INTERFACE

rsif.royalsocietypublishing.org

Research

Article submitted to journal

Subject Areas:
Computational Biology, Biomimetics, Synthetic Biology

Keywords:
Biorobotics, Insect Navigation, Embodiment, Animal Intelligence, Bio-inspired, Insect Locomotion

Author for correspondence:
Shigang Yue; Jigen Peng
e-mail: sy237@leicester.ac.uk; jgpeng@gzhu.edu.cn

I2Bot: an open-source tool for multi-modal and embodied simulation of insect navigation

Xuelong Sun[1,2], Michael Mangan[3], Jigen Peng[1,2*], Shigang Yue[4*]

[1]Machine Life and Intelligence Research Center, Guangzhou University, Guangzhou, China.
[2]School of Mathematics and Information Science, Guangzhou University, Guangzhou, China.
[3]Sheffield Robotics, Department of Computer Science, University of Sheffield, Sheffield, UK.
[4]School of Computing and Mathematical Sciences, University of Leicester, Leicester, UK.

Achieving a comprehensive understanding of animal intelligence demands an integrative approach that acknowledges the interplay between an organism's brain, body, and environment. Insects like ants, despite their limited computational resources, demonstrate remarkable abilities in navigation. Existing computational models often fall short in faithfully replicating the morphology of real insects and their interactions with the environment, hindering validation and practical application in robotics. To address these gaps, we present I2Bot, a novel simulation tool based on the morphological characteristics of desert ants. This tool empowers robotic models with dynamic sensory capabilities, realistic modelling of insect morphology, physical dynamics, and sensory capacity. By integrating gait controllers and computational models into I2Bot, we have implemented classical embodied navigation behaviours and revealed some fundamental navigation principles. By open-sourcing I2Bot, we aim to accelerate the understanding of insect intelligence and foster advances in the development of autonomous robotic systems.

© The Authors. Published by the Royal Society under the terms of the Creative Commons Attribution License http://creativecommons.org/licenses/by/4.0/, which permits unrestricted use, provided the original author and source are credited.

THE ROYAL SOCIETY PUBLISHING

1. Introduction

Unravelling the underlying mechanisms that give rise to animal's intelligent behaviours represents a formidable challenge, one that demands an integrative approach that transcends conventional boundaries. Central to this endeavor is the recognition of the dynamic interplay between an organism's brain, body, and environment [1, 2] – a concept central to the biorobotics approach [3–5]. By bridging the gap between biology and robotics, biorobotics offers a unique opportunity to gain insights into the mechanisms underlying intelligent behaviour [3, 6] while simultaneously inspiring the development of more advanced robotic systems [7–9].

Insect navigation stands as a remarkable example of intelligent behaviour, offering a fertile ground for exploration [10]. The intelligence evidenced by insect navigation [11, 12] which includes both adaptive behaviours that changes dynamically as the environment/stimuli changes [13–15] and cognitive behaviours involving complex decision-making and information integration [16–18]. Despite possessing compact neural architectures with limited computational resources, insects demonstrate an astonishing capacity to navigate complex environments [19, 20], integrating multisensory inputs [16, 21], learning from experience [22], and executing adaptive decision-making processes [23–26]. This ability to solve intricate problems with constrained computational power has sparked profound curiosity and a desire to understand the principles that govern these behaviours, with applications possible in AI and robotics [12].

Computational models play important roles in revealing the neural circuits underlying insect navigation [27–29], and some of these models have been verified using mobile robots [28, 30, 31]. However, these robots could not fully recapitulate the real insects' body due to differences in morphology. Simulations offer significant advantages over real-world robots when studying complex systems, such as the ability to conduct controlled experiments flexibly, modify parameters easily, and rapidly iterate through multiple scenarios [32]. Although neuromechanical simulation tools for *Drosophila* have recently been developed [33–35], there remains a lack of tailored, realistic simulation tools for the systematic and comprehensive validation of insect navigation models, particularly for desert ants, that account for physical constraints.

To address these limitations, by leveraging the user-friendly and open-sourced software - ***Webots*** [36], a widely used robot simulation tools in academia [37–40], we developed a simulation tool that incorporates the morphological characteristics of desert ants, endowing our navigating agent with dynamic vision, olfactory, tactile, and mechanosensory capabilities (see Figure 1). Leveraging this framework, we have implemented simple Forward Kinematics (FK) and Inverse Kinematics (IK) gait controller, enabling the realisation of anatomically constrained computation model for path integration (PI). Additionally, our simulation tool facilitates the integration of vision and olfactory senses, allowing us to explore the sensory-motor closed loop inherent in insect navigation. Through systematic testing, we have uncovered evidence of how body-environment interactions can simplify the design of control models (see Table 1). Compared with other related simulation tools, the proposed open-source tool offers advantages in multimodal sensory capacity, flexibility in constructing 3D worlds, support for multiple programming languages, lower learning costs, and a user-friendly interface (see Table 4 and Section 4 for more detailed discussion).

By open-sourcing our simulation tool, we aim to provide a valuable resource for the research community, accelerating our collective understanding of insect intelligence. Furthermore, we believe that our work holds significant implications for the development of insect-inspired autonomous robots, offering insights that could pave the way for more adaptive and efficient robotic systems [7, 8].

2. Results

Example simulated agent/animal and environments are depicted in Figure 1, illustrating the intricate interactions between brain, body, and environment. The neural models can be integrated into the *Controller* component of the Webots' Robot. The morphology and spatial configurations of individual organs and limbs are reconstructed based on measurements obtained from real desert ants (refer to Section (a) for detailed information). The environments are assembled using Webots' built-in modules (refer to Section (b) for detailed information).

In this section, we will first introduce the sensory-motor system and the fundamental locomotion capabilities of a simulated desert ant. Following this, as summarised in Table 1, we will present several case studies demonstrating the utilisation of the proposed tool to implement classical control scenarios in a sensory-motor loop fashion. These simulations provide compelling examples of how I2Bot can offer new insights into understanding insect intelligence through the dynamic interaction between brain, body, and environment. In the future we hope for the tool to be extended to include additional sensory modalities, brain models, and body structures of other insects (see Table 5).

Note that force sensing (e.g., mechanosensory, tactile) is not utilized in the current implementation of vision- and olfactory-based behaviours. However, the sensory repertoire can be easily extended to incorporate force sensing data from the antennae and legs in specific scenarios, as these capabilities are already available (see Figure 1).

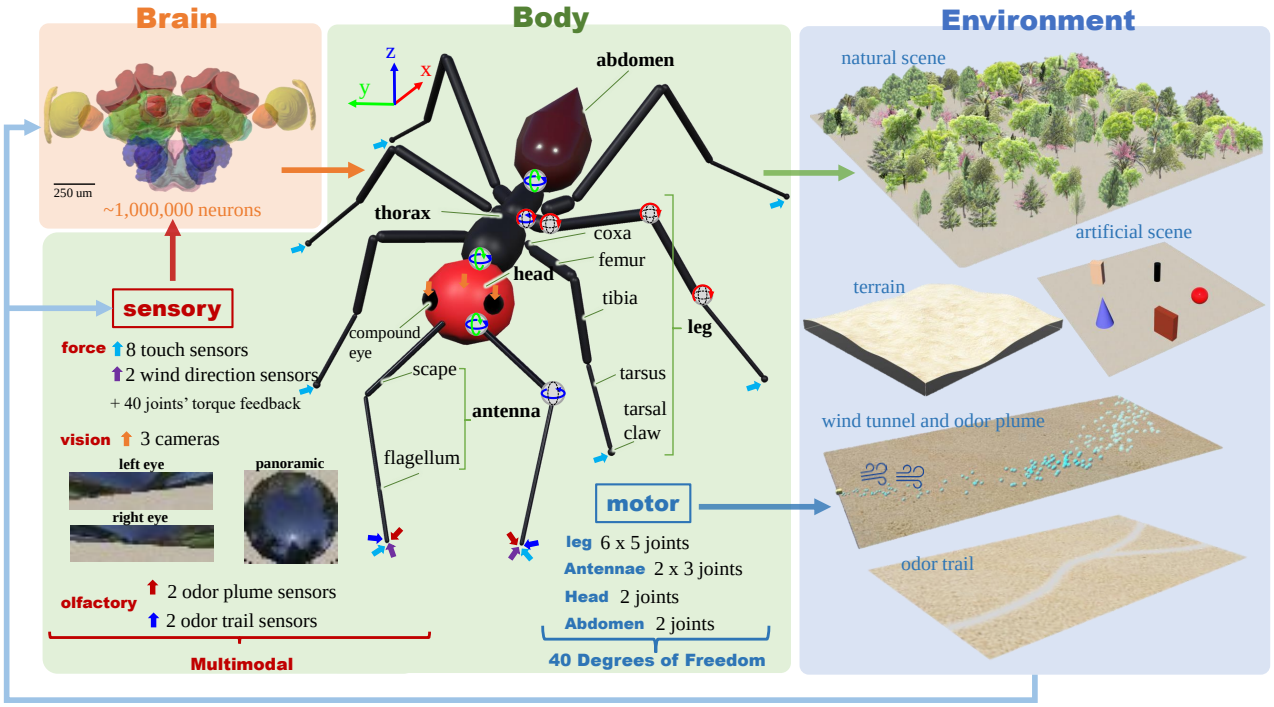

Figure 1. I2Bot overview. The tools aim to integrate simulation of brain, body, and environment, encompassing the sensory-motor loop. Colour-coded arrows indicate the positions of corresponding sensors, while joints are identified by their rotation axes. Note data here is for illustrative purposes, with specific examples shown in subsequent figures.

Table 1. Summary of the presented case studies with I2Bot. Text in red highlights the new insights and contributions.

	Sensor	Motor	Environment
path integration	velocity + direction	6 legs	plain terrain
	Bio-plausible neural output and simple gait control facilitate homing[28, 31]		
visual beacons	binocular/monocular vision	6 legs	plain terrain and 3D trees
	Embodied sensory input effects the performance of controller[1, 2]		
visual compass	panoramic vision	6 legs	plain terrain and 3D trees
	How internal representation of direction could control body orientation[41, 42]		
odour trail following	odour concentration (stable)	6 legs + 2 antenna	optic-simulated odour trail
	Embodied sensors simplify the design of the controller[43]		
odour plume tracking	odour concentration (fluctuated) + wind direction	6 legs	virtual wind and odour puffs
	First implementation of the embodied plume tracking[13]		

(a) Sensors and motors

Insects rely on multimodal sensory information to perceive their surrounding environment and make decisions [44]. Among these modalities, vision, olfaction, and tactile cues are considered crucial for insect navigation. A valuable tool should facilitate easy access to these sensory signals. To address this, the ant robot in I2Bot has been equipped with visual, olfactory, and force sensors. As depicted in Figure 1, two distinct vision sensors—binocular and panoramic—have been incorporated to serve different purposes in visual processing. Similarly, the proposed tool enables simulation of olfactory signals with various spatial-temporal patterns, such as stable odour trails for simulating trail following and dynamic odour plumes for investigating navigation in turbulent olfactory environments. Tactile and mechanosensory input is simulated through force sensors located at the tip of each leg, the tip of each antenna, and torque sensors located in each leg joint (5 per leg), at 3 locations in each antenna, and 2 in the neck and abdomen joints. Finally, wind sensing is simulated by 2 sensors positioning at the tip of the antennae, see Figure 1 for overview.

The locomotion capabilities of insects have long intrigued researchers and have served as inspiration for the design of six-legged robots [4, 8]. To replicate the authentic locomotion of insects, we have defined five degrees of freedom (DoF) for each leg (i.e., front left-FL, middle left-ML, hind left-HL, front right-FR, middle right-MR, hind right-HR). The joints are designated based on the anatomy of body and leg segments. For instance, the joint ThCz refers to the joint connecting the *Thorax* and *Coxa*, rotating along the z-axis. To streamline body movement, the head and abdomen each possess two degrees of freedom, allowing rotation along the y and z axes. Recent studies have emphasised the pivotal role of active olfactory sensing [43, 45, 46] in shaping the navigational behaviours of insects and its application in robotics [47]. Therefore, each antenna of I2Bot is equipped with three degrees of freedom to enable agile movement of the antenna.

(b) Basic locomotion

Locomotion is the basis of complex manoeuvres that have been observed in insects, thus it is crucial for navigation behaviours. To simplify the locomotion control, we have adapted concepts from robotics and implemented a simple gait control algorithm based on forward and inverse kinematics (FK and IK). Unlike higher level locomotion controllers [48, 49] wherein the gait is assumed fixed or emergent, this approach concerns the movement of each joint (i.e., directly set the angle of each joint) for a given leg movement pattern (i.e., a gait). A kind of lower level method without considering the muscles' contraction and relaxation [50]. The forward kinematics (FK) method directly assigns the joint angle which determines the position of the leg tip. While the inverse kinematics (IK) computes each joint angle given the spatial position of the end tip of each leg (for more details see Section (c)). That's to say, in FK, we design the joint angles while in IK, we design the spatial movement of the leg tip and the joint angles are computed through trigonometry. Note that in the current implementation, we only use 3 DoF of each leg (keep angles of the ThCx and the TiTa joint to be constant), which is simpler and more usual for hexapod robots. However, users could in practice apply all the 5 DoF to replicate more biologically realistic locomotion control. In summary, the locomotion models currently included are based on robot control strategies, and could be efficiently upgraded to match models from the biological literature (e.g. [48, 51]) since the joint torque feedback are already accessible.

As depicted in Figure 2 (a) and (b), inverse kinematics-based gait control exhibits greater stability but demands more computational resources. To assess the motion control's performance, we evaluated its ability to navigate uneven terrain and climb walls. Figure 2 (c) illustrates the comparison of walking on floors with varying degrees of unevenness. Here, we observed faster changes in the z-position of the body with negligible differences in joint torques, indicating the six-legged robot's inherent capacity to navigate uneven terrain embedded in its physical dynamics. In climbing vertical walls, insects utilise adhesion which we simulated by dynamically applying force to the leg's ground contact points. Figure 2 (d) presents the climbing performance with varying levels of force applied to the ground support points. Interestingly, larger force doesn't necessarily translate to better wall-climbing performance. This may be due to the competition between the force required to lift the leg and the force needed to maintain contact with the wall. Different gait types are demonstrated in Figure 2 (g,h), with the tripod gait achieving the fastest body movement speed, while the wave tripod is the slowest. It's worth noting that other gait controllers such as CPG and feedback models [4, 48, 52, 53] could also be integrated. Here, we present the simple FK-based gait controller to showcase the proposed tool's feasibility.

(c) Path integration: incorporating locomotion with a bio-constrained neural model

Path integration (PI) stands as a cornerstone of the insect navigation toolkit [54]. Foragers adeptly track the distance and direction to their nest by integrating the series of directions and distances travelled into a *home vector* [19, 55]. Guided by this vector, pointing towards the start position (typically the nest), desert ants can return accurately to their nest even after traversing hundreds of meters (Figure 3a). Recent neuroethological investigations have unveiled the central complex (CX) within the insect brain as pivotal for the computational processes involved in path integration. The CX not only hosts a ring attractor network encoding the animal's heading direction [56, 57] but also receives optic flow information as a velocity encoding [28, 58].

To demonstrate the ease of incorporating biologically constrained neural models with locomotion controllers using I2Bot, we implemented the popular insect path integration model proposed by [28]. We utilised its output as tuning factors for the previously described forward kinematics gait controller. Specifically, the neural activation of the left and right PFL/CPU1 neurons from the insect steering circuit modifies the *hip swing* (speed) and *rotation* (direction) of the gait (Figure 3b,c). This marks the first instance of testing this popular neural model in a hexapod robot with physical constraints and ant-like morphology. Notably, the simulated ant can perform curved walking, unlike a recent biorobotics study [31] where the robot could only rotate on the spot. Several simulations were conducted, and we found that this bio-constrained model performed effectively (see Figure 3d,e), suggesting a robust and efficient solution for robot path integration. Figure 3f&g illustrate the dynamic neural activation of the encoded *home vector* and the agent's velocity during foraging and homing. This exemplifies how neural computation and encoding, coupled with a simple locomotion controller, can generate robust navigation behaviours.

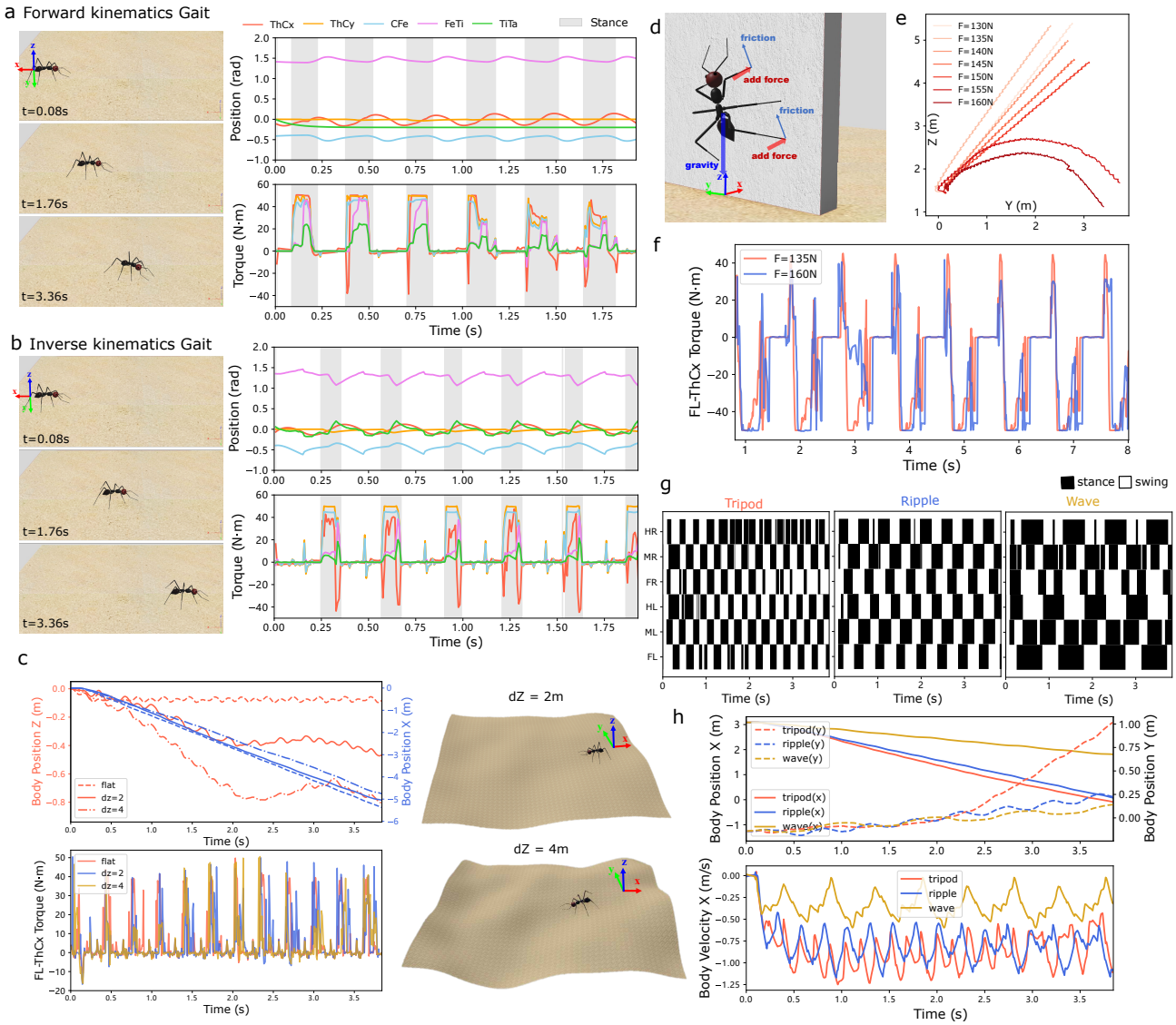

Figure 2. Locomotion and gait control. (a)-(b) Left: snapshots of walking ants at specific times; Right: joint position and torque of the middle right leg based on forward and inverse kinematics. (c) Walking on uneven terrain. (d)-(f) Climbing the wall using adhesion with different force added at the tip of each leg, trajectories are shown in (e) and the torque values are plotted in (f). (g)-(h) Forward kinematics-based implementation of different gait types: tripod, ripple, and wave. Note that when walking on the floor, the robot is initially facing negative x-axis (see the coordinates labelled in Figure 1). All the origins are marked in the figure for clarification.

(d) Vision-guided manoeuvres

Vision serves as one of the primary means for animals to perceive their environment, and insects rely on visual cues for navigation in various ways [59, 60]. In this section, we present scenarios involving monocular, binocular, and widely used panoramic vision to demonstrate vision-motor control in the simulated ant robot.

(i) Monocular and binocular visual beacons

Research has indicated that insects instinctively use visual landmarks as beacons [61, 62]. To showcase the feasibility of using visual images as input with our proposed tool, we implemented a simple visual beacon algorithm (see Figure 4a,b) based on the biologically plausible steering circuit used in neural models of the insect brain [28, 29, 58]. To compare the effects of different types of visual input on motor output, we employed monocular (i.e., solely left or right image) and binocular images as inputs to the visual beacon model. The results are illustrated in Figure 4d, where the simulated ant with binocular image inputs outperforms the others in terms of task completion time. Eye occlusion resulted in ipsilateral motor bias, where agents with the left eye occluded (receiving only right eye information) approached the landmark

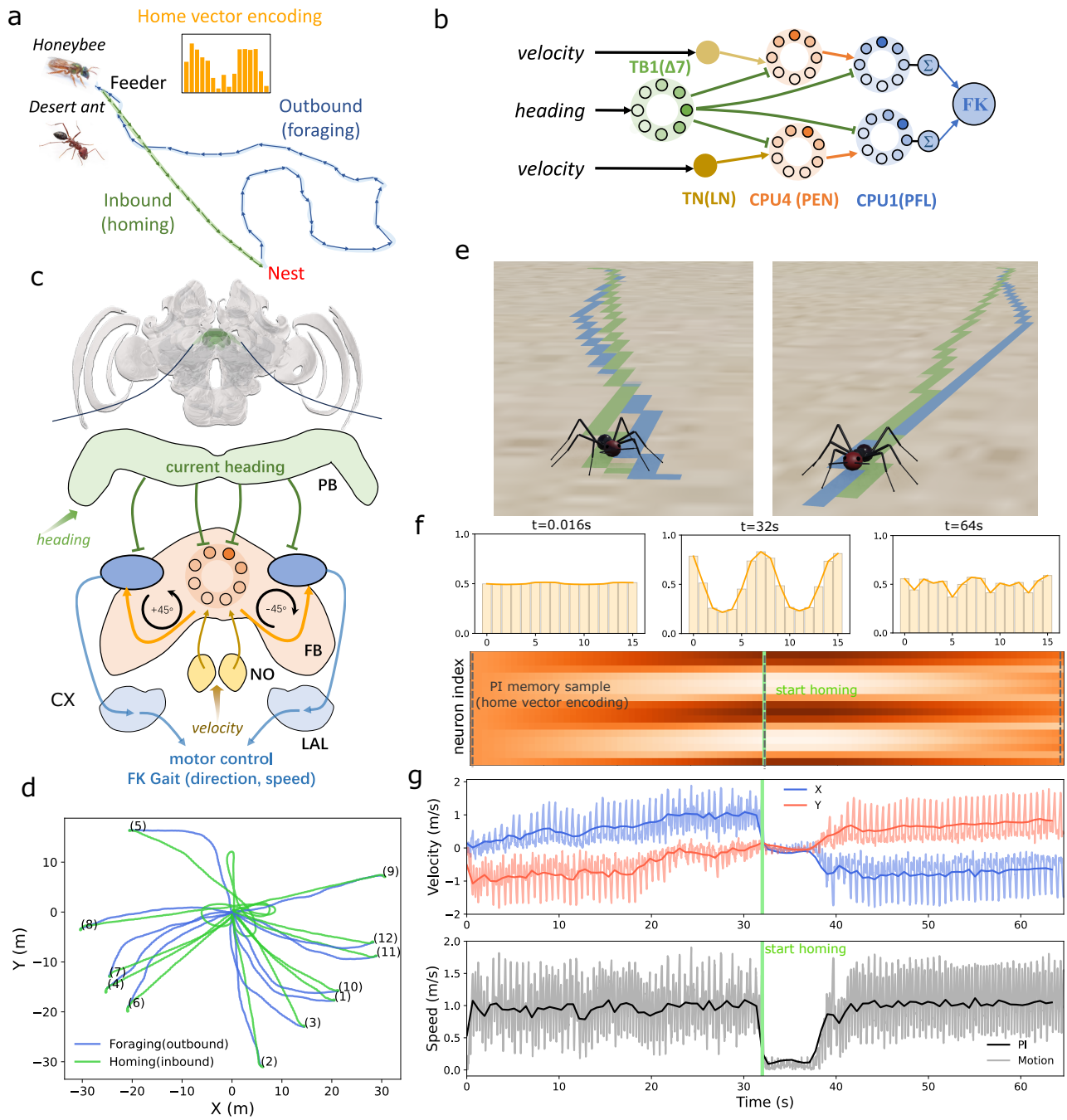

Figure 3. Implementation of path integration model. (a) Schematic diagram of central-foraging insects' path integration. (b) The neural model of path integration adapted from [28]. The output of the model is linked to the forward kinematics gait controller. (c) Insect brain regions within the central complex related to PI computation. (d) Trajectories of simulated ants conducting foraging trips and homing using the proposed PI model. (e) Snapshots of two typical experiments. Trajectories are drawn using the *Pen* function within Webots. (f) Neural activation of PI memory neurons (CPU4 or FPN neurons) during foraging and homing, with sampled points marked by grey dotted lines. Corresponding activation profiles are placed above. (g) Velocity in the x- and y-axis (top) and speed (bottom) during navigation. Dark solid lines indicate the inputs fed into the PI model, while lighter curves show instant velocity measurements of the walking robot.

with left-biased trajectories (red curves in Figure 4c). This illustrates how embodied sensory input influences navigation performance guided by the same neural controller, providing partial insights into the advantages of binocular vision.

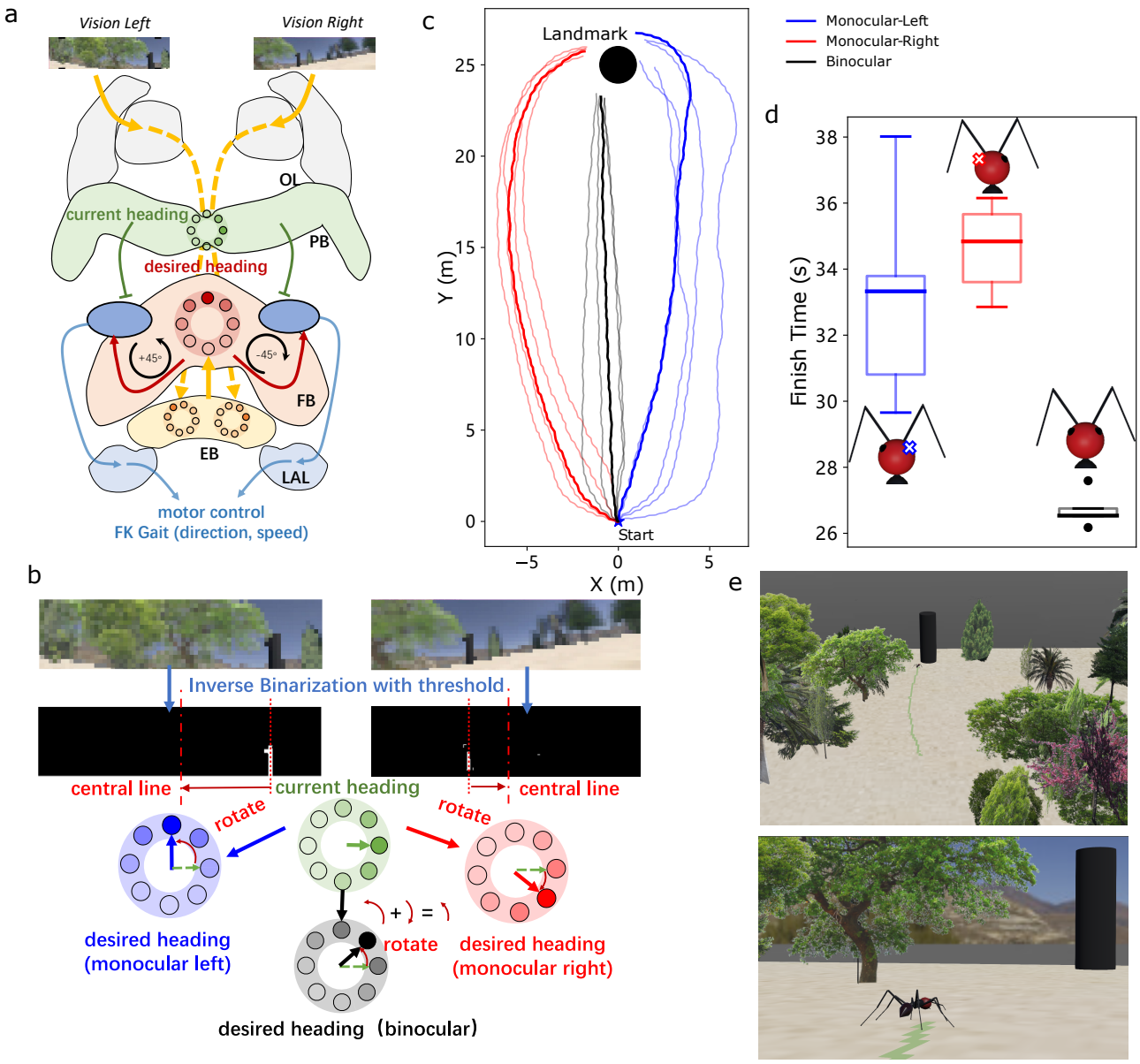

Figure 4. Visual beacons with monocular and binocular vision. (a) Schematic diagram of the visual beacons model overlapped with the insect brain regions. (b) Illustration of the generating desired heading from the visual inputs with monocular and binocular visual input. (c) Trajectories of the visual beacon using left, right, and binocular vision. (d) Time taken by the robot to reach the position near the landmark. (e) Snapshots of the simulated ant robot and environment during visual beacon navigation.

(ii) Panoramic visual compass

In addition to binocular vision, insects are believed to use panoramic views when navigating [63–65]. Computational models also leverage panoramic images to extract useful information for solving visual navigation tasks, particularly in the frequency domain [29, 66, 67]. Here, we demonstrate how processing panoramic views in the frequency domain can aid the agent in sensing direction, known as the visual compass [65, 68]. The methods used to process the panoramic view here are the same as those in [29, 67], which could also serve as the foundation for insect visual route following behaviour (see Figure 5a). The initially extracted phase encoding in the frequency domain is stored as the desired heading direction. When the simulated 3D world is manually rotated, the received panoramic image and the phase encoding rotate accordingly. The difference between the stored phase and the current phase is output to the steering circuit, enabling the agent to turn and continuously maintain its initial heading. To test the performance of the visual compass-based head tracking, we rotated the 3D world in two modes: incremental rotation and jump rotation (sudden and fast). As shown in Figure 5c and supplementary Movie S8-9, the simulated ant efficiently tracked its stored initial heading in both rotation modes, consistently orienting towards the red ball in the 3D visual world.

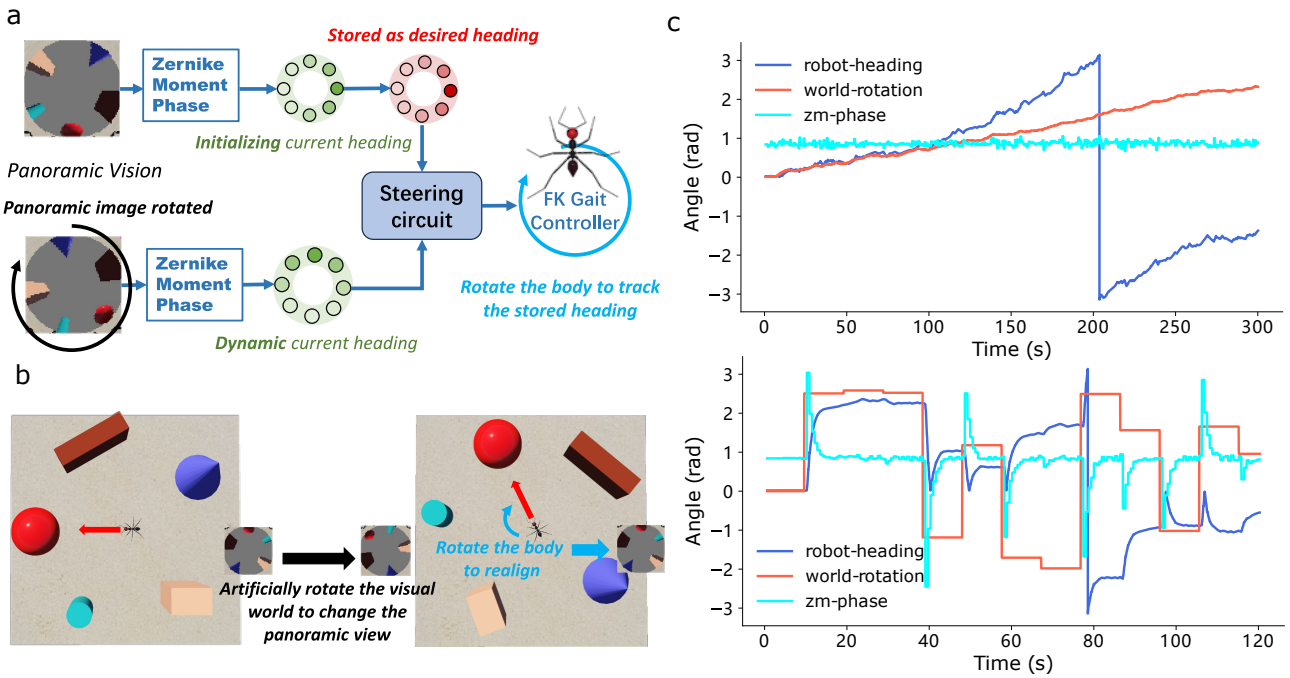

Figure 5. Visual compass using panoramic view. (a) Schematic diagram of the visual compass model. (b) Screenshot of the rotating simulated world from Webots. The rotation mode of the visual scene has two types: incremental (top of (c)) and jump (bottom of (c)). (c) The heading direction of the robot, the visual scene, and the phase of the Zernike moment are plotted when tracking the stored direction by the Zernike moment based visual compass.

(e) Olfactory-guided manoeuvres

Olfaction represents another crucial sensory domain guiding a diverse array of navigational behaviours [13, 69–71]. Different olfactory landscapes lead to varied navigation behaviours [72]. To showcase how the proposed platform facilitates olfactory navigation simulation under environments with different spatial-temporal features, we implemented odour trail following behaviour under a spatially stable distribution [73] and odour plume tracking behaviours in a turbulent olfactory environment [13].

(i) Odour trail following using active sensing

Odour trail following [43] observed in carpenter ants (*Camponotus pennsylvanicus*) was realised to demonstrate how embodiment benefits the design of control strategies. We compared the trail following performance of agents with and without moving their antennae during odour tracking by measuring the time taken to navigate from the same start point to the end of the pheromone trail (finish time as shown in Figure 6d). Intriguingly, we found that agents with moving antennae significantly outperformed those with fixed antennae (Figure 6c,d) when tracking spatially wider trails, although their performance on trails with normal width was similar. This suggests that dynamic antennae movement enhances the adaptability of control strategies to handle a wide range of environmental conditions, as demonstrated in [43]. This finding underscores how embodiment facilitates the design of effective control systems [2].

In addition, I2Bot simplifies body modification. For example, supplementary Video12 demonstrates a preliminary test where ant robots with varied antenna lengths, but identical control algorithms, exhibited different performances in odour trail following. This together with the results presented in the visual beacon simulations (Figure 4) further highlights the impact of body morphology and physical dynamics on the development of control algorithms and neural circuitry. It underscores that intelligence develops through the intricate interaction between body, brain, and environment[1].

(ii) odour plume tracking

To investigate sensory-motor control in turbulent odour environments, we developed a simplified control strategy inspired by behaviours observed in walking flies [13]. While this behaviour has been extensively modeled in previous studies [17, 74], we adapted a conceptual model to validate the proposed tool's capacity to implement dynamic olfactory navigation behaviours. As illustrated in Figure 6e, once the odour is sensed (i.e., the perceived odour concentration exceeds the threshold), the agent orients upwind (aligning with the wind direction detected via the virtual mechanosensory signal). In the absence of odour, the agent performs random rotations in place, simulating scanning/searching manoeuvres. Despite

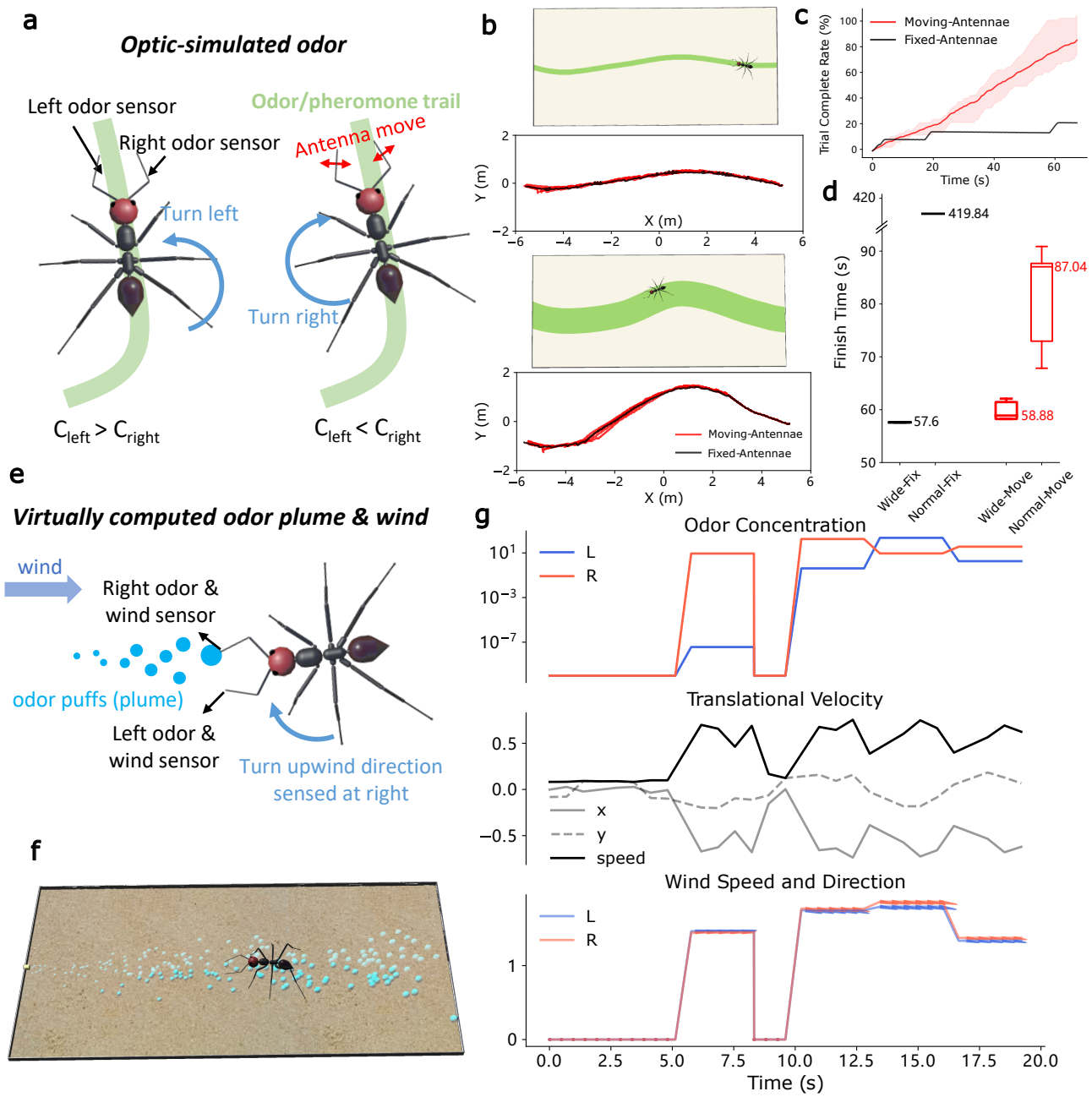

Figure 6. Olfactory-motor closed-loop control. (a)-(d) depict odour trail following, while panels (e)-(g) illustrate odour plume tracking. (a) Schematic diagram illustrating the control strategy of the trail following behaviour. (b) Bird's-eye screenshots of the experiments and the corresponding trajectories of the agent with different widths of the odour trail. (c) Comparison of the completion rate between fixed-antenna and moving-antenna configurations. (d) Box plot of the finish time (n=5). (e) Model of plume tracking. (f) Screenshot of the experiments in the Webots 3D scene. (g) Perceived odour concentration (top), wind (bottom), and the moving velocity of the ant robot (middle) during one experiment.

discontinuous olfactory and mechanosensory cues in this turbulent environment, agents successfully navigate to the odour source using this simple strategy (see Figure 6g and Supplementary Video13).As the first attempt to implement dynamic odour plume tracking in a simulated robot with realistic ant morphology and physical constraints, this case study may provide useful foundational work for future investigations into embodied olfactory-motor control.

3. Materials and Methods

I2Bot is build in the popular open-source robot simulator Webots [36]. The controllers of the demos presented in this paper were programmed in Python 3.9 (compatible for newer version) with packages such as *numpy*, *cv2*, *ikpy*, etc. Note that aside

Table 2. The calculation of ThCx joint range R for yaw-control of forward kinematics gait

	left legs (FL, ML and HL)	right legs (FR, MR and HR)
$\theta \geq 0$	$S_l d(2\theta - 1)$	$S_l d$
$\theta < 0$	$S_l d(2\theta + 1)$	$-S_l d$

from Python, Webots supports multiple programming languages like Java, C/C++ and Matlab scripts. All the source code related to this project is open-sourced at Github `https://github.com/XuelongSun/I2Bot` under the MIT license.

(a) Robot design

The morphology of a *Cataglyphis fortis* desert ant was estimated from images provided [75] (see Supplemental Fig.S1). To simulate the ant body in 3D, we use Webots embedded geometries *sphere*, *cone* and *capsule* to model the body parts as that in [39]. As shown in Supplemental Fig.S1, the length of the hind leg is larger, making it distinctive as the characteristics of desert ants' leg morphology. Note that the size of the ant is scaled 100 times larger for the ease of the physics simulation within Webots.

Binocular Visual sensors are placed at the position of eyes while the abstracted panoramic visual sensor is placed at the top of the ant robot's head to obtain a more ideal field of view. Olfactory sensors are mounted on the end of the antenna to mimic that of animals [46] and make it easy for robotics studies [47]. Eight tactile sensors are mounted on the tip of each leg and each antenna which can detect three dimensional contact force. Each joint (40 in total) has a torque sensor that could be dynamically accessed during locomotion (see Figure 1). This torque feedback could be very useful in designing bio-plausible locomotion controllers like that in [48, 51].

(b) Environment construction

For visual environments, all visible objects, such as basic geometric shapes and trees used in the presented cases, are available in Webots. Vision is constructed using the camera model embedded in Webots. Additionally, simulated visual environments from previous studies on insect navigation can be easily imported into the Webots 3D environment via the *Mesh* shape or *Cadshape* node. See Supplemental Fig.S2 for examples of visual scenes used in [27, 29, 67, 76]. This feature allows researchers to directly obtain visual stimuli from the desired world without needing to configure a camera model, facilitating the comparison of different visual navigation models that share the same visual input.

For olfactory environments, there are two types of simulation: 1) for the odour trail, it is optically simulated as the specific texture of the floor. This texture can be drawn manually thus allows great flexibility. 2) for the odour plume, it is simulated by the filament-based model [77] and its Python implementation called *pompy*. This model mimics both short- and long-timescale features of odour plume evolving in a turbulent flow and has been widely used in olfactory related studies involving insect navigation [69, 78] and robot odour source localization [79].

(c) Models

(i) Forward kinematics

To simply the gait control, we keep the ThCx and TiTa joint angles constant and denote the angles of the left three joints-ThCz, CTr and FTi to be α, β and γ respectively. These values are determined by the hip swing (S_l with unit rad), lift swing (S_h with unit rad) and the moving direction (-1 for backwards and 1 for forwards). To realise yaw control of the robot body, we introduced rotation configuration denoted as θ. Table 2 lists the formulas to calculate the base range of the ThCx joint R.

As different gait types share the same mechanisms, here we just describe how to use FK to generate a tripod gait, other gait could be generated with alternative leg coordination sequence. For the first half phase of a leg during the swing stage, the angles of the three joints are calculated by:

$$\begin{cases} \alpha_{sw1}(k) = & -R + \alpha_0 + \frac{R}{S_n}k \\ \beta_{sw1}(k) = & \beta_0 + \frac{S_h}{S_n}k \\ \gamma_{sw1}(k) = & \gamma_0 + \frac{S_h}{S_n}k \end{cases} , k = 0, 1, ..., S_n \tag{3.1}$$

where $\alpha_0, \beta_0, \gamma_0$ is the initial angle of ThCx, CTr and FTi joint respectively. S_n is the total time steps to lift the leg to its highest position determined by the lift swing S_h. Similarly, in the next half phase, the angles are calculated by:

$$\begin{cases} \alpha_{sw2}(k) = & \alpha_0 + \frac{R}{S_n}k \\ \beta_{sw2}(k) = & \beta_0 + S_h - \frac{S_h}{S_n}k \\ \gamma_{sw2}(k) = & \gamma_0 + S_h - \frac{S_h}{S_n}k \end{cases} , k = 0, 1, ..., S_n \tag{3.2}$$

In this phase, the leg is moved to the frontmost position (α reaches its highest value) and contacts the ground again (β and γ are set to its initial value).

For the stance phase, $\beta(k)$ and $\gamma(k)$ are kept constant (as the leg does not lift) while $\alpha(k)$ is reversed (moves from the front of the body to its back):

$$\begin{cases} \alpha_{st}(k) = & \alpha_0 + R - \frac{R}{S_n}k \\ \beta_{st}(k) = & \beta_0 \\ \gamma_{st}(k) = & \gamma_0 \end{cases} , k = 0, 1, ..., S_n, S_n + 1, ..., 2S_n \tag{3.3}$$

For a tripod gait, the angle of ThCx joint of FR, ML and HL leg should be α_{sw1} -> α_{sw2} -> α_{st} while that of the other three legs (i.e., FL, MR and HR) should be α_{st} -> α_{sw1} -> α_{sw2}. The angles of CTr (β) and FTi (γ) joint follow the same rule.

(ii) Inverse kinematics

Unlike the forward kinematics wherein the joint angles are computed directly given the leg movement parameters (i.e., hip swing S_l, lift swing S_h, etc.), when applying inverse kinematics, we first calculate the positions of the leg tip at a certain time step k in swing and stance state, and then utilize the inverse kinematics (with python package-*ikpy*) to compute the joint angles. The legs of the ant robot were defined by URDF files and can be read by the *ikpy* package. Specifically, the step length and step height is denoted as L and H respectively, then for leg in the swing stage, the joint angles should be:

$$[\alpha_{sw}(k), \beta_{sw}(k), \gamma_{sw}(k)] = \mathbf{ik}(Homo([0, 0, \theta - \frac{\theta}{S_n}k], [L/2 - \frac{L}{S_n}k, 0, -\frac{H}{S_n}k])) \qquad k = 0, 1, ..., S_n \tag{3.4}$$

Where $Homo(\mathbf{rotation}, \mathbf{translation})$ calculates the homogeneous matrix given the rotation and translation input and $\mathbf{ik}()$ denotes the IK function which receives the homogeneous matrix of the leg tip as parameters and returns the calculated joint angles (This function is provided as an API in the *ikpy* package, and the specific calculations are not detailed here, as they follow standard inverse kinematics processes). θ is the rotation configuration. For the stance leg:

$$[\alpha_{st}(k), \beta_{st}(k), \gamma_{st}(k)] = \mathbf{ik}(Homo([0, 0, \frac{\theta}{S_n}k], [-L/2 + \frac{L}{S_n}k, 0, 0])) \qquad k = 0, 1, ..., S_n \tag{3.5}$$

Thus, to generate a tripod gait, the angle of the ThCx joint of FR, ML and HL leg should be α_{sw} -> α_{st} while that of the other three legs (i.e., FL, MR and HR) should be α_{st} -> α_{sw}. The angles ofthe CTr (β) and the FTi (γ) joints follow the same rule.

(iii) Path integration

Path integration requires heading direction and velocity information, this is provided by the **Webots** *Supervisor* in the current implementation. One can use other modalities like vision (e.g. optic flow) to obtain this. The model of path integration is adapted from [28, 29], the output of the summed CPU1 (PFN) neurons are fed into the FK-based gait controller as following:

$$\theta(t) = k_{PI}[\sum_{i=0}^{7} C^i_{CPU1}(t) - \sum_{i=8}^{16} C^i_{CPU1}(t)] \tag{3.6}$$

where $C^i_{CPU1}(t)$ is the membrane potential of the ith CPU1 neuron at time t. Similar with that in [28], the difference between the summed activation of left and right CPU1 neurons modifies the rotation configuration of the gait control (see Table 2 for specific calculation) and thus guides the robot's turning decision via the motor scale k_{PI}. To tune the walking speed, step length S_l is computed by:

$$S_l(t) = S^0_l(t) + S^f_l(t) \times (1 - \min[|\theta(t)|, 1]) \tag{3.7}$$

where the basis step length $S^0_l = 5°$ and the scale factor $S^f_l = 20°$ are held constant. Thus, the step length is confined in the range of $[5°, 25°]$. These parameters are selected empirically based on the ant robot's performance in the designed simulation. For running multiple trials (results presented in Figure 3d), we set the start location to be fixed at $[0, 0]$ while the initial heading varies in range $[0, 2\pi]$.

(iv) Vision

Vision is provided through the embedded camera sensor in Webots. For panoramic view simulating, the projection mode of the camera is set to *spherical* while for monocular and binocular it is set to be *planar*. The parameters of the camera such as resolution, field of view and focal length can be customized in Webots easily through the *scene tree*.

- **Visual beaconing**: The resolution of the left and right eye camera is 74×19 (i.e., the image width $W_{vb} = 74$ and the image height $H_{vb} = 19$) with the horizontal and vertical field of view to be $2rad$ and $0.5rad$ respectively. The motor command is calculated by the 'copy-and-shift' mechanism proposed in [17, 29]. The input image will be first binaries with a given threshold (empirically selected in the range $[10, 25]$ given that the brightness of the image is in the range $[0, 255]$). Then the value of $shift_{vb}$ is computed by:

$$shift_{vb} = \lceil |6\delta / W_{vb}| \rceil \tag{3.8}$$

Where δ is the different between the horizontal image center and the center of the landmark area:

$$\delta = \begin{cases} W_{vb}^{Left} - P_l^{Left} & \text{Left monocular} \\ W_{vb}^{Right} - P_l^{Right} & \text{Right monocular} \\ \frac{(W_{vb}^{Left} - P_l^{Left}) + (W_{vb}^{Right} - P_l^{Right})}{2} & \text{Binocular} \end{cases} \tag{3.9}$$

Where P_l is the averaged horizontal position of the dark landmark in the left and right retina (i.e., image pixel coordinates). The desired heading is then calculated through the *copy-and-shift* mechanism wherein the current heading is copied and shifted by $shift$ amount. Then the turning value $\theta(t)$ is calculated by the steering circuit [17, 28] as that in Equation 3.6 but with a different motor scale of visual beacons k_{vb}. The ThCx joint range R (Table 2) and the hip swing $S_l(t)$ (Equation 3.7) of the FK gait controller then could be calculated to affect the locomotion. For the multiple trials simulation, the ant robot starts from the same point but with different initial headings. Specifically, for agents using the left eye view uniformly sample from the range $[0, \pi/2]$, whereas for agents using the right eye view sample uniformly from the $[\pi/2, \pi]$. For binocular agents, they sample from the range $[\pi/4, 3\pi/4]$.
- **Visual compass**: The resolution of the artificial panoramic vision is 72×72. The direction of the view is extracted by the phrase information of the Zernike moment [29, 67] coefficient (with order $n = 7$ and repetition $m = 1$, $\Phi_{7,1}(t)$). This Zernike phase is compared with the initially stored phase ($\Phi_{7,1}(0)$) and then the difference determines the sign of turning parameter $\theta(t)$ and the hip swing:

$$\begin{cases} \theta(t) = \text{sign}(\Phi_{7,1}(t) - \Phi_{7,1}(0)) \\ S_l(t) = |(\Phi_{7,1}(t) - \Phi_{7,1}(0))/5| \end{cases} \tag{3.10}$$

(v) Olfactory

- **Odour trial following**: As described above, the odour trail is simulated by the texture of the floor, thus the concentration is detected by the *DistanceSensor* (note that the field type should be set to be "infra-red") whose value is modified by a reflection factor depending on the color, roughness and occlusion properties of the object. The detected odour concentration of the odour (returned value of the *DistanceSensor*) from the left and right antenna at time t are denoted as $O^l(t)$ and $O^r(t)$, the difference between the left and right sensed odour concentration is denoted as $O_d(t) = O^l(t) - O^r(t)$. Table 3 shows how the sensed odour concentration determines the locomotion control (i.e., determines the hip swing and rotation of the FK gait control). For multiple trial experiments, all the agents started from the same position $[5, 0]$ with identical initial heading (π).
- **Odour plume tracking**: In this simulation, the agent not only detects the odour concentration (O^l and O^r) but also the wind direction ($\mathbf{W^l} = [w_x^l, w_y^l]$ and $\mathbf{W_r} = [w_x^r, w_y^r]$). These values are computed virtually based on the filament-based model [77]. When there is no odour sensed, the agent conducts a random searching around the spot (set θ to be -1 or 1 randomly and S_l to be a small value like $5°$). While the odour is sensed (i.e., the concentration value exceeds $T_o^p = 10^{-6}$), the agent compares its left and right odour concentration and turns to the up-wind direction sensed with higher odour concentration. Specially, the desired turning orientation is computed by $m(t) = \tan^{-1}(w_y/w_x)$. Then the step length and rotation are set to be:

$$\begin{cases} \theta(t) = & \text{sign}(m(t) - h(t)) \\ S_l(t) = & \min([5 + |(m(t) - h(t))\%5|, 15]) \end{cases} \tag{3.11}$$

where $h(t)$ is the current heading of the ant robot. Note that the agent will conduct 1 gait loop using the parameters defined in the above equation and then going forward ($S_l = 15$, $\theta = 0$) for 2 gait loop.

Table 3. odour trail following. The threshold $T_d = 5$ and $T_o = 900$ are set empirically according to the lookup table of the *DistanceSensor* used. $K_d = 400$ is a constant scale factor.

sensory state	odour sensed		no odour
	$O_d < T_d$	$O_d \leq T_d$	$O^l < T_o$ & $O^r < T_o$
θ (degree)	0	O_d/k_d	-1 or 1 (random)
S_l (degree)	8	4	3

Table 4. Comparison between different simulation tools

	I2Bot	Hector	NeuroMechFly	NeuroMechFly2	CompoundRay
physics	✓ (ODE)	✓ (ODE)	✓ (PyBullet)	✓ (MuJoCo)	✘
vision	✓ (Binocular/Panoramic)	✘	✘	✓- (Binocular)	✓- (Binocular)
olfactory	✓	✘	✘	✓-	✘
tactile	✓	✓-	✓-	✓	✘
Program Language	✓ (Python/Java/C/C++/Matlab)	✓- (Python)	✓- (Python)	✓- (Python)	✓- (Python)
User Interface	✓	✘	✘	✘	✘

4. Discussion

Computer simulations have proven extremely valuable in neuroethology studies of insect vision (e.g. [80, 81]), brain function (e.g. [28, 29]), and locomotion (e.g. [53, 82]) allowing researchers to verify and adapt their hypotheses efficiently and effectively. Yet, as animal behaviour arises from the interaction of each of the above plus information from internal and external cues, new tools are required that integrate as many of these elements as possible. Moreover, for these tools to be adopted widely across this highly multidisciplinary community they should be robust, easy to install, and easy to use.

As a step towards this goal, we presented I2Bot, an open-source tool for studying the mechanisms of multimodal and embodied insect navigation. We have provided a series of case studies that demonstrate the usability and flexibility of the proposed platform, which has the potential to accelerate research in this fast moving research field. Compared to other simulation tools (see Table 4), Hector [83] and NeuroMechFly [33] primarily focus on locomotion. Additionally, I2Bot provides a flexible visual and olfactory world construction and is simple to extend with other tools. For example, CompoundRay [84] could be integrated with the camera model to generate more realistic insect visual inputs, and make use of more realistic 3D environments (e.g., [85–87]). Regarding the physics engine, both the Open Dynamic Engine (ODE) and MuJoCo are popular in the robotics field, each with its own pros and cons for physics simulation (for detailed comparison, see [88, 89]).

(a) Reinforcement Learning and Swarm Intelligence

I2Bot can also be used as a (deep) reinforcement learning (RL/DRL) tool [90–92], which is becoming a more common tool for insect neuroscience [34, 78, 93] and insect-inspired robotics [53, 94], but requires efficient simulation run the large number of trials involved in the optimisation process. The proposed tool could provide an efficient way to conduct this kind of study. The sensory and body kinematic states of the robot can construct the *State Space*, while the motor commands sent to all available joints form the *Action Space*. Our robot acts as an RL *agent* could dynamically interact with the controllable *environment*. *Reward* is scenario dependent, for example, in the visual navigation context, the distance between the agent and the desired spot could be used to form the *reward* while in the odour plume tracking case, *reward* could be determined by the sensed odour concentration. The proposed I2Bot platform offers a high variety of interaction forms that interest researchers from biology and robotics, facilitating research into the underlying mechanisms of intelligent behaviours and the training and testing of RL-based algorithms.

I2Bot is also suitable for simulating multi-agent systems, making it ideal for swarm intelligence and robotics studies [95, 96]. Similar to previous studies using Webots for swarm robotics simulation [97–99], I2Bot can easily be applied to investigate similar scenarios by adding multiple ant robots (see supplementary Video12 for an example of multiple agents

Table 5. Roadmap for developing a community hub of simulation tools in insect neuroethology

	Now (I2Bot currently can offer)	**Next** (implementable immediately)	**Future** (long-term featured goals)
Sensory	Vision (Binocular and Panoramic); Olfactory (Airborne and chemical); Force sensing (joint torque and tactile)	More bio-realistic vision simulation (e.g., embedded CompoundRay[84]); More precise tactile sensing with touch sensor arrays; Compound odour sensing	Multimodal and real-time sensory
Brain Models	Central Complex steering circuit[28]; Heading direction using ring attractor; Central Complex copy-and-shift[17]	Mushroom bodies model for visual navigation[27]; Models of the insect motor centre:lateral accessory lob(LAL)[105]; Central complex multiple cues integration[17], etc.	Bio-plausible neural models of insect brain
Locomotion	First Body model (desert ant); Robotics gait control	Biomimetic locomotor control(e.g., NeuroWalkNet[51]); Body models of other walking insects (e.g.,stick insect [82]); Embedded other body model framework[106], etc.	Agility of walking, flying and jumping
Environment	3D Visual scenes; Odour plume with wind; Stable odour trails; Uneven terrain and walls	Visual scenes reconstructed from real world; Polarise light simulation; Magnetic simulation; simulating multiple odours simultaneously, etc.	Realistic 3D and interactive environment

simulation). This approach can lead to embodied swarm intelligence, as the body morphology and physical dynamics are integrated into this platform.

(b) Roadmap for a community hub: simulation tools in insect neuroethology

The intention of this study was to demonstrate the opportunities that modern computer simulation tools offer to the field of insect neuroethology, taking the classic desert ant navigation problem as a representative use case. Yet, for I2Bot's maximum value to be realised will require adoption by the community to create a virtuous cycle of usage driving tool-development and vice versa. We outline a roadmap of developments that could take place to enhance the tool to make it usable by the broad community of insect neuroethologists in Table 5. Such extensions will allow researchers to continuously enhance the platform's capabilities and apply it to a broader range of studies in insect navigation and robotics. Our hope is to enable community hub to drive] ever faster research cycles as has been realised through tool sharing in field such as machine-learning (such as FastAI[100] and MLlib[101]), robotics (like ROS [102]) and neuroscience (such as FlyWire [103] and insectBrainDB [104]). To this end, we extend an open invitation to the neuroethology community to become active contributors to this open-source project (`https://github.com/XuelongSun/I2Bot`).

A. Supplementary files

Supplementary files are listed as below:

Fig.S1. The design of I2Bot's morphology is based on real desert ant's measurement.
Fig.S2. Examples of importing previous used visual 3D world [14, 27, 29, 67, 76] into I2Bot environment.
Movie S1. Gait control using forward kinematics.
Movie S2. Gait control on uneven terrain.
Movie S3. Wall climbing using adhesion.
Movie S4. Simulation of path integration.
Movie S5. Visual beacons using monocular left eye.
Movie S6. Visual beacons using monocular right eye.
Movie S7. Visual beacons using binocular vision.
Movie S8. Visual compass (jump mode).
Movie S9. visual compass (incremental mode).
Movie S10. odour trail following (normal trial).
Movie S11. odour trail following (wide trail).
Movie S12. odour trail following with varied antenna length.
Movie S13. odour plume tracking.

Ethics. This work did not require ethical approval from a human subject or animal welfare committee.

Data Accessibility. All the source codes, model of the robot and environments are open-sourced via Github `https://github.com/XuelongSun/I2Bot` under the MIT license.

Authors' Contributions. X. Sun., M. Mangan. and S. Yue. conceived the idea and designed the experiments. X. Sun. developed the tool and conducted the experiments. All authors contributed equally to the writing of the manuscript.

Competing Interests. The authors declare that there is no conflict of interest regarding the publication of this article.

Funding. This research has received funding from the National Natural Science Foundation of China under the Grant No 62206066 and No 12031003.

References

[1] H. J. Chiel and R. D. Beer, "The Brain Has a Body: Adaptive Behavior Emerges from Interactions of Nervous System, Body and Environment", Trends in Neurosciences **20**, 553–557 (1997).

[2] A. Wystrach, "Movements, Embodiment and the Emergence of Decisions. Insights from Insect Navigation", Biochemical and Biophysical Research Communications **564**, 70–77 (2021).

[3] B. Webb, "Can Robots Make Good Models of Biological Behaviour?", Behavioral and Brain Sciences **24**, 1033–1050 (2001).

[4] A. J. Ijspeert, "Biorobotics: Using Robots to Emulate and Investigate Agile Locomotion", Science **346**, 196–203 (2014).

[5] N. Franceschini, "Small Brains, Smart Machines: From Fly Vision to Robot Vision and Back Again", Proceedings of the IEEE **102**, 751–781 (2014).

[6] P.-Y. Oudeyer, "On the Impact of Robotics in Behavioral and Cognitive Sciences: From Insect Navigation to Human Cognitive Development", IEEE Transactions on Autonomous Mental Development **2**, 2–16 (2010).

[7] B. Webb, "Robots with Insect Brains", Science **368**, 244–245 (2020).

[8] P. Manoonpong, L. Patanè, X. Xiong, I. Brodoline, J. Dupeyroux, S. Viollet, P. Arena, and J. R. Serres, "Insect-Inspired Robots: Bridging Biological and Artificial Systems", Sensors **21**, 7609 (2021).

[9] A. Matsiko, "Taking Inspiration from Nature Is a No-Brainer", Science Robotics **8**, eadi2720 (2023).

[10] R. Wehner, "The Cataglyphis Mahrèsienne: 50 Years of Cataglyphis Research at Mahrès", Journal of Comparative Physiology A **205**, 641–659 (2019).

[11] R. Menzel and M. Giurfa, "Cognitive architecture of a mini-brain: the honeybee", Trends in cognitive sciences **5**, 62–71 (2001).

[12] G. C. H. E. De Croon, J. J. G. Dupeyroux, S. B. Fuller, and J. A. R. Marshall, "Insect-Inspired AI for Autonomous Robots", Science Robotics **7**, eabl6334 (2022).

[13] E. Álvarez-Salvado, A. M. Licata, E. G. Connor, M. K. McHugh, B. M. King, N. Stavropoulos, J. D. Victor, J. P. Crimaldi, and K. I. Nagel, "Elementary Sensory-Motor Transformations Underlying Olfactory Navigation in Walking Fruit-Flies", eLife **7**, e37815 (2018).

[14] M. Mangan, "Visual homing in field crickets and desert ants: a comparative behavioural and modelling study", thesis, University of Edinburgh (2011).

[15] M. Schilling, J. Paskarbeit, H. Ritter, A. Schneider, and H. Cruse, "From adaptive locomotion to predictive action selection–cognitive control for a six-legged walker", IEEE Transactions on Robotics **38**, 666–682 (2021).

[16] A. Wystrach, M. Mangan, and B. Webb, "Optimal cue integration in ants", Proceedings of the Royal Society B: Biological Sciences **282**, 20151484 (2015).

[17] X. Sun, S. Yue, and M. Mangan, "How the Insect Central Complex Could Coordinate Multimodal Navigation", eLife **10**, e73077 (2021).

[18] M. Schilling and H. Cruse, "ReaCog, a minimal cognitive controller based on recruitment of reactive systems", Frontiers in Neurorobotics **11**, 3 (2017).

[19] R. Wehner, "Desert Ant Navigation: How Miniature Brains Solve Complex Tasks", Journal of Comparative Physiology A: Sensory, Neural, and Behavioral Physiology **189**, 579–588 (2003).

[20] S. Heinze, "Unraveling the neural basis of insect navigation", Current opinion in insect science **24**, 58–67 (2017).

[21] R. Wehner, T. Hoinville, H. Cruse, and K. Cheng, "Steering intermediate courses: desert ants combine information from various navigational routines", Journal of Comparative Physiology A **202**, 459–472 (2016).

[22] S. Heinze, "Visual Navigation: Ants Lose Track without Mushroom Bodies", Current Biology **30**, R984–R986 (2020).

[23] P. Bregy, S. Sommer, and R. Wehner, "Nest-mark orientation versus vector navigation in desert ants", Journal of Experimental Biology **211**, 1868–1873 (2008).

[24] C. Bühlmann, K. Cheng, and R. Wehner, "Vector-based and landmark-guided navigation in desert ants inhabiting landmark-free and landmark-rich environments", Journal of Experimental Biology **214**, 2845–2853 (2011).

[25] E. L. Legge, A. Wystrach, M. L. Spetch, and K. Cheng, "Combining sky and earth: desert ants (Melophorus bagoti) show weighted integration of celestial and terrestrial cues", Journal of Experimental Biology **217**, 4159–4166 (2014).

[26] M. Collett, "How navigational guidance systems are combined in a desert ant", Current Biology **22**, 927–932 (2012).

[27] P. Ardin, F. Peng, M. Mangan, K. Lagogiannis, and B. Webb, "Using an Insect Mushroom Body Circuit to Encode Route Memory in Complex Natural Environments", PLOS Computational Biology **12**, edited by J. Ayers, e1004683 (2016).

[28] T. Stone, B. Webb, A. Adden, N. B. Weddig, A. Honkanen, R. Templin, W. Wcislo, L. Scimeca, E. Warrant, and S. Heinze, "An Anatomically Constrained Model for Path Integration in the Bee Brain", Current Biology **27**, 3069–3085.e11 (2017).

[29] X. Sun, S. Yue, and M. Mangan, "A decentralised neural model explaining optimal integration of navigational strategies in insects", Elife **9**, e54026 (2020).

[30] D. Lambrinos, R. Möller, T. Labhart, R. Pfeifer, and R. Wehner, "A mobile robot employing insect strategies for navigation", Robotics and Autonomous systems **30**, 39–64 (2000).

[31] J. Dupeyroux, J. R. Serres, and S. Viollet, "AntBot: A six-legged walking robot able to home like desert ants in outdoor environments", Science Robotics **4**, eaau0307 (2019).

[32] G. W. Brodland, "How computational models can help unlock biological systems", in Seminars in cell & developmental biology, Vol. 47 (Elsevier, 2015), pages 62–73.

[33]V. Lobato-Rios, S. T. Ramalingasetty, P. G. Özdil, J. Arreguit, A. J. Ijspeert, and P. Ramdya, "NeuroMechFly, a Neuromechanical Model of Adult Drosophila Melanogaster", Nature Methods **19**, 620–627 (2022).
[34]S. Wang-Chen, V. A. Stimpfling, P. G. Özdil, L. Genoud, F. Hurtak, and P. Ramdya, "NeuroMechFly 2.0, a framework for simulating embodied sensorimotor control in adult Drosophila", bioRxiv, 2023–09 (2023).
[35]R. Vaxenburg, I. Siwanowicz, J. Merel, A. A. Robie, C. Morrow, G. Novati, Z. Stefanidi, G. M. Card, M. B. Reiser, M. M. Botvinick, et al., "Whole-body simulation of realistic fruit fly locomotion with deep reinforcement learning", bioRxiv, 2024–03 (2024).
[36]O. Michel, "Cyberbotics ltd. webots™: professional mobile robot simulation", International Journal of Advanced Robotic Systems **1**, 5 (2004).
[37]Y. Xin, X. Rong, Y. Li, B. Li, and H. Chai, "Movements and balance control of a wheel-leg robot based on uncertainty and disturbance estimation method", IEEE Access **7**, 133265–133273 (2019).
[38]J. Knüsel, A. Crespi, J.-M. Cabelguen, A. J. Ijspeert, and D. Ryczko, "Reproducing five motor behaviors in a salamander robot with virtual muscles and a distributed CPG controller regulated by drive signals and proprioceptive feedback", Frontiers in neurorobotics **14**, 604426 (2020).
[39]P. Ramdya, R. Thandiackal, R. Cherney, T. Asselborn, R. Benton, A. J. Ijspeert, and D. Floreano, "Climbing favours the tripod gait over alternative faster insect gaits", Nature communications **8**, 14494 (2017).
[40]Y. Chen, J. E. Grezmak, N. M. Graf, and K. A. Daltorio, "Sideways Crab-Walking Is Faster and More Efficient than Forward Walking for a Hexapod Robot", Bioinspiration & Biomimetics **17**, 046001 (2022).
[41]J. D. Seelig and V. Jayaraman, "Neural dynamics for landmark orientation and angular path integration", Nature **521**, 186–191 (2015).
[42]P. Graham and K. Cheng, "Ants use the panoramic skyline as a visual cue during navigation", Current Biology **19**, R935–R937 (2009).
[43]R. W. Draft, M. R. McGill, V. Kapoor, and V. N. Murthy, "Carpenter Ants Use Diverse Antennae Sampling Strategies to Track Odor Trails", Journal of Experimental Biology, jeb.185124 (2018).
[44]C. Buehlmann, M. Mangan, and P. Graham, "Multimodal interactions in insect navigation", Animal cognition **23**, 1129–1141 (2020).
[45]N. Claverie, P. Buvat, and J. Casas, "Active sensing in bees through antennal movements is independent of odor molecule", Integrative and Comparative Biology **63**, 315–331 (2023).
[46]V. Dürr, V. Berendes, and M. Strube-Bloss, "Sensorimotor ecology of the insect antenna: Active sampling by a multimodal sensory organ", in *Advances in Insect Physiology*, Vol. 63 (Elsevier, 2022), pages 1–105.
[47]T. Hoinville, N. Harischandra, A. F. Krause, and V. Dürr, "Insect-inspired tactile contour sampling using vibration-based robotic antennae", in Biomimetic and Biohybrid Systems: Third International Conference, Living Machines 2014, Milan, Italy, July 30–August 1, 2014. Proceedings 3 (Springer, 2014), pages 118–129.
[48]M. Schilling, T. Hoinville, J. Schmitz, and H. Cruse, "Walknet, a bio-inspired controller for hexapod walking", Biological cybernetics **107**, 397–419 (2013).
[49]S. Steingrube, M. Timme, F. Wörgötter, and P. Manoonpong, "Self-organized adaptation of a simple neural circuit enables complex robot behaviour", Nature physics **6**, 224–230 (2010).
[50]M. H. Dickinson, C. T. Farley, R. J. Full, M. Koehl, R. Kram, and S. Lehman, "How animals move: an integrative view", science **288**, 100–106 (2000).
[51]M. Schilling and H. Cruse, "neuroWalknet, a controller for hexapod walking allowing for context dependent behavior", PLOS Computational Biology **19**, e1010136 (2023).
[52]E. Arena, P. Arena, and L. Patané, "CPG-based locomotion generation in a Drosophila inspired legged robot", in 2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob) (IEEE, 2012), pages 1341–1346.
[53]M. Schilling and H. Cruse, "Decentralized control of insect walking: A simple neural network explains a wide range of behavioral and neurophysiological results", PLoS computational biology **16**, e1007804 (2020).
[54]R. WEHNER, "The desert ant's navigational toolkit: procedural rather than positional knowledge", Navigation **55**, 101–114 (2008).
[55]S. Heinze, A. Narendra, and A. Cheung, "Principles of Insect Path Integration", Current Biology **28**, R1043–R1058 (2018).
[56]S. S. Kim, H. Rouault, S. Druckmann, and V. Jayaraman, "Ring attractor dynamics in the Drosophila central brain", Science **356**, 849–853 (2017).
[57]B. K. Hulse and V. Jayaraman, "Mechanisms underlying the neural computation of head direction", Annual review of neuroscience **43**, 31–54 (2020).
[58]A. Honkanen, A. Adden, J. Da Silva Freitas, and S. Heinze, "The Insect Central Complex and the Neural Basis of Navigational Strategies", Journal of Experimental Biology **222**, edited by B. El Jundi, A. Kelber, and B. Webb, jeb188854 (2019).
[59]M. Collett, L. Chittka, and T. S. Collett, "Spatial memory in insect navigation", Current Biology **23**, R789–R800 (2013).
[60]C. A. Freas and M. L. Spetch, "Varieties of Visual Navigation in Insects", Animal Cognition **26**, 319–342 (2023).

[61]T. S. Collett and J. Baron, "Biological compasses and the coordinate frame of landmark memories in honeybees", Nature **368**, 137–140 (1994).
[62]C. Buehlmann, B. Wozniak, R. Goulard, B. Webb, P. Graham, and J. E. Niven, "Mushroom Bodies Are Required for Learned Visual Navigation, but Not for Innate Visual Behavior, in Ants", Current Biology **30**, 3438–3443.e2 (2020).
[63]J. Zeil, M. I. Hofmann, and J. S. Chahl, "Catchment areas of panoramic snapshots in outdoor scenes", JOSA A **20**, 450–469 (2003).
[64]A. Wystrach, G. Beugnon, and K. Cheng, "Landmarks or panoramas: what do navigating ants attend to for guidance?", Frontiers in Zoology **8**, 1–11 (2011).
[65]T. S. Collett, "Insect navigation: visual panoramas and the sky compass", Current Biology **18**, R1058–R1061 (2008).
[66]W. Stürzl and R. Möller, "An insect-inspired active vision approach for orientation estimation with panoramic images", in International Work-Conference on the Interplay Between Natural and Artificial Computation (Springer, 2007), pages 61–70.
[67]T. Stone, M. Mangan, A. Wystrach, and B. Webb, "Rotation invariant visual processing for spatial memory in insects", Interface focus **8**, 20180010 (2018).
[68]R. Wehner, K. Cheng, and H. Cruse, "Visual navigation strategies in insects: lessons from desert ants", The new visual neurosciences, 1153–1163 (2014).
[69]R. T. Cardé and M. A. Willis, "Navigational Strategies Used by Insects to Find Distant, Wind-Borne Sources of Odor", Journal of Chemical Ecology **34**, 854–866 (2008).
[70]K. Steck, B. S. Hansson, and M. Knaden, "Smells like home: Desert ants, Cataglyphis fortis, use olfactory landmarks to pinpoint the nest", Frontiers in Zoology **6**, 1–8 (2009).
[71]T. J. Steele, A. J. Lanz, and K. I. Nagel, "Olfactory Navigation in Arthropods", Journal of Comparative Physiology A **209**, 467–488 (2023).
[72]G. Reddy, V. N. Murthy, and M. Vergassola, "Olfactory Sensing and Navigation in Turbulent Environments", Annual Review of Condensed Matter Physics **13**, 191–213 (2022).
[73]M. Traner and B. Raman, "Dynamic walking behavior during odor trail-following in locusts", bioRxiv, 2023–02 (2023).
[74]A. M. Matheson, A. J. Lanz, A. M. Medina, A. M. Licata, T. A. Currier, M. H. Syed, and K. I. Nagel, "A neural circuit for wind-guided olfactory navigation", Nature communications **13**, 4613 (2022).
[75]R. Wehner, "Taxonomie, Funktionsmorphologie und Zoogeographie der saharischen Wüstenameise Cataglyphis fortis (Forel 1902) stat. nov", Senckenbergiana biol **64**, 89–132 (1983).
[76]B. Baddeley, P. Graham, P. Husbands, and A. Philippides, "A model of ant route navigation driven by scene familiarity", PLoS computational biology **8**, e1002336 (2012).
[77]J. A. Farrell, J. Murlis, X. Long, W. Li, and R. T. Cardé, "Filament-based atmospheric dispersion model to achieve short time-scale structure of odor plumes", Environmental fluid mechanics **2**, 143–169 (2002).
[78]S. H. Singh, F. Van Breugel, R. P. N. Rao, and B. W. Brunton, "Emergent Behaviour and Neural Dynamics in Artificial Agents Tracking Odour Plumes", Nature Machine Intelligence **5**, 58–70 (2023).
[79]G. Kowadlo and R. A. Russell, "Robot Odor Localization: A Taxonomy and Survey", The International Journal of Robotics Research **27**, 869–894 (2008).
[80]A. Wystrach, A. Dewar, A. Philippides, and P. Graham, "How do field of view and resolution affect the information content of panoramic scenes for visual navigation? A computational investigation", Journal of Comparative Physiology A **202**, 87–95 (2016).
[81]E. Gkanias, B. Risse, M. Mangan, and B. Webb, "From skylight input to behavioural output: a computational model of the insect polarised light compass", PLoS computational biology **15**, e1007123 (2019).
[82]H. Rosano and B. Webb, "The control of turning in real and simulated stick insects", in International Conference on Simulation of Adaptive Behavior (Springer, 2006), pages 150–161.
[83]V. Dürr, P. P. Arena, H. Cruse, C. J. Dallmann, A. Drimus, T. Hoinville, T. Krause, S. Mátéfi-Tempfli, J. Paskarbeit, L. Patanè, et al., "Integrative biomimetics of autonomous hexapedal locomotion", Frontiers in neurorobotics **13**, 88 (2019).
[84]B. Millward, S. Maddock, and M. Mangan, "CompoundRay, an open-source tool for high-speed and high-fidelity rendering of compound eyes", Elife **11**, e73893 (2022).
[85]B. Risse, M. Mangan, W. Stürzl, and B. Webb, "Software to convert terrestrial LiDAR scans of natural environments into photorealistic meshes", Environmental modelling & software **99**, 88–100 (2018).
[86]T. Murray and J. Zeil, "Quantifying navigational information: the catchment volumes of panoramic snapshots in outdoor scenes", PloS one **12**, e0187226 (2017).
[87]W. Stürzl, N. Böddeker, L. Dittmar, and M. Egelhaaf, "Mimicking honeybee eyes with a 280 field of view catadioptric imaging system", Bioinspiration & biomimetics **5**, 036002 (2010).
[88]T. Erez, Y. Tassa, and E. Todorov, "Simulation Tools for Model-Based Robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX", in 2015 IEEE International Conference on Robotics and Automation (ICRA) (2015), pages 4397–4404.
[89]M. Körber, J. Lange, S. Rediske, S. Steinmann, and R. Glück, *Comparing Popular Simulation Environments in the Scope of Robotics and Reinforcement Learning*, 2021.

[90]M. A. Wiering and M. Van Otterlo, "Reinforcement learning", Adaptation, learning, and optimization **12**, 729 (2012).
[91]Y. Li, "Deep reinforcement learning: An overview", arXiv (2017).
[92]R. S. Sutton, "Reinforcement learning: An introduction", A Bradford Book (2018).
[93]S. Lochner, D. Honerkamp, A. Valada, and A. D. Straw, "Reinforcement learning as a robotics-inspired framework for insect navigation: from spatial representations to neural implementation", Frontiers in Computational Neuroscience **18**, 1460006 (2024).
[94]T. Azayev and K. Zimmerman, "Blind hexapod locomotion in complex terrain with gait adaptation using deep reinforcement learning and classification", Journal of Intelligent & Robotic Systems **99**, 659–671 (2020).
[95]M. Dorigo, M. Birattari, S. Garnier, H. Hamann, M. M. de Oca, C. Solnon, and T. Stützle, "Swarm intelligence", Scholarpedia **2**, 1462 (2007).
[96]M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm Robotics: A Review from the Swarm Engineering Perspective", Swarm Intelligence **7**, 1–41 (2013).
[97]B. Khaldi and F. Cherif, "An Overview of Swarm Robotics: Swarm Intelligence Applied to Multi-robotics", International Journal of Computer Applications **126**, 31–37 (2015).
[98]A. L. Newton, C. L. Nehaniv, and K. Dautenhahn, "The robot in the swarm: An investigation into agent embodiment within virtual robotic swarms", in European Conference on Artificial Life (Springer, 2003), pages 829–838.
[99]M. Raoufi, A. E. Turgut, and F. Arvin, "Self-organized collective motion with a simulated real robot swarm", in Towards Autonomous Robotic Systems: 20th Annual Conference, TAROS 2019, London, UK, July 3–5, 2019, Proceedings, Part I 20 (Springer, 2019), pages 263–274.
[100]J. Howard and S. Gugger, "Fastai: a layered API for deep learning", Information **11**, 108 (2020).
[101]X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, et al., "Mllib: Machine learning in apache spark", Journal of Machine Learning Research **17**, 1–7 (2016).
[102]S. Cousins, B. Gerkey, K. Conley, and W. Garage, "Sharing software with ros [ros topics]", IEEE Robotics & Automation Magazine **17**, 12–14 (2010).
[103]S. Dorkenwald, C. E. McKellar, T. Macrina, N. Kemnitz, K. Lee, R. Lu, J. Wu, S. Popovych, E. Mitchell, B. Nehoran, et al., "FlyWire: online community for whole-brain connectomics", Nature methods **19**, 119–128 (2022).
[104]S. Heinze, B. El Jundi, B. G. Berg, U. Homberg, R. Menzel, K. Pfeiffer, R. Hensgen, F. Zittrell, M. Dacke, E. Warrant, et al., "A unified platform to manage, share, and archive morphological and functional data in insect neuroscience", ELife **10**, e65376 (2021).
[105]A. Adden, T. C. Stewart, B. Webb, and S. Heinze, "A neural model for insect steering applied to olfaction and path integration", Neural computation **34**, 2205–2231 (2022).
[106]J. Arreguit, S. T. Ramalingasetty, and A. Ijspeert, "FARMS: Framework for Animal and Robot Modeling and Simulation", bioRxiv (2023).