Cogn Neurodyn (2017) 11:259–281
DOI 10.1007/s11571-017-9426-4

RESEARCH ARTICLE

A plausible neural circuit for decision making and its formation based on reinforcement learning

Hui Wei[1] · **Dawei Dai**[1] · **Yijie Bu**[1]

Received: 28 July 2016 / Revised: 13 December 2016 / Accepted: 10 February 2017 / Published online: 18 February 2017

Abstract A human's, or lower insects', behavior is dominated by its nervous system. Each stable behavior has its own inner steps and control rules, and is regulated by a neural circuit. Understanding how the brain influences perception, thought, and behavior is a central mandate of neuroscience. The phototactic flight of insects is a widely observed deterministic behavior. Since its movement is not stochastic, the behavior should be dominated by a neural circuit. Based on the basic firing characteristics of biological neurons and the neural circuit's constitution, we designed a plausible neural circuit for this phototactic behavior from logic perspective. The circuit's output layer, which generates a stable spike firing rate to encode flight commands, controls the insect's angular velocity when flying. The firing pattern and connection type of excitatory and inhibitory neurons are considered in this computational model. We simulated the circuit's information processing using a distributed PC array, and used the real-time average firing rate of output neuron clusters to drive a flying behavior simulation. In this paper, we also explored how a correct neural decision circuit is generated from network flow view through a bee's behavior experiment based on the reward and punishment feedback mechanism. The significance of this study: firstly, we designed a neural circuit to achieve the behavioral logic rules by strictly following the electrophysiological characteristics of biological neurons and anatomical facts. Secondly, our circuit's generality permits the design and implementation of behavioral logic rules based on the most general information processing and activity mode of biological neurons. Thirdly, through computer simulation, we achieved new understanding about the cooperative condition upon which multi-neurons achieve some behavioral control. Fourthly, this study aims in understanding the information encoding mechanism and how neural circuits achieve behavior control. Finally, this study also helps establish a transitional bridge between the microscopic activity of the nervous system and macroscopic animal behavior.

Keywords Neural circuit · Behaviors · Decision-making · Network flow · Reinforcement learning

Introduction

The key point to understand working mechanisms of the brain is to reveal how knowledge, external stimuli, and behavioral decisions are encoded and processed. Traditional research on information processing of the nervous system began with neural encoding on the neuron-level. Research shows that the firing pattern of a single neuron has some relation to external stimulation. Differences in firing rate or firing time intervals may be directly related to the different behaviors of animals (Kreiman 2004; Britten et al. 1992). However, this kind of research ignores the group behavior of neurons. The "All/None" mode of single neurons is not sufficient to describe complex animal behaviors. With the gradual deepening of research, we now know that a network structure consisting of different types of neurons is the key to perform the basic function. Such a structure involves multiple different types of neurons and a large number of synaptic connections to perform different functions. This is identical to various gate circuits that are

✉ Hui Wei
weihui@fudan.edu.cn

[1] Laboratory of Cognitive Model and Algorithms, Department of Computer Science, Shanghai Key Laboratory of Data Science, Fudan University, Shanghai, China

constituted by basic transistor components in an integrated circuit, where different functional circuits are then formed based on these gate circuits to achieve complex operations.

In nature, both humans and lower insects can adapt to their environments and exhibit stable behavior. For example, insects can use polarized light to navigate (Horváth and Varjú 2013), and bees report the distance and azimuth to nectar or pollen through dance. As long as these behaviors are not stochastic, we can assume that there must be specific neural functional circuits in biological nervous systems that lead to these behaviors. Stable behaviors are controlled by stable neural circuits, and action potential (AP) firing within circuits, as well as collaboration among these neurons, generates basic animal behavior. Since, decision logic reflects the most basic requirement that a behavior can successfully implement, any specific implementation requires these basic functions. We believe that the rules through which animals control their behavior can be described by logic language. For a biological nervous system to achieve specific implementation, its structure must be sufficiently complex to achieve basic logic. Furthermore, biological neural systems must cope with a variety of tasks, and each task has its own internal control rules. Therefore, there must be many types of neural circuits to achieve various logical rules in the nervous system. From the logical perspective, any type of behavioral logic can be formally described by a propositional logic expression. With this reliable and complete formal language, we can accurately describe the basic logical rules with which behaviors comply. This set of logical rules has different implementations in different operating environments. For example, it is a strictly defined program in a computer-controlled automatic control system. How can logical rules be achieved in an organism or, more precisely, in a biological nervous system? Moreover, with different firing patterns of neurons and the synergistic connections between pyramidal neurons and intermediate neurons, can the nervous system assemble a circuit to achieve a set of specific logical rules?

Animals, including humans, are not often one-step making the right decision for some tasks. However, a learning process involving reward and punishment feedback forms a correct decision circuit to complete the task. The implementation of a neural decision circuit may not be initially correct, and may require adjustment. We need to assume what the mechanism is?, which lead that the circuit can always be correctly formed. Dopaminergic neurons distributed in the basal ganglia play an important role in learning (Barron et al. 2010). Research indicates that neuromodulation reconfigures circuit properties, such as changing neuronal functions within seconds, minutes, or even hours (Marder 2012). This conclusion enlightens us to study cases where the aforementioned neural circuits may be initially incorrect, and determine whether the circuit can be corrected based on reward and punishment feedback mechanisms.

There are two main hypotheses for information encoding in the nervous system. The first is that the average firing rate (AFR) encoding view, which considers the AFR of neurons or a neural network as an information supporter in the brain and AFR encoding, has biological experiment foundation that has been confirmed in motor neurons (Hebb 2005; Wang 2007; Weliky et al. 2003; Hirata and Aihara 2009). The second hypothesis is the space-time encoding theory (Malsburg 1995), which proposed the necessity of neural modulation on a fast time scale (2–5 ms). In recent years, biologists have become more convinced in the difficulty to explain mechanisms of complex nervous systems, and even the basic neural circuits, given only physiological and anatomical data. Through designing a reasonable neural circuit and real-time dynamic simulation, we can, as best as possible, reconstruct the process of information processing and propagating in biological neural circuits. Spike firing information from simulated neuron clusters not only can study firing modes from both the frequency encoding and space-time encoding views, but also conduct association analysis of circuit neurons. Designing and simulating a circuit for a specific behavior helps us understand its constitution details, information processing, and the information encoding mechanisms of neural circuits. In this paper, we persist that if we cannot explain, on a detailed level of neuronal activity and neural circuit wiring diagrams, why different behavior can be executed correctly, then we cannot claim to have grasped the mechanisms for information processing.

Related works

As pioneers of artificial neural network (ANN) theory, McCulloch and Pitts (1943) proposed the McCulloch–Pitts (MP) model. However, the working mechanism in the MP model is obviously different from the biological neuron. Firstly, the MP model's activation mode is two-valued, but biological neurons actually experience impulse-firing. Secondly, only one type of Boolean neuron is needed in a MP network, while in a real biological neural network, there are at least two types of vastly different neurons and the proportion of their numbers matters. Thirdly, in the MP model, the thoroughly numerical settings of threshold and connection weights, and being able to adjust them at will, are too idealistic. Lastly, the perfect synchronization of MP neurons in the same layer is also unrealistic. Thus, these differences determine that MP model is unsuitable for modeling the inner neural mechanisms for animal behavior. To address these issues, Hodgkin and Huxley (1952)

proposed the Hodgkin–Huxley model. Not only did the model parameters have biological significance and scalability (Samura et al. 2015), but the model also provided a foundation from which to explore synaptic integration and interactions between ion currents. However, its computational efficiency was poor, and it could only simulate a few neurons in real-time (Izhikevich 2003). A variety of improved models based on the HH model were proposed in later years, such as the FHN model and the HR model (FitzHugh 1961; Hindmarsh and Rose 1984). A simple spiking neuron model that reduced the biophysically accurate but computationally complex HH model to a two-dimensional (2-D) system of ordinary differential equations was also proposed (Izhikevich 2003, 2004), and could simulate many real firing patterns of biological neurons (Zhao et al. 2016; Li et al. 2016) and performed with a high computational efficiency that made real-time simulations of large-scale functional neural networks possible.

Animal behavior arises from the coordinated activity of many interconnected neurons—"many" meaning 302 for *Caenorhabditis elegans*, 20,000 for a mollusk, several hundred thousand for an insect, and billions for humans (Bargmann and Marder 2013). To date, the anatomical wiring diagram of only a single animal (*C. elegans*) nervous system has been obtained. The nematode *C. elegans* provides an excellent model for studying chemotaxis behavior because its 302 neurons are clearly described and neuronal connections are known (White et al. 1986). Many studies have focused on the behaviors of *C. elegans*, parts of which constructed artificial neural networks to replicate some of these behaviors (Ferrée and Lockery 1998, 1999; Karbowski et al. 2008). However, ANN models are just a kind of abstract level and the approximate numerical calculation method, and cannot describe the true mechanism of the nervous system. Xu and Deng (2013) constructed chemotaxis behavioral models biologically extracted from a neural wire diagram rather than an artificial network. This raises the question of how we should approach building behavioral models of large networks without generating neural models. Recent studies indicate that the brain relies on a core set of canonical neural computations (Carandini and Heeger 2012). The nervous system combines and repeats these computations to apply different functions to different problems. Such as thresholding,linear filtering, exponentiation, recurrent amplification, cognitive spatial maps, gain changes resulting from input history, and cognitive demands. However, computations such as divisive normalization is less likely to map one-to-one onto a specific biophysical circuit (Carandini 2012).

To understand neural computations, we must know, in detail, a specific circuit that achieves a specific neural computation. Although it is almost impossible to accurately record spike firing data, information flow direction, logical relationships, and connection strengths of each neuron in a specific circuit, it is possible to design a plausible neural circuit for neural computation. We designed a circuit for canonical neural computation, and assembled complex biological neural networks through combining and repeating these canonical computations for specific behaviors, which helped us build a "bridge" from circuit to behavior.

Behavior neural circuits design

Phototactic behavior in insects

Nocturnal insects display primarily phototaxis behavior (Atkins 1980). Studies indicate that this is because they have the capacity to maintain the ***Angle*** (the angle between its flying direction and light) a fixed value. In Fig. 1a, the light from the moon belongs to approximately parallel light. Since insects can control the Angle to keep a certain angle, they can fly along a straight line, and achieve night navigation; In Fig. 1b, the light from a nearby source is emanative. Insects continue to maintain the ***Angle*** a fixed angle, therefore engaging in phototaxis behavior. As a result, this stable behavior certifies that there exists a neural decision circuit in the nervous system to keep the ***Angle*** a certain value.

Negative feedback system for phototaxis behavior

Since, phototaxis behavior is stable and repeatable, it is certainly a control system, and the behavior is modulated by control rules. From a control theory view, we can imagine a standard negative feedback control system, as shown in Fig. 2. Related terms are defined as follows: (1) The **Controller** issues a series of movement instructions to achieve the behavior decision. In this paper, it is a neural decision circuit, which receives the ***Angle*** and compares it to the fixed angle, and then outputs the command to correct flight direction. The form of instruction is the AFR of the neuron cluster, which can be used to encode the angular speed or line speed; (2) The **Actor** corresponds to the insects muscle that controls wing function, and receives instructions from the Controller to control flying; (3) The **Senor** corresponds to the insect perception system, which can sense the ***Angle*** as an input of the negative feedback system.

The negative feedback system for phototaxis behavior can be described as follows: first, the **Senor** senses the ***Angle*** as system input; then, the **Controller** compares the input with the fixed skew angle to obtain the relative deviation (*E_Angle* in Fig. 2) and outputs the correction instruction (different firing rate spike, *Hz_Spike*) of an angular or linear velocity, according to the deviation; and

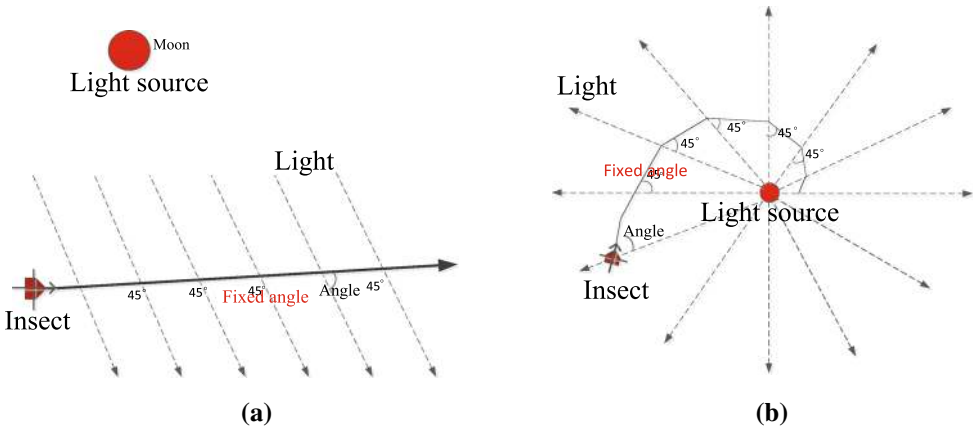


Fig. 1 Flying pattern with different light sources

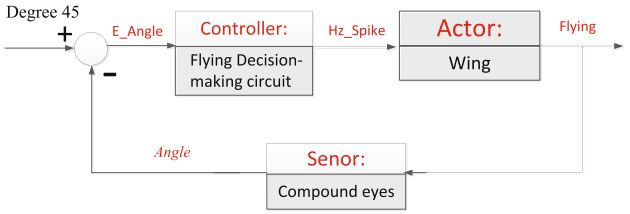


Fig. 2 Negative feedback control system for phototaxis behavior

lastly, the **Actor** executes the commands to maintain or change flying direction or speed. This iteration cycle occurs circularly, enabling phototaxis behavior. Our aim is to explore a type of neural circuit that achieves the function of the **Controller**, and investigate how the circuit implements control rules. Although we use the example of insect flying control, this study can be applied to other animal decision-making behaviors on the basis of the relationship between a neural circuit and its control rules.

Decision-making circuit design

Neural information coding based on neuron firing rate

The present generally accepted view is that spike sequence is the basic unit of biological neural information encoding. In the nervous system, different patterns of spike sequences encode different information (Wang 2007; Weliky et al. 2003; Hirata and Aihara 2009). The definition of neuronal firing rate refers to the number of pulses fired within a certain time window. In most of the neural perceptual system, pulse firing rate increases nonlinearly with increasing stimulus intensity (Kandel et al. 2000). Neural information encoding based on firing rate has been widely used in different types of perceptual research. The following content is also based on the AFR encoding mode, in which a population of neurons fires pulse sequences per second. The definition of firing rate used is as follows.

$$
\begin{aligned}
rate\_neui(t) &= \sum_{t=t_0}^{t=t_0+T} \delta(t) \\
AFR(t) &= \frac{1}{num} \sum_{i=1}^{i=num} rate\_neui(t) \\
delta(t) &= \begin{cases} 0 & \text{else} \\ 1 & \text{if firing a spike at moment t} \end{cases}
\end{aligned}
\tag{1}
$$

T refers to the time window (T = 0.2 s). Calculation step length is 1 ms. $\delta(t)$ is a pulse response function only if the moment t that a neuron is activated and fires an action potential, $\delta(t) = 1$; rate_{neui} represents the pulse firing rate of the i*th* neuron in a population. AFR is based on all individual neurons in a population within time window T.

How to select time window T? The above neural circuit and the following chapters repeatedly involve a circuit model as shown in Fig. 3a. The upstream neurons propagate spike sequence information to the output neurons. We used different output patterns of the circuit to encode different types of information. Therefore, we expected that the AFR of output neurons could easily distinguish with different input spike patterns from A. Thus, this model can serve as an example of how to select a specific T.

Here, we set T as 50, 100, 200, 500, and 1 s. From Fig. 3b, c, we can see that firing rate fluctuates more with a small time window, and firing rate is relatively stable with a larger time window. From Fig. 3d, we can see that with a smaller T, output neuron AFRs following different input patterns are more difficult to distinguish. Conversely, with a larger T, output neurons' AFR are easy to distinguish. However, a larger time window means that the calculations of AFR update more slowly in the computing process, which makes it difficult to control the flying state in real-time. A smaller time window means that firing rate updates quickly in the computing process, but the calculations of firing rate show more fluctuations, which could make the flight process unstable. Compromising between real-time

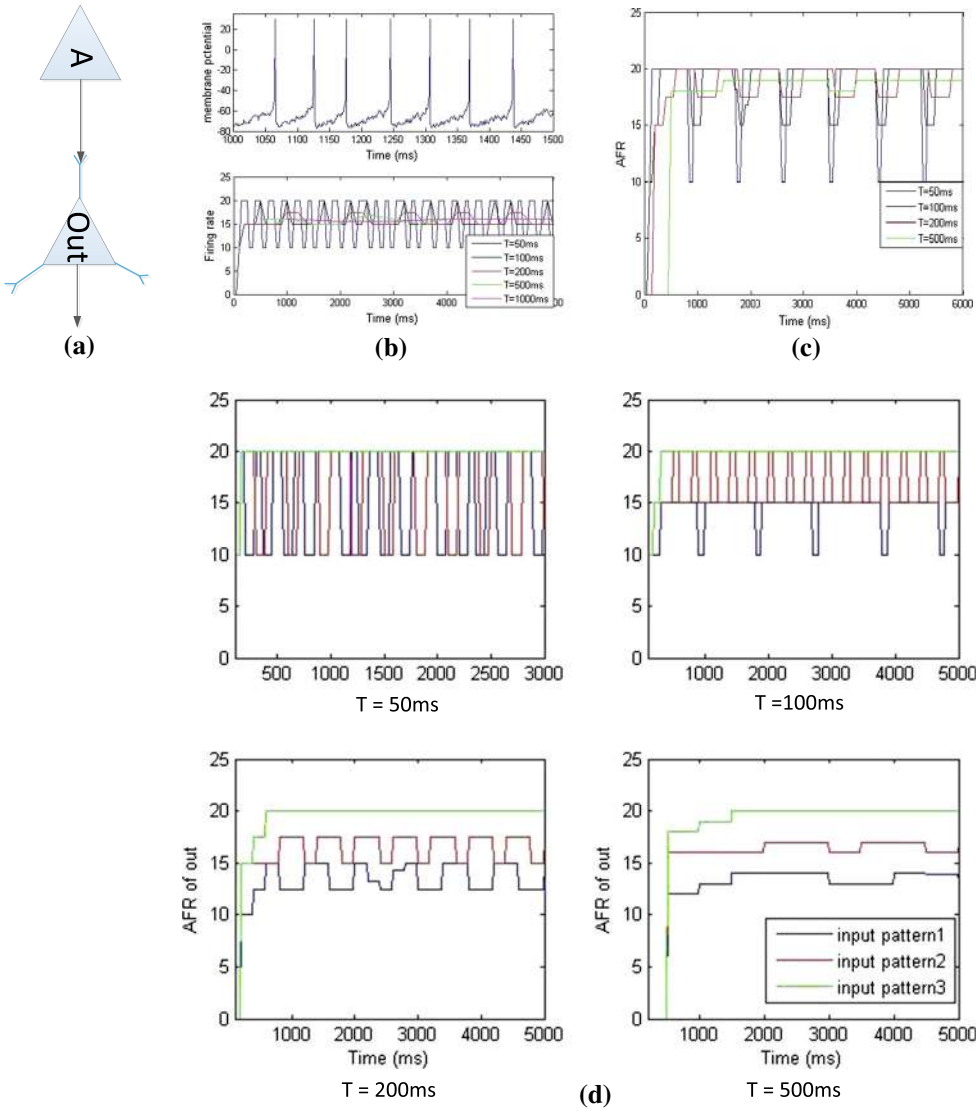


Fig. 3 Different values of time window T affect the calculations of AFR. **a** A neural circuit that shows the effect of different time windows on the AFR of neurons. **b** The *upper panel* shows the spike sequence of a single neuron. The *lower panel* shows pulse firing rate calculations of the spike sequences with different time windows. **c** AFRs of neurons with different time windows. **d** Different values of T in information encoding

and volatility, this paper selected T = 200 ms to calculate AFRs of neurons.

Expressions for behavioral decision logic

Before presenting our proposed neural circuit structure, we must discuss the decision logic of phototaxis behavior, wherein logic language is used to describe basic control rules. Logic language can describe basic control rules behind various behaviors, because the logic reflects the most basic requirement for a behavior to be successfully implemented. Since the biological nervous system is a product of evolution, its structure must be sufficiently complex to achieve basic logic. As shown in Fig. 4a, b, when the ***Angle*** is greater than a fixed angle, the insect adjusts its direction clockwise. Furthermore, the angular velocity increases as the ***Angle*** increases. On the other hand, Fig. 4c, d shows that when the ***Angle*** is less than a fixed angle, the insect adjusts its direction counterclockwise, and the angular velocity decreases as the ***Angle*** increases.

Here, we describe phototaxis behavior decision logic through predicate logic, as follows: Symbol m represents an insect; $Dir\_Diff(m, Fixed\_Angle)$ is a function to obtain the deviation value (E_Angle) of the ***Angle*** and the fixed skew angle (Fig. 4a, c), and we assume that E_Angle is divided into 4 gears, from small to large: $\Delta P1$ (less than 5°), $\Delta P2$ (from 5° to 10°), $\Delta P3$ (from 10° to 20°), and $\Delta P4$ (greater than 20°). If the insect perceives an ***Angle*** greater than a fixed value, the value of the predicate $Over\_Left(m)$ is TRUE; otherwise, it is FALSE. Conversely, if the insect perceives an ***Angle*** less than a fixed value, the value of the predicate $Over\_Right(m)$ is TRUE; otherwise, it is FALSE. $add\_VL(m)$, and $sub\_VL(m)$ indicate increasing and decreasing clockwise angular velocity, respectively. $add\_VR(m)$ and $sub\_VR(m)$ indicate increasing and

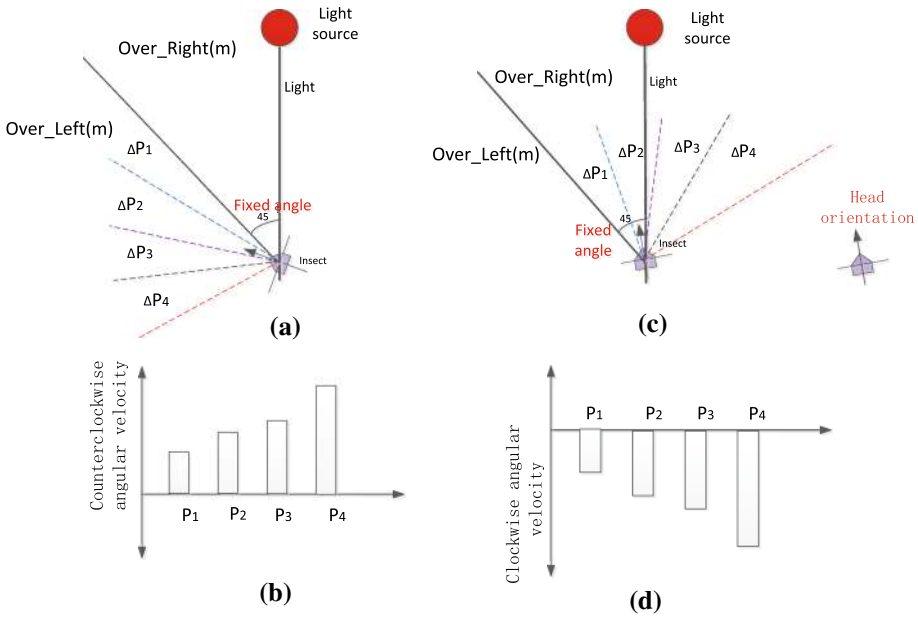


Fig. 4 Schematic diagram of flight decision

decreasing counterclockwise angular velocity, respectively. Based on logic and the simple form of symbols, phototaxis decision logic is described in Table 1.

When $Over\_Left(m) = TURE$ (Items 1, 2, 3, and 4 in Table 1), and when E_Angle is in the magnitude of $\Delta P1$ and $\Delta P2$ (Items 1, and 2), the insect's clockwise angular velocity decreases; and when E_Angle is in the magnitude of $\Delta P3$ and $\Delta P4$ (Items 3, and 4), the insect's clockwise angular velocity increases. Conversely, when $Over\_Right(m) = TURE$ (Items 5, 6, 7, and 8 in Table 1), and when E_Angle is in the magnitude of $\Delta P1$ and $\Delta P2$ (Items 5, and 6), the insect's counterclockwise angular velocity decreases; and when E_Angle is in the magnitude of $\Delta P3$ and $\Delta P4$ (Items 7, and 8), the insect's counterclockwise angular velocity increases. Such a set of predicate logic expressions simplify and clearly depict the control rules behind insect phototaxis behavior. Moreover, the predicate or function for this group of control rules is almost in its simplest; if the formal semantics were further simplified, normal functionality would not be possible. Therefore, we have sufficient reason to believe that the neural control system of insects is, at least, equivalent to this group of expressions.

Decision-making circuit

Any neural circuit that can achieve decision functions are composed of biological neurons, and therefore the structure unit and connection type should abide by neural biology and electrophysiological findings. This is the essential difference from ANN models. Studies have indicated that the central complex of insects may handle a great deal of sensory information (Pfeiffer et al. 2005; Ritzmann et al. 2008; Ridgel et al. 2007), and that this central complex is connected to thoracic motor neurons. Furthermore, the central complex participates in some advanced function, such as motion control, and polarized light navigation (Horváth and Varjú 2013; Heinze and Homberg 2007).

Table 1 Formal description for insect flight control logic

Type	Range	Corresponding logic expression for decision-making
Larger than fixed angle	$\Delta P1$	$Over\_Left(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P1) \rightarrow subVL(m)$ (1)
	$\Delta P2$	$Over\_Left(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P2) \rightarrow subVL(m)$ (2)
	$\Delta P3$	$Over\_Left(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P3) \rightarrow addVL(m)$ (3)
	$\Delta P4$	$Over\_Left(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P4) \rightarrow addVL(m)$ (4)
Less than fixed angle	$\Delta P1$	$Over\_Right(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P1) \rightarrow subVR(m)$ (5)
	$\Delta P2$	$Over\_Right(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P2) \rightarrow subVR(m)$ (6)
	$\Delta P3$	$Over\_Right(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P3) \rightarrow addVR(m)$ (7)
	$\Delta P4$	$Over\_Right(m) \wedge (Dir\_Diff(m, Fixed\_Angle) = \Delta P4) \rightarrow addVR(m)$ (8)

We designed a biological neural circuit composed of both excitatory and inhibitory neurons to achieve the aforementioned decision control rules. This is a plausible structure based on neuromechanism. As shown in Fig. 5, the circuit is divided into two areas (I, and II), in which the area I network is responsible for correcting clockwise angular velocity, while the area II network is responsible for correcting counterclockwise angular velocity. Both areas receive *E_Angle* information, we assume that when $E\_Angle > 0$, information was projected to area I; otherwise, information was projected to area II. Output neuron clusters (Pyramidal neurons) OutVA1 and OutVA2 encode the clockwise and counterclockwise turning angular velocity instructions, respectively, through changes in Spike firing rate. In area I for example, when the ***Angle*** is larger than the fixed skew angle ($E\_Angle > 0$, Fig. 4a), the AFR of OutVA1 encodes the clockwise flight angular velocity (Fig. 4c). Similarly, in area II, when the ***Angle*** is less than the fixed skew angle ($E\_Angle < 0$, Fig. 4b), the AFR of OutVA2 encodes the counterclockwise flight angular velocity (Fig. 4d).

To avoid misfiring, we designed a set of inhibitory neurons between neuron clusters OutVA1 and OutVA2, so that the two clusters could inhibit each other. When insects must fly counterclockwise, the neurons that issued the clockwise flight command should be inhibited, and vice versa. Conclusively, the two groups of output neuron clusters, OutVA1 and OutVA2, act as the ***Controller*** (Fig. 2), of the output of which acts on the ***Actor***, resulting in phototaxis behavior.

To further illustrate our proposed structure, area I, for example, is responsible for dealing with the situation in Fig. 3a. We assume that the *E_Angle* is divided into 4 adjustable gears ($\Delta P1$, $\Delta P2$, $\Delta P3$, and $\Delta P4$). Area I

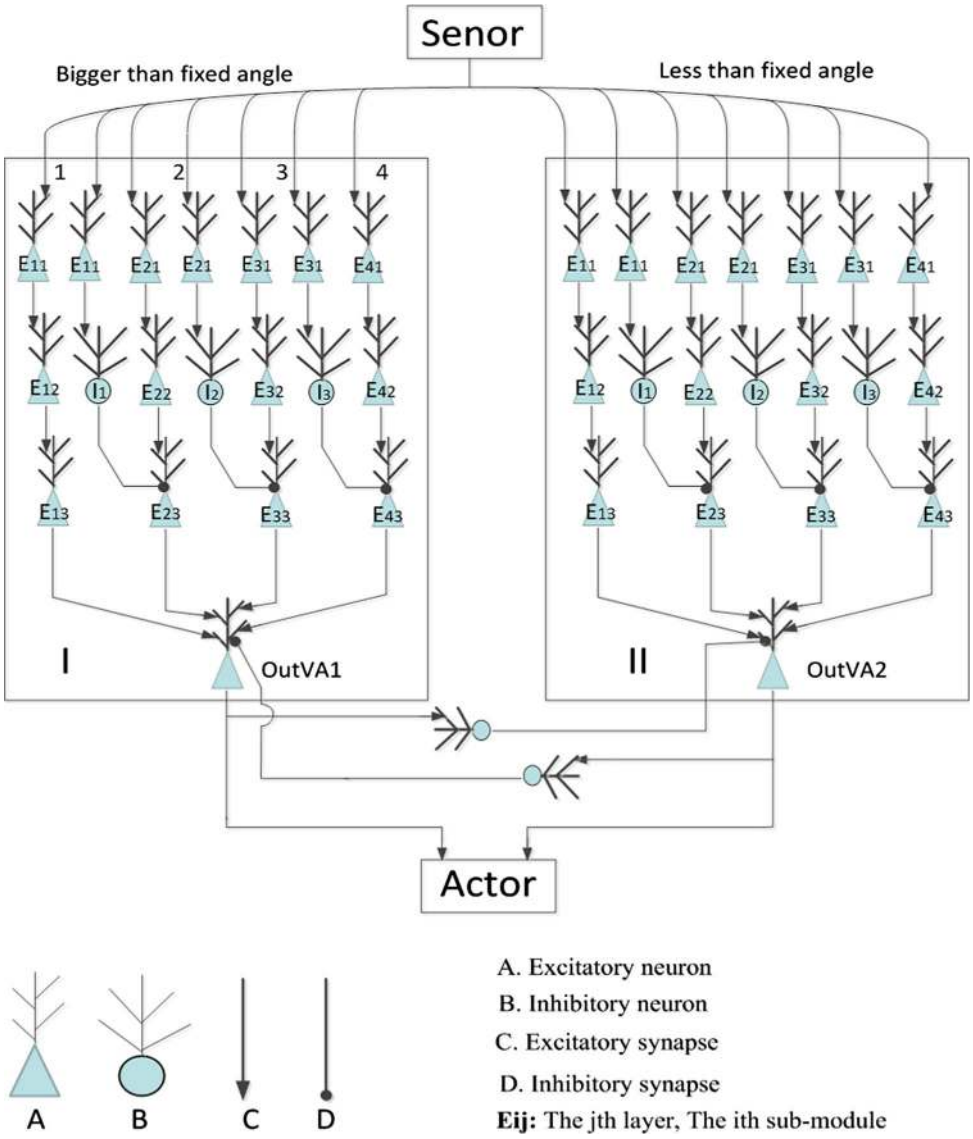


Fig. 5 A possible neural circuit for flight decision-making

contains four longitudinal submodules (1, 2, 3, and 4), which respectively corresponds to the treatment of four different levels ($\Delta P4$, $\Delta P3$, $\Delta P2$, $\Delta P1$) of input. Each longitudinal submodule consists of 3 layers of neurons to transfer the signal from input to output, and whether through relay channels or a signal significance discriminant channel, passes through the same number of neurons and achieve synchronous output. The second layer of neurons contains a number of inhibitory neurons, which are designed to inhibit other submodules. Taking the activity of neural circuits in area I as an example, the details of the signal propagating processes with different input magnitudes are described in Table 2.

Table 2 shows that when signal strength is in the $\Delta P1$ magnitude, it activates only E41 and its downstream neuron cluster. Conversely, when signal strength is large enough, such as a $\Delta P4$ magnitude, it activates the E11, E21, E31, E41 and their downstream neuron clusters. This means that if the case cannot be constrained, except for from E43, another three synchronous commands are issued, which are not expected. However, this reasoning is incorrect. Instructions activated by different magnitude input should be exclusive. That is, one type of magnitude input should activate only one type of instruction, and therefore avoid motor neuron disorder resulting from conflicting instructions common in high-level decision-making. Therefore, we introduce a lateral inhibitory mechanism to the network structure.

For experimentation, we designed the parameters of neuron models for different submodules (E11, E21, E31, and E41) such that they were activated upon different magnitude inputs. The AFR of submodules E11, E21, E31, and E41 were about 20, 17, 12, and 10 Hz, respectively. When input signal strength was in the $\Delta P4$ magnitude, E11, E21, E31, and E41 were all activated. Conversely, neurons E23, E33, and E43 were not activated due to inhibition from inhibitory neurons I1, I2, and I3. Therefore, only E11 propagated information to OutVA1, and thus, the AFR of OutVA1 was about 20 Hz. To illustrate another example, when input signal strength was in the $\Delta P3$ magnitude, neurons cluster E21, E31, and E41 were all activated. Due to inhibition from inhibitory neurons I2 and I3, only E21 propagated information to OutVA1, and therefore, the AFR of OutVA1 was about 17 Hz. Similarly, when the input signal was in the $\Delta P2$ magnitude, the AFR of OutVA1 was about 12 Hz, and when the input signal was in the $\Delta P1$ magnitude, the AFR of OutVA1 was about 10 Hz.

Neural circuit redundancy

In Fig. 5, all cells represent a population of neurons. This design has a certain redundancy for the following two reasons: Firstly, neurons can fire either an excitatory postsynaptic potential (EPSP) or an inhibitory postsynaptic potential (IPSP). As the postsynaptic potential produced by a single presynaptic neuron is very small and insufficient for reaching the threshold potential of the postsynaptic neuron, but the accumulation of EPSP can achieve threshold and activate the postsynaptic neuron. For example, if an AP generated an EPSP with amplitude of 0.2 mV, then at least 50 synchronous APs can produce more than 10 mV and reach threshold. Secondly, generally, neurons in a human brain over 26 years old begin to gradually decrease in number. To maintain normal working of neural circuits, redundant neurons are required to perform mutual backup.

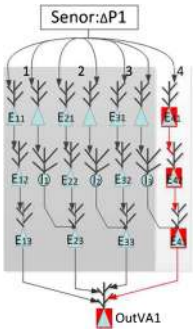
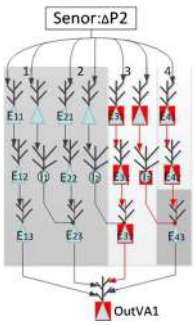
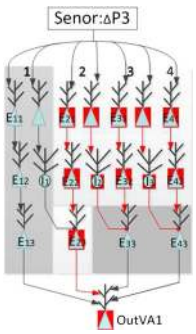
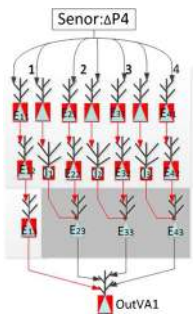
The above neural circuit and the following chapters repeatedly involve a model as shown in Fig. 6a, in which the upstream neurons propagate spike frequency information to the downstream neurons. A and B are shown as populations of neurons rather than two individual neurons because the circuit is designed to take into account a functional requirement and redundancy. For the functional requirement, the upstream neurons activate the downstream neurons. For redundancy, if some upstream neurons (A) cannot work properly, the circuit can still work well and the spike sequence shows little change. Therefore, neuron number plays an important role in a circuit, we must consider the effect of number of upstream neurons on the firing rate of downstream neurons. The AFR of A was 20–25 Hz, and the AFR of B varied with different number neurons in A as shown in Fig. 6b. In Fig. 6c, d, the AFR of A neurons was 15–20 and 10–15 Hz, respectively. Overall, when the number of neurons in A was 60–120, the AFR of B was close to that of A and relatively stable. Therefore, unless stated otherwise, a population contains 80 neurons. For example, in Fig. 5, all pyramidal cells represent a population of 80 excitatory neurons. With this setting, the downstream neurons, B, generate stable output. In addition, although some neurons (fewer than 1/4) in the circuit do not work, the B group can still function properly. Thus, the circuit has a certain level of redundancy.

Computational simulation on the neuron level

Neuron computational model

Izhikevich proposed a simple spike model that reduced the HH model to a 2-dimensional differential system (Izhikevich 2003). This model not only has the biological significance of the HH model, but also has high computational efficiency and easy mathematical analysis. This model is used in this paper and its equation is shown below:

Table 2 Neural activities in area I with different input

Input	Description
	The sensor system senses the deviation value (*E_Angle*), and through a number of pyramidal neurons, the different AFR spikes that encode the *E_Angle* information are projected to input layers. When the signal strength is in the $\Delta P1$ magnitude, neuron cluster AFR is lowest, and activate only E41. E41 propagates the excitatory signal to E42, which propagates the signal to E43, which activates output neuron cluster OutVA1. Finally, OutVA1 generates an angular velocity adjustment instruction corresponding to the $\Delta P1$ input
	When signal intensity is in the $\Delta P2$ magnitude, E31 and E41 are both activated. When E31 is activated, it propagates the excitatory signal to the downstream neuron cluster E32 and inhibitory neuron cluster I3. Then, E32 activates E33, which activates output neuron cluster OutVA1. Meanwhile, when inhibitory I3 is activated by E31, it propagates the inhibitory signal to E43, inhibiting its activation. As a result, although E41 propagates excitatory signals to E42, E43 does not activate. Inhibition from I3 to E43 ensures that OutVA1 only receives information from E33. Finally, OutVA1 generates an angular velocity command corresponding to the $\Delta P2$ input
	In the case of a $\Delta P3$ magnitude signal, E21, E31, and E41 are simultaneously activated. E21 propagates the excitatory signal to the downstream E22 and inhibitory neuron cluster I2; then, E22 activates E23, which activates output neuron cluster OutVA1. Meanwhile, inhibitory neuron cluster I2 propagates the inhibitory signal to E33, inhibiting its activation. As a result, although E31 propagates excitatory signals to E32, E33 does not activate. Similarly, E43 also does not activate. Inhibition from I3 and I2 ensures that E33 and E43 do not interfere with OutVA1
	When the $\Delta P3$ magnitude signal coming, E11, E21, E31 and E41 are simultaneously activated. E11 propagate the excitatory signal to the downstream neurons E12 and inhibitory neurons I1; then, neuron cluster E12 activates E13, which activates the output neuron cluster OutVA1. Meanwhile, inhibitory neuron cluster I1 propagated the inhibitory signal to E23, inhibiting its activation. Similarly, I2 and I3 inhibit E33 and E43, which achieve exclusive competitive output

$$\begin{aligned} v' &= 0.04v^2 + 5v + 140 - u + I \\ u' &= a(bv - u) \\ \text{if } \mathrm{v} \geq 30 \text{ mV, then} &\begin{cases} \mathrm{v} \leftarrow \mathrm{c} \\ \mathrm{u} \leftarrow \mathrm{u} + \mathrm{d} \end{cases} \end{aligned} \tag{2}$$

Here, v and u are dimensionless variables, and a, b, c, and d are dimensionless parameters. The variable v represents the membrane potential of the neuron, and u represents a membrane recovery variable. Synaptic currents or injected DC currents are delivered via variable I. With

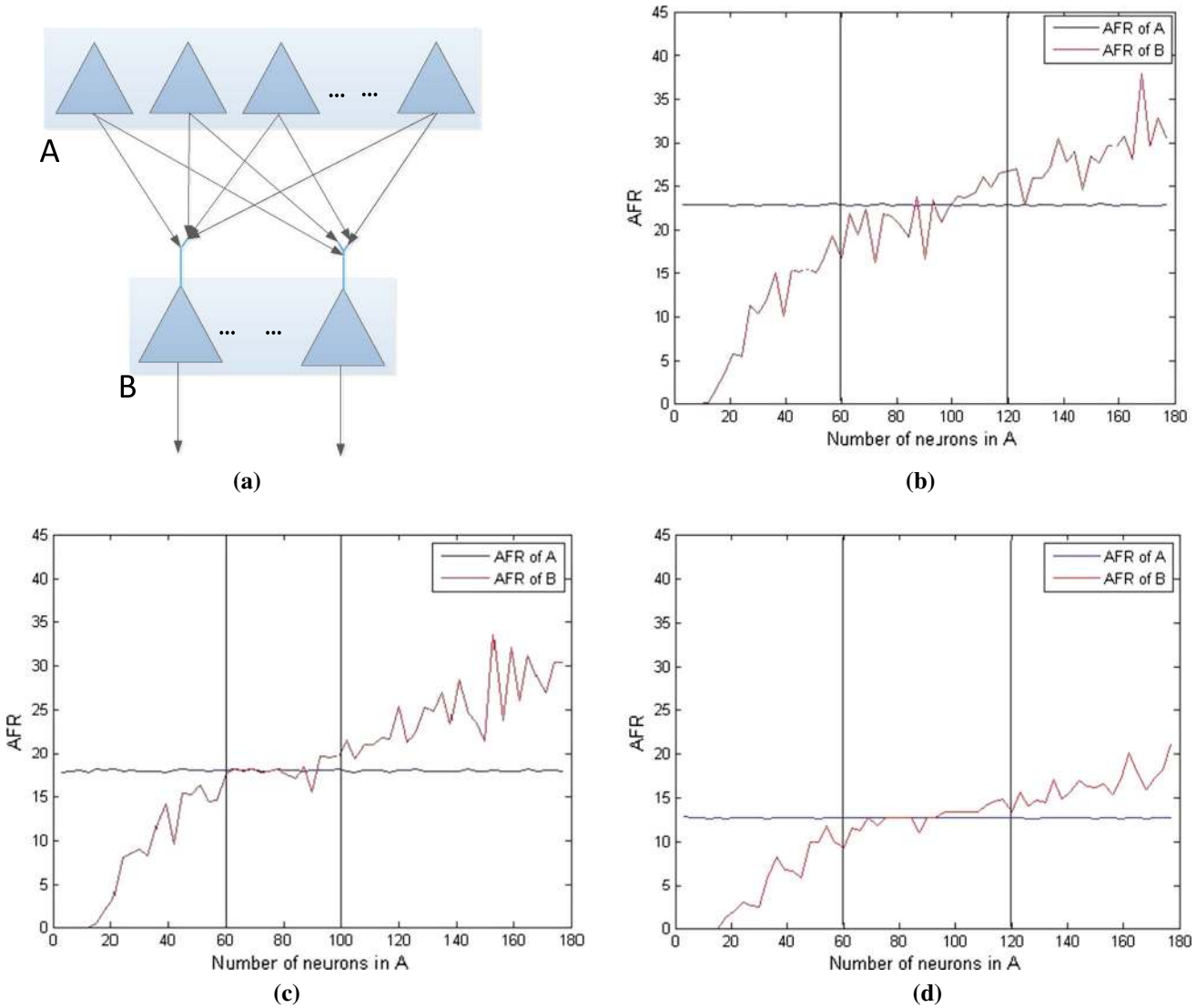


Fig. 6 The effect of the number of neurons in A on the AFR of B. **a** A neural circuit that shows the influence of number of upstream neurons on the AFR of downstream neurons. A is the upstream neuron cluster and B is the downstream neuron cluster, and the neurons in A are fully connected to the neurons in B. **b–d** AFR of B varies with the number of neurons in A

excitatory neurons, typical parameter values were: a = 0.02, b = 0.25, c = −65 and d = 8. The firing rate of pyramidal neuron AP was between 0 and 21 Hz. With inhibitory neurons, typical parameter values were: a = 0.1, b = 0.2, c = −47 to −50 and d = 2. The firing rate of intermediate neuron AP was between 0 and 200 Hz. Two typical firing modes (Li et al. 2016) are shown in Figs. 7 and 8. From Figs. 9a, c, 9b, d, we can see that neuronal membrane potential increases steadily with continuous stimulation, and membrane potential gradually returns to resting potential with no stimulation.

Time delays in AP transmission

Due to synapse distribution along axons, or different positions of dendrites receiving APs, different delays occur when an AP propagates from presynaptic to postsynaptic neurons (Tolnai et al. 2009). Haberly (1985) found that a wide range of time delays (up to 20 ms) could occur (Fig. 10). Since most previous studies did not relate to specific behavioral control logic or decision logic, this fact was easy to ignore. Thus, previous studies did not reflect real information processing and transfer in the nervous system. We must not ignore AP transmission has timing influence on behavioral control. Compounding on this, the AP's arrival at the postsynaptic neuron may be time-critical or time-sensitive, which should also not be ignored. In this paper, delays in AP transmission may be similar to "time multiplexing" in signal processing, and plays an important role in behavioral decision logic. In our study, we used different queue lengths to simulate different delays.

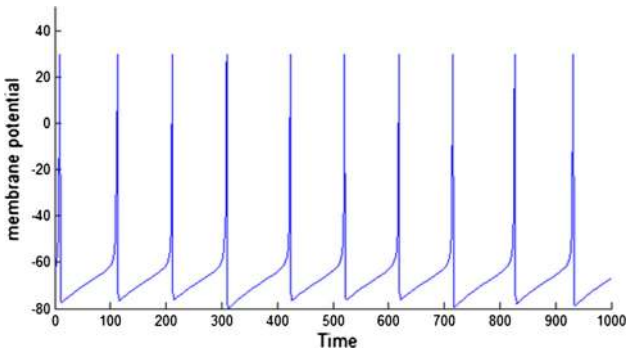


Fig. 7 Firing pattern of excitatory neurons

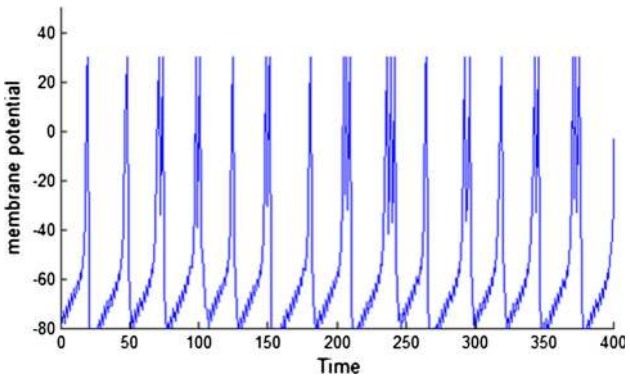


Fig. 8 Firing pattern of inhibitory neurons

Although the firing rate of excitatory neurons was lower, random firing suggests that there may always be a certain number of excitatory neurons in a cluster firing at a given moment. To inhibit excitatory neurons and avoid misfiring, inhibitory neuron clusters must generate a certain degree of inhibitory signals. The different propagating time delays during AP propagation (see the following sections) may play an important role in asynchronous firing of neuron clusters.

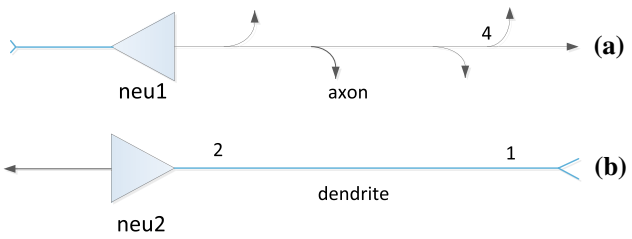


Fig. 10 Delays in presynaptic and postsynaptic neurons. Delays in presynaptic neurons are shown by points 3 and 4 in **a**. Milliseconds of difference occur when the neu1 AP propagated from the cell body to positions 3 and 4. Delays in postsynaptic neurons are shown by points 1 and 2 in **b**. Postsynaptic APs from position 1 propagating to the neu2 cell body require more time than those from position 2

Adjustable firing rate of pyramidal neurons

Complex information processing involves cooperative activities that are performed under different oscillation rates for different behavior states. Researches indicate that intermediate neurons take part in regulating the firing rate of a neural network (Pi et al. 2013). Intermediate neurons either affect firing rate of pyramidal neurons directly (as show in Fig. 11a) or indirectly by inhibiting other intermediate neurons (as show in Fig. 11b). Perhaps other neural networks can also achieve this firing rate

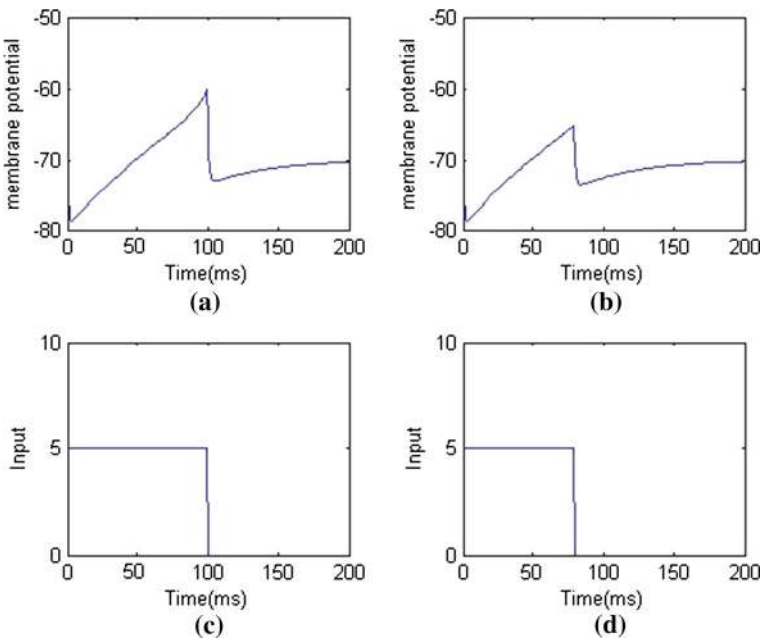


Fig. 9 Membrane potential gradually returns to resting potential with no stimulation. **a** Membrane potential changes with the input pattern shown in **c**. **b** Membrane potential changes with the input pattern shown in **d**

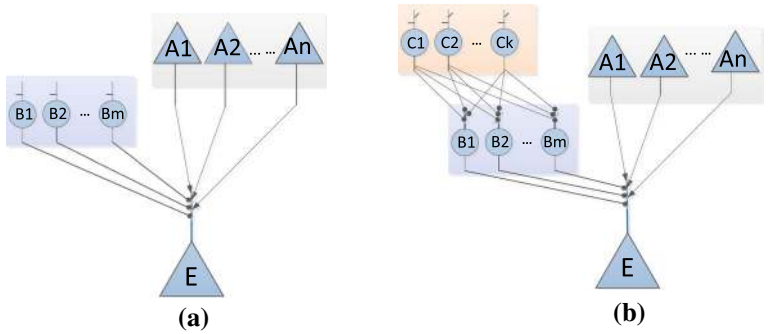


Fig. 11 Intermediate neurons regulating the firing rate of pyramidal neurons. In **a**, neuron cluster A propagated different frequency spike train to pyramidal neuron E, which was directly regulated by inhibitory neuron cluster B; In **b**, neuron cluster C, made up of inhibitory neurons, indirectly regulated the firing rate of E by inhibiting neuron cluster B

regulation. This cooperation between excitatory and inhibitory neurons regulating the firing rate of downstream neurons allows the nervous system to perform its most basic functions.

In Fig. 11a, if neuron cluster A projected APs with a stable firing rate to pyramidal neuron E, cluster B joining the network can regulate E's firing rate directly. As shown in Fig. 12, the firing rate of E varies corresponding to the firing rate of B. In Fig. 11b, cluster C modulates B's firing rate. Changing C's firing rate subsequently affected B's firing rate, which regulates the firing rate of E indirectly, as shown in Fig. 13. This basic law reveals that nervous systems achieve output demand through the precise configuration of types of neurons and connections, and the overall firing pattern of the neural network.

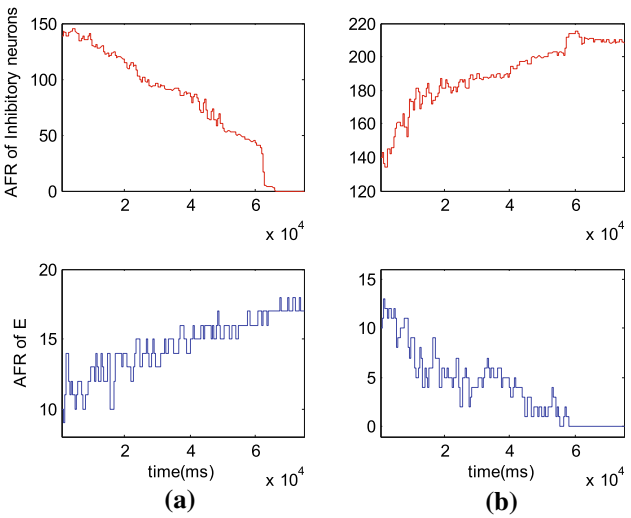


Fig. 12 Intermediate neurons of cluster B regulating the firing rate of pyramidal neuron E. **a** Firing rate of neuron E increases as firing rate of B decreases; **b** Firing rate of neuron E decreases as firing rate of B increases

Distributed simulation

Hardware platform

A local area network of 25 PCs with COREi5 processors and Windows 7 operating system was constructed using a fast Ethernet switch (Fig. 14). This network was used for real-time simulation of information processing in biological neural circuits. We used Visual Studio 2010 as the IDE and C# as the programming language. Each PC simulated 1 neuron cluster; each pyramidal neuron cluster had 80 excitatory neurons, while each interneuron cluster contained 160 inhibitory neurons. The PC (neuron cluster) communicates (propagating AP) with each other through UDP protocol. Our platform performs real-time

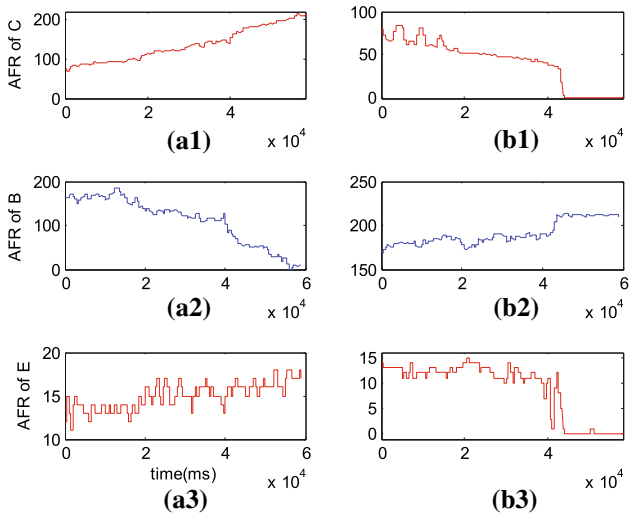


Fig. 13 Neuron cluster C indirectly regulating E's firing rate by inhibiting B. As C's firing rate increases (in **a1**), the intensity of inhibition to B increases, leading to a decrease in B's firing rate (in **a2**) and therefore decreasing inhibition to E decreasing. Finally, this results in an increase in E's firing rate (in **a3**). Conversely, when C's firing rate is decreases, inhibition to B also decreases, leading to increased inhibition of E, and therefore a decrease in E's firing rate (**b1**–**b3**, respectively)

simulations of both AP integration and propagation in a neural decision-making circuit. However, to make the simulation as identical as possible to the real biological neural network, we must solve the following problems.

Why do we use a distributed PC array?

Although there are some excellent neural network simulation platforms, such as NEURON, GENESIS, and Brain, they are not without their limitations. Firstly, simulation results of these softwares are primarily graphically displayed, which is not conducive to data mining and numerical analysis for large-scale networks. Secondly, they do not support distributed parallel simulation. Therefore, when the number of neurons is large, it is inevitable that each CPU nucleus simulates the operation of a large number of neurons, which makes ensuring neuron independence difficult. Thus, it is necessary to design a distributed parallel simulation platform for the large-scale neural network. Such a distributed parallel simulation platform is also a cheap and efficient simulation platform compared to a high performance computer.

The simulated neural circuit contains over 4000 physically independent neurons, and the circuit contains approximately 320 thousand synaptic connections. Taking excitatory neurons with a low firing rate (such as 10 Hz) as an example, the computer must access all synapses 3 million 200 thousand times per second. If the network scale expands, its computational complexity exponentially increases and it is unable to perform real-time simulations so fine-grained parallel computing by little number of CPU.

It is difficult for an ordinary PC machine to simulate information integration and propagation processes at the millisecond level, which is composed of more than 100 neurons. Here, we used a distributed PC array to construct a LAN simulation platform, which not only ensures independence between neurons, but also satisfies the real-time requirement through multi-PCs bearing the large-scale network computation. This simulation platform is convenient for expanding to simulate larger neural network circuits. For example, when the scale of a neural network increases, simply adding PC nodes to the PC array meets that scales demand, and the requirements of single node PC performance is not high.

Fig. 14 Distributed simulation hardware platform

Computational synchronization in PC array

In this paper, the computer simulates the calculation that neurons complete can be described as: When the cell bodies or dendrites receive an AP from upstream neurons, the membrane potential raises. When the membrane potential of this neuron reaches a certain threshold, a new AP is generated and rapid propagates through the axon and synapse to downstream neurons. If neurons do not receive any AP, the membrane potential gradually returns to resting potential. We assume that the neural circuit contains a total of n independent neurons; the process of one calculation of m neurons in one PC is described in Table 3.

In the practice, due to differences in computers hardware configuration, software systems, and operational environments, the same simulation in different PCs require a different amount of time. The maximum time differences (in Millisecond-level) may vary a hundred times. This difference greatly impacts computing the AFR of neuron clusters in different PCs. For example, two neuron clusters with a large number of synaptic connections are simulated in two PCs. PC1 runs computations 1000 times within one second, but PC2 runs computations only 500 times. The neuron cluster simulated in PC2 receives APs from the neuron cluster simulated in PC1, but because PC2 runs faster than PC1, input of neurons in PC2 may be incorrectly expanded.

To solve this problem, we implement a control thread to send the computational command to the PCs simulating a neuron cluster at a random time interval. Each neuron cluster has a computation command queue; when the command queue is not empty, the neuron cluster begins calculations. Each command can be used only one time. If the queue is empty, then the neuron cluster waits for the computational command. This ensures that the simulation calculation interval is roughly equal across all PCs.

Simulating AP transmission delays

In this paper, we simulated AP propagation delays using different queue lengths. For example, four different queue lengths, as shown in Fig. 15b, Queue 1 4, simulated the different delays of an AP propagating from the cell body to positions 1 4 in Fig. 15a. If the length of a queue is k, then the AP is delayed k ms. Four queues with sequential

Table 3 One calculation process of m neurons in one PC

Define:

1. N-dimensional matrix ***Adjacency*** is the adjacency matrix of neurons. If Adjacency[i, j] == 1, a synaptic connection exists between neuron i and neuron j; otherwise, there is no connection.

2. Neuron[i] represents the ith neuron, Membrane_v[i] represents the current membrane potential of the ith neuron, and Iap[i]represents the total number of APs that Neuron[i] receives from upstream neurons at this particular moment.

Begin:

```
Parallel For i = 1,2,...,m // Each PC uses multiple threads to simulate m neurons.
    Update(Iap[i]); // Receiving the AP propagating to the Neuron[i]
    //Computing the current membrane potential using the computational model
    Membrane_v[i] = NeuModel(Iap[i]);
    If Membrane_v[i] ≥ 30, Then // Neuron[i] generates AP
    // If a connection exists between Neuron[i] and Neuron[j], then send the AP to
the PC, i.e., where Neuron[j] was.
        For j = 1,2,3,...,n
            If Adjacency[i, j] == 1, Then: Udpsend(j);
        End for
    End if
End Parallel for
```

increases in length indicated that as synapse location moved away from the cell body, delays increased. If an AP was generated in the presynaptic neuron, we added 1 to the head of the queue; otherwise, we added 0. When the end of a queue element was 1, it indicated that the postsynaptic neuron received an AP. Since, delays of each neuron were limited; Fig. 15c presents a possible means of dealing with greater delay requirements.

Taking the inhibitory neuron cluster I1 of area I as an example, assume that there are n different delays from E11 to I1, and we use n different queue lengths to simulate these delays. Each delay from E11 to I1 involves roughly same number of neurons. Figure 16 shows that the value of n affects the asynchronism of I1's activity and greatly influences the stability of the output neuron cluster's AFR (Fig. 17). The firing moment of neuron cluster I1 when the input of the first layer neuron cluster was in the $\Delta P1$ magnitude is shown in Fig. 16a, b. With only four different delays (2, 4, 6, and 8 ms) from E11 to I1, the firing moment of I1 has a certain synchronization and the number of firing neurons is small and even no at some moment (Fig. 16a). As a result, I1 fails to inhibit downstream neurons at some moment. With 8 different delays (2, 4, 6, 8, 10, 12, 14, and 16 ms) from E11 to I1, I1 firing was distributed in most computing moment. Therefore, I1 completely inhibits downstream neurons.

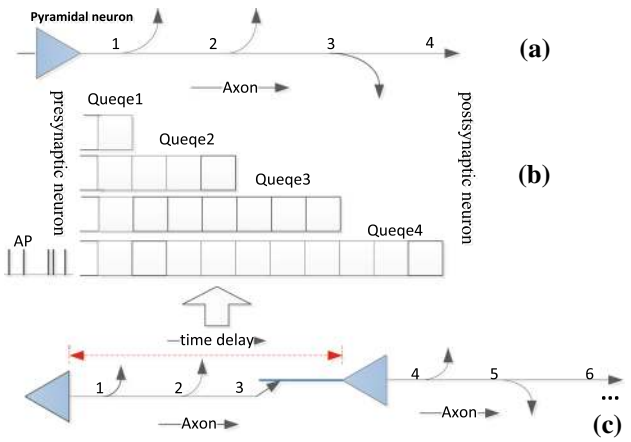


Fig. 15 Simulating delays in AP transmission along an axon using queues

As n increased, the firing moment of inhibitory neuron cluster I1 became gradually distributed across each computing moment (i.e., asynchronous, as shown in Fig. 16b). As shown in Fig. 17a–d, the inhibitory effect of I1 on pyramidal neurons gradually strengthened, resulting in a stable AFR of OutVA1. For example, when the input to the first layer neuron cluster was in the $\Delta P4$ magnitude, inhibitory neuron clusters I1, I2, and I3 inhibited neuron clusters E23, E33, and E43 so that only the AP from neuron cluster E13 was propagated to neuron cluster OutVA1. When n was small, it is possible that, at some moment, the number of firing inhibitory neurons was too small to completely inhibit the activities of E23, E33, or E43, and therefore, neuron cluster OutVA1 received APs from multiple neuron clusters. This resulted in an unstable AFR, which resulted in sudden variations in angular velocity during flying. Consequently, flight was also unstable.

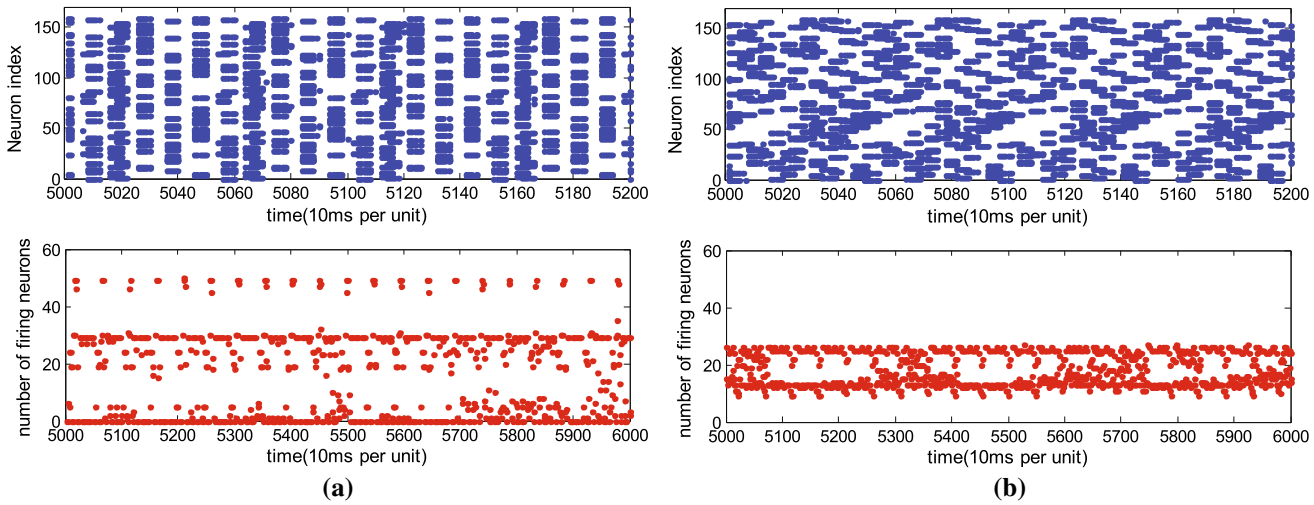


Fig. 16 Firing moment distribution of I1. **a** n = 4; **b** n = 8. In the **a**, **b** *up*-subgraphs, the longitudinal coordinate is the neuron index, which represents the firing neuron index at each computing moment; In **a**, **b** *down* subgraphs, the longitudinal coordinate indicates the number of firing neurons at each computing moment

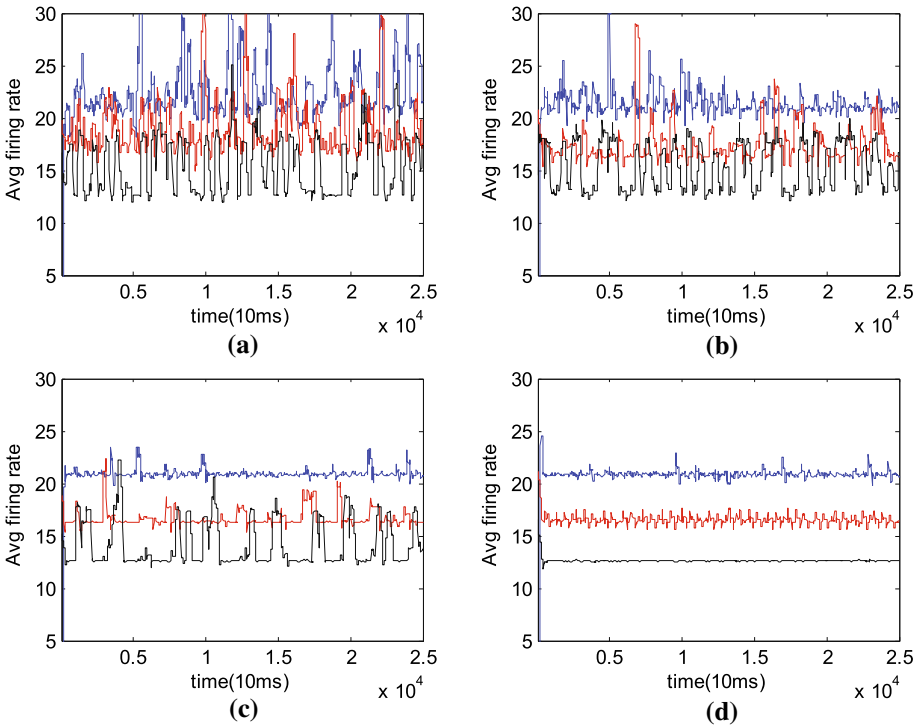


Fig. 17 AFRs of output neuron cluster OutVA1. The blue curve represents the AFR when the input of the first layer neuron clusters in area I was in the $\Delta P4$ magnitude. *Red* and *black curves* represent the AFRs resulting from $\Delta P3$ and $\Delta P2$ magnitude inputs, respectively. The *x-axis* represents the moment of calculation, while the *y-axis* represents the AFR. The AFRs when $n = 2$, 4, 6, and 8, are shown in **a**–**d**, respectively. (Color figure online)

The lager the value of n, the more dispersed the synapses on the dendrite. In this way, the moment in which inhibitory neurons received the AP was more decentralized, and the activities of inhibitory neurons were more asynchronous. This improved inhibitory effects, and thus resulted in a more stable AFR of the output neuron cluster. Conversely, the smaller the value of n, the more concentrated the synapses on the dendrite, and the more synchronous neuron cluster activities. This reduced inhibitory effects, and thus resulted in a less stable AFR of the output cluster. In this paper, set the n range from 8 to 12 for inhibitory neurons and range from 1 to 2 for excitatory neurons. One typical setting for queue length was 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24,..., in pyramidal neuron cluster for downstream inhibitory neurons and 2, 3 for downstream excitatory neurons.

Real-time simulation and analysis

Environment simulation

As shown in Fig. 18, the red point represents a light source. The Senor, as the insect's perception system, senses the

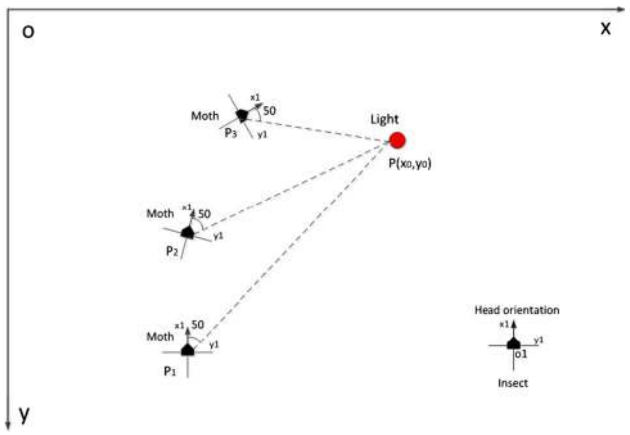


Fig. 18 Simulation environment

E_Angle. Here, the simulation calculation was used to replace the Senor to get *E_Angle*. To keep the ***Angle*** at a certain value (50°) and modulate flying direction, the decision-making circuit output the different AFRs encoding different *E_Angle*. In this way, we simulated flying under both near and far light sources conditions. During experimentation, we added random noise to the perception process, and set error decision probability between 0.1 and 0.3.

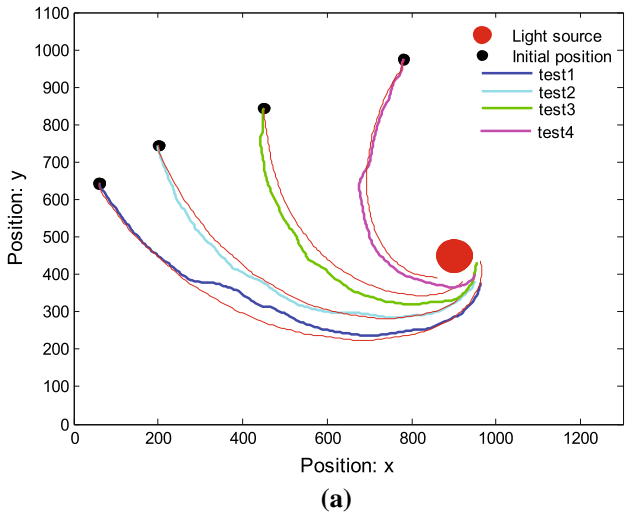
Simulation in near light source and far light source environments

Four flight paths with 4 groups of different initial position and direction were simulated under a near light source environment. As shown in Fig. 19a, each group also contained an additional experiment including gauss noise interfere. Simulation showed that insects can still perform a spiral progressive to light under noise interference. On the other hand, Fig. 19b shows flight paths simulated under a far light source. In this case, the insect should fly in a straight line. We assumed the angle between the parallel light and the horizontal was 60°, and the insect's neural circuit maintained the ***Angle*** at about 50°. The AFR and instantaneous firing rate (IFR) of OutVA1 and OutVA2 in the decision-making circuit, clockwise and counterclockwise angular velocity, and the Angle of test1 and test4 in Fig. 19a are shown in Figs. 20 and 21 respectively, that of test1 in Fig. 19b were shown in Fig. 22.

These simulation experiments illustrate that output neuron clusters OutVA1 and OutVA2 in the decision-making circuit use the AFR to encode different angular velocities. When the ***Angle*** was greater than 50°, OutVA1 was activated; and the larger the ***Angle***, the larger the AFR is, and the greater the clockwise angular velocity. On the contrary, the closer the ***Angle*** is to 50°, the smaller the AFR is. When the ***Angle*** was less than 50°, OutVA2 was activated and OutVA1 was inhibited. The smaller the ***Angle***, the larger the OutVA2's AFR, and the larger the counterclockwise angular velocity. In our simulations, although we added interfere in each link of the negative feedback control system, the circuit still made correct decisions in most moments, and exhibited stable behavior.

Why can the brain always learn a proper circuit for a decision-making task?

Humans and other animals can perform stable behaviors through learning or training. The behavior arises from the coordinated activities of interconnected neurons in the



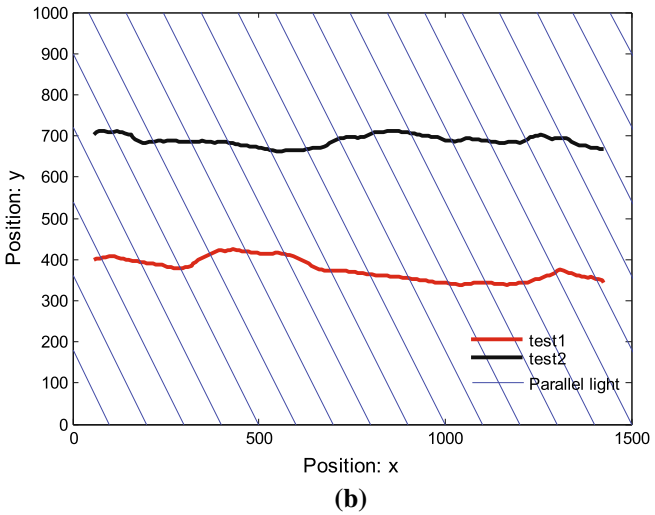
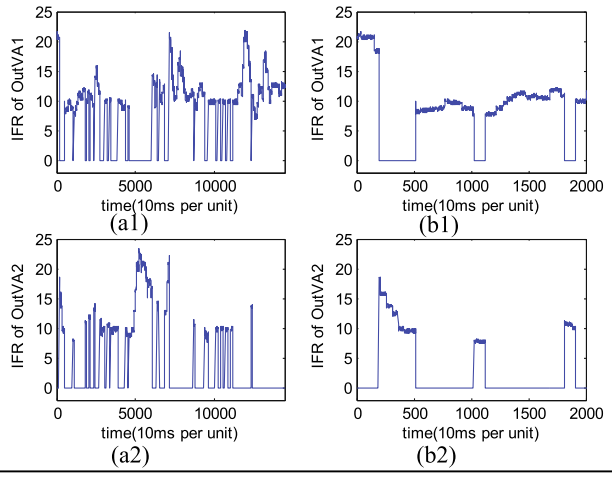
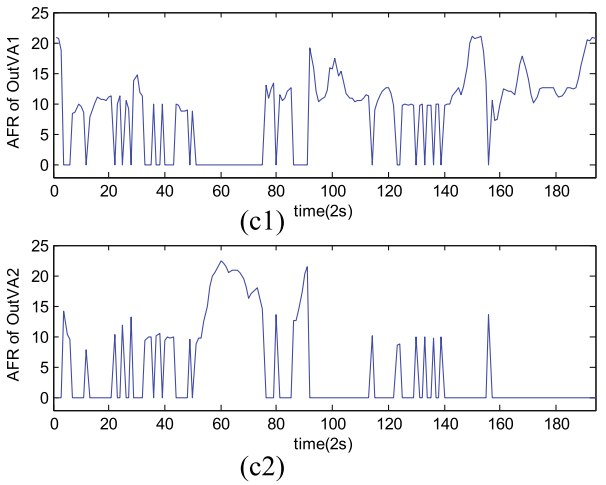


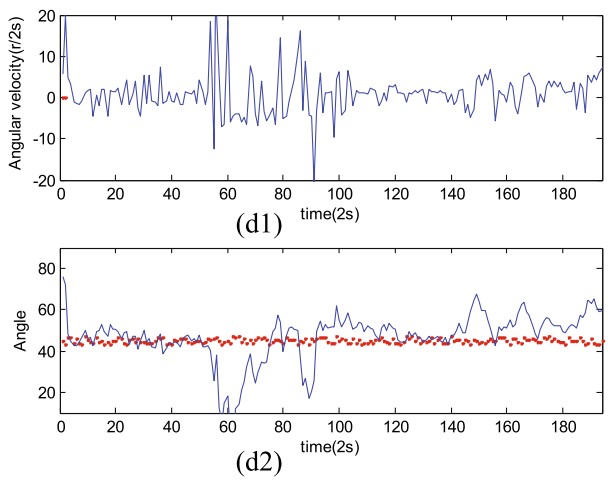
Fig. 19 Flight trajectories under near light source and far light source environments. **a** 4 groups of flight paths with different initial positions and direction under a near light source. *Red lines* indicate results under ideal conditions; *colored thick lines* indicate results under noise interference. **b** 2 groups of flight paths with different initial positions and direction under a parallel light environment (far light source). (Color figure online)



The initial ***Angle*** was 75.8, larger than 50 degrees, and therefore, E_Angle belonged to the $\Delta P4$ magnitude input. This input activated neuron cluster OutVA1 with about 20 Hz AFR, while OutVA2, which controls flight speed in the opposite direction, was not activated. (a) The IFR of OutVA1 and OutVA2 during the flight process. (b) IFR of the first 2000 computing moments, illustrating mutual inhibition of OutVA1 and OutVA2.



Differencs in OutVA1 and OutVA2 AFR during phototaxis simulation. The activities of OutVA1 and OutVA2 are mutually exclusive.



In d1, a positive value indicates clockwise angular velocity, while a negative value indicates counterclockwise angular velocity. The d2 shows the ***Angle*** changing during flying. With clockwise flying, the ***Angle*** gradually decreased, along with the OutVA1's AFR decreasing. When the ***Angle*** was less than 50 degrees, OutVA2 was activated, which inhibited OutVA1, and thus, triggered counterclockwise flying to keep the ***Angle*** near 50 degrees.

Fig. 20 Test1's result in Fig 19a

nervous system, and determining the connectivity of these neurons has always been a part of neuroscience (Bargmann and Marder 2013). To describe the circuit clearly, the network flow model (Ford and Fulkerson 1956) is used here. We take neuron cluster as nodes and take the connections between the nodes as directional pathway. Then, the neural circuit can be analogy of a directed graph, where spikes propagation in the graph was seen as the network

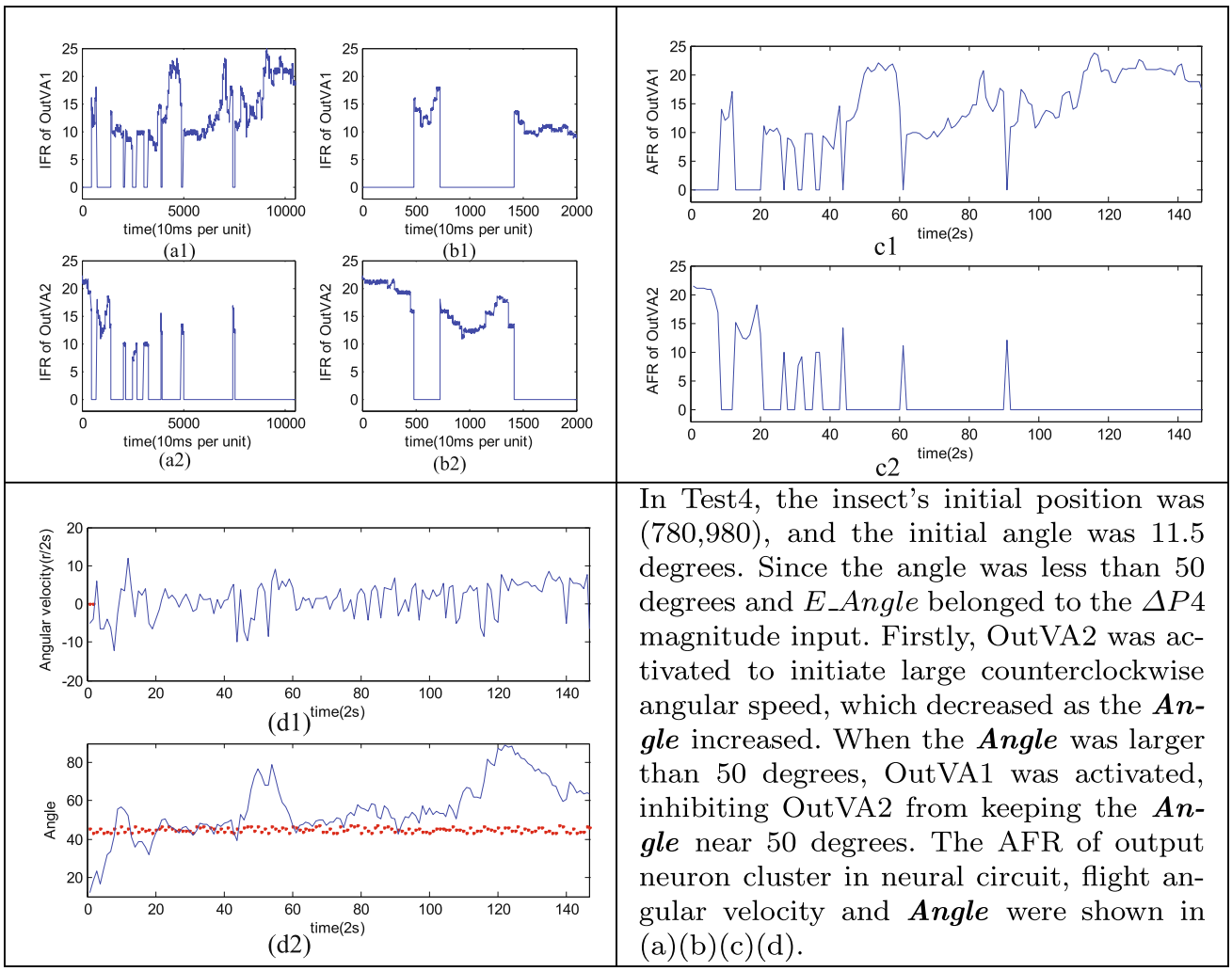


Fig. 21 Test 4's result in Fig 19a

flow model. To form the decision circuit, we must find the correct path from the input node to the expected output node in a directed graph. Assuming that signal flow starts from multiple points, connections of a circuit is mostly incorrect at the beginning, which results in signal flow sometimes arriving at the wrong end. What is the learning mechanism drives the nervous system to form a neural circuit containing the correct control rules? Research indicates that neuromodulation reconfigures circuit properties, and change neuronal functions over seconds, minutes, or even hours (Marder 2012). Under some modulatory conditions, synaptic connections may be functionally silent, only to be strengthened under other modulatory conditions (Marder 2012). The neuromodulation released by dopaminergic neurons in the basal ganglia, which represents the reward or punishment signal, could be responsible for correcting this signal flow path.

To address this question, we apply the reinforcement learning mechanism of Skinner's theory of operant conditioning to interpret this learning mechanism. Initially, as the circuit is running, doing well will yield reward, while doing otherwise results in punishment. Feedback signal from the reward circuit results in two conflicting adjustment trends. First, due to the different receptors and internal responses of the target cell, the arrival of dopamine produces an excitatory effect on some cells, but inhibitory effects on others (Neve 2009; Horvitz 2002). This strengthens the correct firing pattern and suppresses the wrong pattern. Second, "reinforce" can be interpreted as "the prior firing circuit should be enhanced", and "inhibition" can be interpreted as "the prior firing circuit should be weakened", which, together, can be translated into a variety of specific synaptic connections and pulse firing levels. These conflicts present the question of which specific signal transmission paths should be strengthened or weakened.

The basic idea of path modification based on the reward mechanism is as follows. A punishment signal indicates

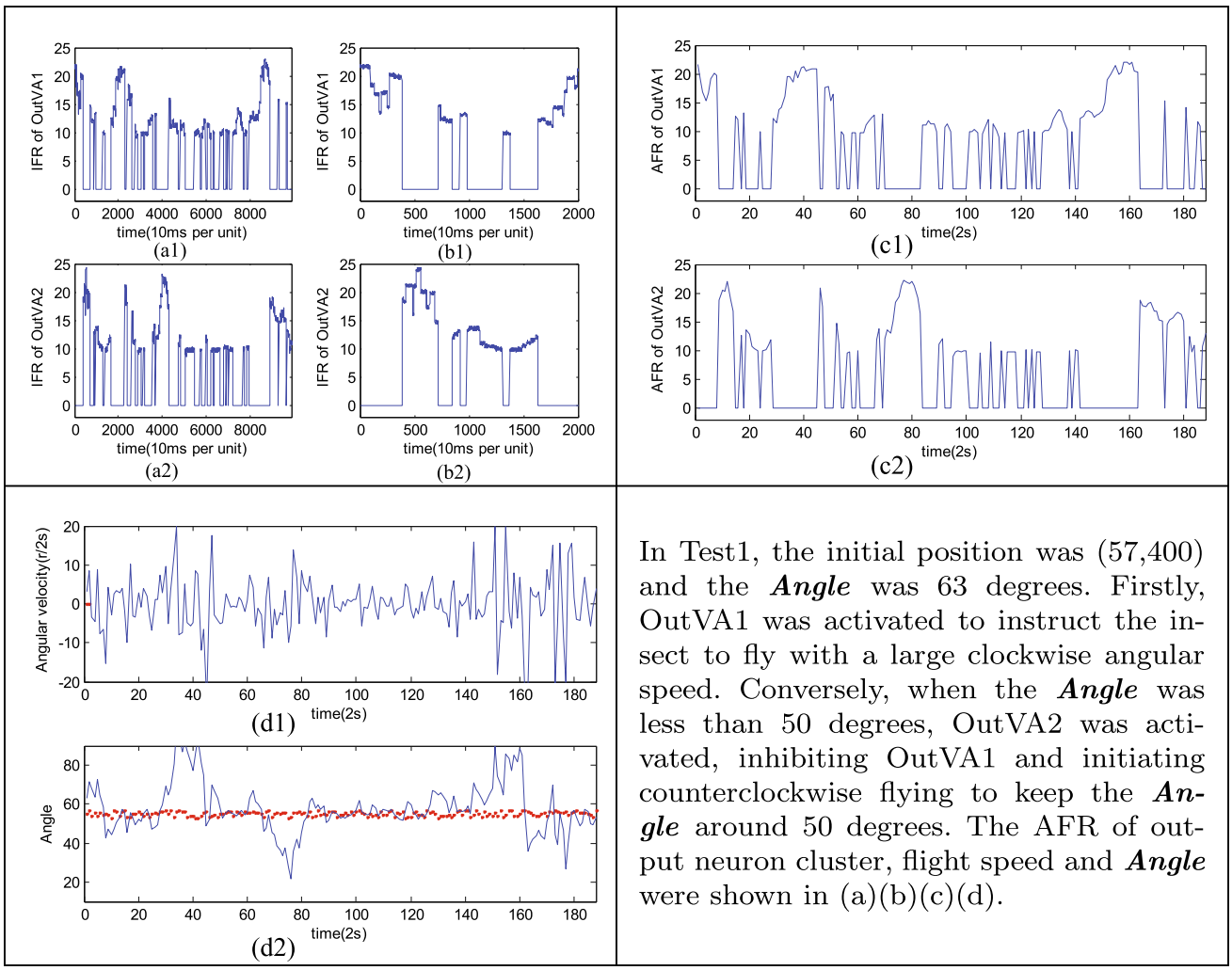


Fig. 22 Results of Test1 from Fig. 19b

arrival at an incorrect end, which increases the threshold or weaken the synaptic connection so that the node can just barely be activated. Meanwhile, this blocking effect is propagated back along the signal path in accordance with Hebb's rule. Conversely, a reward signal indicates arrival at the expected output, which decreases the threshold or strengthen the synaptic connection so that the node more easily activated. Similarly, this effect is propagated back along the signal path in accordance with Hebb's rule. Furthermore, hyperpolarizing neurons (increasing the threshold) and depolarizing neurons (decreasing the threshold) are opposite processes that can be easily achieved. For example, neurotransmitter release near the synapse can make postsynaptic neurons either hyperpolarized or depolarized. This can be the result of dopaminergic neuron activity. Circuit regulation by reinforcement learning is achieved at the level of neural activity. To model this, we changed the downstream signal processing path by activating or inhibiting circuit nodes.

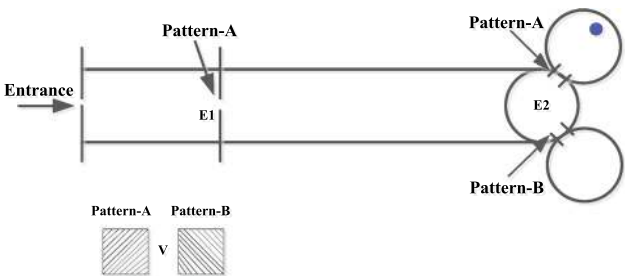


Fig. 23 Environmental bee experiment [corresponding to Zhang et al. (2005)]

Demonstrating feasibility of the above circuit modification process using a bee behavioral experiment

Description of one of the honey bee decision-making behavioral experiments (as shown in Fig. 23): In a long tunnel, bees encounter visual pattern A at position E1.

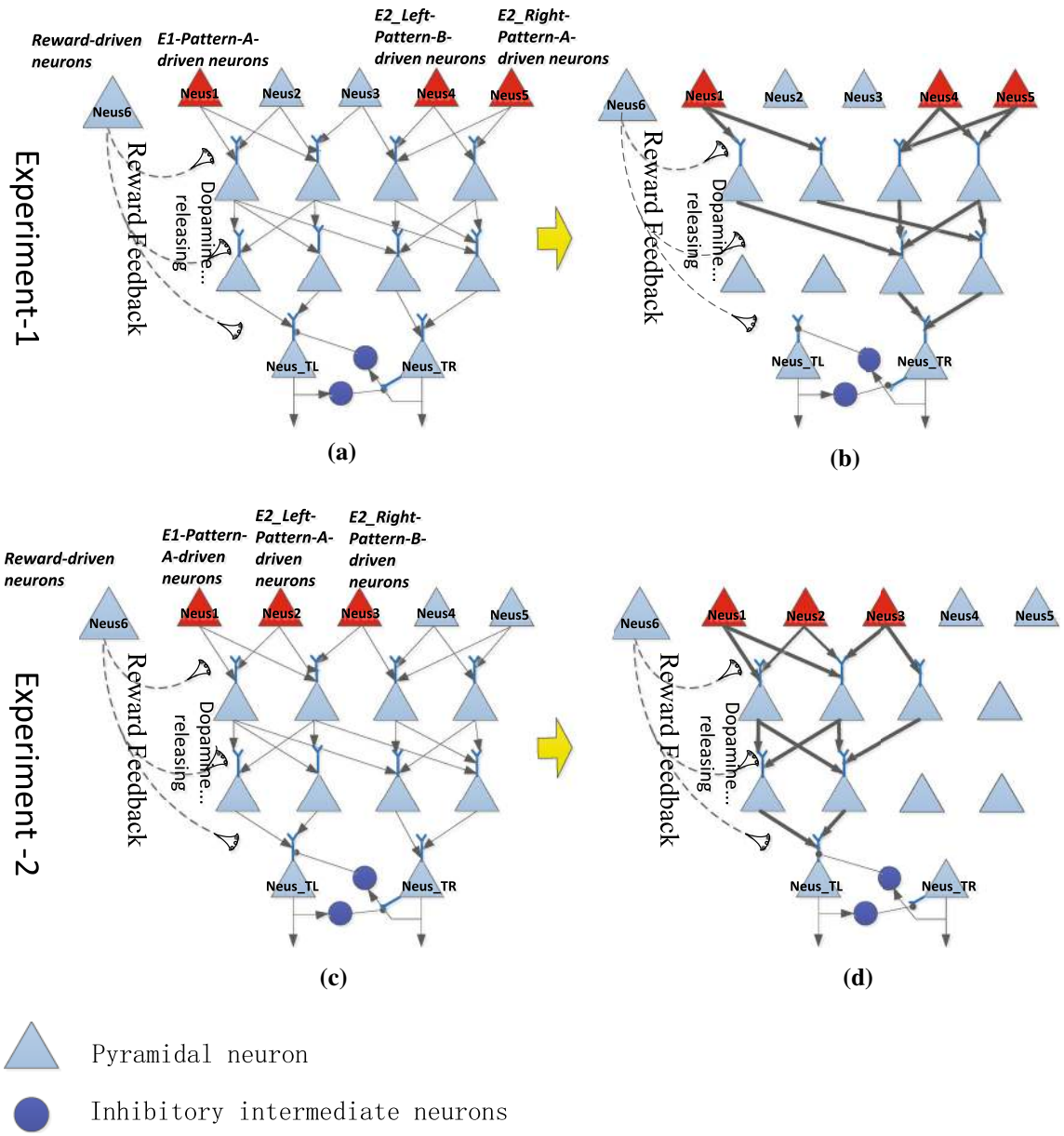


Fig. 24 Reward training and formation of the correct circuit. In Experiment-1, **a** when bees were making decisions, neuron groups Neus1, Neus4, and Neus5 were activated and fired with high frequency. During this stage, connections in the circuit were still set randomly. **b** Connections in the circuit after 300 epochs of reward training. Only a few strengthened connections were retained. These connections formed the correct decision-making path. In Experiment-2, **c** when bees were making decisions, neuron groups Neus1, Neus2, and Neus3 were activated and fired with high frequency. During this stage, connections in the circuit were the same as the Experiment-1. **d** Connections in the circuit after 300 epochs of reward training. Only connections that correctly reflected the causal association were retained

Then, after a slightly longer distance, bees reach position E2, which is at the fork of this Y-shaped tunnel. Visual patterns A and B are placed at the two entrances. Only if bees choose the left entrance (i.e., bees choose pattern A that was seen before), do they get a sugar reward. After multiple training sessions, bees stably choose the left entrance at position E2. However, without the reward training, bees randomly choose left and right entrances (Zhang et al. 2005).

To study the bee decision-making circuit that is involved in executing this behavior, we designed the connection structure among pyramidal neurons and interneurons as shown in Fig. 24. The initial connections were random. In this experiment, a neuronal AFR greater than 10 Hz was defined as high frequency, and an AFR less than 5 Hz was defined as low frequency. We use neuron group represent information. Here, the neuron group Neus1 was activated and fired with high frequency

only if bees encountered pattern A at position E1. Neuron group Neus2 was activated and fired with high frequency only if bees saw pattern A on the left at position E2. Neuron group Neus3 was activated and fired with high frequency only if bees saw pattern B on the right at position E2. Neuron group Neus4 was activated and fired with high frequency only if bees saw pattern B on the left at position E2. Neuron group Neus5 was activated and fired with high frequency only if bees saw pattern A on the right at position E2. Neuron group Neus6 was activated and fired with high frequency only if bees received a sugar reward. When bees receive a sugar reward, amines are produced and can affect connections between neurons in the circuit by remote projection. When neuron group Neus_TL's firing frequency was greater than Neus_TR's firing frequency, bees chose to turn left. Otherwise, bees chose to turn right. Table 4 describes the settings in two decision-making experiments, Experiment-1 and Experiment-2.

Development of the above decision-making circuit in Experiment-1: The initial decision-making circuit is shown in Fig. 24a (other connection settings also can be set in the initial circuit). During the early stage of training (the first 50 training epochs in Fig. 26a), the information cannot correctly flow from neuron groups Neus1, Neus4, and Neus5 to Neus_TR. With random background noise, the intensity of neuron group Neus_TR's AFR and Neus_TL's AFR was in an alternating mode. In this stage, bees did not choose to turn right every time at position E2; they randomly turned left or right. When Neus_TR's AFR was higher than Neus_TL's AFR (i.e., bees chose to turn right), bees received a sugar reward. Then, Neus6 from the reward circuit was activated and fired with high frequency, and amines released to this area by remote projection strengthened the connections in the path from input neuron groups Neus1, Neus4, and Neus5 to Neus_TR. When Neus_TL's AFR was higher than Neus_TR's AFR, bees chose to turn left. Then, bees did not receive a sugar reward, and firing of Neus6 was weak. Furthermore, the strength of connections in the path from input neurons to Neus_TL was weakened. With training, the path from neuron groups Neus1, Neus4, and Neus5 to Neus_TR was strengthened gradually, and the path from neuron groups Neus1, Neus4, and Neus5 to Neus_TL was weakened gradually (Connections strength W in the training process were shown in Fig. 25a). Then, random background noise input was no longer the main factor. The AFR of Neus_TR increased, and the probability of choosing to turn right at position E2 gradually increased as well. After 300 epochs of computer simulation training, the circuit in Fig. 24b was developed (weak connections were ignored). During this time, bees consistently chose to turn right (as the last 100 training epochs in Fig. 26a. The forming process of the decision-making circuit in Experiment-2 was shown in Figs. 24c, d, 25b, and 26b.

At the circuit level, the training process of bees is reflected in development of the correct path from input neuron groups to output neuron groups. For example, the objective of training in Experiment-1 is to form a path from Neus1, Neus4, and Neus5 to Neus_TR. In Experiment-2, the objective of training is to form a path from Neus1, Neus2, and Neus3 to Neus_TL. The dynamic changes of output neurons in these 2 experiments are shown in Fig. 25a, b. During the early stage of training, the intensity of neuron group Neus_TR's AFR and Neus_TL's AFR was in an alternating mode in which bees randomly chose to turn left or right. When bees chose the correct direction and received a sugar reward, the path from the input neuron to the correct output neuron was strengthened. Conversely, the path from the input neuron to the undesired output neuron was weakened. After repeated training, the probability of a bee making the desired decision increased, and the AFR of neurons for the correct decision was persistently and stably higher than the AFR of neurons for the incorrect decision

Table 4 Description of Experiment-1 and Experiment-2

	Experiment-1	Experiment-2	Description
Conditions			When a bee makes a correct decision, then it receives a sugar reward. With training, a bee can form a decision circuit that can make the correct decision in most cases
Visual mode A presented at E1	✔	✔	
Visual mode A presented to the left of E2		✔	
Visual mode B presented to the right of E2		✔	
Visual mode A presented to the right of E2	✔		
Visual mode B presented to the left of E2	✔		
Expected decision			
Turn right	✔		
Trun left		✔	

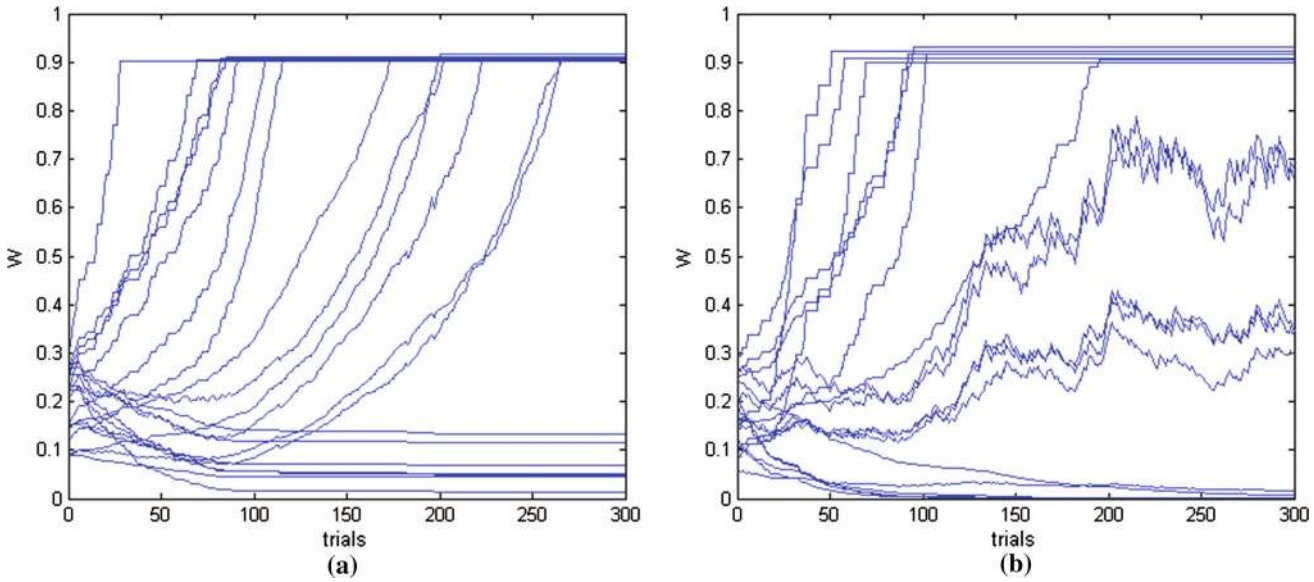


Fig. 25 **a** The learning process of connections strength W in Experiment-1 of Fig. 24; **b** the learning process of connections strength W in Experiment-2

Conclusions

Under the premise of obeying the physiological characteristics of biological neurons, we describe the control rules for phototactic behavior from the logic view, and designed a neural circuit to achieve these control rules. The AFR of the output neuron cluster was used to encode flying angular velocity. Between the input and the output layers, a series of excitatory and inhibitory neurons were used to modulate AFR of output neuron cluster according to different input. Then, we simulated AP firing and propagation of each neuron in the circuit using a distributed PC array. To verify the circuit's feasibility, we also conducted a real-time simulation of insect flight.We also explored how a correct neural decision circuit is generated through a bee's behavior experiment based on the reward and punishment feedback mechanism. By simulating and simultaneously recording circuit firing activities, we obtained a large degree of AP firing data that approximation to the data from multi-electrode array output. It also is convenient to test the validity of data mining methods in analyzing the correlation between neurons. It is helpful for us to better understand cooperative conditions that achieve some control rules. It is valuable to understand the information processing mechanism of the nervous system.

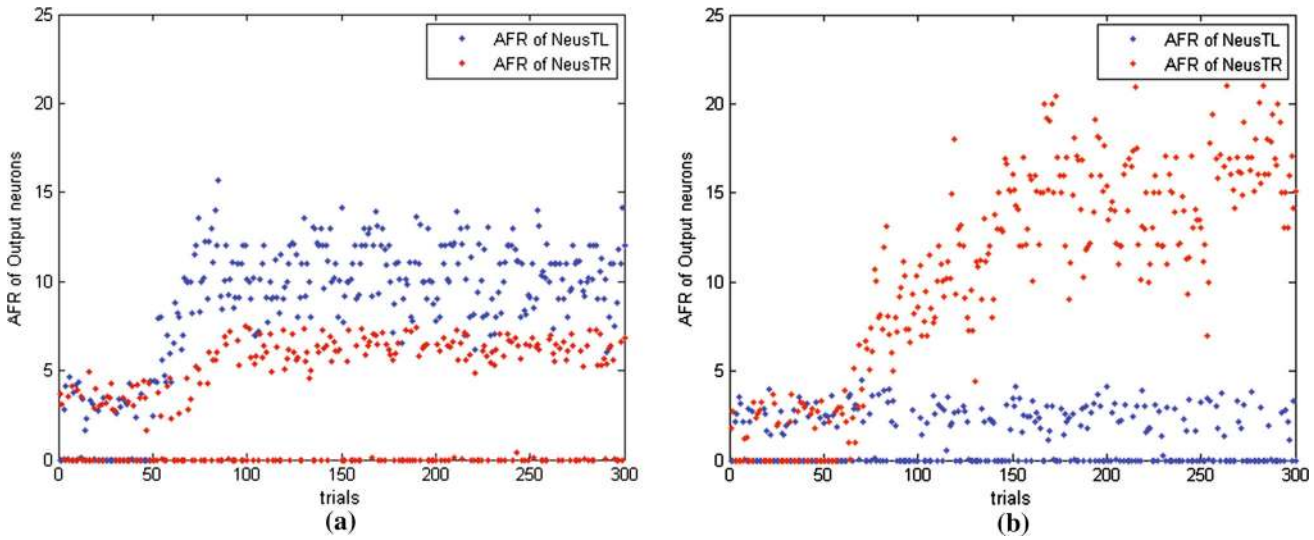


Fig. 26 **a** AFR of two neuron clusters in 300 decision-making trials in Experiment-1. **b** AFR of 2 neuron clusters in 300 decision-making trials in Experiment-2

Carandini (2012) considered that we lack a bridge theory from microscopic neural activity to macroscopic behavior. Moreover, this theory must explain how the structure of the nervous system adjusts its function. For example, how do microscopic activities of neurons and logical relationships in circuits support cognitive ability? The aim of this study was to construct a biological neural network for behavioral control rules and, at the same time, treat neurons and principles of circuit design from logic perspective that strictly followed electrophysiological characteristics and anatomy of biological neurons. Conclusively, our findings and detailed circuit presentation may be useful for this gradual transition from microscopic neural activity to macroscopic behavioral control.

Acknowledgements This work was supported by NSFC project (Project No. 61375122), and in part by Shanghai Science and Technology Development Funds (13dz2260200, 13511504300). We thank LetPub (www.letpub.com) for its linguistic assistance during the preparation of this manuscript.

References

Atkins MD et al (1980) Introduction to insect behaviour. Macmillan Publishing Co., Inc, New York

Bargmann CI, Marder E (2013) From the connectome to brain function. Nat Methods 10(6):483–490

Barron AB, Søvik E, Cornish JL (2010) The roles of dopamine and related compounds in reward-seeking behavior across animal phyla. Front Behav Neurosci 4:163

Britten KH, Shadlen MN, Newsome WT, Movshon JA (1992) The analysis of visual motion: a comparison of neuronal and psychophysical performance. J Neurosci 12(12):4745–4765

Carandini M (2012) From circuits to behavior: a bridge too far? Nat Neurosci 15(4):507–509

Carandini M, Heeger DJ (2012) Normalization as a canonical neural computation. Nat Rev Neurosci 13(1):51–62

Ferrée TC, Lockery SR (1998) Chemotaxis control by linear recurrent networks. In: Computational neuroscience. Springer, pp 373–377

Ferrée TC, Lockery SR (1999) Computational rules for chemotaxis in the nematode *C. elegans*. J Comput Neurosci 6(3):263–277

FitzHugh R (1961) Impulses and physiological states in theoretical models of nerve membrane. Biophys J 1(6):445

Ford LR, Fulkerson DR (1956) Maximal flow through a network. Can J Math 8(3):399–404

Haberly LB (1985) Neuronal circuitry in olfactory cortex: anatomy and functional implications. Chem Senses 10(2):219–238

Hebb DO (2005) The organization of behavior: a neuropsychological theory. Psychology Press, New York

Heinze S, Homberg U (2007) Maplike representation of celestial E-vector orientations in the brain of an insect. Science 315(5814):995–997

Hindmarsh J, Rose R (1984) A model of neuronal bursting using three coupled first order differential equations. Proc R Soc Lond B Biol Sci 221(1222):87–102

Hirata Y, Aihara K (2009) Representing spike trains using constant sampling intervals. J Neurosci Methods 183(2):277–286

Hodgkin AL, Huxley AF (1952) A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol 117(4):500

Horváth G, Varjú D (2013) Polarized light in animal vision: polarization patterns in nature. Springer, Berlin

Horvitz JC (2002) Dopamine gating of glutamatergic sensorimotor and incentive motivational input signals to the striatum. Behav Brain Res 137(1):65–74

Izhikevich EM et al (2003) Simple model of spiking neurons. IEEE Trans Neural Netw 14(6):1569–1572

Izhikevich EM (2004) Which model to use for cortical spiking neurons? IEEE Trans Neural Netw 15(5):1063–1070

Kandel ER, Schwartz JH, Jessell TM, Siegelbaum SA, Hudspeth A (2000) Principles of neural science, vol 4. McGraw-Hill, New York

Karbowski J, Schindelman G, Cronin CJ, Seah A, Sternberg PW (2008) Systems level circuit model of *C. elegans* undulatory locomotion: mathematical modeling and molecular genetics. J Comput Neurosci 24(3):253–276

Kreiman G (2004) Neural coding: computational and biophysical perspectives. Phys Life Rev 1(2):71–102

Li X, Chen Q, Xue F (2016) Bursting dynamics remarkably improve the performance of neural networks on liquid computing. Cogn Neurodyn 10:415–421

Marder E (2012) Neuromodulation of neuronal circuits: back to the future. Neuron 76(1):1–11

McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5(4):115–133

Neve K (2009) The dopamine receptors. Springer, Berlin

Pfeiffer K, Kinoshita M, Homberg U (2005) Polarization-sensitive and light-sensitive neurons in two parallel pathways passing through the anterior optic tubercle in the locust brain. J Neurophysiol 94(6):3903–3915

Pi HJ, Hangya B, Kvitsiani D, Sanders JI, Huang ZJ, Kepecs A (2013) Cortical interneurons that specialize in disinhibitory control. Nature 503(7477):521–524

Ridgel AL, Alexander BE, Ritzmann RE (2007) Descending control of turning behavior in the cockroach, *Blaberus discoidalis*. J Comp Physiol A 193(4):385–402

Ritzmann RE, Ridgel AL, Pollack AJ (2008) Multi-unit recording of antennal mechano-sensitive units in the central complex of the cockroach, *Blaberus discoidalis*. J Comp Physiol A 194(4):341–360

Samura T, Ikegaya Y, Sato YD (2015) A neural network model of reliably optimized spike transmission. Cogn Neurodyn 9(3):265–277

Tolnai S, Englitz B, Scholbach J, Jost J, Rübsamen R (2009) Spike transmission delay at the calyx of Held in vivo: rate dependence, phenomenological modeling, and relevance for sound localization. J Neurophysiol 102(2):1206–1217

Von Der Malsburg C (1995) Binding in models of perception and brain function. Curr Opin Neurobiol 5(4):520–526

Wang X (2007) Neural coding strategies in auditory cortex. Hear Res 229(1):81–93

Weliky M, Fiser J, Hunt RH, Wagner DN (2003) Coding of natural scenes in primary visual cortex. Neuron 37(4):703–718

White JG, Southgate E, Thomson JN, Brenner S (1986) The structure of the nervous system of the nematode *Caenorhabditis elegans*. Philos Trans R Soc Lond B Biol Sci 314(1165):1–340

Xu JX, Deng X (2013) Biological modeling of complex chemotaxis behaviors for *C. elegans* under speed regulation dynamic neural networks approach. J Comput Neurosci 35(1):19–37

Zhang S, Bock F, Si A, Tautz J, Srinivasan MV (2005) Visual working memory in decision making by honey bees. Proc Natl Acad Sci USA 102(14):5250–5255

Zhao J, Deng B, Qin Y, Men C, Wang J, Wei X, Sun J (2016) Weak electric fields detectability in a noisy neural network. Cogn Neurodyn 11:81–90