A computational model for motor learning in insects

Paolo Arena, Sergio Caccamo, Luca Patané and Roland Strauss

Abstract—The aim of this paper is to propose a computational model, inspired by Drosophila melanogaster, able to handle problems related to motor learning. The role of the Mushroom Bodies and the Central Complex in solving this problem is analyzed and plausible biologically inspired models are proposed. The designed computational models have been evaluated in simulation using a dynamic structure inspired by the fruit fly. The obtained results open the way to new neurobiological experiments focused to better understand the underlined mechanisms involved, to verify the feasibility of the hypotheses formulated and the significance of the obtained results.

I. INTRODUCTION

Learning capabilities are fundamental mechanisms of living beings from Aplysia to mammals [1]. Different forms of learning are present in animals: they have to show adaptive capabilities to survive in changing environments. For our purposes, motor learning is defined as the process by which we acquire precise, coordinated movements needed to solve a task. We can speculate that motor learning may have been one of the earliest type of associative learning to evolve. It has been demonstrated in the ventral nerve cord of insects [2] and is probably universal in moving animals. Motor learning is classified as an implicit procedural learning. Errors are monitored and used to guide adaptive changes without conscious awareness. Instead of learning a concept, the subject learns an accurate and appropriate motor response. Distinctions have also been made between short- and long-term motor memory (separated by the process of consolidation). In performing motor learning agents apply operant strategies in which a movement is made and sensory feedback is used to evaluate its accuracy [3].

Motor learning was largely studied in mammals with a number of behavioural paradigms [4]. It has been demonstrated that different motor areas of the cortex, the cerebellum, the striatum and the spinal cord are all involved in different forms of motor learning.

In insects, beside their tiny body and brain, there are different examples of learning and adaptive motor schemes depending on the performed task. For instance, in honeybees, different types of motor learning can be identified. An interesting case consists in the changes of antennal movements after the prolonged presentation of an obstacle. The animals showed antennal movements for several minutes that were correlated with the position and shape of the removed object.

This form of “behavioural plasticity” can be interpreted as a typical example of motor learning [5].

These observations suggest that bees acquire information about the position of an object during scanning. This acquisition of information occurs without sucrose reward and might represent a simple form of associative learning.

Compared to antennal motor learning, acquisition of information during associative reward-dependent learning is much faster. While an animal has to scan an object for at least 10 min for significant motor learning, a bee which is rewarded after scanning an object shows significant behavioural changes already after 3-4 rewards. This is equivalent to 10-15 s of scanning the object. This comparison shows that reward-dependent learning is much more efficient than motor learning [5]. The results of motor learning and reward-dependent conditioning suggest that bees have tactile spatial memory. Other examples of motor learning in insects are reported in [6] where the role of proprioceptive signals is investigated in locust flight behavior. In an interesting paper on flies capabilities [7] a series of properties have been identified during learning: the system has a desired state to be reached; to perform this task the system randomly activates a range of motor programs; efference copies of the motor programs are compared with those sensory inputs which represent the deviation from the desired state; if a significant correlation is detected for a certain motor program, this is applied to the system. Other studies were also carried out on bumblebees [8] and butterflies [9].

Looking at Drosophila melanogaster, one of the first studied forms of learning is related to olfaction. These learning processes are well localized in the Mushroom Bodies (MBs), the neuropile responsible for both the short-term and long-term component of olfactory memory. Several experiments demonstrate how, through classical conditioning, flies can associate a meaning with either positive or negative reinforcement signals to olfactory inputs [10]. Olfaction is an ancestral sensory modality that massively involves the MBs; however, recent studies identified MBs as also responsible for other learning processes. In fact, MBs are well known in bees and flies, beside for their role in performing associative learning and memory in odor conditioning experiments [11], [12], also for handling other sensory modalities and different forms of learning mechanisms [13], [14].

Considering the MBs as a multimodal learning center, models able to show properties like attention [15], expectation [16] and delayed-matching to sample tasks [17] have been proposed recently. Modeling the MB neuropile as a pool of spiking neurons branching into lobes and introducing the synaptic connections identified among MBs and the other structures directly involved like the Projection Neurons (PN)
and the Lateral Horn (LH), it is possible to investigate the emergence of interesting neural activities that can generate specific behaviors shown in flies. The role of MBs in motor learning is also known from ongoing experiments: in particular the short-term memory components is not obtained if MBs plasticity is inhibited in mutant flies. MBs can be modeled as a reward-driven parameter adapter that improves the fly performance while a task like climbing over a chasm is continuously repeated for multiple trials [18]. Besides the MBs, the Central Complex (CX) is another well-known insect neuropile that is mainly involved in visual learning and orientation. Looking at the ongoing experimental campaign performed with *Drosophila melanogaster*, the long term component for motor learning may need an intact CX.

In real life, motor learning and other forms of associative learning can be mixed together and it is not simple to distinguish between each other. Wang and coworkers proposed an experiment with wild-type and mutant flies in a flight simulator. The obtained results demonstrate that both visual associative learning and motor learning were involved in the visual operant conditioning of *Drosophila* and that the two learning forms could be dissociated and they might have different neural bases [19].

Starting from these behavioral experiments and topological information on the fruit fly brain structures, models able to explain the known data on interesting fruit fly motor learning and memory capabilities are developed.

II. MODELING MOTOR LEARNING

Analyzing the information acquired from literature on motor learning, the key issues identified for a computational model design and implementation are here discussed in details. Considering a task that needs a specialization of the insect motor skills to be solved in an optimal way, a mechanism for searching the suitable system parameters to be applied is needed. The generation of pseudo-random parameters constrained by the insect body is an important element needed to improve the system motor activity. This process will produce a series of successful attempts that improve the system performance stabilizing the new set of optimal parameters through the generation of an internal reward signal. In insects, the generation of these trials can be performed at the level of the thoracic ganglia, but the statistic shaping and the final selection of the successful parameters that drive the basic behaviors is mediated by the MBs. These learning processes are at the basis of the implementation of a short-term working memory. It is important to notice that the visual input processed by the MBs is certainly not so detailed as for the CX; it could be rather poor and rough, so behavioral adaptation is only due to a reward signal. Input signals come in parallel to the whole fly brain. This means that meanwhile the MBs process the loosely visual input signals (essentially event driven signals), to try to associate a rewarded sequence of parameter associations useful to obtain a reward, the CX is contemporarily processing the complete visual input.

Let us assume that inside the insect brain, the MB and CX work together, even without any direct connection, to acquire relevant information from the environment, in particular: MB is involved in random search in the space of motor parameters guided by a reward function (i.e. formation of a short-term memory); CX is looking at the visual scene, trying to learn visual associations between the visual stimuli (in the case of gap climbing, for example, the near and far side of the border, which is the rewarding side, since the fly is forced to climb) and place specific goodness or badness areas within that visual scene (for example a narrow area just above the border). This is the starting of the CX learning mainly based on visual input, the long-term component of the memory is contained here, and is being formed incrementally. The motor attempts, while shaping the successive behavioral choices, concurrently give the possibility to the CX to learn successful visual associations between each visual frame, where attention is being currently paid (for example the opposite gap border in the case of a gap climbing scenario) [18], and the visual area where, for example, the front legs should be targeted in order to generate a good reward signal (notice that this is a visuo-tactile or visuo-spatial targeting association similar to the Morris water maze [20]). In fact, it should emerge from the trials that only a suitable front leg targeting in a precise visual area next to the gap border will contribute to success: the CX will learn this visual association, which, of course is far slower than the MB learning process. Moreover, while the MB could create a sequence of parameter adaptations triggered by a virtual time signal, loosely related to the received rough visual input, the CX, working at a frame-size, in principle does not need time but has to associate, in the case of gap climbing, to particular image frames of interest (for example the near gap shape, far gap edge, etc) specific visual target areas where to place the front legs. The CX could so have the role of transforming the sequence of parameters found by the MBs and probably transferred to the ganglia as a new library of motor behaviors, into a frame-related visual association. This could explain the speeding up of the whole process which, essentially, associates a visual image to an area of targeting, eliciting the corresponding, already learned, motor behavior. We tentatively assume that the horizontal fiber system is involved in this type of motor learning as it has recently been shown to fulfill acute visual targeting tasks during climbing [21]. Of course, if this horizontal fiber system is experimentally prevented from learning, the visual-motor associations cannot take place, and the already learned behaviors cannot be associated to the visual stimuli. This could explain the lack of motor improvements and the missing long-term memory. Further down the motor system, the horizontal fiber system, which ends at the ventral lobes, might in its intact state trigger a cascade of plastic changes in the lower motor centers in the thoracic ganglia to make the improvements permanent. As an advantage of this modeling process we are leaving to the CX the classical visual learning capabilities. In the fan-shaped body (FB) the rewarding visual
The robot, do not show particular climbing skills, since neurons. A forward FG) is adaptation for the effectiveness of the performed actions in solving the on-going task. An event detector is used to trigger the evaluation.

The generated reinforcement signal is provided to the MBs where a Random Function Generator (RFG) is used to update a set of control parameters acting on the CPG. To develop a long-term memory of the optimal parameters found during learning and to allow interpolation capabilities to generate feasible behaviors also in situations similar to others seen in the past, a recurrent neural network was included. In particular an Echo-state network (ESN) has been used to memorize the optimized temporal evolution of the tuned parameters to improve the final motor output of the system acquired through learning.

ESNs are recurrent neural structures were a readout map can be updated through a supervised learning mechanism to reproduce the time evolution of the system taken into account [24]. The main idea consists of supplying an input signal to a randomly connected recurrent neural network called a reservoir that will provide a nonlinear response signal. By combining the output of each neuron of the network with a linear weighted function, it is possible to obtain the desired output and to tune the weights of the readout map. A selector block is finally used to determine when a random trial has to be performed; alternatively the output of the ESN can be used to guide the motor actions.

III. MOTOR LEARNING: APPLICATION TO CLIMBING

Motor learning in a multi-limb system can be applied to a variety of different tasks that can involve a large number of degrees of freedom, requiring the fine tuning of multiple parameters. Among the scenarios that can be considered, in the proposed simulation a case study related to step climbing has been considered. The simulation environment used for the step-climbing experiment is shown in Fig. 3. The problem is common in nature: insects have to deal with complex terrains and are able to adapt their motor responses to climb over obstacles. Typically, flying insects, like Drosophila melanogaster, do not show particular climbing skills, since they are used to flying over gaps. However, if the wings are cut off, the fly is able to show distinct motor learning abilities. The goal of the motor learning procedure used in the experiments is to improve the climbing capabilities of the simulated robot by modulating over time a set of parameters used in the motor control layer.

The task is quite complex and involves an optimization procedure of all the joint movements. The task has been divided into different phases as depicted in Fig. 4.

The approaching phase is guided by the visual sensors that identify the distance from the obstacle and its height.
When the distance from the step is below a given threshold the Phase 1 is triggered and the front leg parameters are modified through the Random Function Generator to extend the swing phase for a safe touch down onto the top of the step. For the sake of simplicity, a subset of key parameters was trained in this step of learning; in particular the value of the bias for the coxa and tibia joints and the gain for the femur and tibia joints in the front legs. If the expected result is obtained, that is the front legs are stable on the step and the body has lifted off, a reward function is calculated on the basis of the energy spent in fulfilling the subtask. In our case a function of the angular movement covered by the leg joints was considered as the task for learning. The reward is then compared with the previously found optimal value and, when an improvement is obtained, the new sets of functions are stored in the ESN readout map. In the considered simulation the ESN consists of a pool of internal neurons with a single input that corresponds to the height of the step to be climbed and multiple different corresponding outputs, each one providing the time evolution of each specific leg parameter that is subject to learning. For the considered task we have a single lattice with one input (i.e. step height) and a total of ten read-out maps, one for each parameter to be learned. The joint position evolution and the effect of the parameter adaptation for the front legs is reported in Fig. 5(a).

The subsequent phase follows a similar procedure considering as relevant parameter to be adapted those ones referring to the hind legs: bias of femur and tibia joints to facilitate the climbing of the middle legs. The evaluation parameter used in this stage is the horizontal position of the center of mass. The effect of the parameter adaptation for the middle legs is reported in Fig. 5(b). The third phase consists in elevating the hind legs on the step acting on parameters of the middle legs (i.e. bias of the femur and gain of the coxa joint) and of the hind legs (coxa and femur joint gain) as shown in Fig. 5(c).

In the actual experiments the function randomly generated to modify the joint parameters can assume an exponential, a quarter sinusoidal or a sigmoidal shape and reach a steady state value in a given time that is below the stepping time. To better understand the effect of the parameter adaptation on the CPG, the block scheme of the motor layer is shown in Fig. 6 that refers to the middle leg controller. Without discussing in details the structure (see [25] for a complete description), the CPG generates coordinated signals for the legs guiding the phase locking within the different gaits. These waveforms are adjusted to the kinematics of each leg through a set of parameters and used to control the position of coxa, femur and tibia joints of the robot. The parameters of the middle legs subject to learning during the last phase are the coxa gain ($K_{M,C}$) and the femur bias ($I_{M,F}$). In the proposed simulation the integration time is $dt = 0.01s$, the stepping time is about $1.5s$ and the parameters reach the steady state in a range within $[20-60]$ steps. Fig. 7 shows the signals used to evaluate the transition between phases. Fig. 8 underlines the results of a successful climbing behavior obtained through the adaptation in time of the leg parameters. To evaluate the optimal configuration for the ESN, a series of tests was performed. We decided to fix the spectral radius to 0.9 and to include a low level of feedback from the output neurons with a scaling factor of 0.05. Concerning the internal units, a statistical analysis was performed to determine the suitable configuration. Finally the number of internal neurons...
Fig. 5. Effects of the parameter adaptation on the leg joints (only the left side is shown). A limited number of parameters can be subject to learning in the three phases of the obstacle climbing procedure: (a) in phase 1 only the front legs are involved, (b) the hind legs in phase 2 and (c) both middle and hind legs in phase 3. The effect of the parameters on the leg joint trajectory is limited to the current phase.

Fig. 6. Network devoted to control a middle leg. The CPG generates the coordinated signals that are used to control the leg joint position after a suited conditioning. During phase 2 the coxa gain ($K_{M,C}$) and the femur bias ($I_{M,F}$) of the middle legs are subject to learning.

Fig. 7. Shift among phases during learning. The triggered events used to evaluate the successful of a motor action in a complete task phase are the $y$ and $z$ coordinate of the center of mass of the robot body and the distance sensor placed in the front part of the system used to detect the presence of an obstacle. The robot can proceed to the successive phase if the expected conditions are verified, in this case a reward is calculated and used for the learning, otherwise if the phase is not completed in a given time window, the system is reset to the starting initial condition.

was set to 140 in the simulations.

Looking at the learning process, Fig. 9 depicts the trend of reward and delta reward functions (i.e., the error between the best-to-date and the current reward function value) during the trials for the three climbing phases while learning to overcome a step with a height of 12 units. The Reward is calculated on the basis of the energy spent to fulfill the subtask: in detail, the absolute value of the relative angular position of each joint involved in parameter adaptation is integrated until the rewarding event for the subtask occurs. If the subtask is not fulfilled then the parameters used are not stored in the ESN. While trying to overcome the obstacle in each phase, the robot is allowed to change the parameters for a maximum of three times; then, if the next phase is not reached, the simulation is reset and the robot starts again from the approaching phase. The parameter searching strategy is substituted by a test every three climbing attempts using the actual stored optimal values. The switch between phases during the learning trials is shown in Fig. 10 where the capability to reach the last phase improves during learning.
it is seen how the performance obtained during testing is significantly higher with respect to the learning phase with the exception of the highest step tested that is unsurmountable for the Drosophila-inspired robot.

During learning different parameter evolutions can be stored in the ESN but, when a better solution is found, the network is trained again with the new signals. Fig. 11 shows the temporal evolution of the learned parameters during different trials. The postures of the fly-inspired robot model during the different phases of the climbing task are reported in Fig. 12. Tab. I reports the learning (L) and testing (T) performances obtained while trying to overcome steps with different heights. The robot receives learning sessions for obstacles with height 12 and 18 units, and has been tested also with obstacles of 15 and 20 units of height to evaluate the interpolation/extrapolation capability of the architecture. The number of successful attempts in percentage with respect to the trials performed is indicated for the three phases and also for the complete behavioral procedure. It can be noticed that the interpolation performance is within that one obtained in learning. Furthermore, given that the highest step is unsurmountable for the robot, the generated parameters allow to fulfill the first two phases but not the last one. The time evolution of this performance index for the complete climbing behavior during learning is shown in Fig. 13 for two obstacle heights: the complexity of the task increases with the step height and consequently the performance slightly decreases. The time evolution of the four parameters used during the first phase are reported in Fig. 14. In this case the network received learning patterns only for steps with heights between 12 and 18 units and produced outputs also for steps of 15 and 20 units of height interpolating and extrapolating the knowledge stored in the synaptic output weights.

### Table I

<table>
<thead>
<tr>
<th>Step height (units)</th>
<th>12 L</th>
<th>12 T</th>
<th>15 T</th>
<th>18 L</th>
<th>20 T</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1</td>
<td>74%</td>
<td>100%</td>
<td>94%</td>
<td>48%</td>
<td>50%</td>
</tr>
<tr>
<td>Phase 2</td>
<td>67%</td>
<td>68%</td>
<td>61%</td>
<td>29%</td>
<td>42%</td>
</tr>
<tr>
<td>Phase 3</td>
<td>25%</td>
<td>81%</td>
<td>90%</td>
<td>41%</td>
<td>6%</td>
</tr>
<tr>
<td>Phase 1-2-3</td>
<td>14%</td>
<td>50%</td>
<td>52%</td>
<td>6%</td>
<td>5%</td>
</tr>
</tbody>
</table>

IV. CONCLUSIONS

Learning capabilities for improving motor skills are fundamental in living beings as well as in robots that have to interact with unstructured environments. Taking inspiration from insects and in particular from Drosophila melanogaster, a computational model for motor learning has been designed and implemented in a simulation environment. The structure is based on the mushroom bodies and central complex neuropile which seem to play a crucial role in active motor learning and long-term memory. For a long-term storage of the parameter time evolution, an Echo-state network was used to evaluate the proposed architecture for motor learning.
Fig. 10. Evolution of the phases reached by the robot when learning to climb a step with height 12 units (a), in testing for the same step (b), during learning a step of 18 units (c), in testing a never seen obstacle with height 15 units (d), and in testing an unsurmountable step of 20 units (e). Green markers indicate a success whereas red ones a failure.

Fig. 11. Time evolution of the output generated by the ESN during the learning phase of the step with height 18 units. The parameters tried during time for the phase 1 (a) 2 (b) and 3 (c) are reported and the final waveforms stored at the end of the learning are depicted in bold lines.

The simulated fly has to learn how to climb over steps of different heights. The obtained results demonstrate that it is possible to learn through a reward-based mechanism the time evolution of several parameters to improve the robot capabilities in climbing an obstacle. The structure was also able to deal with step heights never seen before using the interpolation capabilities of the ESN.

The model introduced follows the idea that brain is a parallel computing architecture, where environmental signals are processed concurrently by the different neural structures devoted to handle with each sensory modality. Faster sensory motor loops learn basic reflex-based skills that can become targets to learn new, possibly more complex sensorimotor capabilities and behavioral responses that help to improve the agent capabilities [26]. The results obtained open the way to new experiments on wild type and mutant flies to further assess and improve the model introduced.
Fig. 12. Snapshots of the Drosophila model while trying to approach the obstacle (a), reach the top of the step with the front legs (b), pull the middle legs on the top (c) and fulfill the task with the hind legs (d).

Fig. 13. Performance index based on the percentage of successful attempts to complete the climbing behavior obtained during the learning phase for a step with height of 12 units (a) and 18 units (b). A sliding window of 40 attempts has been used for the plot.

ACKNOWLEDGEMENT
This work was supported by EU Project EMICAB, Grant N 270182.

REFERENCES

Fig. 14. Output of the ESN for the first phase after learning steps of heights 12 and 18 units. The time evolution of the four parameters for heights 15 and 20 is a result of the interpolation capability of the recurrent network.
