

## Chapter 2

# A Computational Model for the Insect Brain

P. Arena, L. Patanè and P. S. Termini

**Abstract** As seen in the Chap. 1, the fruit fly *Drosophila melanogaster* is an extremely interesting insect because it shows a wealth of complex behaviors, despite its small brain. Nowadays genetic techniques allow to knock out the function of defined parts or genes in the *Drosophila* brain. Together with specific mutants which show similar defects in those parts or genes, hypothesis about the functions of every single brain part can be drawn. Based upon the results reported in the Chap. 1, a computational model of the fly *Drosophila* has been designed and implemented to emulate the functionalities of the two relevant centres present in insects: the Mushroom Bodies and the Central Complex. Their actions and inter-actions are adapted from the neurobiological perspective to a computational implementation. A complete block scheme is proposed where the proved or conjectured interactions among the identified blocks are depicted. Several simulations results are finally provided to demonstrate the capability of the system both considering specific parts of the complete structure for comparison with insect experiments, and the whole model for more complex simulations.

### 2.1 Introduction

In the bio-inspired robotics field, robots can be used to reproduce animal behavior in order to study their interaction with the environment. Robots help to improve the understanding of animal behavior and animals help to create efficient and robust robotic systems. The study of animal brains leads to new control systems that could

---

P. Arena (✉) · L. Patanè · P. S. Termini  
Department of Electrical, Electronic and Computer Science Engineering,  
University of Catania, 95125 Catania, Italy  
e-mail: parena@diees.unict.it

L. Patanè  
e-mail: lpatane@diees.unict.it

allow robots to be able to orient themselves in complex environments, to take decisions, to accomplish dangerous missions, in order to become completely autonomous. Robotic implementation of biological systems could also lead to the introduction of new models for basic sciences, in particular when investigating the emergent properties of models. Several attempts are present in literature related to algorithms or bio-inspired networks able to mimic the functionalities of parts of the brain. A lot of work has been done in several animal species belonging to mammals, mollusks and insects [1]. Looking into the insect world different research groups around the world are trying to design models which are able to reproduce interesting behaviors shown by insects: cooperation mechanisms in ants [2], navigation strategies in bees [3], looming reflex in locusts [4], homing mechanisms in crickets [5], central pattern generator and obstacle climbing in cockroaches [6, 7], reflex-based locomotion control in the stick insect [8], just to cite some examples. It is evident that the effort is focused on specific peculiarities associated with the different insect species that can be also useful for robotic applications. Nevertheless, a more challenging task consists of trying to model the main functionalities of an insect brain, looking from an higher level, trying to identify the mechanisms involved in the sensing-perception-action loop. The proposed work is focused on the development of an insect brain computational model mainly focused on the *Drosophila melanogaster*, the fruit fly. The insect brain architecture, structured in functional blocks, has been developed in a complete software/hardware framework in order to evaluate the capabilities of this bio-inspired control system on both simulated and real robotic platforms. In order to develop an useful and suitable architecture, the proposed framework is flexible and robust and presents a structure suitable to decouple simulations from control algorithms. The functional separation helps to isolate the application itself from graphic interfaces and the underlying hardware. The main aim is to develop an extensible and general purpose architecture. The insect brain model has been evaluated in scenarios strictly linked to the neurobiological experiments to make a direct comparison. Moreover the available data on wild type flies and mutant brain-defective flies allows to identify the main role of each neural assembly in performing specific tasks like visual orientation, olfactory learning, adaptive termination of behaviours and others. Finally the main guidelines used for the definition of evaluation criteria and the creation of benchmarking scenarios where the system performance can be evaluated, are also reported.

## 2.2 Insect Brain Cognitive Architecture and Learning Issues

In the previous chapter a first model of the interplay between MB and CX was presented. In that model there is not a specific block representing a specific function of the MB or of the CX; the interest was focussed to functional aspects. In the following a model useful in view of a robotic implementation will be first considered. This builds upon a previously designed model [13] and outlines sensory motor pathways with the addition of learning and representation blocks. Subsequently biological aspects

will be more and more included leading to the most recent scheme of a complete insect brain computational model, which will be simulated and implemented for robotic experiments.

In this section a preliminary description of the insect brain cognitive architecture developed on the basis of the *Drosophila* experiments is given. An overview of the general architecture is reported and particular attention is devoted to the learning strategies that are envisaged inside the cognitive structure.

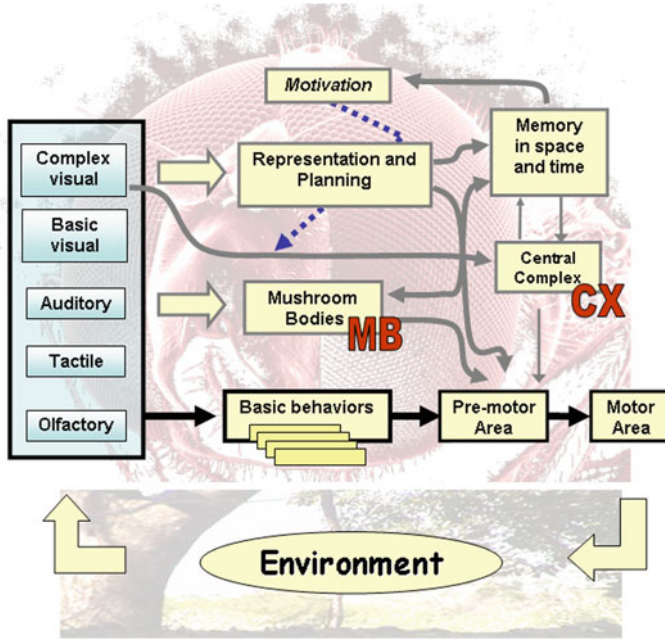
### 2.2.1 Various Steps in Modeling the Insect Brain

The actual insect brain computational model is the result of a number of previous versions, that were further refined and updated, once new results, especially on the neurobiological perspective, were available. The first perceptual architecture proposed was designed and organized in various control levels consisting of functional blocks, acting either at the same level, as competitors, or at distinct hierarchical levels showing the capability to learn more complex, experience-based behaviors [9].

The control architecture (as reported in Fig. 2.1) consisted of series of parallel sensory-motor pathways (i.e. basic behaviours) that were triggered and controlled by specific sensory events in a reflexive way, giving the knowledge baseline to the system. Going up in the hierarchical scheme, two relevant centers of the insect brain were considered: the Mushroom Bodies (MBs) and the Central Complex (CX). Taking into account the known facts about these centres, from a biological/neurogenetic point of view and their role in perceptual processes [9–11], some preliminary main functions were initially focussed, to be assessed and refined during the project activities. In particular, a function ascribed to MBs was to have a role, due to their learning capabilities, in the enhancement of causal relations arising among the basic behaviours, by exploiting the temporal correlation between sensory events; information storage and retrieval in the case of the olfaction sense; resolving contradictory cues through the visual sense by imposing continuation or adaptive termination of ongoing behaviour. Relevant functions ascribed to the CX were integration and elaboration of visual information, storing and retrieving information on objects and their position in space, controlling the step length in order to approach or avoid such objects; motor control, landmark orientation and navigation, orientation storage and others.

These learning aspects were treated using causal Hebbian rule in an array of spiking neurons for anticipation [12], on the basis of what already studied in [13], where memory structures based on Recurrent Neural Networks were considered.

At a higher level of the scheme, a representation layer was introduced, able to process sensory information in order to define the final behavior. Here we introduced a lattice of non spiking neurons. This neural lattice shows distinct characteristics of complex dynamical systems. The emerging patterns of neural states take on the meaning of percepts. These ones are then associated to suitable modulations of the basic behaviors. This modulation is performed through an unsupervised learning process which creates associations among sensory stimuli and patterns. In this way,



**Fig. 2.1** Functional block diagram of the initial version of the insect brain cognitive architecture. The interaction between the robot and the environment is realized by direct sensory-motor pathways, the *basic behaviors*, which are modulated by the representation layer. MB and CX are relevant centers of the insect brain devoted to temporal correlation, information storage and retrieval, and other functionality summarized in a correlation layer. Finally the high level functions of the representation layer consists of a *preprocessing block*, a *perceptual core*, a *selection network*, while the *Reward function* drives the learning process

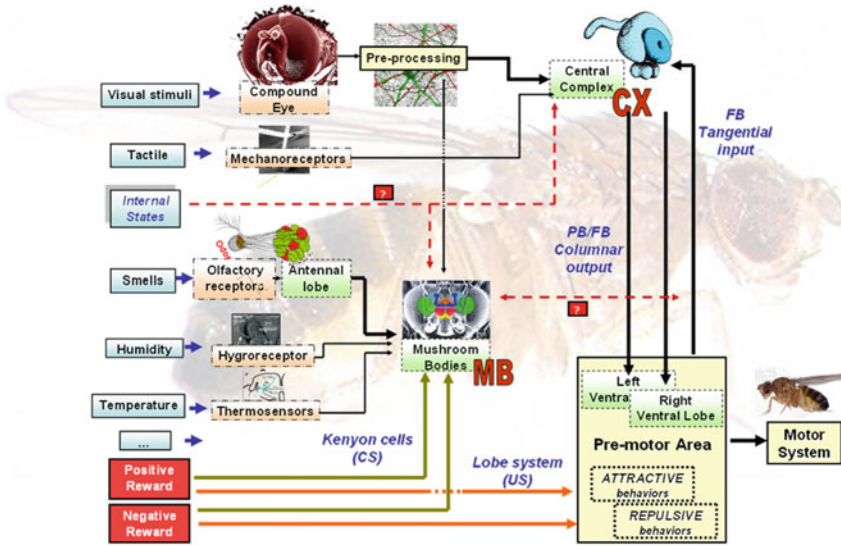
at the end of the leaning stage, each pattern represents a particular behavior modulation, while its trained basin of attraction represents the set of all the environment conditions, as recorded through the sensors, leading to the emergence of that particular behavior modulation. The modulation parameters associated with each pattern are learned through a reinforcement learning: here the reinforcement signal is provided by a motivation layer implementing the degree of satisfaction of the robot. This depends on the local satisfaction of the single basic behaviors with the addition of other terms that reflect the robot mission. The presence of additional information into the motivation layer, not used by the basic behaviors can be exploited by the Representation layer in order to increase the robot performance.

Memory was distributed in the whole architecture, but a specific block was also considered (i.e. Memory in space and time in Fig. 2.1). This block develops a contextual layer, like in [14]. Here sequences of successfully emerged patterns can be memorized to be retrieved when needed. The whole architecture was conceived in such a way that the basic behaviors, which are often life-saving sensory-motor pathways, are progressively enriched with emergent capabilities which incrementally

increase the animal skills. The main focus was therefore on the application of complex dynamics to obtain a proper, complex, context-learned modulation of the basic skills.

In a first attempt, following the results reported in [13], this process of complex emerging of situation related percepts was an important characteristic of our approach which makes it different from the other control strategies, based on the subsumption architecture proposed by [15]. The latter in fact, uses a high level approach to face with the design of both basic behaviors and the coordination block. In our strategy, complex dynamical systems are successfully used. Both architectures use a behavioral decomposition of the system to exploit parallel computation, although the Subsumption network makes a rigid hierarchy among the basic behaviors: the lower ones cannot influence the upper ones, whereas the latter can act on the former. In our scheme all the basic behaviors are sensory-motor pathways elicited by only one sensory modality and on the same hierarchical level: knowledge is incrementally built upon their modulation, giving importance to one or the other, depending on the context. Under this perspective the proposed architecture resembles the Motor Schemas, introduced by [16]. Turing Patterns in RD-CNN are hosted, in our architecture, within a layer here called *Representation Layer*. This term is here not referred to a place where a predictive model of the body-environment interaction is learned. This is rather a layer where the single-sensory motor modalities, constituted by the parallel sensory motor pathways, are modulated in a feedforward way, taking into account all the incoming sensory stimuli. This leads to the emergence of a contextually self organising activity, focusing at modulating the basic behaviors.

This was the insect brain model inspired by previous results [13] and object of an intense speculation phase. The aim of the research activity was to tightly link the emergent approach to cognition, based on nonlinear complex dynamics, to the knowledge gained from insect neurobiology. Therefore, the initial approach, briefly discussed above, mainly based on emergence and self organization, was modified to take into account the biological perspective. From a deep analysis of the state of the art and direct experiments performed, current knowledge from insect Neurobiology provided precious information on the details of the (mainly) low level information processing (i.e. excluding the representation level). Therefore our effort moved toward the lower level blocks. Once defined these parts, a suitable connection with the higher layers is envisaged. This is clear since the details about how insects gain a structuring of the whole information for decision making and planning is really a challenge, and at the present stage, it is unknown. So to cope with the lack of this information, the high level representation layer could be used to complete the architecture. Following such a path, the functional block scheme in Fig. 2.1 has been modified to include details from Neurobiology that led to a modification on handling the different sensory inputs and the sequence of processing steps involved, adding new details, mostly at the basic sensorimotor pathways and medium level of information processing. Moreover two different learning mechanisms were identified, which are mainly involved in this process: Classical Conditioning through positive and negative reward signals, and Operant Conditioning at the pre-motor area level. Fig. 2.2 shows the relevant elements of the updated insect brain architecture taken



**Fig. 2.2** Revised block diagram of the insect brain architecture. The diagram proposed in Fig. 2.1 is here modified to further match with the biological counterpart. Question marks indicate hypothesized connections that have to be still assessed

into consideration and the interaction among them. The architecture includes a series of sensorial stimuli acquired and preprocessed by the insect, that are successively handled by the two main structures taken into consideration: MBs and CX.

As it can be seen in Fig. 2.2, input sensory modalities are divided into different sensorimotor pathways:

- visual stimuli, through the compound eye and a pre-processing phase are further processed by the CX, which, with all its constituent parts, contributes to vision related functions like orientation (through direct connections to the ventral lobe), object detection, classification and memory.
- Tactile stimuli: there is behavioral evidence for mechanosensory information to be present in the *Drosophila* CX from the legs. Mechanosensation is the next best proven modality to be represented in the CX, besides the visual one.
- Smells and their connections to the MBs: MBs play a relevant role in olfactory processing, through input from the antennal lobe and olfactory receptors. They play also a role in context generalization starting from visual information. As far as the learning aspects are concerned, Classical conditioning mechanisms are constantly used in insects, and MBs seem to be the main center where learning takes place. Two distinct paths for positive and negative rewards exist. Another important learning mechanism occurs in the pre-motor area and it is basically an instrumental learning. Experiments shown how a long term memory can be created and the sleep phase is fundamental to stabilize and improve the learning process; for these reasons a kind of internal model is expected.



- Humidity and temperature conditioning are included into the model but not further directly exploited for the insect brain computational model as themselves. Indeed the information about the role and specific functions of MBs in these processing (like responses to temperature gradients) was used to perform specific experiments to study a memory effect called *memotaxis* in flies, in view of the robot implementation.
- Other functionalities can be considered taking information from data acquired through neuron recording in freely moving cockroaches and involving MBs processing. Their results suggest several until now unrecognized functions of the MBs: extrinsic neurons that discriminate between imposed and self-generated sensory stimulation, extrinsic neurons that monitor motor actions, and a third class of extrinsic neurons that predict episodes of locomotion and modulate their activity depending on the turning direction. The relevant neurons sent their processes generally ascend to other areas of the protocerebrum. Their results support the idea of multiple roles for the MBs. These include sensory discrimination, the integration of sensory perception with motor actions, and a role in place memory.
- Internal states and motivations are considered in the process but hypotheses and speculations will be done due to the lack of specific neurobiological evidences.

The evolution of the various stages of the model improvement leads to the scheme shown in Fig. 2.3.

In the insect brain block scheme it is possible to distinguish four main sensorial pathways; the olfactory and the visual pathways allow to perceive the environment, whereas gustation and nociception are indispensable to obtain information about the goodness or badness of the current situation. In particular the gustatory sensory modality, placed in the front legs of the fly, is reproduced in robotic experiments through signals coming from light sensors placed in the ventral part of the robots, facing with the ground. This modality is used in experiments like the adaptive termination of behaviours. Nociceptory signals, used for punishment, are reproduced through sound signals (or through the ventrally placed light sensors) and applied in such experiments as visual/odour learning. These sensorial pathways are linked together to make the system able to perform anticipatory actions to improve efficiency in finding rewards and to avoid dangerous situations. In the actual structure, learning is attained using mechanisms based on classical as well as operant conditioning. Olfactory and visual inputs, due to their complexity, are considered as pre-processed at the sensory level. Olfaction, has been studied at the aim to derive the corresponding MB neural models. Regarding the olfactory sensors, since the artificial ones are still too slow and difficult to be efficiently characterized, they were substituted by sound sensors, which are more reliable and able to provide both unconditioned and conditioned inputs to the neural processing network. Soon after the visual pre-processing stage we can find the Central Complex neuropil model, containing all its main components:

**PB** The Protocerebral Bridge (PB), which, in our model, performs its three main functions (Object Detection, Distance Estimation and Object Position extraction), as drawn by the biological experiments and neuro anatomical evidence;

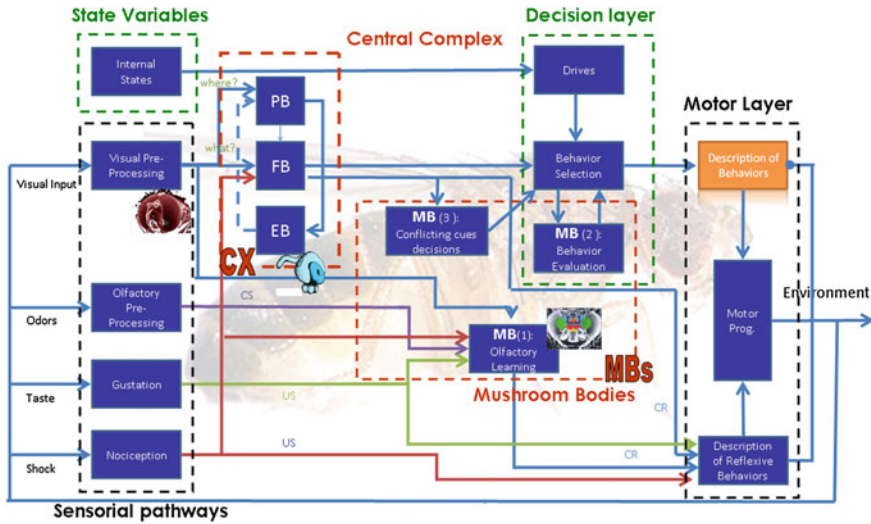


Fig. 2.3 Block diagram of the insect brain model

FB the Fan-shaped Body (FB), which performs two main functions: feature extraction (color, orientation, size, center of gravity, wideness, height) and feature evaluation/learning (the robot collects features and is able to associate those features to punishment or neutral situations).

EB the Ellipsoid Body (EB), where the robot spatial and the newly discovered decision making memory is formed and contained.

The other fundamental neuropil of the insect brain is constituted by the Mushroom Bodies (MBs). MBs were found to influence many different aspect of the insect brain plasticity. The main function of MBs is olfactory learning: this was implemented in our architecture through a hebbian or STDP based learning scheme in spiking networks. The other function experimentally found in MBs is behaviour evaluation, mainly acting at the decision level. For this reason this MB functionality, here called MB2 (Fig. 2.3) is included into the decision layer and implemented as a separate block with respect to the olfactory learning block. Another addressed function is decision making: this function was discovered working with MB defective flies which were unable to make a sharp choice among two different contradictory visual features (color and shape) in front of the fading of the preferred one (color). This is a function that, involving visual learning, cannot be ascribed to the conventional functionality of MBs (olfactory representation and leaning). So this function was modeled as a separated block (MB3) and placed at the decision layer (Fig. 2.3) [17]. Moreover, from the block-size perspective, direct connections among FB and MB cannot be directly drawn for the lack of experimental evidence in the fly. So it is hypothesized that particular visual information, like color saturation reaches the MBs indirectly through other brain parts (like for example the Lateral Horn) and



gives it the possibility to concurrently act on the Behaviour Selection block at the level of decision making.

A series of internal states are monitored through a set of virtual proprioceptive sensors; these internal states undergo a continuous interaction with the ongoing external state of the agent, recorded through the exteroceptive sensors. Internal states are chosen according to the applications prepared, discussed within each experimental scenario. An internal state (like hunger or “need for charging”, need to sleep, etc.) is supposed to be directly related to drives which are typically reference control signals for the following Behaviour Selection Network (BSN) block (like desired battery level, zero home distance, etc.). In order to satisfy its drives, the robot has to choose a precise behaviour from a pre-defined set of available behaviours, each one oriented to satisfy one or several contemporary drives. Up to this stage, the BSN is implemented through a spiking network with dynamic synapses, leaving opened the possibility to learn other behaviours better satisfying the strongest drive. This functionality within the BSN takes place at the highest layer in the insect brain architecture. Till now, there are not yet specific experiments that can demonstrate the existence of such a network in the *Drosophila* brain; therefore the hypothesized artificial BSN was maintained to represent the highest level control functionality. The BSN was endowed, at this stage, with auto-excitatory synapses to avoid a continuous switching among the selected behaviours.

The other block residing at the decision layer is Behavior Evaluation. Experiments on the MB less flies show that this function is ascribed to MBs, even if apparently separated by the common MB functionality. So also this block was modeled separately with respect to the main MB block and so called MB2, as also mentioned above. This block evaluates the capability of the selected behaviour to satisfy the active drive, represented by a given setpoint to be reached. As soon as a given behaviour is initiated (behaviour initiation is ascribed as a specific CX role) the MB2 block starts an increasing inhibitory function on the ongoing behaviour in order to completely inhibit this one if the drive is not satisfied within a certain time window. In this case another behaviour wins the competition and is selected.

The Motor layer contains the following blocks: The Description of Reflexive Behaviours describes the fixed actions that allow the robot to take the right direction in the case of punishment. Here additional functions are included, considering the fact that a fly, repetitively punished, can reach a “no-motion state”: i.e. the insect is frozen for a certain amount of time.

The Description of Behavior block describes the available behaviours that the robot can follow. The type and number of the possible behaviours the robot can exhibit depends on the robot applications. As an example implemented is the targeting behaviour. This behaviour, when selected in the BSN, causes a series of actions focussed at moving the robot towards the visual target that elicited that behaviour, while maintaining it at the centre of the visual scene.

The Motor Programs block contains all the possible elementary actions the robot can perform. They are supposed, up to now, to be pre-programmed unless a wide space for hosting learning strategies exists, which is currently under investigation. This block is strictly dependent on the robotic architecture to be used. It contains a series

of control signals for the wheels/legs in order to realize the desired advancement, steering or rotation. In particular, dealing with legged robots, the central Pattern Generator paradigm was taken into account. This approach was recently accompanied to some powerful theoretical conditions which a-priori guarantee global exponential convergence to any imposed gait that the structure is asked to show [18].

As it can be derived from the analysis of the new scheme, almost all of the blocks enhanced are truly biologically driven, from experiments on flies or other insects. The last item outlined, i.e. the path internal states—drives—behavior selection—behavior evaluation (the really high level functionalities) up to now does not find any specific experimentally driven model. The behavior evaluation is addressed to the MBs, and indeed such areas are recognized to have a role in decision making, resolving contradictory cues, imposing adaptive termination or continuation of ongoing behaviors, but how this is linked to the choice of the behaviors and to the internal states and drives, is unknown. This information flow can really be included into the *Representation layer* and could be modeled using the Reaction-diffusion approach discussed at the highest layer in the former scheme of Fig. 2.1. In this part of the insect brain architecture could well find place all the spatial-temporal dynamics leading to the experience based control of ongoing behaviors.

## 2.3 Memory and Learning Mechanisms in Nature

Artificial agents (i.e. simulated and real robots) need learning algorithms to construct a knowledge and to improve their basic capabilities. In Nature, learning mechanisms are part of living beings and act at different form and level of complexity. According to a general definition, “*learning* is an adaptive change in behavior caused by experience” [19].

It is important to notice that learning mechanisms, in whichever form, need a memory structure to correctly work. Memory is the storage and recall of previous experiences. *Memory* is necessary for learning; it is the mechanism whereby an experience is incorporated into the organism, so that it can later be used to bring about adaptive changes in behavior [19].

There are a number of different types of learning and memory. Table 2.1 lists the main categories. In other words, learning is one of the most important mental function present in humans, animals and artificial cognitive systems. It relies on the acquisition and processing of different types of knowledge supported by perceived information. It leads to the development of new capacities, skills, values, understanding.

### 2.3.1 Memory and Learning in MBs and CX

In relation with the Insect Brain Model, memory elements and learning mechanisms are distributed on the whole architecture. The instruments used to unravel

**Table 2.1** Main categories of learning and memory [19]

Types of learning	Types of memory
<i>Simple</i>	Immediate
Habituation	Short-term
Sensitization	Long-term
<i>Associative</i>	Specific
Passive (classical)	
Operant (instrumental)	
One-trial (aversion)	
<i>Complex</i>	
Imprinting	
Latent	
Vicarious	

**Table 2.2** Memory and learning in insect brain

Experiments	Functionality	STM	LTM	Memory/Learning
Exploration	Increase of mean free path	?	?	Working memory
Olfactory-based navigation	Olfactory	MBs	MBs	Classical conditioning
Visual-based navigation	Visual learning	FB (CX)	?	Operant conditioning
Detour paradigm	Path integration	EB (CX)	–	Working memory
Heatbox learning	Orientation memory	?	–	Operant conditioning
Gap climbing	Motor learning	MBs	CX	Operant learning

the information flow that characterizes this complex structure are focused biological experiments that can be used to distinguish involved blocks and functionalities. A summary of the results obtained is reported in Table 2.2 where a series of experiments are used to identify which parts of the insect brain are involved in performing specific behaviors. In particular the table includes the locations of the short term (STM) and long term (LTM) components of the memory system involved and the learning mechanism used.

The identification of the role of each center in the cognitive process that characterizes the fly is not always easy to obtain. An example could be the analysis of the exploration phase. Through experiments a centrophobic behavior has been identified: the fly adopts an exploration strategy that probably includes the increasing of its mean free path. The resulting behavior in a closed arena consists in reaching and following the external walls. This kind of behavior needs a working memory but up to now the location of this element is unknown.

As far as the olfactory learning is concerned, the fly is able to navigate into an environment following a smell. Both the short term and long term memory can be located in the MBs and a simple but efficient classical conditioning is performed to associate a meaning to specific smells that can be either rewarded or punished.

Similarly to the olfactory learning where the MBs are the structure responsible in the fly, visual processing and the corresponding memory and learning structures is

completely associated to the Central Complex. The fly is able to learn both through classical and operant conditioning to classify objects, extracting from the segmented image a series of characteristic features. The *fan-shaped body* (FB) is the structure of the CX devoted to the STM: the fly associates a meaning to objects, depending on its past experience.

Another important neural structure belonging to the CX is the *ellipsoid body* (EB), that is responsible for orientation memory.

In the detour experiment (addressing the capability to aim to a formerly seen target, even after being luring away) the parameters characteristic for path integration have been identified. The capability to retain, recall and integrate positional information about the target into guiding behavior has been summarized under the term spatial working memory and is ascribed to the EB.

Finally as concern the motor learning, relevant information were obtained using the gap climbing experiment where the fly is forced to climb several consecutive gaps. The strategy used to overcome the obstacle can be improved through operant learning both in the short term (the STM was located in the MBs) and in the long term after a sleeping phase (the LTM was located in the CX). The modeling of the mechanisms used for motor learning is complex task that up to now represents an open issue and will be investigated.

As previously discussed, important functions can be referred to the MBs and CX: all the experiments carried on up to now reveal that these two areas are not directly connected, although many indirect connections are present. Therefore olfactory and visual stimuli could be treated independently but, finally, they converge to the pre-motor area. The final decision will be taken but we do not know where. Some experiments envisaged that the CX decides what to do but the MBs modulate the intensity of the response.

## 2.4 Description of the Computational Architecture

The main parts of the insect brain have been modeled and integrated into a computational architecture. This global computational model inspired by the *Drosophila* brain is presented in Fig. 2.3. It has been designed in order to be directly linked to a robotic implementation [20, 21]. The main parts of the whole architecture are described in the following.

### 2.4.1 Sensorial Pathways and Internal States

In the model, it is possible to distinguish four main sensorial pathways; the olfactory and the visual pathways allow to percept the environment while the gustation and the nociception are indispensable to obtain information about the goodness or badness of the current situation. The interaction among the sensorial pathways allow the

emergence of anticipative actions to improve efficiency in finding rewards and to avoid dangerous situations. Learning mechanisms based on classical and operant conditioning are used in the architecture [22]. Olfactory and visual inputs, due to their complexity, are pre-processed to be easily handled by the spiking networks used for the learning processes. The internal states of the system are also important and can be monitored through a set of virtual proprioceptive sensors. These states are chosen according to the application; for example, autonomous navigation-based tasks need an accurate monitoring of the batteries level of the robot.

### 2.4.2 Drives and Behavior Selection Network

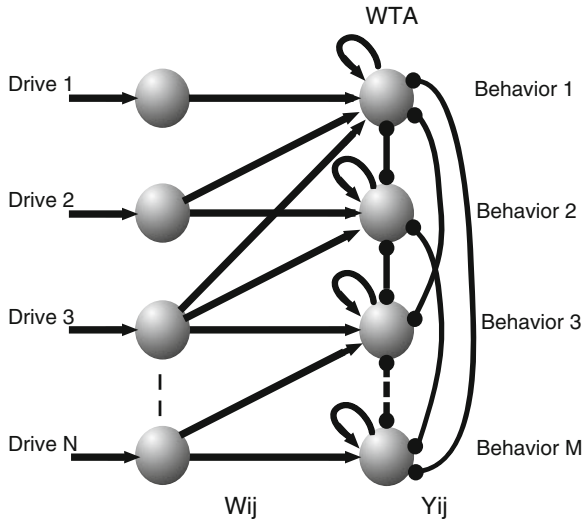
External stimuli and internal states are essential information for living beings to survive in unstructured environments. An internal state is supposed to be directly related to drives like hunger, thirst, the will to sleep that are used by animals to adapt the behavioral responses. In a robotic implementation, drives will be chosen following the robot applications: a classical example is the need for power supply. In order to satisfy its needs, the robot has to choose a behavior from a pre-defined number of available behaviors. Behavior is meant like a sequence of programmed actions. Each behavior is oriented to satisfy one or more drives. The aim is to make the robot able to choose the right behavior that can satisfy the strongest drives. Even if there are not specific experiments that can demonstrate the existence of such a network in the *Drosophila* brain, an artificial Behavior Selection Network (BSN) was envisaged and implemented. The BSN was thought as a two-layers neural network, in which each unit is an Izhikevich Class I spiking neuron [23], having the following equation:

$$\begin{cases} \dot{v} = 0.04v^2 + 5v + 140 - I \\ \dot{u} = 0.02(-0.1v - u) \end{cases} \quad (2.1)$$

if  $v \leq 0.03$ , then  $v \leftarrow -0.055$  and  $u \leftarrow u + 6$

where  $v$  is the membrane potential of the neuron,  $u$  is a recovery variable and  $I$  is the synaptic current. A typical example of a BSN structure is shown in Fig. 2.4.

The number of neurons in the first layer matches the number of drives the robot has to satisfy whereas, in the second layer, each neuron corresponds to the available behaviors. Every drive introduces a current, that is then converted in a spike-rate by the corresponding first layer neuron. The weight of the synapses connecting the first and the second layer neurons are chosen according to the capacity of each behavior to satisfy each drive. Synaptic weights  $W_{ij}$  represent the importance of drive  $i$  for the behavior  $j$ . Synaptic efficiencies are fixed: no learning is considered at this step. The second layer is a Winner-Takes-All (WTA) network; during every simulation step the neurons in the second layer are competing and only one neuron can win the competition: the behavior represented by the winning neuron is the selected behavior for the next robot step. To avoid a continuous switching among



**Fig. 2.4** Example of a spiking network used to simulate the behavior selection functionalities. Drives are represented by input currents. Each drive can excite more than one behavior. Synaptic efficiencies between the input layer and the WTA layer represent the influence that each drive has in each behavior. Only the most excited behavior can win the competition and can be selected

the selected behaviors, a self-excitatory synapse has been introduced in each neuron of the second layer of the BSN. In this way, if a behavior is been selected during a simulation step, the probability to be selected again is increased during the next step. Synaptic weights  $Y_{ij}$ ,  $i \neq j$ , represent the inhibitory synapses between neuron  $i$  and  $j$  in the WTA layer. Synaptic weights  $Y_{ii}$  represent the self-excitatory synapses of neuron  $i$  in the WTA layer. The last point to clarify is how to transform drives in an input current. Considering for instance the drive “recharge”, the robot analogue to “sleep”, strongly connected to an internal sensor that measures voltage in batteries. It is possible to implement a transfer function that takes as input the battery level and gives as output a numerical evaluation of the “sleep” drive. Other methods for behavior selection have been used in literature, in particular for sequence learning [16]. The proposed approach could be modified in order to implement sequence learning, even if, up to now, there are no biological evidences about the capabilities of *Drosophila* in learning sequences of behaviors.

### 2.4.3 Central Complex Model

#### 2.4.3.1 Protocerebral Bridge Model

Object detection and distance estimation are functions related to the PB in fruit flies. Mronz and Strauss proposed a simple model based on parallax motion [24] that can



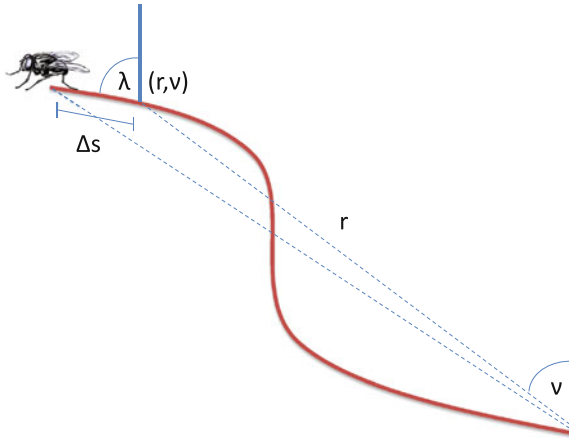
be used to model these functions and a hardware implementation for an autonomous roving robot has been proposed in [25]. However, it is possible to use a generalized PB model, realized by a cascade of three simple blocks:

- *Object Detection Block.* This block takes input from the visual system and is used to detect the presence of an object.
- *Distance Estimation Block.* When an object has been detected, this block estimates its distance from the robot. In real flies, distance is estimated using a parallax motion approach.
- *Object Position Block.* It is possible to reproduce fly behavior assuming as interesting objects those standing in the compartments ranging from the frontal direction to  $\pm 100^\circ$  in the two lateral sides and repulsive ones those standing in the compartments from  $\pm 100^\circ$  to  $\pm 160^\circ$  on the rear part of the robot. Objects at angles exceeding  $160^\circ$  cannot be seen.

#### 2.4.3.2 Fan-Shaped Body Model

The fan-shaped body model has been designed as a cascade of two sub-blocks: a feature extraction and a feature evaluation element.

- *Feature Extraction.* Once an object has been detected, the Feature Extraction Block classifies it by using a series of features. As underlined in focused experiments with flies [26], the following features can be considered:
  - *Color.* Using a HSV representation, it is assumed to consider only the Hue value.
  - *Orientation.* Orientation is meant as the angle between the vertical direction and an axis that represents the direction in which an object is mainly distributed.
  - *Size.* Size is meant as the portion of total visual area of the robot occupied by the object, normalized with respect to the distance from the robot.
  - *Center of Gravity.* This feature is given by the height of the center of gravity normalized with respect to the vertical dimension of the visual area and the distance from the robot.
  - *Wideness.* Wideness is meant as the maximal horizontal extension of the object, normalized with respect to the total horizontal dimension of the visual area and the distance from the robot.
  - *Height.* Height is meant as the maximal vertical extension of the object, normalized with respect to the total horizontal dimension of the visual area and the distance from the robot.
- *Feature Evaluation.* The robot is able to associate object features to punishment or neutral situations. Every feature has a *Punishment Value*: if this value exceeds a threshold, the robot escapes every time it meets an object with that feature. The Punishment Value of a feature decreases if the robot is not punished when that feature is encountered. When the robot meets an object, it evaluates its Escaping Value: this is a weighted sum of the Punishment Values of the features of that object. When the Escaping Value is high enough, the robot escapes from the object, even



**Fig. 2.5** Path Integration scheme. The values of  $r$  and  $\nu$  represent the position of the robot from the object. Every robot step  $r$  and  $\nu$  are updated according to the last robot movements, in direction  $\lambda$  by a path increment  $\Delta s$

if not punished. This is the simplest way to implement a classifier. Other, more performing and sophisticated algorithms, either bio-inspired, or more information theory-bases, like the Neural Gas [27], could also be taken into account to improve the system plasticity. A neural model has been finally used to implement visual learning capabilities in flies [28].

### 2.4.3.3 Ellipsoid Body Model

Neuser [29] described the role of the *Drosophila* ellipsoid body in the visual short term memory. That functional analysis leads to the implementation of a model able to create a spatial memory in a robot. By using polar coordinates to code the robot position in the environment, it is possible to design neural architectures inspired by the ant's path integration [30]. However other solutions could be based on a mathematical implementation of a polar path-integration algorithm and this kind of approach (easier and more robust) has been taken into consideration. A scheme describing the path integration mechanism is shown in Fig. 2.5. Supposing  $\Delta s \ll r$ , defining  $\lambda$  as the direction of the current robot movement and  $\delta = \lambda - \nu$ , the approximation of the current robot position is recursively given by:

$$\begin{cases} r_{n+1} = r_n + \Delta r_n = r_n + \Delta s_n \cos(\delta_n) \\ \nu_{n+1} = \nu_n + \Delta \nu_n = \nu_n + \Delta s_n \sin(\delta_n) / r_n \end{cases} \quad (2.2)$$

where  $\Delta s_n$  is the length of the robot step of index  $n$  and  $r$  and  $\nu$  are the coordinates that represent the position of the robot with respect to the object that is supposed to be the origin of the polar coordinate system.

### 2.4.4 Mushroom Bodies Model

The MBs are a key structure of the insect brain. In particular, two main functions are related to this structure. Mushroom Bodies are primarily involved in olfactory learning [31, 32] and in a more complex function that will be called *behavior evaluation*. Because experiments are not able to demonstrate the connection between the two functions, two uncoupled models were implemented.

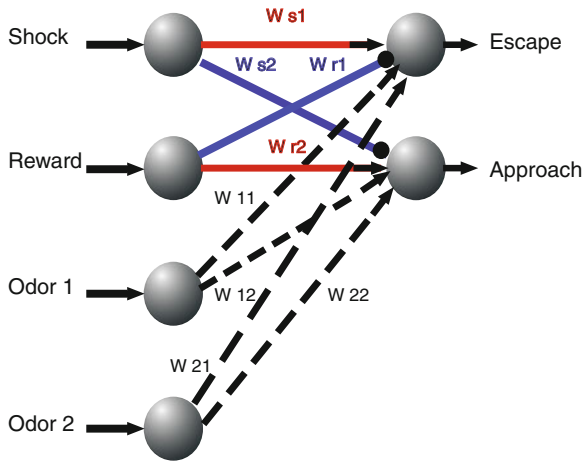
#### 2.4.4.1 Olfactory Learning Model

A two layer spiking neural network was designed and implemented to model the olfactory learning function. The Spike Timing Dependent Plasticity (STDP) has been applied as learning algorithm [33, 34]. This algorithm can reproduce Hebbian learning in biological neural networks. The algorithm works on the synaptic weights, modifying them according to the temporal sequence of spikes occurring. The algorithm is represented by the following formula:

$$\begin{cases} \Delta W = A_+ \exp(\Delta t / \tau_+), & \text{if } \Delta t < 0 \\ \Delta W = -A_- \exp(\Delta t / \tau_-), & \text{if } \Delta t \geq 0 \end{cases} \quad (2.3)$$

where  $\Delta t$  is the time delay between pre and post synaptic spikes. In this way the synapse is reinforced if the pre-synaptic spike happens before the post-synaptic one, it is weakened in the opposite situation. Parameters  $\tau_+$  and  $\tau_-$  represent the slope of exponential functions, while positive constants  $A_+$  and  $A_-$  represent the maximal variations of the synaptic weight.

Each neuron is modeled by an Izhikevich Class I neural model [23]. A scheme of the neural model is shown in Fig. 2.6. The Shock (Punishment) Neuron takes as an input a current proportional to the value of the robot punishment, while the Good (Reward) Neuron takes as input a current proportional to the reward. In experiments with *Drosophila*, the punishment could be represented by an electrical shock, while the reward is represented by sugar. Each of the remaining neurons of the first layer takes as an input a current proportional to the odors the robot can detect in the environment. Each odor has a corresponding receptor and a neuron that converts the current in a spiking-rate if the current is high enough above the threshold for the Class I Izhikevich model. In a real robotic implementation odors can be substituted with other sensorial inputs, according to the application. Specific neural network composed by Izhikevich neurons and STDP learning were already implemented to realize approaching or escaping behaviors [12, 35]. In the present implementation Synapses between the Shock and the Reward Neuron and the output layer have a fixed value. Outputs of the second layer neurons are connected to a Motor Program block. The robot escapes from the actual position if the Escape Neuron is firing, while it begins an approaching algorithm if the Approach Neuron is firing. Synapses between unconditioned stimuli (i.e. shock and reward) and motor neurons are fixed



**Fig. 2.6** Olfactory Learning Model. Solid line (dashed) connections correspond to fixed (plastic) synapses; arrows (bullet) correspond to excitatory (inhibitory) connections. The model here presented can be easily extended to the desired number of odors

and represent the inherited knowledge whereas connections between conditioned stimuli (i.e. odors) and the motor system are subject to learning, according to the STDP rule. If not reinforced, the efficiency of a synapse decays with time.

#### 2.4.4.2 Behavior Evaluation Model

When a behavior is selected, the robot defines a setpoint to be reached. A setpoint is meant as a desired value for a vector of physical quantity linked to the definition of the drives: to satisfy its needs, the robot has to minimize the error between this setpoint and its actual state. For example, let us assume that the robot has a low battery voltage and that a charging-station is present in the environment. In this situation the robot could choose to go to the base station whereas the desired battery level would be the setpoint. If the selected behavior is not able to allow the robot to satisfy its needs, that behavior has to be inhibited: in the opposite situation, if the selected behavior leads to satisfy its needs, that behavior has to be excited. The Behavior Evaluation block inhibits or excites the actual behavior depending on the amount of time already spent and the level of success in reaching the setpoint. Inhibition or stimulation is easily implemented sending a current to the neuron of the WTA layer in the BSN associated to the ongoing behavior. Moreover some additional plasticity could be implemented into the Behavior Selection Network through the Behavior Evaluation model. In particular, the synapses between the neuron related to the selected behavior and drives that represent the setpoint could be reinforced (or weakened) if the last selected behavior has been able (or not) to reach the last setpoint. However, there are no biological evidences about this point.

### ***2.4.5 Motor Programs and Description of Behaviors***

The Motor Program block describes all the possible elementary actions that the robot can perform. Motor learning is not considered up to now, although it is envisaged to be investigated and added in the near future.

### ***2.4.6 Reflexive Behaviors***

When the robot is punished in some way it needs to escape as fast as possible from the object responsible for the shock. The *Description of Reflexive Behaviors* is a simple high level block that allows the robot to take the right direction in the case it is punished.

### ***2.4.7 Complex Behaviors***

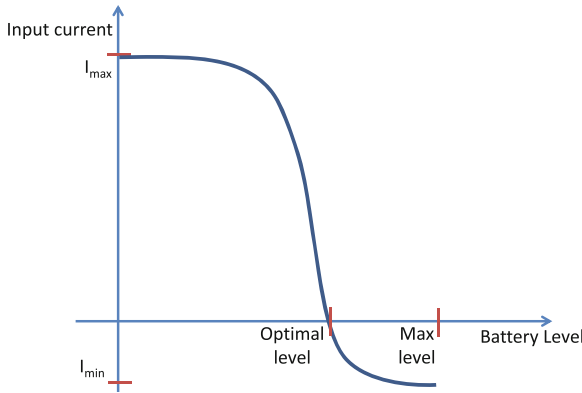
The *Description of Behavior* block is a high level part of the complete model that describes the available behaviors that the robot can follow. The choice of the possible behaviors the robot can exhibit depends on the robot applications. Applying a searching strategy to find a charging station could be an example of a typical behavior. The description of each behavior, however, could depend on the robotic structure and the embedded sensorial system.

## **2.5 Simulation Description**

The implementation of drives and behaviors on a real robot strongly depends on the field of application in which that robot is involved. The simulation of the general model of an insect brain requires a simulated environment, where behaviors and drives, which the robot has to satisfy, need to be defined. The focus is to simulate the model in a context that can present analogies with the *Drosophila* real experimental set-up, in order to obtain the experimental validation of the model. Details about the implementation of the complete model are reported in the following.

### ***2.5.1 Drives***

A brief description about the drives chosen in simulations and their analogies with real fruit flies is presented here.



**Fig. 2.7** Example of the transfer function used to model the sleep drive. A low battery level leads to a high value of the current related to the *Sleep* drive. The optimal battery level is represented by the point in which the drive *Sleep* is equal to zero. If the battery level exceeds the optimal point, the drive became negative, in order to inhibit a possible dangerous battery charging

- *Sleep*—The drive *Sleep* is assumed to be the need for a robot to charge its batteries. In real fruit flies sleep is indispensable for learning [36]. In common robots we can quantify the drive sleep using a function of the battery level:

$$I_{drive} = K_{drive} \tanh(\Lambda - \chi) + \psi \quad (2.4)$$

where  $K_{drive}$ ,  $\Lambda$ ,  $\chi$  and  $\psi$  are the parameters of the function. Through these parameters it is possible to set the maximum and the minimum value of the current and the optimal battery level. An example is shown in Fig. 2.7. The optimal battery level is represented by the point in which the drive *Sleep* is equal to zero. If the battery level exceeds the optimal point, the drive became negative, in order to inhibit a battery charging. In order to simulate the battery level, it is convenient to implement a virtual sensor. The output of such sensor is the estimated battery level. The battery level must decrease each step in which the robot is far from the charging station, and reaches the Max Battery Level after a given time spent in the charging station area. A sleep drive is indispensable for every time it is necessary to have a completely autonomous robot, which must be able to find power supply sources and use them to move for a long time.

- *Hunger*—The need of food can be reproduced putting inside the environment objects or landmarks that the robot should periodically find and/or visit. The drive *Hunger* could be thought as proportional to the time the robot left the object. This drive is indispensable to obtain a behavior that can match with reality but also to force the robot to find objects that can be periodically useful. The drives *Hunger* and *Sleep* have some similarities; their differences will be remarked according to the application.



- *Shelter*—When in danger, a fly looks for a safe place. A fly in open spaces often has the tendency to protect itself, typically aiming to approach and follow walls. Shelter can be related to the distances of the robot from the walls.
- *Curiosity*—The drive Curiosity allows a fly to search for other resources when the other drives are satisfied. Curiosity can be quantified with a constant value. From a robotic point of view, curiosity leads the robot to explore the environment and to acquire information about the detected objects.

### 2.5.2 Behaviors

To make the robot able to satisfy its drives, the following behaviors have been implemented. They constitute the output of the Behavior Selection Network:

- *Exploration*. During an exploration behavior the robot tries to find new resources. In flies, the environment exploration is characterized by an increase of the mean-free path algorithm [37]. As in real flies, during an exploration behavior it is possible to distinguish two behaviors [38]:
  - Sitter larvae behavior: short path length and tight turning angles.
  - Rover larvae behavior: long path length and wide turning angles.

Exploration can be thought as a default behavior: the robot could choose this behavior when no particular drives are enabled. Usually curiosity is the drive that mainly influences the choice of an exploration behavior. The implementation of the exploration behavior requires also the management of the obstacle avoidance and object detection. Moreover, the robot must be able to update its position at each step; in our case, the ellipsoid body model is involved.

- *Homing*. The homing behavior consists in returning to the charging station, where the simulation is started. Of course, the position of the Home must be known and updated every step. An obstacle avoidance algorithm has to be implemented during the homing behavior. Homing behavior is needed to have an autonomous robot, able to charge its battery before its autonomy is compromised.
- *Landmark Recalling and Achievement*. During the navigation the robot meets objects: if some objects are associated to food, the robot must remember their position in order to reach them when it is needed. The biological plausibility of this behavior is evident. The robot must use the path integration system to update its position from each interesting object.
- *Centrophobism*. A centrophobic behavior has been found in flies [39]. From a biological point of view, centrophobism in flies could be a consequence of the increase of mean-free path in the exploration behavior. A fly uses centrophobic behavior to protect itself in dangerous environments. Shelter is the drive that mainly influences the choice of a centrophobic behavior.

### 2.5.3 The Robot and the Simulator

The robot used in the first experiments is a Pioneer P3-AT differential-drive roving robot. The platform operates as a server in a client-server environment; the onboard PC is used to host the control architecture.

MobileSim is the software used for simulating the Pioneer P3-AT roving robot in a virtual 2D environment. This simulation environment has been used to evaluate the performance of the proposed control system.

### 2.5.4 Implementation of Odors, Punishments and Rewards in the Simulator

In order to implement olfactory classical conditioning it is necessary for the robot to have sensors that can detect odors and that can monitor rewards or punishments given to the robot. In a simulation environment it is convenient to implement virtual sensors. For instance, if an object releases an odor called Odor1, it is convenient to assume the output of the olfactory sensor as a Gaussian function of the distance  $d$  from the robot to that object:

$$f_{od}(d) = K_{od}e^{-d/\tau_{od}} \quad (2.5)$$

where  $K_{od}$  is a constant gain and  $\tau_{od}$  represents the decay of the sensor output when the robot goes away from the object.

It is possible to use a similar strategy to determine the output of a punishment sensor and the output of a reward sensor:

$$f_{pun}(d) = K_{pun}e^{-d/\tau_{pun}} \quad (2.6)$$

$$f_{rew}(d) = K_{rew}e^{-d/\tau_{rew}} \quad (2.7)$$

The values of the constants can be determined in order to obtain a tighter Gaussian function for the output of the punishment and reward sensors: in this way, if the robot is approaching the object, it will first detect the odor and then it will be rewarded or punished if that object is not neutral.

## 2.6 Simulation Results

This section presents the experiments made in order to perform a first evaluation for each model of the general computational architecture of the *Drosophila melanogaster* brain.

**Table 2.3** Summary of the characteristic of the objects in the MBs model test

Object	Odor	P/N/R
Object A	Odor 1	Punishment
Object B	Odor 2	Reward
Object C	Odor 2	Punishment
Object D	Odor 2	Neutral

### 2.6.1 Mushroom Bodies and Olfactory Learning

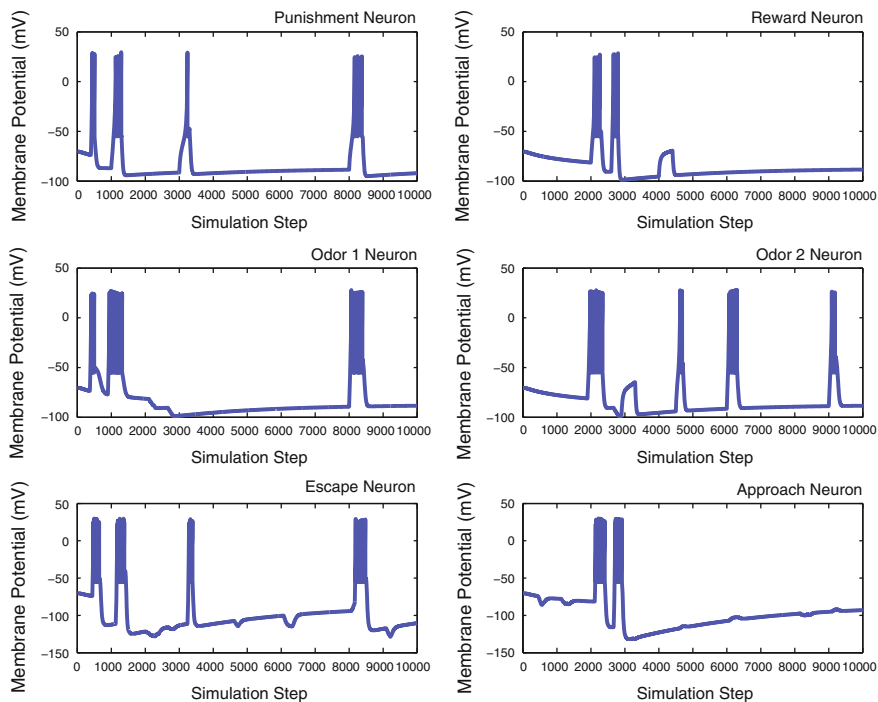
The following simulation shows how the MBs model for odor learning works. The simulation of the model has been done using the Euler integration method with a constant integration time of 20 ms. The synapses time constant is 800 milliseconds and synaptic weights are initialized to the value of 0.05. A priori known information is codified in the fixed synaptic weights that have been initialized to the value of 10 (excitatory) and -3 (inhibitory). A decay rate has been introduced: every 100 simulation steps all synaptic weights are decreased by 1 % of their value. The implemented network is the same as shown in Fig. 2.6. This simulation was performed to verify the capability of the MBs model to make the right associations between odors and rewards or punishments in a complex environment. The robot is introduced into a square arena,  $10 \times 10$  m, in which four objects are present. There is an odor spreading out from each object in the environment. In particular, Eq. 2.5 has been used. Two different odors are associated to these objects and a reward or a punishment is given to the robot when one of the objects is reached, following the association reported in Table 2.3.

Exploring the environment, the robot has to learn that there is a strong association is between the Odor 1 and the punishment: in a testing phase, the robot will be able to escape when detecting that odor, before the shock occurs. The behavior of the network neurons during the simulation is shown in Fig. 2.8 while Fig. 2.9 presents the trend of the synaptic weights during the simulation. The network evolves for 100 simulation steps for each robot action.

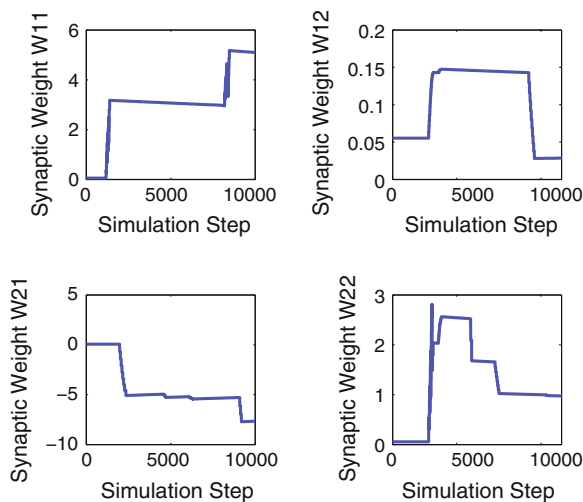
At the end of the simulation the robot explored the environment completely and it is able to make the right association. Other experiments were performed, obtaining similar results.

### 2.6.2 Protocerebral Bridge and Fan-Shaped Body

Through a functional analysis of *Drosophila* protocerebral bridge and fan-shaped body, it is possible to suppose that object detection and distance estimation are mainly performed by the PB, while the FB is related to feature extraction and classification. In the following experiments, the properties of the PB and FB models and the capabilities of the robot in terms of visual learning are presented.

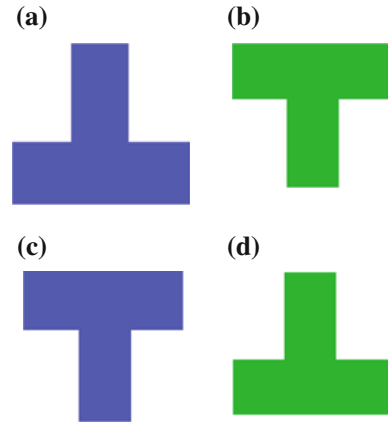


**Fig. 2.8** Simulation results of the olfactory learning model: behavior of the neuron membrane potential during simulation. The implemented network is shown in Fig. 2.6

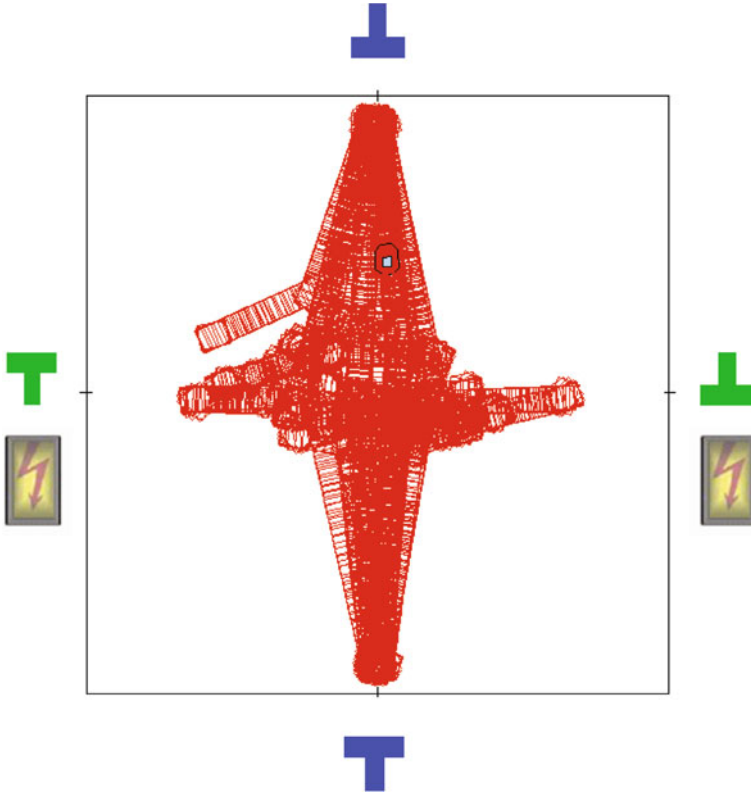


**Fig. 2.9** Simulation results of the olfactory learning model: trend of the synaptic weights during the simulation. The implemented network and the parameter meaning are illustrated in Fig. 2.6

**Fig. 2.10** Objects used in the fan-shaped body model simulation. **a** *blue* inverted T-shape; **b** *green* upright T-shape; **c** *blue* upright T-shape; **d** *green* inverted T-shape



In particular, the proposed simulation is inspired by the experiment designed by Liu and collaborators on real flies [40] about visual learning and object recognition. The robot has to explore a square arena, ( $10 \times 10$  m), in which four objects are present. Even if these objects are different, they can have some similar features. The objects used in this simulation are shown in Fig. 2.10. Every time the robot meets an object, it tries to recognize that object, extracting features and comparing them with the stored ones. If the robot meets an object for the first time, it extracts and stores the new features. It has been assumed to consider six features: color (in the Hue Saturation Brightness representation, here only the Hue value is considered), orientation, size, center of gravity position, wideness and height. The PB model has been set so that the robot is able to detect objects in a range of 4 m. Objects associated to a punishment shock the robot if its distance from these objects is less than 2.7 m. In this experiment the color “green” is a bad feature: the robot will be punished every time it tries to approach a green object. The robot has to learn to avoid green objects. The arena and the simulation results are shown in Fig. 2.11. At the beginning of the simulation, the robot tries to approach every object standing in its visual range. If punished, the robot increases the punishment value of the features of the approached object. If an object is neutral, the punishment value of the features associated to that object decreases. If the escaping value of an object reaches a threshold, the robot will escape when that object is detected. In this simulation the robot learns correctly to avoid green objects. Figure 2.12 shows also the Punishment Value of the bad feature (green color) and the Punishment Value of a neutral feature, the wideness, that is the same for all the objects. In order to implement a hysteretic response, when the Punishment Value exceeds 2, it is simply raised to 7. In this way the robot will remember this bad feature association for a long time, even if the learning is not reinforced. If the robot detects an object, the Punishment Value of all the features that do not belong to that object will remain the same. A time-dependent decay rate could also be introduced.



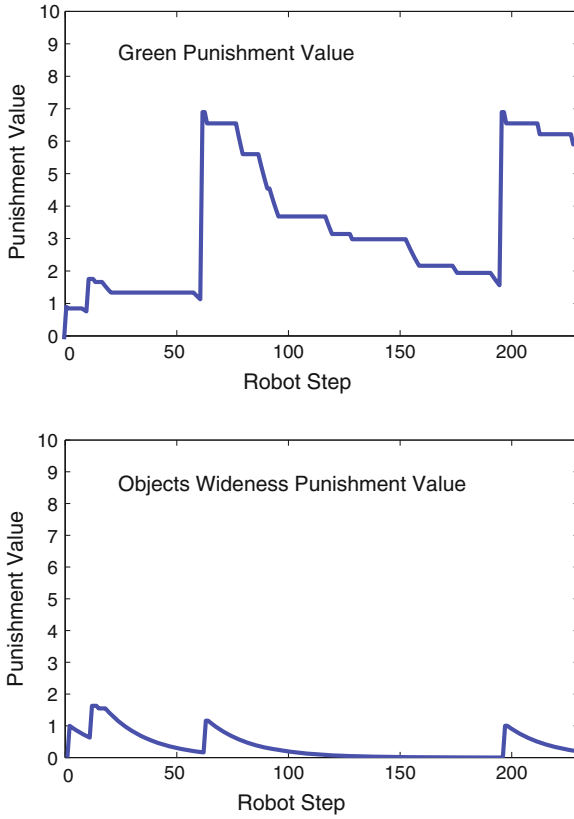
**Fig. 2.11** Robot trajectories obtained during the testing of the fan-shaped body model. After being punished enough times, the robot is able to isolate the dangerous feature (the *green color*) and escapes when a *green* object is detected

In a second experiment, the robot has to learn to avoid each “T” object. Color is now neutral for the robot. Even if the shape is not a feature, a T is different from an inverted T because of the different center of gravity. This experiment leads to the same conclusion of the first experiment; the robot is able to recognize bad features and to avoid them.

### 2.6.3 Ellipsoid Body

In real fruit flies, the ellipsoid body is necessary for a visual short-term memory and orientation. In the following simulation, the behavior of the EB model while the robot is moving around the environment is evaluated. In the simulated environment an odometry error has been introduced, to make the results more realistic. In this first

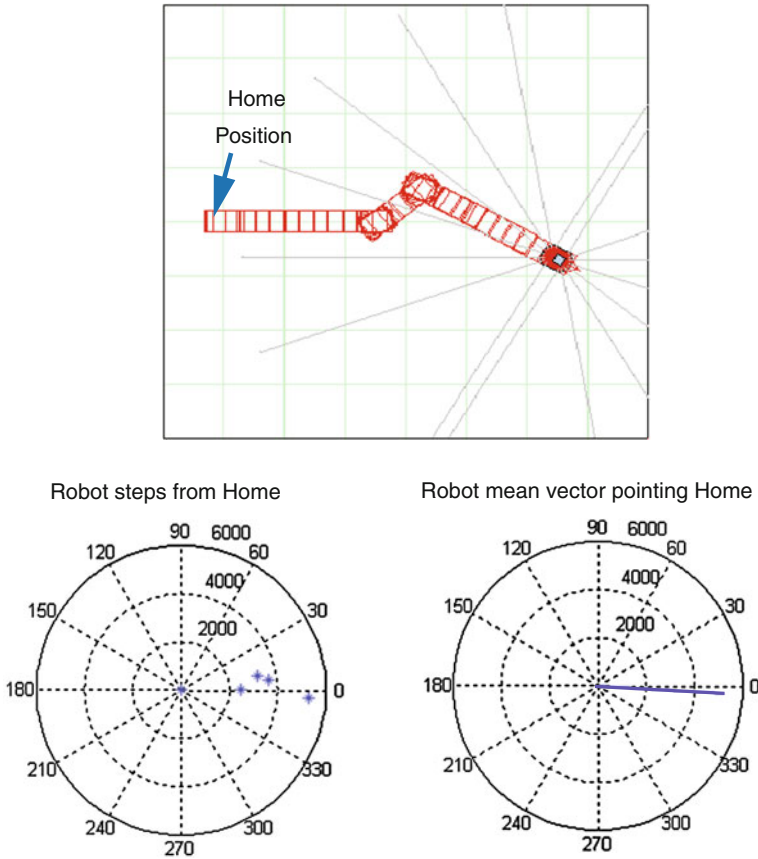




**Fig. 2.12** Comparison between the punishment value of the dangerous feature and the punishment value of a neutral feature, the wideness. Using the punishment value algorithm the robot is able to discriminate the dangerous feature. The decreasing of the punishing value of the dangerous feature is due to the steps in which the robot can detect a green object but is not so near to be punished

experiment we want to show how the ellipsoid body model works: the robot must be able to update its position while moving into a square arena ( $8 \times 8$  m). The robot starts from the Home position and moves randomly into the arena: its capability to update its relative position with the Home is analyzed. Of course the coordinates stored into the robot memory will be different from the real ones, because of the odometry errors and the approximation of the path integration method. Figure 2.13 show an example of trajectory and the response of the ellipsoid body. The same test has been repeated many times, in order to make a better analysis of the model.

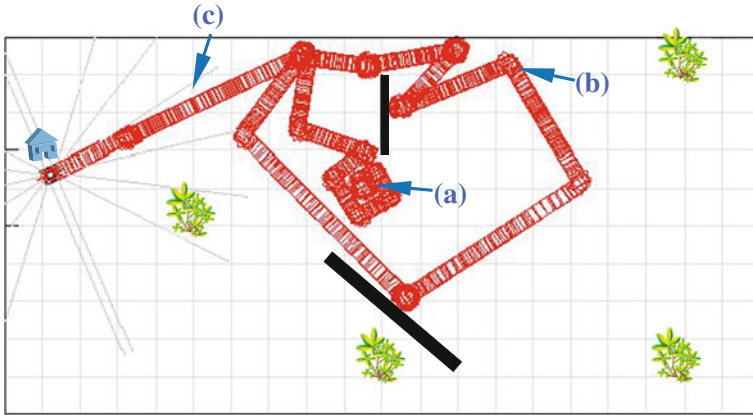
In order to test the capability of the model in real situations, it is convenient to simulate the robot behavior and the EB response in more complex arenas. In the following experiment the robot has to explore a large arena, in which several objects are present. The robot starts from the Home and initially it moves randomly: this



**Fig. 2.13** Information stored in the EB. The relative position of the robot with respect to the Home is represented in polar coordinates and it is indicated in millimeters (distance) and degrees (angular position). In this case the current robot position is  $r = 5730$  mm,  $\nu = -12^\circ$

behavior is created to simulate a typical escaping reaction of real flies when newly introduced into an arena.

After that, the robot starts an Exploration behavior. If the robot meets objects it is able to learn about their danger or neutrality, thanks to the MBs model. During the exploration, the robot updates its position from the Home. An obstacle avoidance mechanism was also implemented. During this experiment two behaviors are available: Exploration and Homing. The level of the battery decreases while the robot explores the arena. A virtual battery sensor has been implemented. If the level of the battery is too low, the BSN switches the selected behavior to the Homing Behavior. If the stored position is correct, the robot must be able to return to the Home position. Obstacle avoidance is used also during the Homing behavior. Simulation results are shown in Fig. 2.14. The robot starts to move and, after the escaping reaction imple-



**Fig. 2.14** The robot starts to move and, after the escaping reaction implemented to match the biological experiments with real flies, it begins an exploration phase (a). The escaping reaction from the Home position to position (a) is not outlined for clarity reasons. After fifteen exploration steps, the battery level is low and the robot starts its homing behavior (b). Using an obstacle avoidance algorithm, the robot is able to return to the Home (c). In order to have a more complex simulation, some objects have been also introduced into the arena

mented to match the biological experiments with real flies, it begins an exploration (a). The escaping reaction from the Home position to position (a) is not outlined for clarity reasons. After fifteen exploration steps, the battery level is low and the robot starts its homing behavior (b). Using an obstacle avoidance algorithm, the robot is able to return to the Home (c).

#### 2.6.4 Behavior Selection

In order to allow the robot to choose the more suitable behavior, the Behavior Selection Network (BSN) has been implemented. The BSN has been tested and its properties have been analyzed. In a real implementation of the model the drives are the inputs of the first layer of the network. In the following simulations drives have been simulated in order to study the response of the BSN in different possible situations. This experiment shows how the Behavior Selection Network works. It has been assumed to have four behaviors and four drives, and to represent these drives with four input currents.

In this first example a perfect symmetry in the complete network has been supposed:  $W_{ij} = 1.5$ ;  $W_{ii} = 10$ ;  $Y_{ij} = -3$ ;  $Y_{ii} = 3$  (see Fig. 2.4 for the network topology). A random Gaussian noise has been added in the input currents ( $\sigma = 2$ ). Fig. 2.15 presents the behavior of the neurons of the network. When a second layer (WTA layer) neuron is firing faster than the others, the respective behavior is selected.

The network has been simulated for ten thousand simulation steps, with an integration step of 20 milliseconds. During a short transient, all the WTA neurons are firing: this situation is due to the response of the synapses between WTA neurons. After this transitory period, only one neuron can win the competition.

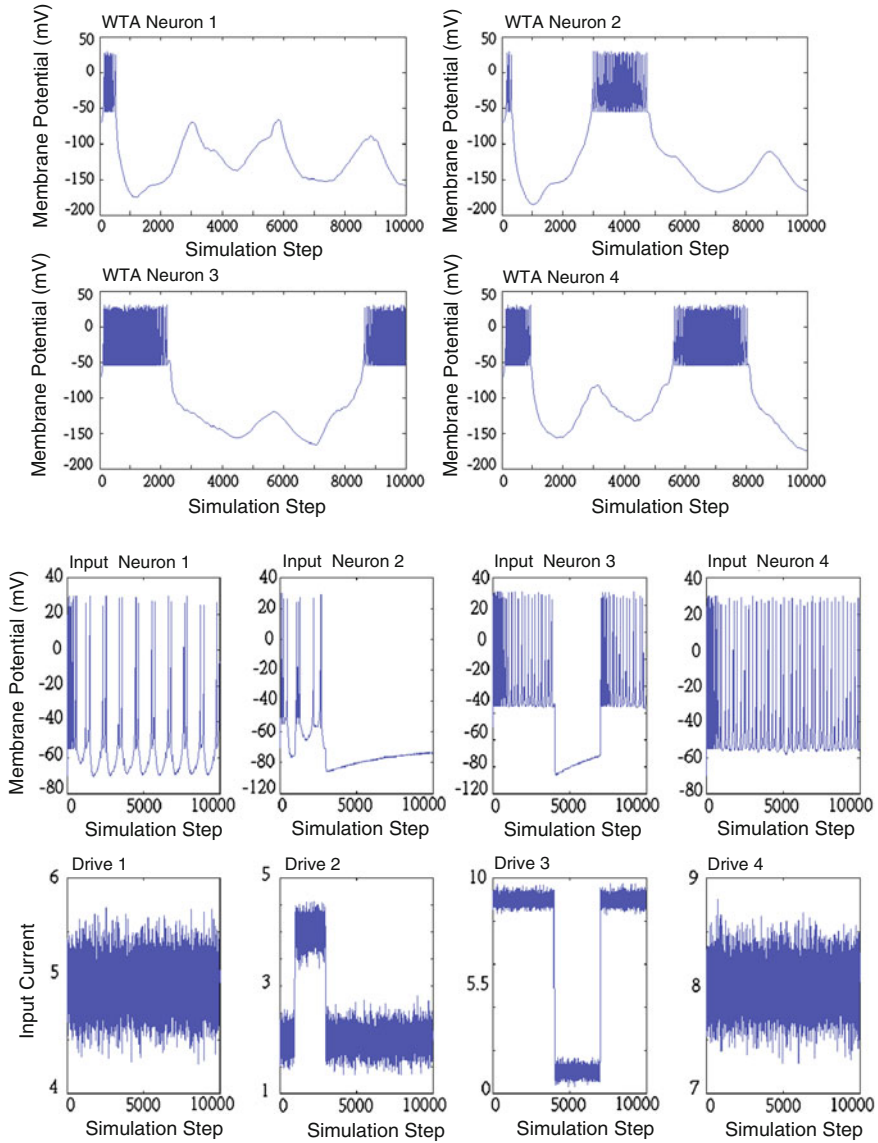
## 2.7 Real Life Scenario Application

In the Sect. 2.6.4 a model of the main parts of a fly brain computational model has been introduced. Herewith the capability of the model to solve realistic tasks is shown. Modifying the behavior repertoire but maintaining the conceptual structure of the general model we can obtain a versatile robot that is able to learn about the environment, to make choices and to face potentially dangerous situations. The experiment presented in this section is only one example of the real applications of the insect brain model, and it could be easily modified or generalized [41, 42].

### 2.7.1 Description of the Experiment

Let us imagine to have a critical situation in which, after a disaster (earthquake, fire) it is necessary to rescue people trapped in a place. Often situations like this are very dangerous both for survivors and people who try to help them. Now let us imagine to have a smart robot able to explore the environment and which can learn, recognize people and remember their position. Such a robot could manage a critical situation acquiring the information needed to solve it. In the present experiment an environment that can represent a place after a disaster has been implemented into a robot simulator. The robot has to explore the environment, find some good objects that it is able to recognize, remember their position and learn about all kinds of dangers present in the environment. At the end of the exploration, the robot must escape from the environment and give all the information useful for humans to know the position of the survivors and organize a safe rescue. In order to solve this problem, the behavior repertoire of the robot has been limited to two possible behaviors, exploration (rover type) and homing. In the same way, two drives are considered, Curiosity and Sleep, the latter indispensable for the robot to understand when to leave the environment and return home; for this, a virtual battery level sensor is used. The MBs model was also simplified: only the olfactory learning model will be considered. The synaptic weights, the synapses time constant and the integration step are the same of the previous simulations. Every robot step of the robot includes only one hundred simulation steps of the MBs and BSN neural networks.

The arena implemented for the simulation and the results are shown in Fig. 2.16. The Home represents the starting point for the robot exploration and the point the robot has to reach at the end of the simulation.



**Fig. 2.15** Results of the simulation of the BSN. After a short transient in which all the WTA neurons are firing, only one neuron can win the competition. The transient is a consequences of the time response of the synapses between WTA neurons. Variation of the drives could also lead to new transient, in which the WTA neurons compete. A low value of the auto-excitatory synapses weights in the WTA layer can cause a continuous switching of the selected behavior, while a too high value leads to a conservative behavior selection

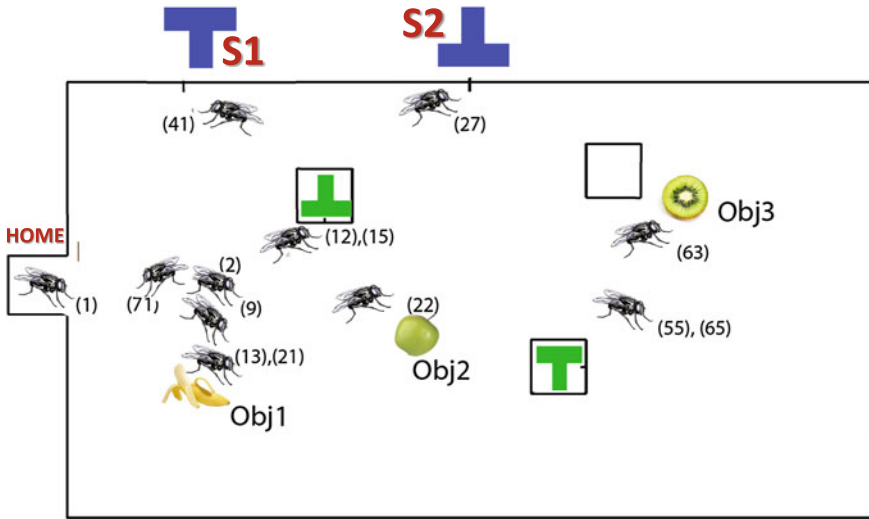
$S_1$  and  $S_2$  represent the position of the targets: let us assume the robot considers them as interesting objects and, after an approach, it is able to recognize them. Let us assume that the targets are a blue T-shaped object and a blue inverted T-shaped object. Identical objects  $Obj_1$ ,  $Obj_2$  and  $Obj_3$ , are also considered. The robot cannot see them, but can sense them thanks to another sensorial system (i.e. olfactory). The robot is punished every time it tries to approach them. In the environment, two other objects are present, a green upright T-shaped object and a green inverted T-shaped object. The robot can detect them with the visual system. The robot is punished only when it tries to approach the first one, while the second one is neutral.

After a long exploration, the robot must be able to detect the targets, learn to avoid as soon as possible the objects  $Obj_1$ ,  $Obj_2$  and  $Obj_3$ , understand that the green upright T-shaped object is dangerous and finally reach the Home and give the position of the targets at the end of the exploration. Mushroom Bodies model will be used for the learning involving  $Obj_1$ ,  $Obj_2$  and  $Obj_3$ ; the protocerebral bridge model will be used for the detection of the objects and the fan-shaped body model for the visual learning; the ellipsoid body model is indispensable for homing and remembering the position of the targets. For this simulation, the capabilities of real flies have been extended, for instance, improving the performances of the EB that is now able to store multiple target information in a long time memory. This is an example of how the elementary functions of the *Drosophila* brain that allows the insect to face with its world can be easily extended in a modular way to make a robot able to fulfill more complex tasks, not affordable for the real fly. The Behavior Selection Network is useful to select the homing behavior if the battery level is too low. The parameters of the model have been set so that the robot can sense odors if its distance is lower than 3 m away from the nearest odor source, while it is punished if its distance from that source is  $< 1$  m. In the same way, the visual system of the robot can detect objects if they are closer than 2.5 m. It is punished if an object is closer than 1.5 m. The arena used for the simulation is 28 m long and 15 m wide.

### 2.7.2 Results and Discussion

In this section experimental results obtained in a typical simulation are shown, discussing step by step the behavior of the robot. Only the most relevant robot steps are depicted in Fig. 2.16, whereas Fig. 2.17 shows the MBs model response during the whole simulation.

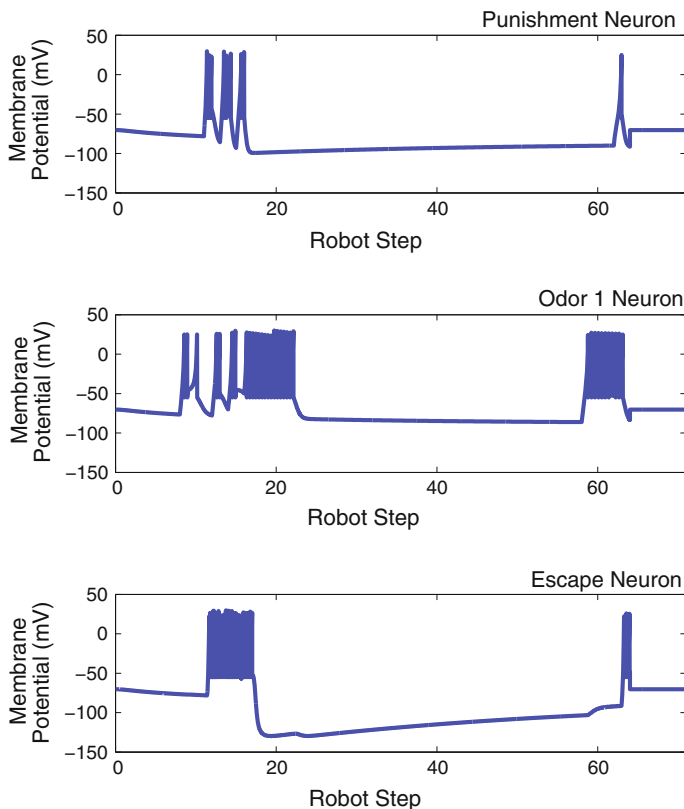
At step 1 the robot starts the simulation from the Home position. At the second step the robot enters the arena and begins an exploration behavior. The ellipsoid body model updates the position of the robot. Neurons of the MBs model are not stimulated and they lie in their silent state. At the following step (step 5, not shown), the robot uses the increase of the mean free path algorithm. The EB model updates the position of the robot. During the exploration, the robot must find objects and sense odors. At step 9, the robot senses  $Odor_1$ , but it is not punished, because it is not close enough to  $Obj_1$ . In the following step the robot continues its exploration following



**Fig. 2.16** Most relevant robot steps of the proposed simulation. After the exploration of the environment the robot returns to the Home and gives the position of the target  $S_1$  and  $S_2$ . Moreover, information about the dangers in the environment are stored in the FB and the MBs model

the increase of the mean free path algorithm, while the EB model updates the position it has stored. At step 11 (not shown), the robot detects the green T-shaped object. The FB model extracts features from this object and the robot tries to approach it. While the robot is approaching the new object, it is punished (step 12). After being punished, the robot escapes from the green upright T-shaped object (step 13). It has to be notice an unexpected situation: the robot sensed  $Odor_1$  and was punished after two subsequent steps, due to the punishing visual input, and not for the odor. So, even if not planned in this way, the robot has made an association between  $Odor_1$  and Punishment. This situation is plausible and it is a natural consequence of the correlation based on STDP learning.

As a consequence, the association between  $Odor_1$  and the need to escape is reinforced. While it is escaping, the robot again detects  $Obj_1$ , senses  $Odor_1$ , is punished and escapes again in the opposite direction (step 14, not shown), reaching once more the green inverted T-shaped object (step 15). The robot is then punished for the third time. At step 16 (not shown), the robot is escaping again. At step 20 and 21 the robot is sensing  $Odor_1$  again, without being punished. It is very interesting to analyze how the MBs model responds to this contradictory situation. Studying the firing of each neuron of the MBs model, it is possible to see that at a first time the robot was punished immediately after sensing  $Odor_1$ , while at a second time it senses  $Odor_1$  but it is not punished. In this way, at a first time the robot made an association between punishment and  $Odor_1$ , but at a second time this association was weakened. However, the synaptic weight between the  $Odor_1$  neuron and the Escape neuron of the olfactory learning model was not high enough to make the

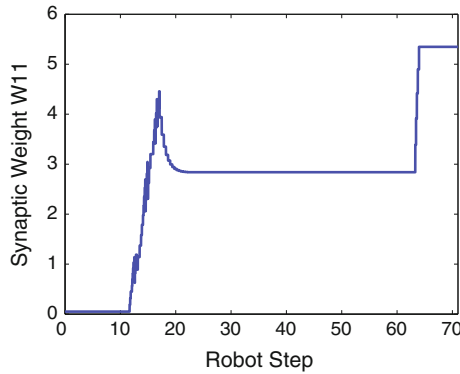


**Fig. 2.17** Mushroom Bodies model response during the simulation

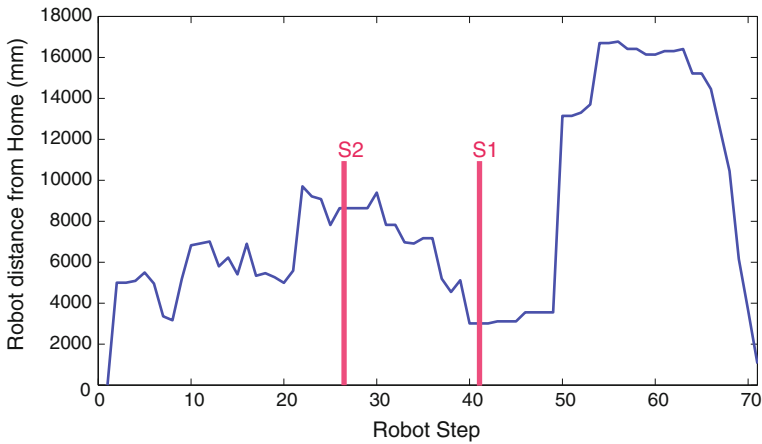
robot escape when sensing again  $Odor_1$ , without being punished. Now the robot continues its exploration of the arena. At step 22 the robot is near  $Obj_2$ , it is sensing  $Odor_1$  again but it is not close enough to be punished. The association between punishment and  $Odor_1$  must decrease again. While exploring, the robot detects the first target (step 27). The fan-shaped body analogue extracts the features of the object, the robot recognizes the target and tries to approach it. The EB model stored the position of the robot. The target  $S_2$  is now reachable in the future. The robot leaves the object and begins another exploration.

After many steps, the robot detects and reaches the target  $S_1$  and stores its position (step 41). After leaving the second target, the robot begins another long exploration. At step 55, the robot is into the area of detection of the green T-shaped object, but in this case the PB model leads the robot to consider that object repulsive because it is standing in the rear of the robot, therefore the robot leaves the object. The robot continues its exploration and, detecting  $Obj_3$ , the robot senses  $Odor_1$  again at step 59. At step 63, the robot is close enough to  $Obj_3$  to be punished. Because of the position of the robot, the punishment is not so strong, but the robot is sensing  $Odor_1$





**Fig. 2.18** Trend of the synaptic weight of the synapses between the *Odor*<sub>1</sub> receptor neuron and the Escape neuron, in the pre-motor area. The synaptic weights of the MBs olfactory learning model are subject to STDP learning. The higher the value of the weight is, the faster the robot will escape if punished while sensing that odor. If the weight exceed a certain threshold, the robot sensing that odor will escape even if not punished at all. For clarifications about the parameters refers to Fig. 2.6



**Fig. 2.19** During the simulation, the EB model estimates the distance of the robot from the Home position. Errors in the position are due to the simulated odometry error and to the path integration method approximations

and it is recalling the association with punishment: even if the Punishment neuron only spikes once, the robot escapes.

Analyzing MBs response and the synaptic weights at step 64, it is evident how the robot reinforced the association between *Odor*<sub>1</sub> and Punishment, as shown in Fig. 2.18. Learning allowed the robot to escape fast, without strong punishment. After escaping, at step 65, the robot meets again the green T-shaped object.

While the robot tries to approach it, the low output level of the virtual battery sensor determines the behavior and initiates the homing procedure. The EB model

is involved to remember the Home position. The response of the EB model at step 65 is shown in Fig. 2.19. At steps 67, 69 (not shown) and 71 the robot tries to return to the Home position. At the end of the simulation, the robot can communicate the approximated position of the targets. Moreover, the robot is aware of the association between an odor and a danger. Nevertheless, in this simulation, the robot was not able to certainly associate a visual feature with reward or punishment, because it has been punished only once while approaching a landmark.

## 2.8 Conclusions

The concept of cognitive abilities is commonly associated to humans and animals like mammals, birds and others. Nevertheless, in the last years several research groups have intensified the studies on insects that possess a much simpler brain structure even if they are able to show interesting memory and learning capabilities. In this chapter, some results toward the design and implementation of a model of the insect brain inspired by the *Drosophila melanogaster* have been presented. Particular attention was paid to the main neural centers the Mushroom Bodies and the Central Complex. In this chapter the parts of the model have been presented and simulation results are reported. In the following Part III of the book a Software/Hardware framework, where the complete architecture could be tested and evaluated by using both simulated and real robots, is presented.

## References

1. B. Webb, T.R. Consi, *Biorobotics: Methods and Applications* (AAAI Press/MIT Press, Menlo Park, 2001)
2. M. Dorigo, T. Stutzle, *Ant Colony Optimization* (MIT Press, Cambridge, 2004)
3. M.V. Srinivasan, S. Zhang, M. Altwein, J. Tautz, Honeybee navigation: nature and calibration of the odometer. *Science* **287**(5454), 851–853 (2000)
4. T.W. Secomb F.G. Barth, J.A.C. Humphrey, Locust's looming detectors for robot sensors. in *Sensors and sensing in biology and engineering*, ed. by F. Rind, R. Santer, J. Blanchard, P. Verschure (springerwien, newyork, 2003)
5. B. Webb, T. Scutt, A simple latency dependent spiking neuron model of cricket phonotaxis. *Biol. Cybern.* **82**(3), 247–269 (2000)
6. P. Arena, L. Fortuna, M. Frasca, L. Patané, M. Pavone, Realization of a CNN-driven cockroach-inspired robot, in *International Conference on Circuits and Systems (ISCAS)*, 2006
7. J.T. Watson, R.E. Ritzmann, S.N. Zill, A.J. Pollack, Control of obstacle climbing in the cockroach, *Blaberus discoidalis*. I. Kinematics. *J Comp Physiol A* **188**, 39–53 (2002)
8. H. Cruse, T. Kindermann, M. Schumm, J. Dean, J. Schmitz, Walknet a biologically inspired network to control six-legged walking. *Neural Networks* **11**, 1435–1447 (1998)
9. B. Webb, J. Wessnitzer, Multimodal sensory integration in insects—towards insect brain control architectures. *Bioinspiration Biomimetics* **1**, 63 (2006)
10. U. Homberg, Structure and functions of the central complex in insects. in *Arthropod Brain, Its Evolution, Development, Structure and Functions*, vol. 347–367, ed. by A.P Gupta, (Wiley, New York, 1987)

11. G.O. Lopez-Riquelme, W. Gronenberg, Multisensory convergence in the mushroom bodies of ants and bees. *Acta. Biol. Hung.* **55**, 31–37 (2004)
12. P. Arena, L. Fortuna, M. Frasca, L. Patané, Learning anticipation via spiking networks: application to navigation control. *IEEE Trans. Neural Networks* **20**(2), 202–216 (2009)
13. P. Arena, L. Patané, (eds.), Spatial temporal patterns for action-oriented perception in roving robots. in *Cognitive Systems Monographs*, vol. 1 (Springer, Berlin, 2009)
14. P.F.M.J. Verschure, T. Voegtlin, R.J. Douglas, Environmentally mediated synergy between perception and behaviour in mobile robots. *Nature* **425**, 620–624 (2003)
15. R. Brooks, New approaches to robotics. *Science* **253**, 1227–1232 (1991)
16. R.C. Arkin, *Behavior-Based Robotics* (MIT Press, Cambridge, 1998)
17. P. Arena, L. Patané, P.S. Termini, Decision making processes in the fruit fly: a computational model. in *Frontiers in Artificial Intelligence and Applications—Proceedings of the 21st Italian Workshop on Neural Nets*, vol. 234 (Seville, Spain, 2011), pp. 284–291
18. E. Arena, P. Arena, L. Patané, Efficient hexapodal locomotion control based on flow-invariant subspaces. in *18th World Congress of the International Federation of Automatic Control (IFAC)*, Milan, Italy, 2011
19. G.M. Shepherd, *Neurobiology* (Oxford University, New York, 1997)
20. P. Arena, C. Berg, L. Patané, R. Strauss, P.S. Termini, An insect brain computational model inspired by *Drosophila melanogaster*: architecture description, *WCCI 2010 IEEE World Congress on Computational Intelligence* (Barcelona, Spain, 2010), pp. 831–837
21. P. Arena, M. Cosentino, L. Patané, A. Vitanza, Sparkrs4cs: a software/hardware framework for cognitive architectures, *5th SPIE's International Symposium on Microtechnologies* (Czech Republic, Prague, 2011), pp. 1–12
22. P. Arena, L. Patané, Simple sensors provide inputs for cognitive robots. *Instrum. Measur. Mag.* **12**(3), 13–20 (2009)
23. E.M. Izhikevich, Simple model of spiking neurons. *IEEE Trans. Neural Networks* **14**(6), 1569–1572 (2003)
24. R. Strauss, M. Mronz, Visual motion integration controls attractiveness of objects in walking flies and a mobile robot, *International Conference on Intelligent Robots and Systems* (Nice, France, 2008), pp. 3559–3564
25. L. Alba, P. Arena, S. De Fiore, L. Patané, R. Strauss, G. Vagliasindi, Implementation of a *Drosophila*-inspired model on the eye-ris platform. in *Proceedings of the IEEE International Conference on CNNA* (Berkley, 2010)
26. G. Liu, H. Seiler, A. Wen, T. Zars, K. Ito, R. Wolf, M. Heisenberg, L. Liu, Distinct memory traces for two visual features in the *Drosophila* brain. *Nature* **439**, 551–556 (2006)
27. H. Hoffmann, R. Moller, An extension of neural gas to local pca. *Neurocomputing* **62**, 305–326 (2004)
28. P. Arena, S. De Fiore, L. Patané, P.S. Termini, R. Strauss, Visual learning in *Drosophila*: application on a roving robot and comparisons, *5th SPIE's International Symposium on Microtechnologies* (Czech Republic, Prague, 2011), pp. 1–12
29. K. Neuser, T. Triphan, M. Mronz, B. Poeck, R. Strauss, Analysis of a spatial orientation memory in *Drosophila*. *Nature* **453**, 1244–1247 (2008)
30. R. Wehner, G. Hartmann, The ant's path integration system: a neural architecture. *Biol. Cybern.* **73**, 483–497 (1995)
31. R.L. Davis, X. Liu, Insect olfactory memory in time and space. *Curr. Opin. Neurobiol.* **6**, 679–685 (2006)
32. J. Dubnau, C. Margulies, T. Tully, Deconstructing memory in *Drosophila*. *Curr. Biol.* **15**, 700 (2005)
33. L.F. Abbott, S. Song, K.D. Miller, Competitive hebbian learning through spike-timing-dependent plasticity. *Nat. Neurosci.* **3**, 919–926 (2000)
34. L.F. Abbott, S. Song, Cortical development and remapping through spike timing-dependent plasticity. *Neuron* **32**, 339–350 (2001)
35. P. Arena, S. De Fiore, L. Patané, M. Pollino, C. Ventura, Stdp-based behavior learning on tribot robot. in *Proceedings of IEEE/RSJ International Conference SPIE* (2009)

36. L. Vine, L. Gottschalk, P.J. Shaw, L. Seugnet, Y. Suzuki, D1 receptor activation in the mushroom bodies rescues sleep-loss-induced learning impairments in *Drosophila*. *Curr. Biol.* **18**, 1110–1117 (2008)
37. R. Biesinger, K.G. Gotz, Centrophobism in *Drosophila melanogaster*. Physiological approach to search and search control. *J. Comp. Physiol. A* **156**, 329–337 (1987)
38. M.B. Sokolowski, S. De Belle, Heredity of rover/sitter: alternative foraging strategies of *Drosophila melanogaster* larvae. *Heredity* **5**(9), 73–83 (1987)
39. J.R. Martin M. Besson, Centrophobism/thigmotaxis, a new role for the mushroom bodies in *Drosophila*. *J. Neurobiol.* **62**, 386–394 (2005)
40. A. Wen, T. Zars, K. Ito, R. Wolf, M. Heisenberg, L. Liu, G. Liu, H. Seiler, Distinct memory traces for two visual features in the *Drosophila* brain. *Nature* **439**, 551–556 (2006)
41. P. Arena, L. Patané, P.S. Termini, A. Vitanza, R. Strauss, Software/hardware issues in modelling insect brain architecture. in *4th International Conference on Intelligent Robotics and Applications (ICIRA)*, Aachen (Germany), 2011
42. P. Arena, L. Patané, P.S. Termini, An insect brain computational model inspired by *Drosophila melanogaster*: simulation results, *WCCI 2010 IEEE World Congress on Computational Intelligence* (Barcelona, Spain, 2010), pp. 838–845

<http://www.springer.com/978-3-319-02361-8>

Spatial Temporal Patterns for Action-Oriented  
Perception in Roving Robots II

An Insect Brain Computational Model

Arena, P.; Patanè, L. (Eds.)

2014, XIV, 371 p. 256 illus., 205 illus. in color.,

Hardcover

ISBN: 978-3-319-02361-8