the object change in time.

The whole trajectory followed by the robot during the experiment is reported in Fig. 6.a. The time used for the learning was about 30 minutes, on the hardware architecture outlined in the previous section, in which the robot performed 400 actions (including basic navigation) 45 behaviours. The percentage of successful actions, cumulates in windows of 50 control steps, is shown in Fig. 6.b. During learning, the percentage grows to 100% that means that the robot performs the correct action at the first trial. The effect of learning is also evident in



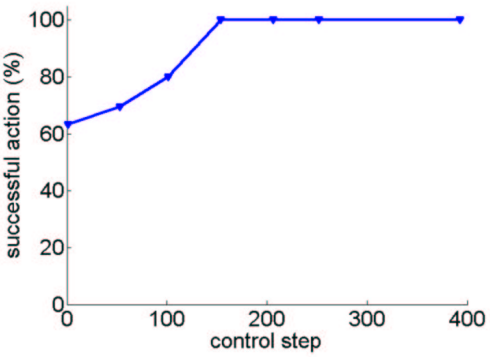
(a)



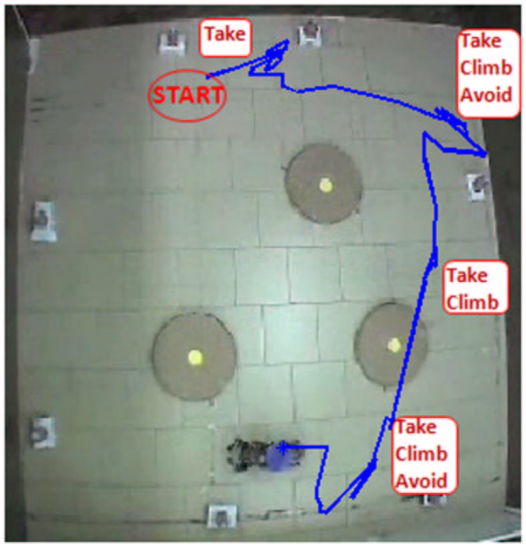
(b)

Figure 5. After few steps the correct correlation scheme is learnt. (a) when a contact sensor is activated (grey signals are related to the right contact and black signals to the left contact), the robot tries to perform different actions, in figure the circle indicates "avoid", the square "climb" and the triangle "take". At the beginning of the learning phase more than one behaviour are performed on the same object. (b) The trend of learnable synaptic weights shows the association scheme created during the learning phase. The time is discretizated in control steps that corrispond to the action executed by the robot.

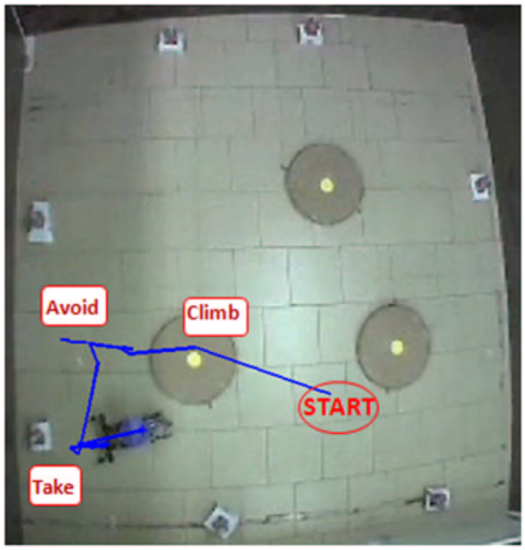
(a)



(b)



(c)



(d)

Figure 6. Some experimental results: (a) trajectory followed by the robot during the complete learning phase (about 30 minutes); (b) percentage of successful actions performed; (c) (d) trajectories followed by the robot between [8,49] and [372,393] control steps respectively, also the actions are reported.

Fig. 6.c and Fig. 6.d, where parts of the whole trajectory with the actions learned are reported showing the robot behaviour at the beginning and at the end of the learning phase.

Video of experiments are available on line.[15]

6. CONCLUSIONS

In this paper a network of spiking neurons for robot navigation control is introduced. The learning mechanism of this network is provided by STDP, which allows unsupervised learning to be realized in a simple way. The technique is applied to a new hybrid robot, named TriBot. Thanks to the mechanical structure, TriBot is able to interact with the environment in different ways. In particular, the basic behaviours here used are: taking, climbing and avoiding an obstacle. A complete neural architecture that allows to control the robot in order to find an object and to perform the basic behaviours available for the robot has been designed and experimentally evaluated. The experimental results show the efficiency of the algorithm. The system is able to learn the association among visual features and basic behaviours through the STDP rule in a reasonable time.

We believe that the introduced approach can provide efficient navigation control strategies with very simple unsupervised learning mechanisms. Further works will include the extension of neural architecture to include new sensors and basic behaviours.

The scalability of the networks is simplified by the modularity of the approach. Moreover hybrid robot TriBot can be enriched with new sensors and new behaviours can be programmed and learned as a sequence of the already available basic behaviours.

ACKNOWLEDGMENTS

The authors acknowledge the support of the European Commission under the project SPARK II "Spatial-temporal patterns for action-oriented perception in roving robots: an insect brain computational model".

REFERENCES

1. S. Song, K. D. Miller, L. F. Abbott, "Competitive Hebbian learning through spike-timing-dependent plasticity", *Nature Neurosci.*, Vol. 3, pp. 919–926, 2000.
2. B. Webb, T.Scutt, "A simple latency dependent spiking neuron model of cricket phonotaxis", *Biological Cybernetics*, 82(3): 247-269, 2000.
3. E.M. Izhikevich, "Solving the distal reward problem through linkage of STDP and dopamine signaling", *Cerebral Cortex Advance*, 2007.
4. O. Jensen, J.E. Lisman, "Hippocampal sequence-encoding driven by a cortical multi-item working memory buffer", *TRENDS in Neurosciences*, vol.28, no.2, 2005.
5. N. Burgess, J. O'Keefe, "Neuronal computations underlying the firing of place cells and their role in navigation", *Hippocampus*, no. 6, pp. 749–762, 1996.
6. P. Arena, F. Danieli, L. Fortuna, M. Frasca, L. Patané, "Spike-timing-dependent plasticity in spiking neuron networks for robot navigation control", *Int. Symposium on Microtechnologies for the New Millennium (SPIE 05)*, Sevilla (Spain), 9-11 May 2005.
7. P. Arena, L. Fortuna, M. Frasca, L. Patané, D. Barbagallo, C. Alessandro, "Learning high-level sensors from reflexes via spiking networks in roving robots", *Proc. of 8th International IFAC Symposium on Robot Control (SYROCO 2006)*, Bologna (Italy), September 6 - 8, 2006.
8. WHEGS Robot, online at *http://biorobots.cwru.edu/projects/whegs/whegs.html*
9. E. M. Izhikevich, "Simple Model of Spiking Neurons", *IEEE Transactions on Neural Networks*, Vol. 14, No. 6, 1569–1572, 2003.
10. E. M. Izhikevich, "Which Model to Use for Cortical Spiking Neurons?", *IEEE Transactions on Neural Networks*, Vol. 15, No. 5, 1063–1070, 2004.
11. S. Song, L.F. Abbott, "Cortical development and remapping through Spike Timing-Dependent Plasticity", *Neuron*, vol. 32, pp. 339-350, 2001.
12. P.F.M.J. Verschure and R. Pfeifer, "Categorization, Representations, and the Dynamics of System-Environment Interaction: a case study in autonomous systems", in J.A. Meyer, H. Roitblat, S. Wilson (Eds.) From Animals to Animats: Proceedings of the Second International Conference on Simulation of Adaptive Behavior, Cambridge, MA, MIT Press, pp. 210–217, 1992.

13. P.F.M.J. Verschure, B.J.A. Kröse, R. Pfeifer, "Distributed adaptive control: The self-organization of structured behavior", *Robotics and Autonomous Systems*, vol. 9, pp. 181-196, 1992.
14. Datasheet ATMEL ATMEGA64 Microcontroller: *http://www.atmel.com*
15. EU Project SPARK II, online at *http://www.spark2.diees.unict.it.*
16. P. Arena, L. Fortuna, M. Frasca, L. Patané, "Learning anticipation via spiking networks: application to navigation control", *IEEE TRANSACTIONS ON NEURAL NETWORKS*, VOL. 20, NO. 2, pp. 202-216, February 2009.