

Spiking neural controllers in multi-agent competitive systems for adaptive targeted motor learning

Paolo Arena^{a,b}, Luca Patané^a, Alessandra Vitanza^a

^a*Dipartimento di Ingegneria Elettrica, Elettronica e Informatica, University of Catania, viale A. Doria 6, 95125 Catania, Italy*

^b*National Institute of Biostructures and Biosystems (INBB), Viale delle Medaglie d'Oro 305, 00136 Rome, Italy*

Abstract

The proposed work introduces a neural control strategy for guiding adaptation in spiking neural structures acting as nonlinear controllers in a group of bio-inspired robots which compete in reaching targets in a virtual environment. The neural structures embedded into each agent are inspired by a specific part of the insect brain, namely Central Complex, devoted to detect, learn and memorize visual features for targeted motor control. A reduced-order model of a spiking neuron is used as the basic building block for the neural controller. The control methodology employs bio-inspired, correlation based learning mechanisms like *Spike timing dependent plasticity* with the addition of a reward/punishment-based method experimentally found in insects. The reference signal for the overall multi-agent control system is imposed by a global reward, which guides motor learning to direct each agent towards specific visual targets. The neural controllers within the agents start from identical conditions: the learning strategy induces each robot to show anticipated targeting actions upon specific visual stimuli. The whole control structure also contributes to make the robots refractory or more sensitive to specific visual stimuli, showing distinct preferences in future choices. This leads to an environmentally induced, targeted motor control, even without a direct communication among the agents, giving robots, while running, the ability to perform adaptation in real-time. Experiments, carried out in a dynamic simulation environment, show the suitability of the proposed approach. Specific performance indexes, like Shannon's Entropy, are adopted to quantitatively analyze diversity and specialization within the group.

Key words: Neural control, multi-robot applications, bio-robotics, spike-frequency adaptation, STDP learning

1. Introduction

Adaptive motor control in a dynamically changing, life-like environment, involves a continuous evaluation of the external constraints imposed by the environment and a comparison with the internal states which drive intended behaviors. The capability of efficiently react to external stimuli and re-adapt behavioral responses improves the living being performance and life expectations. This overall perspective poses the agent-environment interaction as a complex adaptive non-linear control system, where the single agent exploits the possibilities available in the environment and records the imposed constraints. This is even more important when a group of agents coexists in the same environment. Multiple individual control architectures have to compete and cooperate to improve individual and collective performance. Among adaptive properties shown by each individual, *diversity* refers to the possibility to show differences among the single agents in terms of behavioral responses: this could derive from a different body structure, hardware constraints or controller architecture. Another possible adaptive behavior, which could be a consequence of the previous property, is *specialization*: this generally arises in identical agents through learning processes and adaptation mechanisms. Its consequence is an improvement of performance related to the ongoing task. Nature offers a superb source of inspiration to face with this complex control task: insect colonies show extraordinary emerging adaptation capabilities embedded into simple brain structures. Often, these are the result of a modulation of individual insect behaviors (Staddon, 1983). For example, ants modify their paths according to environmental changes, succeeding, for instance, to avoid an obstacle, or to recover from a faulty situation. Recent progresses in Ecology heavily contributed to identify the presence of different personalities in bee colonies, furthermore ants have shown to be able to develop behaviours not only within colony, but also individually. The intra-colony variability is vital to survive and to respond to environmental changes in a flexible manner. Several researchers tried to formulate the possible theory about the origin and co-existence of different personalities. Raine *et al.* (2006) argue about population variability, whereas Nettle (2006) discusses how heterogeneity and variation of environments seem to play an important role to induce this diversity.

The novelty of the proposed work derives from the consideration that often swarm intelligence is not only the result of cooperation among ‘simple’ individ-

uals, but depends on ‘individual abilities’. Research on distributed and swarm intelligence often underestimates the capabilities of the individuals. For instance insects are not reflex automata, though they are often considered as such. Rather they show interesting capabilities, as individuals, like attention, categorization, capability to distinguish the concept of sameness and difference, to find new solutions, and others (Chittka and Niven, 2009). The authors have been involved in modeling and designing an insect brain computational model inspired by *Drosophila melanogaster*. Despite its tiny brain, it is able to fulfill complex tasks, and there are genetic tools used to modify specific areas and functionalities of the brain in order to understand their role in different behaviours (Arena and Patané, 2014). While trying to model the single behaviors from a bio-mimicking neural perspective, using the most advanced control methods for implementing locomotion (Arena *et al.*, 2011a) and cognition dynamics (Arena *et al.*, 2012a,b,c), the authors have been more and more aware of the richness of behaviours in fly individual, in terms of learning and decision capabilities. These enable the insects to actively interact with the environment and decide appropriate strategies. *Drosophila melanogaster* is definitely not a social insect: it does not have a nest, the only social behaviors, discovered up to now, are mating and aggression. It is therefore the best case to study interaction from the individual viewpoint. Individual capabilities are posed in interaction with the external world. Each individual namely interacts with the environment as if it is alone: the other agents are considered within the context as part of the environment, and their own behavior can be exploited to increase the individual satisfaction. This is exactly the finality of mating or aggression.

This paper analyzes the adaptive behavior of robots in a particular noisy environment, where noise is caused by actions of other robots: this leads to a competitive/cooperative scenario. Here the adaptive control strategy implemented through interrelation among synaptic learning and neural dynamic adaptation within the individual control system can lead to the emergence of a flexible collective behavior which gradually increments, in real-time, the performance of each individual and the overall performance of the group.

In order to analyze the interaction between the two learning mechanisms, namely neural adaptation and synaptic plasticity, a number of simulation experiments have been considered investigating the advantages of applying them simultaneously or separately. A realistic situation is that a robot interacts with a noisy environmental condition due to the presence of the other robots. Performance in such cases were evaluated: only two robots are considered in this paper for the sake of clarity and for a thorough description. This approach has a direct rela-

tion with the ethological counterpart: individual arthropods (e.g. bees, ants, fruit flies) can mutate their behaviours in a flexible way, exploiting different learning mechanisms (Robinson, 1992; Dornhaus and Franks, 2008). In these scenarios, the individual capabilities subject to learning in a dynamic environment induce the emergence of specialization: this is an adaptive dynamical and environment-driven process (Li *et al.*, 2002; Potter and Jong, 1995; Tuci *et al.*, 2008) with interesting consequences in the global performance. For this reason, the results experimentally obtained in this paper potentially become the inspiration for future research in bio-inspired computation in robot colonies, focusing attention on the individual capabilities rather than on the population.

The manuscript is organized as follows: the control structure within each robot will be depicted, starting from the dynamics of the neuron embedded into the robot controller, passing through the learning algorithm, and then describing the network architecture. Then the problem definition will be presented through the description of the experimental setup, the robot mechanical and sensor structure as well as the environment characteristics are described. Thus, the results obtained through extensive simulation campaigns will be presented. Finally, conclusions and future perspectives will be drawn.

2. The Neural Control scheme

Insects show several different learning capabilities for adaptation in the real world (Menzel and Giurfa, 1996; Chittka and Niven, 2009; Liu *et al.*, 1999).

Among the different insect species we focused our attention on the development of an insect brain computational model mainly focused on the *Drosophila melanogaster*, the fruit fly (Arena *et al.*, 2011b). Within the insect brain two important neuropiles are the Mushroom Bodies and the Central Complex (CX). The latter is the brain structure mainly involved in visual tasks; it is composed by three neuropils, namely the Protocerebral Bridge (PB), the Ellipsoid Body (EB) and the Fan-shaped Body (FB). In particular, The PB allows the fruit fly to know where the visual object of interest is located in the visual range; in other species, like the flesh fly, the PB is also a center where mechanosensor information is processed (Phillips-Portillo, 2012). The FB, on the other hand, has an important role in storing object features to be used for learning. The FB can be modeled as a multi-network control architecture based on spiking structures where unconditioned stimuli (i.e. reward and punishment) are used to trigger a correlation based learning, as shown below. The Ventral Lobes, an accessory area of the CX, are believed to convey pre-processed visual information from the CX to motor areas for

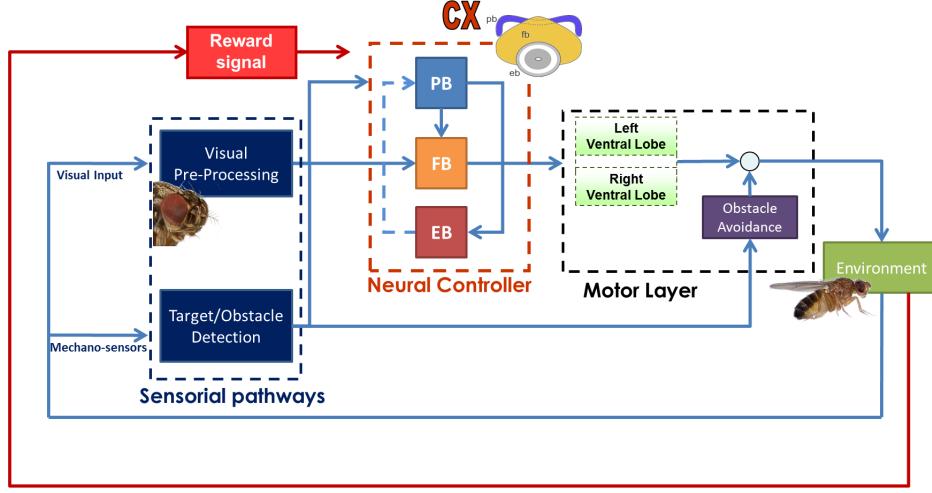


Figure 1: Block diagram of the control system architecture. Three main blocks are depicted: *Sensorial pathway* where the information from the sensors is collected, a *Neural Controller*, resembling the Central Complex in the insect brain, where the spiking network evolves on the basis of the acquired sensory signals and a *Motor Layer* where the action is selected and executed in the dynamic simulated *Environment*.

steering control (Triphan *et al.*, 2010). In the following a neural model of the FB has been implemented. It was designed using a biologically inspired neural network with spiking neurons. Due to their properties Spiking Neural Networks are specifically suitable to fast and efficient applications, where the timing of input signals carries important information, providing powerful tools for analysis and investigations of processes in the brain structures. Moreover, spiking neurons are computationally more powerful and biologically plausible than perceptrons and sigmoidal gates (Maass and Graz, 1997; Maass, 1997). For these structures, bio inspired strategies were developed to modulate synaptic connections. As a consequence, spiking versions of supervised, unsupervised and reinforcement learning techniques can be introduced. A number of control signals are used to obtain learning and specialization. The former is a function of the external reward signal; it implements an adaptive proportional control gain on the neuron bandwidth, facilitating or inhibiting its spiking activity. The latter contributes to the synaptic plasticity, as will be deeply discussed below.

Fig.1 shows the block diagram of the overall control system: the *Neural Controller*, modelling through spiking network the basic functionalities of the CX, acquires external information from the *Sensorial pathways* including, in particu-

lar, the *Visual* and *Mechano sensory inputs*. The output of the *Neural Controller* guides the *Motor Layer* modulating the appetitive/aversive behaviour of the system. In particular, *Obstacle Avoidance* is a reflex induced reaction which is automatically performed whenever a robot meets walls or other robots. The Reward signal (R_w), generated by the environment, triggers the learning mechanisms on the Neural Controller, as detailed below, to specialize the system on the basis of the actions previously performed. A hearing source or a flashing light are examples of potential signals stimulating a reward or a punishment.

Since a foraging task has been implemented, the different environment setup impose a series of realistic constraints to each robot. Multiple targets with different color (Col1 = yellow and Col2 = blue) are present in the environment and generally the task is composed of four phases:

Phase 1 - only yellow targets are visible in the arena;

Phase 2 - a robot reaches one yellow target: then blue targets become visible to the robots. After reaching a yellow target, the robot remains there for a given time to simulate feeding conditions;

Phase 3 - a robot reaches one of the blue targets. If concurrently another robot is on one yellow target the condition *two robots-two colors* is realised. The reward signal is then activated;

Phase 4 - blue targets are now disabled and a new foraging task begins.

These simple rules, imposed by the environment, prevent a robot from reaching a rewarded state independently from the actions of the other robots, even if each robot does not have any information about the other robot behaviors. This leads to specialize in different behavioral actions. The arousal of this specialization is reached thanks to the fact that, in case of reward, each robot undergoes a learning phase, leading to increase its willingness to find the color currently reached (considered as the source of reward) and decreases its interest in the other. Learning details are given below.

2.1. The Neuron model

The spiking neural network introduced in this paper is made-up of class I excitable Izhikevich neurons: they reproduce the main firing patterns and neural dynamical properties of a biological neuron, guaranteeing, at the same time, low computational burdens (Izhikevich, 2003). Class I configuration is particularly

suitable for implementing sensing neurons, being the spiking rate proportional to the amplitude of the stimulus. The basic dynamics of the neuron is depicted in the model equations as follows:

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

with the spike-resetting

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

where v , u represent, respectively, the neuron membrane potential and the recovery variable, and they are dimensionless variables. a , b , c and d are system parameters and in particular, $a = 0.02$, $b = -0.1$, $c = -55$, $d = 6$. I models the pre-synaptic input current. This is composed of two main contributions and will acquire relevance for the learning method, presented below. The time unit is ms .

2.2. Neural control through STDP learning

To model the synapse connecting a neuron j with a neuron i , we assumed that the synaptic input to neuron j is given by the following equation:

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

where t_i indicates the instant in which a generic neuron i , connected to neuron j emits a spike, w_{ij} represents the weight of the synapse from neuron i to neuron j and the function $\varepsilon(t)$ describes the contribution of a spike from a presynaptic neuron emitted at $t = 0$ (Floreano and Mattiussi, 2001), according to:

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

In our simulations $\tau = 5ms$. In order to model the biological synaptic modulation, the Spike Timing Dependent Plasticity (STDP) was adopted (Song *et al.*, 2000). In this method synaptic weights are controlled through a function of correlations between pre- and post-synaptic spikes. In details it holds:

$$w(k+1) = w(k) + \sum_{l=1}^n \Delta w_l\tag{5}$$

where Δw_l represents the weight variation associated to the l -th couple of pre-post synaptic spike ($l = 1, \dots, n$) and k denotes the index of the time window used for weight update, here fixed to $300ms$.

$$\Delta w_l = \begin{cases} A_+ e^{\frac{\Delta t_l}{\tau_+}} & \text{if } \Delta t_l < 0 \\ A_- e^{\frac{-\Delta t_l}{\tau_-}} & \text{if } \Delta t_l \geq 0 \end{cases} \quad (6)$$

$\Delta t_l = t_{pre} - t_{post}$ represents the difference between the spiking time of the pre-synaptic neuron (t_{pre}) and the post-synaptic one (t_{post}), referred to the l -th spike couple. During a learning cycle, the synapse is reinforced when $\Delta t_l < 0$, (i.e. when the post-synaptic spike occurs after the pre-synaptic spike); otherwise when $\Delta t_l \geq 0$, the synaptic weight is decreased. The terms A_+ and A_- represent the maximum values, obtained for equal pre- and post- spiking times. The synaptic rule in eq. 4 may lead to an unrealistic growth of the synaptic weights, it was proposed in Song and Abbott (2001) and Izhikevich (2007) to fix upper limits for the weight values, whereas in Verschure *et al.* (1992) and Verschure and Pfeifer (1992) a decay rate in the weight update rule was introduced. This solution tries to avoid the increase of weights and allows a continuous learning to be implemented. In our experiments the parameters are chosen in a range compatible with biological findings available in literature (Arena *et al.*, 2009b; Drew *et al.*, 2006) as follow: $A_+ = 0.8$, $A_- = -0.8$, $\tau_+ = 7ms$, $\tau_- = 2ms$, whereas the upper bound of the weights value is set to 32, the lower bound is set to 0 and the decay rate is set to 2% each robot performed action (i.e. robot step).

2.3. Neural control through threshold adaptation

Threshold adaptation has a solid biological background: in fact, it can be seen as a consequence of the nonlinearities presented in the neuron membrane dynamics (Izhikevich, 2004). Input-output function adaptations for auditory neurons involved in sound coding were accurately detected and studied (Dean *et al.*, 2005). Moreover this mechanism seems to produce emergent cooperative phenomena in a large population of neurons, and seems to be responsible for contrast adaptation (Dragoi *et al.*, 2000; Greenlee and Heitger, 1988), or for scaling adaptation to varying stimuli in the somatosensory cortex (Garcia-Lazaro *et al.*, 2007). In our control model the reward signal will act on specific neurons in the neural controller by increasing or decreasing their input current. In particular, the threshold of the punished neurons will be incremented: repeated punishment will make neurons unable to elicit any response. On the other hand, the threshold of the

rewarded neurons will be decreased: this will lead to an increasing facilitation to emit spikes when stimulated.

Biological studies have reported its relevance in such cases as the forward masking of weaker stimuli (Sobel and Tank, 1994) or the selection response to fast stimuli (Peron and Gabbiani, 2009). In Neurobiology, the adaptation mechanism should be modeled as either an adaptation current or a dynamic threshold: both mechanisms result in a similar adapting spiking rate and the two methods are mostly comparable (Liu and Wang, 2001). Moreover, adaptation current reproduces the properties of the more realistic conductance-based model for integrate-and-fire neurons, even if the two mechanisms have a qualitatively different effect on the neuron transfer function. Furthermore, spike-frequency adaptation can be modeled using adaptation current, because dynamic firing threshold seems not to be the cause of spike-frequency adaptation, but it is a secondary effect, i.e. the result of an adaptation current action (Benda *et al.*, 2010).

In our implementation, the input current I in eq. 1 is split in two contributions: an input I_i , that accounts for both external stimuli (e.g. sensorial stimuli) and synaptic inputs, and the I_A , that represents a bias subject to the adaptation effects. The I in the eq. 1 becomes:

$$I = I_A + I_i \quad (7)$$

In particular, the Threshold adaptation process can be modelled as a voltage-dependent current and so the term I_A in eq. 7 can be expressed as $I_A = g_A V_{thresh}$, defining g_A as an activation-conductance. The current can be modified to hyperpolarize or depolarize ($I_A \leftarrow I_A \pm \Delta I_A$) neurons. I_A acts as an incremental adaptation current depending on the reward signal (Benda *et al.*, 2010; Benda and Herz, 2003). In details, I_A is modulated if the target sensory neurons are active (through the input I_{Target}) and concurrently a reward signal comes from the environment. The adaptation scheme here proposed is based on a solid biological background. Neuro-control schemes were in fact introduced recently which explain olfactory conditioning in insects using feedforward signals modulated by external reward/punishment events (Gerber *et al.*, 2004; Arena and Patané, 2014).

2.4. The Neural Feedforward Controller Structure

In this section the *Neural Controller* and the *Motor Layer* blocks, are explained in details (see Fig. 1). The spiking network (Arena *et al.*, 2009a,b), used as starting point for this new learning mechanism, consists of two main parallel pathways: one acting as a classical feedback controller, and another acting as a

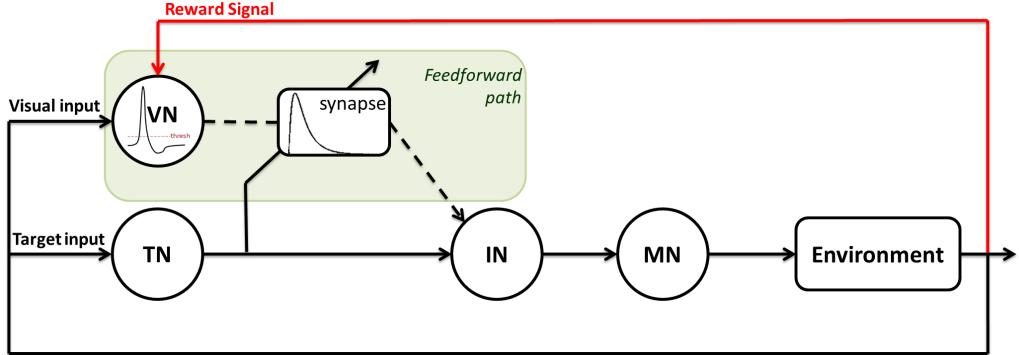


Figure 2: The feedback/feedforward control loop element within the neural controller.

trainable feedforward path. The feedback controller is not subject to learning and it is used as reference dynamical model for the trainable feedforward one. Fig.2 shows the details of the control loop in a simplified case involving the neuron chain. Feedback signals from the environment are here divided into low level and high level signals. Low level are provided by color sensors fixed in the ventral area of the agent. They respond only if the agent is located on a specific colored area. High level sensors are here represented by a camera used to detect colored objects from a distance spanning the whole arena size. Low level sensors constitute the input for the Target Neuron (TN), whose activity certifies that the robot has reached a given colored target. Concurrently high level sensor signals constitute the input to the Vision Neuron (VN) which fires whenever a specific colored target is within the visual range. Learning takes place when the agent reaches the colored target: now both TN and VN fire. Since synapses in the feedforward path are not trained, initially only TN is able to elicit a motor neuron (MN) response through interneuron (IN) fire. Now both VN and IN get fired and this elicits STDP for the synapse from the VN to IN. Therefore TN-IN dynamics constitutes the reference for the VN-IN neural group and this feedforward controller will learn to stimulate the MN with a targeting action whenever VN is active. This trainable feedforward synapse is drawn in dashed line. A detailed analysis neuro-computational aspects of this control method are reported in (Arena *et al.*, 2009a). Another control action is here added to the loop: if specific conditions are satisfied in the environment, for example if we are in *phase 3* previously presented, the reward signal activation acts by controlling the input current to the VN, making it more or less prone to fire in front of visual stimuli. This causes specialization of the agent to specific stimuli. The two control actions, namely specialization and STDP, can be used

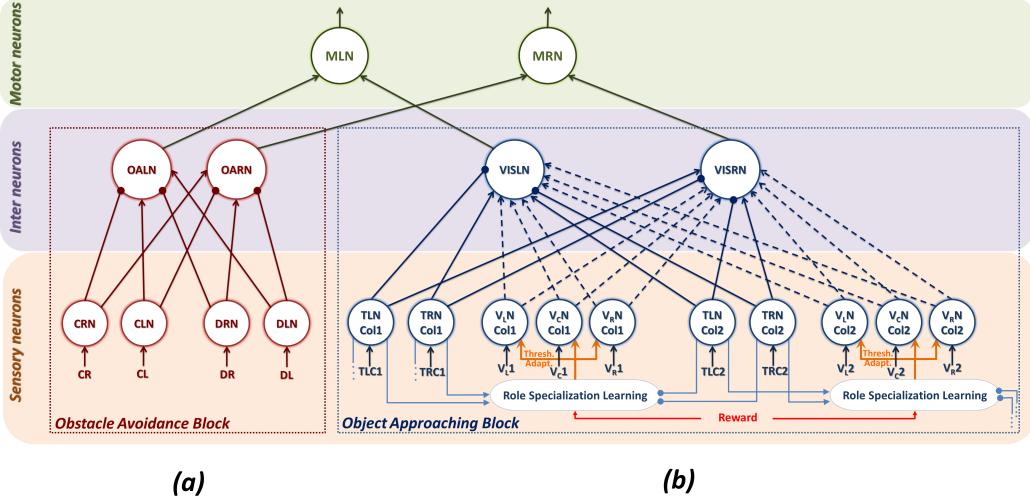


Figure 3: The neural network used to control each robot is composed by two subnetworks: **(a)** Obstacle avoidance subnet - *CL/CR*: contact left/right signal; *DL/DR*: distance left/right signal. *CLN/CRN* and *DLN/DRN* are the sensory neurons to modulate obstacle avoidance actions through the inter-neurons *OALN/OARN*. **(b)** Object approaching subnet - *TLCx/TRCx*: target left/right for color x ; $V_{L/C/R}N_x$: vision (left,central,right) sensor signals for Color x ($x = 1,2$); these are the stimuli whose motor action has to be learned. *TLN/TRN* and *V_{L/C/R}N* are sensory neurons that act on the motor neurons (*MLN/MRN*) through the inter-neurons *VISLN/VISRN*.

separately or concurrently, as will be discussed in the following Sections.

The whole neural controller is shown in Fig.3. Here are visible the details for the obstacle avoidance network (Fig. 3 (a)) and for the subnetwork related to visual targeting (Fig. 3 (b)). Trainable feedforward synapses are here again depicted in dashed lines. Both sub-networks contribute to control the robot movements, through motor-neurons that modulate the velocity of the wheels on the left ($M_L N$) and right ($M_R N$) side of the robot. Walls and other robots in the environment are considered as obstacles and the robot learning is performed managing also obstacle detection and overcoming (through the obstacle avoidance block), whose contribution heavily influences the final robot specialization. The obstacle avoidance subnet has higher priority in order to avoid collision if an obstacle is detected. The block dedicated to visual target detection and object approaching permits the robot to identify and reach specific targets. The Role Specialization learning starts when a reward function is activated and the Threshold adaptation of the activated vision neurons is performed, in order to reinforce or weaken the corresponding sub-network.

The network consists of three layers and generalises the scheme of Fig.2 with

the addition of an obstacle avoidance block. The first layer is constituted by sensory neurons that are connected to motor-neurons (*motor layer*) through a layer of inter-neurons. Referring to Fig. 3, contact ($CL-CR$) and target ($TLCx-TRCx$) sensory neurons elicit reflexive reactions (i.e. obstacle avoidance and target reaching): they are not subject to learning. The network learns targeting motor responses to vision inputs ($V_LCx-V_CCx-V_RCx$). These occupy specific positions on the robot, as will be clarified in the Section 3. All the trainable synapses (depicted in dashed lines in Fig. 3) act so as to incrementally generate, through STDP learning, feedforward pathways from the visual input signals to the motor neurons, as explained above. The feedforward path will, at the end, take the lead, anticipating motor targeting as soon as the target will appear in the robot vision field. The weight of the anticipative effect will depend on the value of the synaptic efficiency. In fact at the beginning, visual stimuli are not able to trigger any response and motor control is managed only by the target sensor network. The learning process incrementally activates the new path to the motor layer modifying the corresponding synaptic weights by STDP. The additional control, acting on neuron Threshold through I_A modulation, works as follows: if the target sensors fired, for example $TLN\ Col1$ stimulate by Col1, and the reward signal is activated, due to the concurrent actions of another robot, the currents I_A of the Col1 subnet are increased, whereas the I_A currents for neurons sensitive to Col2 are weakened. Consequently, the neurons related to the inhibited subnet become less sensitive to Col2, until no response occurs in presence of Col2. All I_A currents are initialized to the same value: $I_A = 20$; instead I_i can assume two possible values: $I_i = 0$ if no input occurs or $I_i = 8$ otherwise. Using this initialization, together with a threshold value for the input current $I_{A_{th}} = 22$, at the beginning the presence of a target is abundantly able to elicit a neuron response. The learning process influences the parameter I_A so, when this value sufficiently decreases, the contribution of I_i cannot overcome $I_{A_{th}}$ and the neuron is no longer sensitive to external stimuli. For the following arguments, it is important to underline that within each robot step the proposed SNN is simulated for a time window of 300ms: in our case one robot step corresponds to 2500 simulation steps with an integration time of 0.12ms. The robot motion direction will be selected according to the number of spikes generated by the left and right motor-neurons during the simulation time window.

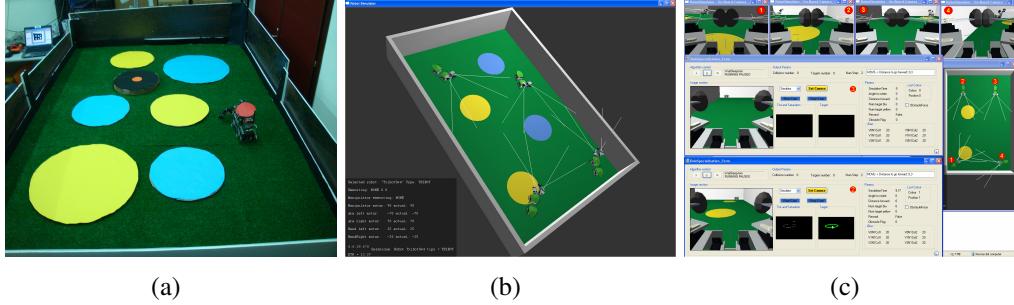


Figure 4: (a) The real environment with the prototype of the TriBot robot. (b) The simulated environment: an arena $3\text{m} \times 2\text{m}$ with four targets (two yellow and two blue) and four simulated TriBot robots. (c) The simulated environment and the GUI of SPARKRS4CS. On the left-middle and left-bottom side, the graphical interface related to two robots are shown. Subfigures from the top left to the top right show the robot on-board cameras, whose field of view was fixed to $\pm 45^\circ$.

3. Experimental Setup

The experimental case taken into account is related, as reported above, to a classical foraging task. The robot experiments were implemented using the SPARKRS4CS architecture, a software/hardware framework realized to allow the development of cognitive systems with the addition of an *ad hoc* 3D Dynamic Robotic Simulator (Arena *et al.*, 2011c). The framework is able to perform experiments both in real and simulated environments; however, due to the missing of multiple prototypes of the TriBot robot, all the reported results refer to dynamical simulations. Fig. 4 (b) shows a screenshot of the overall dynamic environment used to perform the experiments. It directly resembles the real arena used for real robot experiments (Fig. 4 (a)). Simulation tests were carried out in an arena ($3\text{m} \times 2\text{m}$) filled with a number of two different targets (yellow and blue) distributed on the floor. In these scenarios, the initial position of the robot and targets are randomly determined.

The robots used in the experiments are the simulated version of the bio-inspired hybrid mini-robot, named TriBot I (Arena *et al.*, 2010; SPARK I Project, 2007; SPARK II Project, 2011). It is a hybrid robot developed to investigate cognitive capabilities inspired by insects. TriBot I is composed of three modules, the first two contain wheels made of three-spokes appendages that improve the dexterity of the structure (Schroer *et al.*, 2004). The front module is composed of two standard legs with 3 degrees of freedom, each one connected to the main body.

The characteristic capabilities were already tested on the robot in previous works (Arena *et al.*, 2006), and could be useful to boost the cooperation abil-

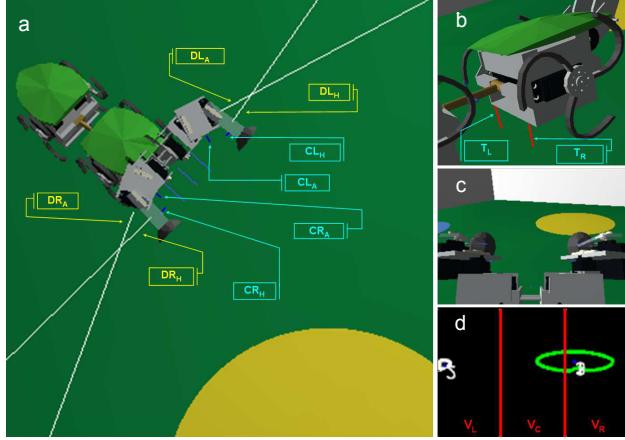


Figure 5: Overview of the TriBot robot model, where simulated sensors are highlighted. **(a)** Distance sensors: DL_A (Distance Left Arm), DL_H (Distance Left Hand), DR_A (Distance Right Arm) and DR_H (Distance Right Hand). Contact sensors: CL_A (Contact Left Arm), CL_H (Contact Left Hand), CR_A (Contact Right Arm), CR_H (Contact Right Hand). **(b)** Low level sensors: T_L (Target Left), T_R (Target Right). **(c)** on-board camera captured image. **(d)** processed image.

ties. Fig. 5 gives an overview of on-board sensors and camera embedded on TriBot I. Obstacle avoidance is performed using four distance sensors as shown in Fig. 5 (a): DL_A (Distance Left Arm), DL_H (Distance Left Hand), DR_A (Distance Right Arm) and DR_H (Distance Right Hand). On each side, the lowest distance value calculated by the sensors is used to provide the DL and DR input value to the neural network, respectively (see Fig. 3(a)). TriBot is also endowed with contact sensors: CL_A (Contact Left Arm), CL_H (Contact Left Hand), CR_A (Contact Right Arm), CR_H (Contact Right Hand); if at least one of the two sensors is under a given threshold, the respective CL/CR input is triggered (see Fig.3(a)). Furthermore the robot is endowed with Low level colour sensors, T_L (Target Left) and T_R (Target Right), used to detect when a robot reach one of the targets placed in the area (Fig. 5(b)). Each robot includes an on-board camera (see Fig. 4(c)) that acquires information (Fig. 5(c)) that is then processed by the visual system (Fig. 5(d)). Using a standard image processing algorithm, the image is partitioned in three sectors to identify the position of the object in the visual field for orientation purposes. The sector is selected depending on the centroid position of the detected target in the scene. V_L , V_C and V_R represent respectively the left, central and right sectors of the segmented visual field according to visual information is spatially mapped on the visual neurons (see Fig.3(b)). If multiple targets are detected in the scene, the object with the largest area is selected. The field of view

is $\pm 45^\circ$. The characteristics used for the simulated camera (e.g. field of view, position on the robot, etc.) are related to the real device equipped on the robot. Here visual information is extracted using a bio-inspired smart camera able to process in parallel and with a high frame rate the visual flow, following the paradigm of the Cellular Neural Networks (Arena *et al.*, 2011d; Alba *et al.*, 2009, 2010).

4. Results and analysis

STDP and threshold plasticity have been applied together. This is useful to investigate the interplay between Role Specialization mechanism and STDP learning, to estimate the improvement in speeding-up Role Specialization and to evaluate the possible improvement of the learning convergence.

The first series of experiments includes robots already trained through STDP: they are able to recognize and reach all the targets (Arena *et al.*, 2009b): so here only the effect of Threshold adaptation can be evaluated. The rules are the same as reported above: the robots start searching targets; when a robot reaches the target area a second target area is made visible, and so the other robot can reach it. When the final target is reached and the *two robots-two targets rule* is satisfied, the reward function is activated. In these conditions if a robot has currently reached a specific target (say Col1), the corresponding target neurons ($TLNC_{Col1}$, $TRNC_{Col1}$) activate their own role specialization learning block (see Fig. 3(b)) in order to increase I_A for vision neurons belonging to the same subnet (V_LNC_{Col1} , V_CNC_{Col1} , V_RNC_{Col1}). Moreover $TLNC_{Col1}$ and $TRNC_{Col1}$ will act on all the other role specialization learning blocks to decrease I_A . In this way, the *Role Specialization* learning mechanism will reward the active subnet and punish the other subnets. At the end of the learning process, each robot will be specialized to reach a specific colored target. In the second series of experiments, the weights are not pre-trained. At the beginning, no targets are known and robots perform only default behavior, i.e. exploration, which simply consists of going straight forward in the arena and avoid obstacles. When a robot by chance walks on a colored target, the corresponding target sensors are activated concurrently with visual sensors: this induces a weight update through STDP. If the reward is also activated, a threshold learning phase is triggered. To validate the efficacy of our learning method, a set of testing simulations have been performed and some simulation results are discussed to show the influences and interactions between the two learning methods.

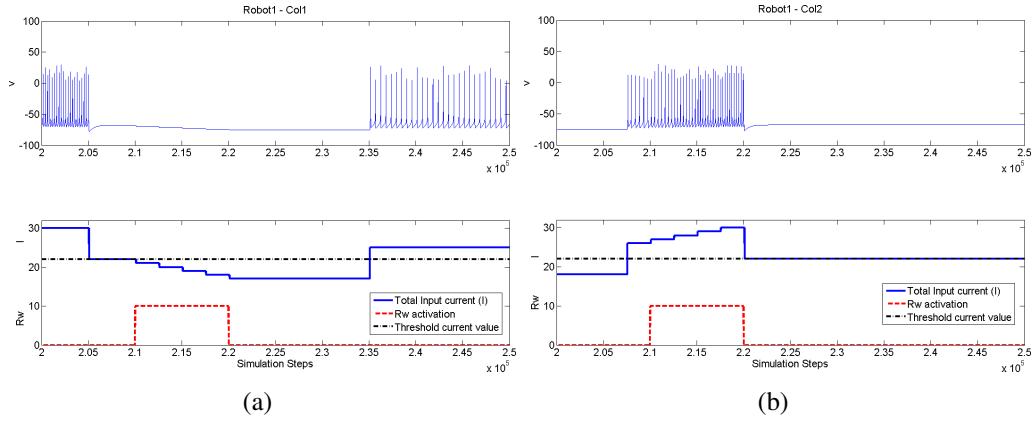


Figure 6: Threshold adaptation effects: Membrane Potential, Reward activation and Total Input Current ($I = I_A + I_i$) of a vision neuron for Robot 1 are shown. (a) - Hyperpolarization: the input current decreases during the reward activation. The spike train frequency tends to decrease in front of subsequent stimulations; this will lead to emit no spikes at the end of the learning. (b) - Depolarization: the adaptation current increases reaching the upper bound, increasing the neuron spiking rate.

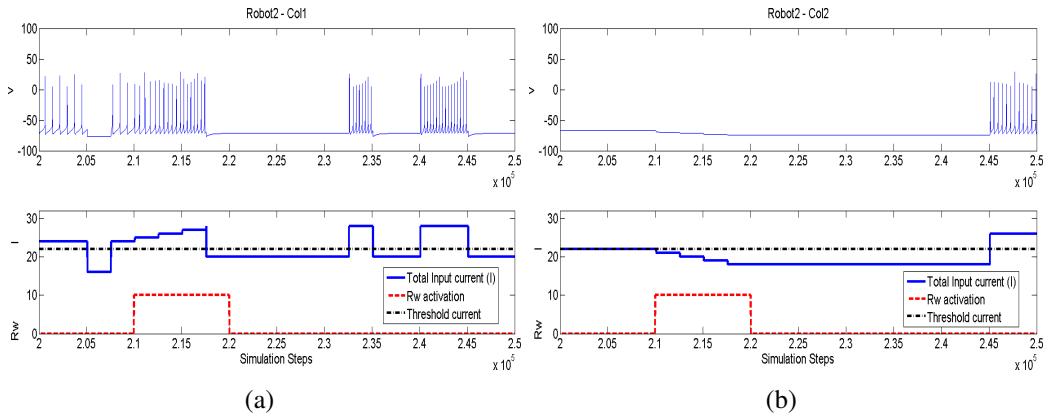


Figure 7: Example of simulation results: Membrane Potential, Reward activation and Total Input Current of a vision neuron for Robot 2 are shown. (a) - In this case a depolarization of the neuron occurs in front of a reward signal. The adaptation current increases during an input excitation, and the spike train increases. (b) - Hyperpolarization of a vision neuron. Even if no input is present, a decrease of the current is learned during the reward activation, producing a decrease in the spike train frequency.

4.1. Role Specialization through Threshold adaptation

In these series of experiments the capabilities of threshold adaptation are evaluated. Initially each robot is sensitive to two targets, characterized by two different colors. This kind of learning will lead to a specialization towards a single colored target. Fig.6 and Fig.7 show an example of the learning mechanism: the reward function is elicited when Col2 is activated, after the activation of Col1. In the time window around $2.0 * 10^5$ simulation steps, a specific vision neuron for Robot 1 is currently stimulated by Col1 (Fig. 6(a)), but no threshold adaptation takes place, since, at the same time, Robot 2 is not stimulated by Col2; clearly in this time window both robots are in the same colored target (Fig. 6(a) and Fig. 7(a)). The reward signal is therefore silent. On the contrary, around $2.07 * 10^5$ simulation steps, Robot 1 vision neuron is stimulated by Col2 (Fig. 6(b)) and concurrently Col1 target is within the visual range of Robot 2. So its vision neuron fires (Fig. 7(a)). Around $2.1 * 10^5$ Robot 1 arrives to Col2 target (Fig. 7(b)): the *two robots-two targets rule* is satisfied and the reward signal is activated. Robot 1 vision neuron sensitive to Col2, is depolarized (by increasing I_A) whereas Col1 vision neuron is hyperpolarized. On the other side, in Robot 2 the effect for each subnet is exactly the opposite. The emerging behavior is a specialization effect for Robot 1 to Col2, for Robot 2 to Col1, respectively.

4.2. Role Specialization with threshold adaptation and STDP Learning

Another set of experiments use threshold adaptation in combination with synaptic plasticity. Initially the robots need to learn the correct behavior in front of a visible object. The two learning mechanisms are here combined in order to perform a complex global learning procedure. During the experiment, as said above, if the robot reaches a target previously seen through the visual system, target neurons are stimulated and the update of the corresponding synaptic weights for the feedforward (visual) pathway is performed. In addition, when the global *reward function* is activated, a cycle of the Specialization mechanism is performed through I_A update. All the tests performed have shown the remarkable positive influence of the threshold adaptation over the STDP learning in the synapses related to the vision neurons for the color which the specific robot is specialized for. This result was predictable since initially the retrieval of the target is random, being the environment and the objects unknown by the robots. At the beginning, robots show no reaction to visual stimuli but, during the learning phase, the increase of synaptic efficiency and the threshold adaptation work concurrently. In this way, a speed-up in the learning process is obtained: the initial randomness encourages one of the potential roles, since robots are prone to find the first rewarded target

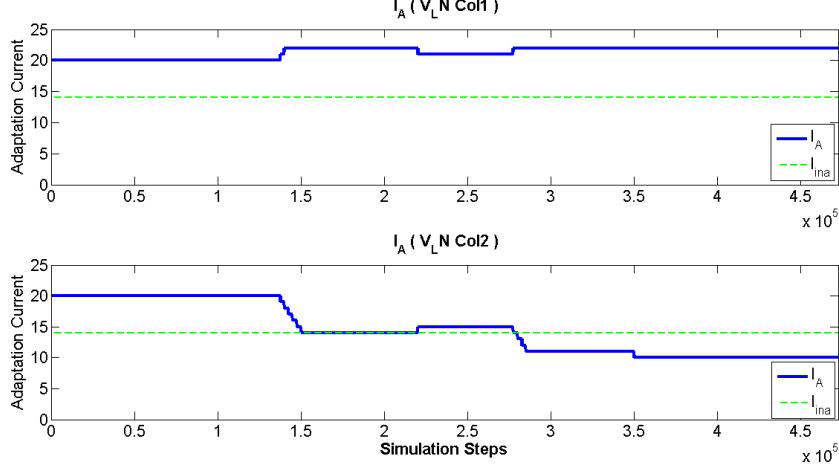


Figure 8: Trends of adaptation current: high values of I_A are maintained for V_LNCol1 . This Vision neuron remains sensitive to its input stimulus, whereas I_A of V_LNCol2 is decreased even below the Inactivation current level I_{ina} (dashed line). This neuron during time loses sensitivity to its input.

more frequently than other targets. At the same time the relative synapses are updated and contemporarily the other behavioral choices are inhibited, improving the learning convergence. Fig. 8 shows the neuron current (I_A) for a Vision neuron related to the subnet that is going to be reinforced (V_LNCol1) and for a Vision neuron of the subnet that will be inhibited (V_LNCol2). In the first one, the current is maintained at high values and so the sensitivity of this neuron is maintained: whereas, the currents of the other neurons are decreased. During time, for these last neurons it becomes more and more difficult to reach and exceed the threshold value ($I_{A_{th}} = 22$); the effect consists in an ever decreasing spiking rate until the lower bound current value (I_{ina}) is overcome and no more spikes are emitted, even in presence of a visual input. As regards synaptic weights, Fig. 9 displays how only the subnet devoted to the selected target is trained, whereas the weights of the neurons related to the other target are maintained very low. Fig. 10 shows the trend of the number of targets (N_Y for yellow targets, N_B for blue ones) found by the two robots during a simulation. In Fig. 10(a) Robot 1 specializes in yellow targets. Results show clearly that N_Y is much higher than N_B . Fig. 10(b) shows the outcome of Robot 2 that specializes in the blue targets at the end of the learning. During exploration in the arena it can happen that robots detect targets not as a result of targeting, but accidentally. Fig. 11 shows the trend of the synaptic weights when only STDP learning was applied, showing experimental evidence

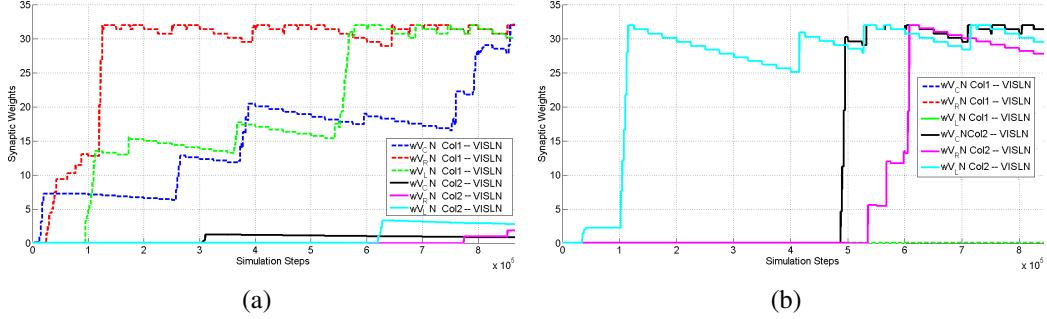


Figure 9: Role Specialization with STDP Learning: **(a)** - trends of the synaptic weights for the robot specialized in yellow (Col1) targets. The weights related to Col1 reach upper bounds, whereas the weights of blue targets (Col2) keep lower values. **(b)** - trends of the synaptic weights for the robot specialized in blue targets. In this case, the weights related to Col2 reach upper bounds, whereas the weights of Col1 remain negligible: the robot quickly becomes sensitive to the blue target. Even if it reaches a yellow one, no learning mechanisms will be activated.

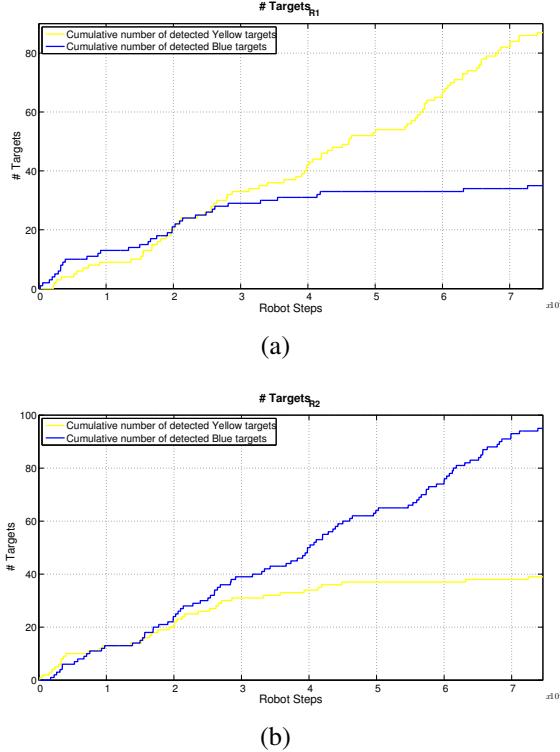


Figure 10: Cumulative number of reached targets: **(a)** related to the robot which is going to specialize in yellow (Col1) target, **(b)** related to the robot which is going to specialize in blue (Col2) target.

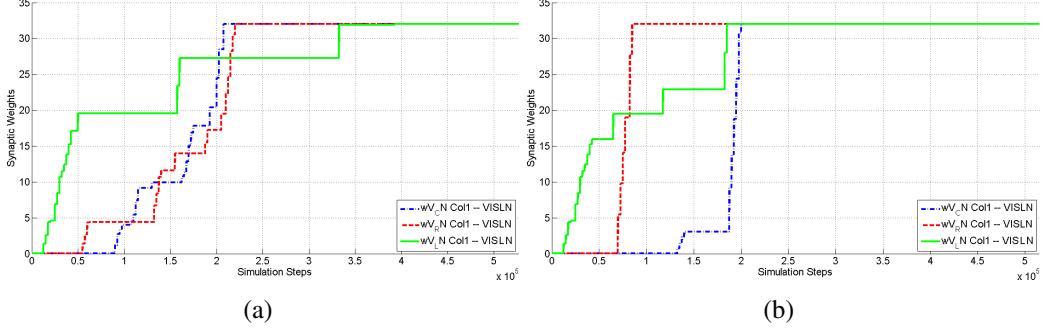


Figure 11: **(a)** STDP learning without threshold adaptation: trends of the synaptic weights for yellow target (Col1). The weights reach their upper bound after 3.3×10^5 simulation steps (corresponding to 132 robot steps). **(b)** - Threshold adaptation and STDP Learning: trends of the synaptic weights for Col1. The weights saturate around 2×10^5 simulation steps (corresponding to 80 robot steps).

of the improvement in convergence when both learning methods are applied.

4.3. Specialization with and without STDP: Comparisons

To perform statistical comparisons, a set of simulation tests was performed. No decay effect was applied to the synaptic weights: in this way, learning is considered complete when all values reach their upper bounds. The results reported in Fig. 12 refer only to one of the two robots. In this case, we can see that using only STDP, the weights require more simulation steps to reach their steady-state values, than when applying also Threshold adaptation. This comparison shows an improvement in the range of 30 – 50% in simulation steps. The results are shown in Table 1. The same comparison was carried out to evaluate the performance of the Threshold adaptation with and without the STDP learning concurrently active. The results show that the specialization becomes slowly convergent when acting together with STDP learning. This aspect is directly related to the global reward and its activation. In fact, to perform a Threshold adaptation learning cycle, both robots must reach different targets. In the first steps, when the weights are not tuned, the robots spend a lot of time exploring the arena, so the specialization learning starts to converge later.

Fig.13 displays a comparison between the current I_A for a vision neuron which is going to loose sensitivity to its input in presence of the concurrent action of Threshold adaptation and STDP, and when threshold adaptation was applied on an already trained network. In the first case, the specialization was completed after 180 robot steps, whereas in the second case just after 41 robot steps. This is a clear

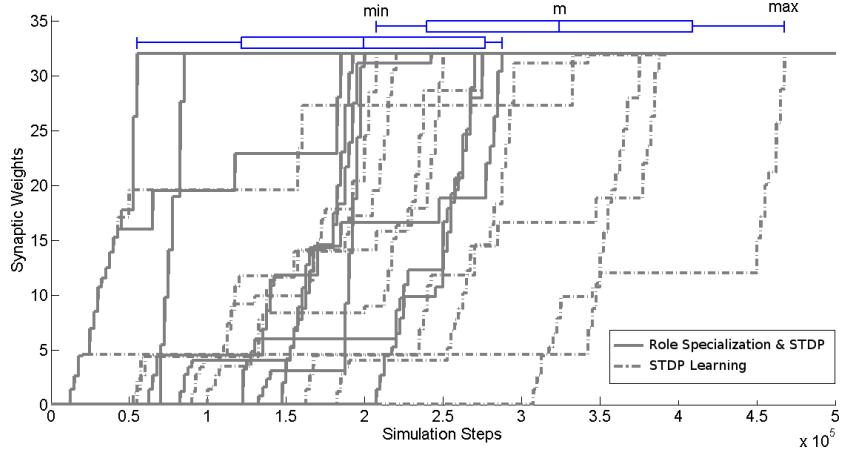


Figure 12: Statistical comparison of results. Trend of the synaptic weights with and without the activation of the threshold adaptation during STDP: results of ten trials for each case. The weights of the network trained only via STDP are reported in dotted line, whereas in solid line the trend of weights with STDP learning in association with threshold adaptation are shown. The event used to compare performance is the simulation step when weights reach their upper bound.

Table 1: Statistical comparisons between STDP learning with and without Role Specialization. The mean value of the time elapsed for the convergence of the synaptic weights together with other statistical parameters are reported.

Specialization with STDP	simulation steps	robot steps
mean value (m)	$2 * 10^5$	80
standard deviation (σ)	$0.78 * 10^5$	31
min	$0.55 * 10^5$	22
Max	$2.9 * 10^5$	115

Specialization without STDP	simulation steps	robot steps
mean value (m)	$3.2 * 10^5$	130
standard deviation (σ)	$0.85 * 10^5$	34
min	$2.1 * 10^5$	83
Max	$4.7 * 10^5$	187

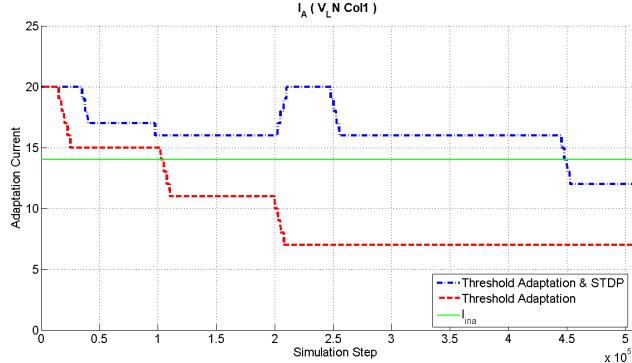


Figure 13: Comparison of the adaptation current (I_A) evolution with active STDP learning and with already tuned weights. The inactivation current level is $I_{ina} = 14$. When the current falls below this lower bound (around 4.5×10^5 and 1×10^5 simulation steps for upper and lower line respectively) the neuron does no longer emit spikes, even in presence of an input I_i .

result: in a network already trained to reach all the targets, it is relatively simple to specialize the agent to a particular target class. This is a potential benefit, since preferences could be easily modified by adjusting the reward function, without changing completely the system knowledge. This is particularly interesting in case of dynamically changing environment.

Fig. 14 shows a typical trajectory before and after learning, respectively, for an experiment with two targets and two robots. Here Role Specialization is tested using already tuned weights. At the beginning of learning, robots are both attracted by all targets present in the arena. This is evident from the initial trajectory: both robots try to reach the first active (yellow) target (placed in the bottom right part of the arena). These trajectories show the presence of competition among robots, indicated with a circle in Fig. 14(a). Here the two robots, attracted by the same target, interact with each other and Robot 1 turns right to avoid the other agent, that will be able to reach the yellow target. After learning, in Fig. 14(b) each robot is interested only in a specific type of target. Here, at the beginning of the simulation the blue target is not yet visible, so Robot 2 goes straight (i.e. the default exploration strategy) whereas the yellow target attracts Robot 1. At 10 robot steps the blue target is enabled, since Robot 1 has just reached the yellow target. Even if Robot 2 can see both targets, it proceeds directly towards the blue one. Finally, at 18 robot steps Robot 1, which has already reached the yellow target starts to explore the environment, whereas Robot 2 directs toward the blue one to complete the mission.

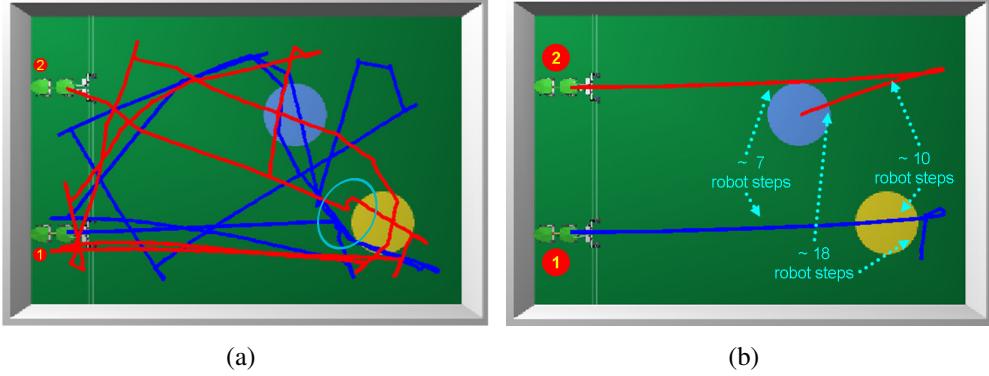


Figure 14: Example of robot trajectories during Threshold adaptation with tuned weights. At the beginning of learning (a), robots are sensitive to both targets. The threshold adaptation causes gradual specialization to only one target. After learning (b), Robot 1 is specialized in yellow targets and Robot 2 in blue ones.

5. Specialization Indexes

In this section, we discuss about the metrics for measuring specialization in our system. Traditionally, quantify the global performance corresponds to identify a functional index to optimize. For this reason, a simple and general criterion is to calculate the number of rewards triggered during the trials, i.e. how many times the task is properly accomplished, according to a problem-specific difference measure. Following these concepts, we can define the following Specialization index:

$$S(t) = \frac{\# \text{Task accomplished events}}{\# \text{Tot. events}} \quad (8)$$

A typical trend of the index is shown in Fig.15, where eq.(8) is used to probe the evolution of specialization during an experiment. The task is considered accomplished if the targets reached when the reward is triggered match the final behaviors of the robots. A statistical analysis over 20 experiments is shown in Fig.16. According to the figure, we can assert that statistically after 150 R_w successful events (N_{Rw}^w) all experiments converged to the final solution.

Another typical metric is to cluster agents with similar task-specific measure and look at the obtained pattern in the feasible solution space; the number of clusters represents the measure of diversity in the system. In ecology, diversity indices are used to provide information about agents' composition and relative abundances of different species. Since specialization is quantified in term of di-

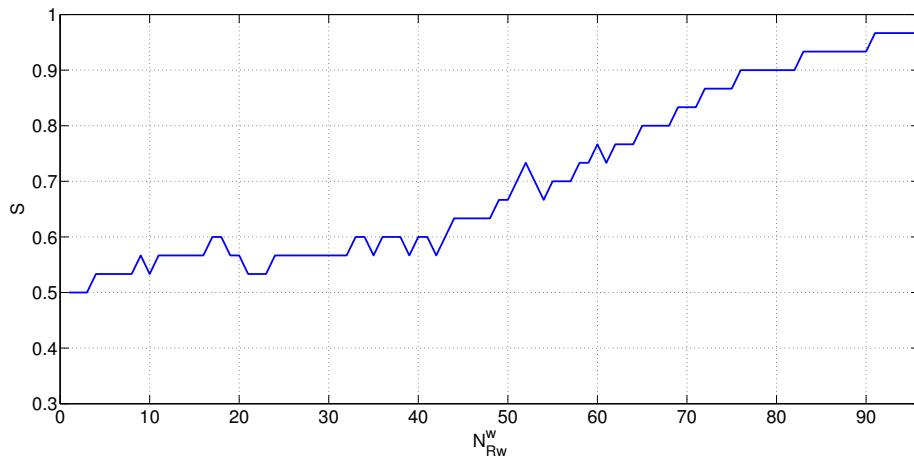


Figure 15: Evolution of Specialization index $S(t)$. A typical evolution of Specialization index during an experiment. A sliding window (N_{Rw}^w) of 30 reward events is used.

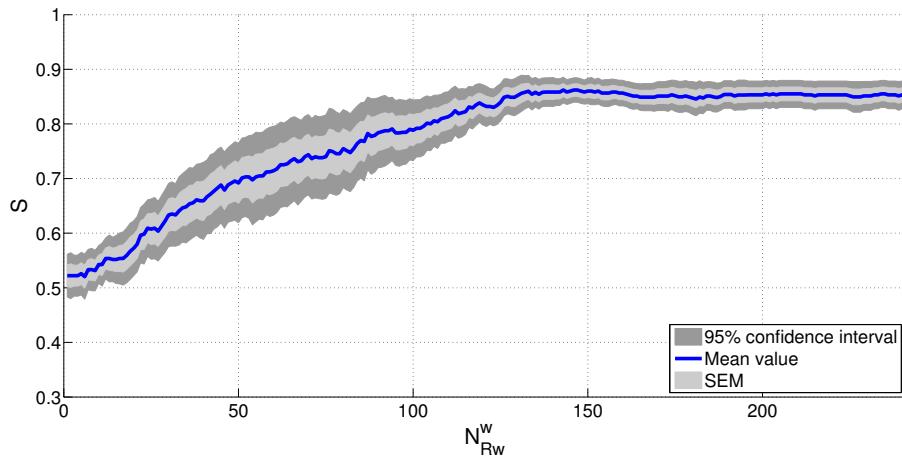


Figure 16: Statistical trend of Specialization index $S(t)$. The blue line represents the mean of the specialization evolution in a set of 20 experiments. The light gray area represents the standard error of the mean (SEM), whereas the dark gray area represents the 95% confidence interval. The index is calculated using a sliding window of w reward events ($w = 30Rw$).

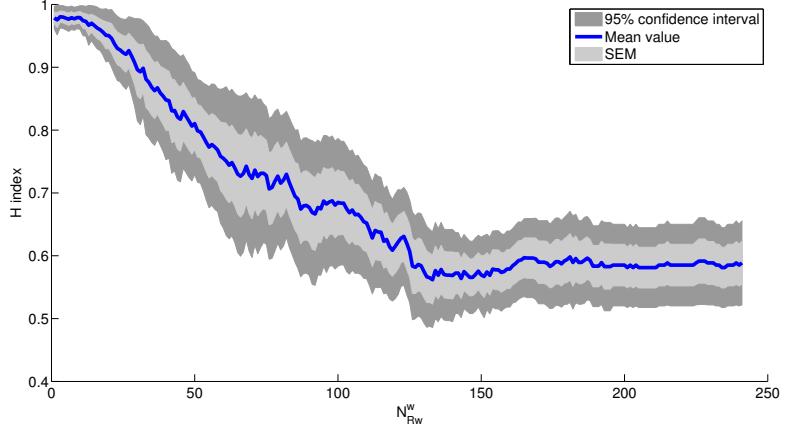


Figure 17: $H(t)$ index. The blue line represents the mean of the specialization entropy. The light gray area represents the standard error of the mean (SEM), whereas the dark gray area represents the 95% confidence interval.

versity, Shannon’s entropy and Simpson’s diversity (Shannon, 1948; Simpson, 1949) are two interesting metrics used to evaluate the evenness (equitability or diversity) in relation to the distribution among different species. Successively Balch (2002) introduced a metric to measure diversity, called *simple social entropy*, as an application of Shannon’s entropy to scenarios involving multiple agents.

It is defined as:

$$H(\chi) = \sum_{i=1}^n p_i \log_2(p_i) \quad (9)$$

where n are the different sub-groups identified in the system χ and p_i represents the portion of agents in the i -th cluster. Starting from these concepts, we decided to use the $H(\chi)$ metrics in a different way: we evaluate the entropy ($H(k)$) of our system χ during the experiment, starting from the eq.(9) as

$$H(k) = -H(\chi, k) = -\sum_{i=1}^n p_i(k) \log_2(p_i(k)) \quad (10)$$

where k represents the time evaluated in terms of sliding windows of reward events and $n = 2$ represents the subgroups in which individuals can be divided (respectively: sensible to both targets, sensible to a single target). Following this assumption the evolution of the trend in the different number of clusters followed by the agents is our diversity measure (see Fig. 17). According to the social entropy index, which makes a meaningful measure by incorporating information

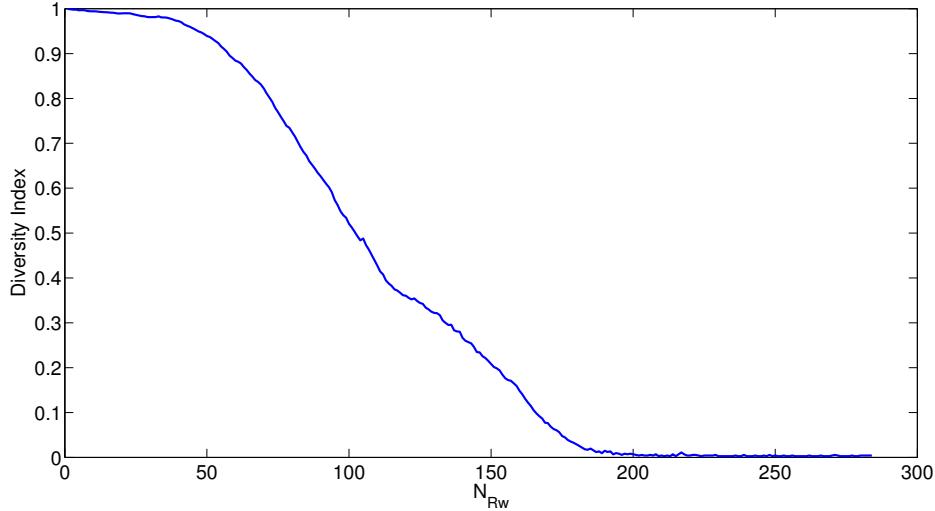


Figure 18: Simpson’s Index. The value ranges between 0 and 1, where the greater value represents evenness, on the contrary the lower value corresponds to diversity. The index shows how Threshold adaptation’s uptime induces different learning convergence.

about the spatial distribution of the clusters, H starts from a high value that indicates the robots, sensible to both targets, are included in the same subgroup. During time the index value decreases theoretically toward 0, when the system is totally specialized and the individuals are splitted in two opposite and decoupled states.

In our case the steady state value is around 0.5 because even small fluctuations in the subgroup division are emphasized by the presence of a logarithmic function. To resolve this problem, according to Simpson (1949) and Meyer and McIntosh (1992), we can correlate the decrease of the index directly to the diversity as here indicated:

$$D(k) = 1 - \sum_{i=1}^s q_i^2(k) \quad (11)$$

where $s = 4$ is the number of adaptation currents involved in the Threshold adaptation learning. It depends on the number of targets, whereas q_i represents the total sensitivity to whole targets, proportionally to the normalized sum of adaptation currents of the agents in the system. Accordingly with this definition, the index, in Fig.18, shows how our system starts with a complete evenness and converges to a total diversity at the end of the training.

6. Conclusion

In this paper a nonlinear controller realised using a spiking neural network is presented. The neural network, designed to control motion in a group of agents roving in a simulated environment, is a reduced scale structure of the Central Complex, a neuropile met in the insect brain and acting as visual object classifier and high level motion controller. The STDP based learning algorithm contributes to generate a neural feedforward controller able to anticipate basic motor actions computed by a neural feedback controller in each single robot. Another control law, acting on the threshold of specific sensory neurons contributes to modulate robot sensitivity to specific targets, implementing an adaptive real-time targeting motor control on all the robots, which therefore show specialization for specific targets. Simulations, statistical results and specific performance indexes assess the suitability of the approach, showing how neural control strategies, within each single agent, produce individual adaptation and a kind of labour division without needing any direct communication among the robots. The task presented in this paper considers the simple case of specialization to color preference: the method can be extended to control artificial swarms and colonies, where each agent can learn, and possibly re-learn, its favorite role within the group for the emergence of teams.

Acknowledgement

This work was supported by EU Project EMICAB, grant no. 270182.

References

- Alba, L., Arena, P., De Fiore, S., Patané, L., Strauss, R. and Vagliasindi, G. (2010). Implementation of a Drosophila-inspired orientation model on the Eye-RIS platform. In *12th International Workshop on Cellular Nanoscale Networks and their Applications (CNNA10)*, Berkley, pp.1-6, 3-5 February 2010.
- Alba, L., Morillas, S., Listan, J., Jimenez, A., Arena, P., and Patané, L. (2009). Embedding the AnaFocus Eye-RIS vision system in roving robots to enhance the action-oriented perception. In *Proc. of Microtechnologies for the New Millennium (SPIE 09)*, Dresden (GERMANY), 4-6 May 2009, 7365-08.
- Arena, E., Arena, P., and Patané, L. (2011a). Efficient hexapodal locomotion control based on flow-invariant subspaces. In *18th World Congress of the International Federation of Automatic Control (IFAC)*, Milan, Italy.

- Arena, P. and Patané, L. (2014). *Spatial Temporal Patterns for Action-Oriented Perception in Roving Robots II: an insect brain computational model*. Springer, Series: Cognitive Systems Monographs, Vol. 21.
- Arena, P., Fortuna, L., Frasca, M., Patané, L., and Pavone, M. (2006). Implementation and experimental validation of an autonomous hexapod robot. In *Proc. of the IEEE International Symposium on Circuits and Systems*, pages 401–406, Kos, Greece.
- Arena, P., Fortuna, L., Frasca, M., and Patané, L. (2009a). Learning anticipation via spiking networks: Application to navigation control. *IEEE Transactions on Neural Networks*, **20**, 202–216.
- Arena, P., de Fiore, S., Patané, L., Pollino, M., and Ventura, C. (2009b). STDP-based behavior learning on the TriBot robot. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*.
- Arena, P., Patané, L., Pollino, M., and Ventura, C. (2010). Tribot: a hybrid robot for cognitive algorithm implementation. In *18th IEEE Workshop on Nonlinear Dynamics of Electronic Systems*, (NDES 2010), 26-28 May 2010, Dresden.
- Arena, P., Patané, L., Termini, P. S., Vitanza, A., and Strauss, R. (2011b). Software/hardware issues in modelling insect brain architecture. In *Proceedings of the 4th international conference on Intelligent Robotics and Applications - Volume Part II*, ICIRA’11, pages 46–55.
- Arena, P., Cosentino, M., Patané, L., and Vitanza, A. (2011c). SPARKRS4CS: a software/hardware framework for cognitive architectures (invited paper). In *Proceedings of the SPIE - The 5th SPIE’s International Symposium on Microtechnologies*, volume 8068, pages 8068A–18, Prague, Czech Republic.
- P. Arena, S. De Fiore, L. Patan, L. Alba, S. Strauss (2011d). Drosophila-inspired visual orientation model on the Eye-Ris platform: experiments on a roving robot. In *Proceedings of the SPIE - The 5th SPIE’s International Symposium on Microtechnologies*, volume 8068, Prague, Czech Republic.
- Arena, P., Patané, L., and Termini, P. (2012a). Learning expectation in insects: a recurrent spiking neural model for spatio-temporal representation. *Neural Networks*, **32**, 35–45.

- Arena, P., Patané, L., and Termini, P. S. (2012b). Modeling attentional loop in the insect mushroom bodies. *International Joint Conference on Neural Networks (IJCNN 2012)*.
- Arena, P., Patané, L., Stornanti, V., Termini, P., Zaepf, B., and Strauss, R. (2012c). Modelling the insect mushroom bodies: Application to a delayed match to sample task. *Neural Networks (special issue on Autonomous Learning)*, doi: **10.1016/j.neunet.2012.11.013**.
- Balch, T. (2002). *Robot teams: From diversity to polymorphism*, chapter Measuring robot group diversity, pages 93–135. Balch, T. and Parker, L.E., natick, ma: a k peters edition.
- Benda, J. and Herz, A. V. M. (2003). A universal model for spike-frequency adaptation. *Neural Comput.*, **15**(11), 2523–2564.
- Benda, J., Maler, L., and Longtin, A. (2010). Linear Versus Nonlinear Signal Transmission in Neuron Models With Adaptation Currents or Dynamic Thresholds. *Journal of Neurophysiology*, **104**(5), 2806–2820.
- Chittka, L. and Niven, J. (2009). Are Bigger Brains Better? *Current Biology*, **19**(21), R995–R1008.
- Dean, I., Harper, N. S., and Mcalpine, D. (2005). Neural population coding of sound level adapts to stimulus statistics. *Nature Neuroscience*, **8**(12), 1684–1689.
- Dornhaus, A. and Franks, N. R. (2008). Individual and collective cognition in ants and other insects (hymenoptera: Formicidae). *Myrmecological News*, **11**(August), 215–226.
- Dragoi, V., Sharma, J., and Sur, M. (2000). Adaptation-Induced Plasticity of Orientation Tuning in Adult Visual Cortex. *Neuron*, **28**(1), 287–298.
- Drew, Patrick J. and Abbott, L. F. (2006). Extending the effects of spike-timing-dependent plasticity to behavioral timescales. *Proceedings of the National Academy of Sciences*, **103**(23), 8876-8881.
- Floreano, D. and Mattiussi, C. (2001). Evolution of spiking neural controllers for autonomous vision-based robots. In *T. Gomi (Ed.), Evolutionary Robotics IV*, Berlin. Springer-Verlag.

- Garcia-Lazaro, J., Ho, S., Nair, A., and Schnupp, J. (2007). Shifting and scaling adaptation to dynamic stimuli in somatosensory cortex. *European Journal of Neuroscience*, **26**(8), 2359–2368.
- Gerber, B., Tanimoto, H., and Heisenberg, M. (2004). An engram found? Evaluating the evidence from Fruit Fly. *Curr. Op. in Neurobiol.*, **14**(6), 737–744.
- Greenlee, M. W. and Heitger, F. (1988). The functional role of contrast adaptation. *Vision Research*, **28**(7), 791–797.
- Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, **14**(6), 1569–1572.
- Izhikevich, E. M. (2004). Which Model to Use for Cortical Spiking Neurons? *IEEE Trans. on Neural Netw.*, **15**(5), 1063–1070.
- Izhikevich, E. M. (2007). Solving the Distal Reward Problem through Linkage of STDP and Dopamine Signaling. *Cerebral Cortex Advance*, **17**(10), 2443–2452.
- Li, L., Martinoli, A., and Abu-mostafa, Y. S. (2002). Emergent specialization in swarm systems. pages 261–266. Springer-Verlag.
- Liu, L., Wolf, R., Ernst, R., and Heisenberg, M. (1999). Context generalization in *Drosophila* visual learning requires the mushroom bodies. *Nature*, **400**, 753–756.
- Liu, Y. H. and Wang, X. J. (2001). Spike-frequency adaptation of a generalized leaky integrate-and-fire model neuron. *Journal of Computational Neuroscience*, **10**, 25–45.
- Maass, W. (1997). Noisy spiking neurons with temporal coding have more computational power than sigmoidal neurons. In M. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 211–217. MIT Press (Cambridge).
- Maass, W. and Graz, T. U. (1997). Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, **10**, 1659–1671.
- Menzel, R. and Giurfa, M. (1996). Dimensions of cognitive capacity in an insect, the honeybee. *Behav. Cogn.*, **5**, 24–40.

- Meyer, P. and McIntosh, S. (1992). The usa today index of ethnic diversity. *International Journal of Public Opinion Research*, **4**(1), 56–58.
- Nettle, D. (2006). The evolution of personality variation in humans and other animals. *American Psychologist*, **61**(6), 622–631.
- Peron, S. and Gabbiani, F. (2009). Spike frequency adaptation mediates looming stimulus selectivity in a collision-detecting neuron. *Nature Neuroscience*, **12**, 318–326.
- Phillips-Portillo, J. (2012). The central complex of the flesh fly, *Neobellieria bullata*: Recordings and morphologies of protocerebral inputs and small-field neurons. *The Journal of Comparative Neurology*, **520**(14), 3088–3104.
- Potter, M. A. and Jong, K. A. D. (1995). Evolving neural networks with collaborative species. In *In Proceedings of the 1995 Summer Computer Simulation Conference*, pages 340–345.
- Raine, N., Ings, T., Dornhaus, A., Saleh, N., and Chittka, L. (2006). Adaptation, genetic drift, pleiotropy, and history in the evolution of bee foraging behavior. *Adv. in the Study of Beh.*, **36**(06), 305–354.
- Robinson, G. E. (1992). Regulation of division of labor in insect societies. *Annual Review of Entomology*, **37**(1), 637–65.
- Schroer, R. T., Boggess, M. J., Bachmann, R. J., Quinn, R. D., and Ritzmann, R. E. (2004). Comparing cockroach and whegs robot body motion. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3288–3293, New Orleans.
- Shannon, C. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, **27**, 379–423.
- Simpson, E. (1949). Measurement of diversity. *Nature*, **163**, 688–688.
- Sobel, E. C. and Tank, D. W. (1994). In Vivo Ca²⁺ Dynamics in a Cricket Auditory Neuron: An Example of Chemical Computation. *Science*, **263**(5148), 863–826.
- Song, S. and Abbott, L. (2001). Cortical Development and Remapping through Spike Timing-Dependent Plasticity. *Neuron*, **32**(2), 339–350.

- Song, S., Miller, K. D., Abbott, L. F., and Program, N. G. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neurosci*, **3**, 919–926.
- SPARK I Project (2004–2007). SPARK: Spatial-temporal Patterns for Action-oriented perception in Roving robots, EU ICT-FP6 Project.
- SPARK II Project (2008–2011). SPARK II: EU ICT-FP7 Project.
- Staddon, J. (1983). *Adaptive Behaviour and Learning*. Cambridge University Press.
- Triphan, T., and Poec B., and Neuser, K., and Strauss, R. (2010). Visual Targeting of Motor Actions in Climbing Drosophila. *Current Biology*, **20**(April, 13), 663–668.
- Tuci, E., Ampatzis, C., Vicentini, F., and Dorigo, M. (2008). Evolving homogeneous neurocontrollers for a group of heterogeneous robots: Coordinated motion, cooperation, and acoustic communication. *Artif. Life*, **14**(2), 157–178.
- Verschure, P. and Pfeifer, R. (1992). Categorization, representations, and the dynamics of system-environment interaction: a case study in autonomous systems. In *From Animals to Animats: Proceedings of the Second International Conference on Simulation of Adaptive behavior*, pages 210–217. MIT Press.
- Verschure, P., Kröse, B. J., and Pfeifer, R. (1992). Distributed adaptive control: The self-organization of structured behavior. *Robotics and Autonomous Systems*, **9**(3), 181–196.