








Swarm navigation of cyborg-insects in unknown obstructed soft terrain

Received: 6 June 2024

Accepted: 3 December 2024

Published online: 06 January 2025

 Check for updates

Yang Bai ^{1,2,4}, Phuoc Thanh Tran Ngoc ^{3,4}, Huu Duoc Nguyen^{3,4}, Duc Long Le³, Quang Huy Ha³, Kazuki Kai³, Yu Xiang See To³, Yaosheng Deng ¹, Jie Song ¹, Naoki Wakamiya ¹ ✉, Hirotaka Sato ³ ✉ & Masaki Ogura ^{1,2} ✉

Cyborg insects refer to hybrid robots that integrate living insects with miniature electronic controllers to enable robotic-like programmable control. These creatures exhibit advantages over conventional robots in adaptation to complex terrain and sustained energy efficiency. Nevertheless, there is a lack of literature on the control of multi-cyborg systems. This research gap is due to the difficulty in coordinating the movements of a cyborg system under the presence of insects' inherent individual variability in their reactions to control input. Regarding this issue, we propose a swarm navigation algorithm and verify it under experiments. This research advances swarm robotics by integrating biological organisms with control theory to develop intelligent autonomous systems for real-world applications.

The research on swarm navigation, referring to guiding a collective of agents to traverse an environment together, has captured increasing attention in recent years^{1–6}. By harnessing the collective intelligence of autonomous entities, swarm navigation not only facilitates efficient traversal of unexplored terrains^{7–9} but also extends its utility to various sectors such as logistics^{10,11}, disaster response¹², and agriculture¹³. In logistics, swarm navigation allows us to optimize route planning among multiple vehicles, enhancing transportation efficiency and reducing costs^{10,11}. In disaster response, it enables coordinated efforts among robotic teams to support disaster monitoring and survivor searching¹². In agriculture, it supports precision farming techniques, monitoring crop health, and automating tasks to increase productivity and minimize environmental impact¹³.

However, using conventional robotic platforms such as UAVs and UGVs in swarm navigation has notable drawbacks, including their relatively large size and limited mobility. These drawbacks pose practical challenges, especially in constrained or crowded environments where spatial limitations impede effective task execution. In addition, their ability to operate in large terrains is directly constrained by the battery capacity, which limits endurance and overall efficiency.

This study introduces cyborg insects as a solution to the limitations of traditional robots in swarm navigation. Cyborg insects (Fig. 1a), which combine living insects with miniature electronic controllers, offer several advantages over conventional robots^{14–18}. One of the advantages is their energy efficiency. Unlike robots, which typically rely on power-consuming drive mechanisms such as motors for locomotion, cyborgs exploit the natural mobility of insects and require less energy consumption^{16,18}. Cyborgs can also adapt to complex terrains^{14,15}. They can effortlessly maneuver around obstacles of various shapes and sizes and easily traverse through narrow spaces. Additionally, cyborgs are equipped with sophisticated sensing systems, which enable them to rapidly perceive and respond to the environment¹⁷.

Because of these features, research on cyborgs has recently gained increasing attention. Studies on cyborgs in^{19,20} have mainly demonstrated their functionality in obstacle-free environments. Works in^{21–23} allow a single cyborg to navigate in rather complicated terrain by manually manipulating it along a preplanned path. In addition, the cyborg developed in¹⁸ can navigate to predetermined destinations and autonomously traverse unknown terrain. However, despite the aforementioned achievements in single-cyborg control, to the best of our knowledge, the multi-cyborg navigation problem has not been

¹Graduate School of Information Science and Technology, Osaka University, 1-5 Yamadaoka, Suita 5650871 Osaka, Japan. ²Graduate School of Advanced Science and Engineering, Hiroshima University, 1-3-2 Kagamiyama, Higashi-Hiroshima 7398511 Hiroshima, Japan. ³School of Mechanical & Aerospace Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore. ⁴These authors contributed equally: Yang Bai, Phuoc Thanh Tran Ngoc, and Huu Duoc Nguyen. ✉ e-mail: wakamiya@ist.osaka-u.ac.jp; hirosato@ntu.edu.sg; oguram@hiroshima-u.ac.jp

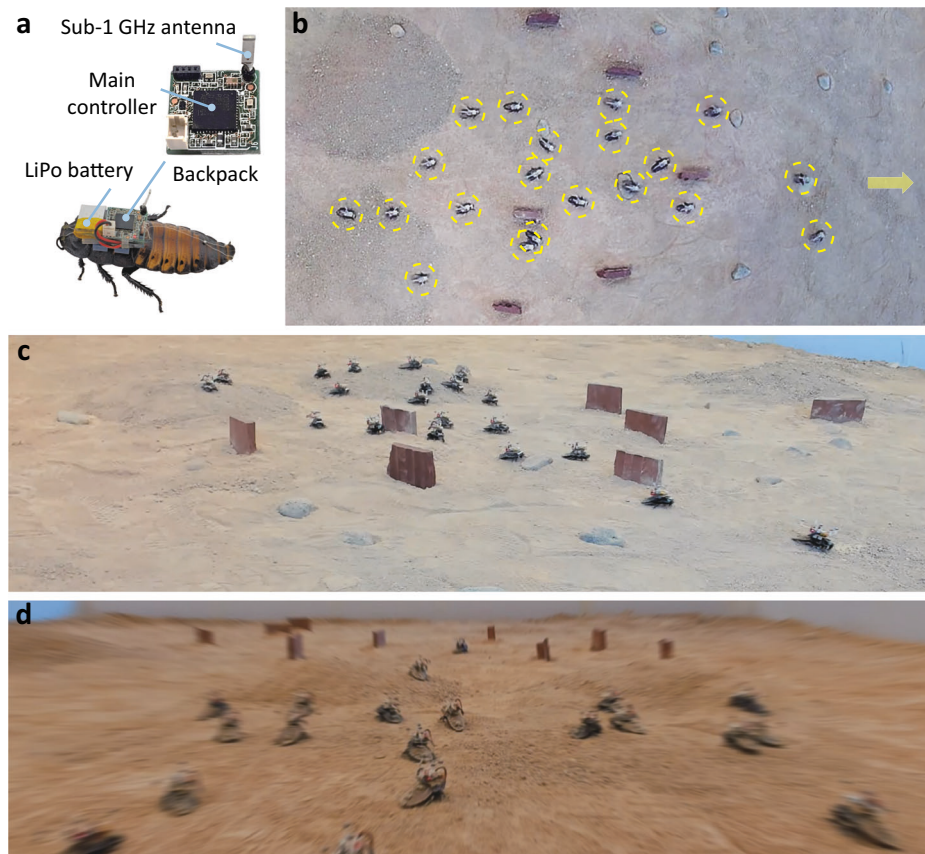


Fig. 1 | Overview of the cyborg swarm navigation. **a** A cyborg insect. In this work, the Madagascar hissing cockroach (*Gromphadorhina portentosa*) was chosen for the cyborg system. A backpack circuit board attached to the insect was designed to house the necessary systems, including the locomotion control system and wireless communication module. It was powered by a rechargeable LiPo battery. How to prepare a cyborg insect is detailed in the supplementary materials. **b** An illustration

of cyborg swarm navigation (top view). A decentralized algorithm was proposed to navigate the cyborg swarm to a designated goal through a sandy area in the presence of hills and obstacles. All the cyborgs have no information about obstacles and hills in the field. **c** The side view and **(d)** the back view of a cyborg swarm. A comprehensive presentation of this work can be found in Supplementary Video 1.

addressed in the literature (an illustration of a cyborg swarm can be seen in Fig. 1b–d).

A multi-agent system presents numerous advantages compared to a single-agent system, such as fault tolerance^{24,25}, distributed problem-solving^{26,27}, and applicability to large-scale problems²⁸. Furthermore, the multi-cyborg system offers unique benefits, particularly in terms of enhanced robustness and intelligence. Regarding robustness, the interactions among the cyborgs in a multi-cyborg system can improve their ability to recover from an overturn because a well-designed swarm control algorithm can allow overturned cyborgs to be surrounded by neighbors that provide support points and aid in their rapid recovery. Additionally, the swarm behavior of a multi-cyborg system contributes to its intelligence. When a cyborg encounters a potentially difficult area, it slows down and attempts to navigate through or around the obstacle. Neighboring cyborgs will maneuver around the slowed cyborg in the same way they would avoid obstacles. Consequently, this maneuvering leads the neighboring cyborgs to bypass the challenging area that caused the slowdown. This cooperative behavior is similar to common human strategies, such as selecting shorter checkout lines in supermarkets for faster service. Furthermore, with appropriately designed swarm control algorithms, cyborgs that bypass the challenging zone can collectively assist the individuals trapped therein, resulting in the passage of the entire swarm.

While multi-cyborg systems offer various advantages over single-cyborg systems, controlling multiple cyborg insects presents many challenges. As partly biological entities, cyborg insects retain natural instincts that limit control precision compared to robots, with each

insect responding differently to the same stimulus. Managing a single cyborg may be feasible despite this variability, but controlling a swarm presents a challenge. Advanced techniques such as adaptive control may be able to drive cyborgs to desired positions, but they often require intensive stimulation. This level of stimulation increases the risk of habituation, where the responsiveness of cyborgs to control signals diminishes over time. The process is believed to involve synaptic changes in the nervous system, where the neurons become less responsive to the stimulus due to decreased neurotransmitter release or receptor sensitivity²⁹. Furthermore, certain innate behaviors of cyborg insects can even hinder the application of conventional swarm control logic. For instance, efforts to use stimulation for separating cyborg insects in close proximity often prove unsuccessful, frequently causing one insect to climb onto another, which results in entanglement. The entanglement presents a potential risk, as it may damage the cyborg's backpack or its connection to the insect, making the cyborg non-functional. Collectively, these issues demonstrate that conventional control methods are not well-suited for managing cyborg swarms.

To address these challenges, we propose a swarm control algorithm specifically tailored for cyborg insects. This algorithm not only adapts to the unique characteristics of cyborg insects but also leverages these traits for effective swarm navigation. It offers improvements over conventional methods in terms of safety, endurance, and adaptability. Specifically, our algorithm harnesses the insects' natural instincts, enabling efficient navigation through complex terrains without the need for precise locomotion control. The intermittent control inputs generated by our algorithm help mitigate

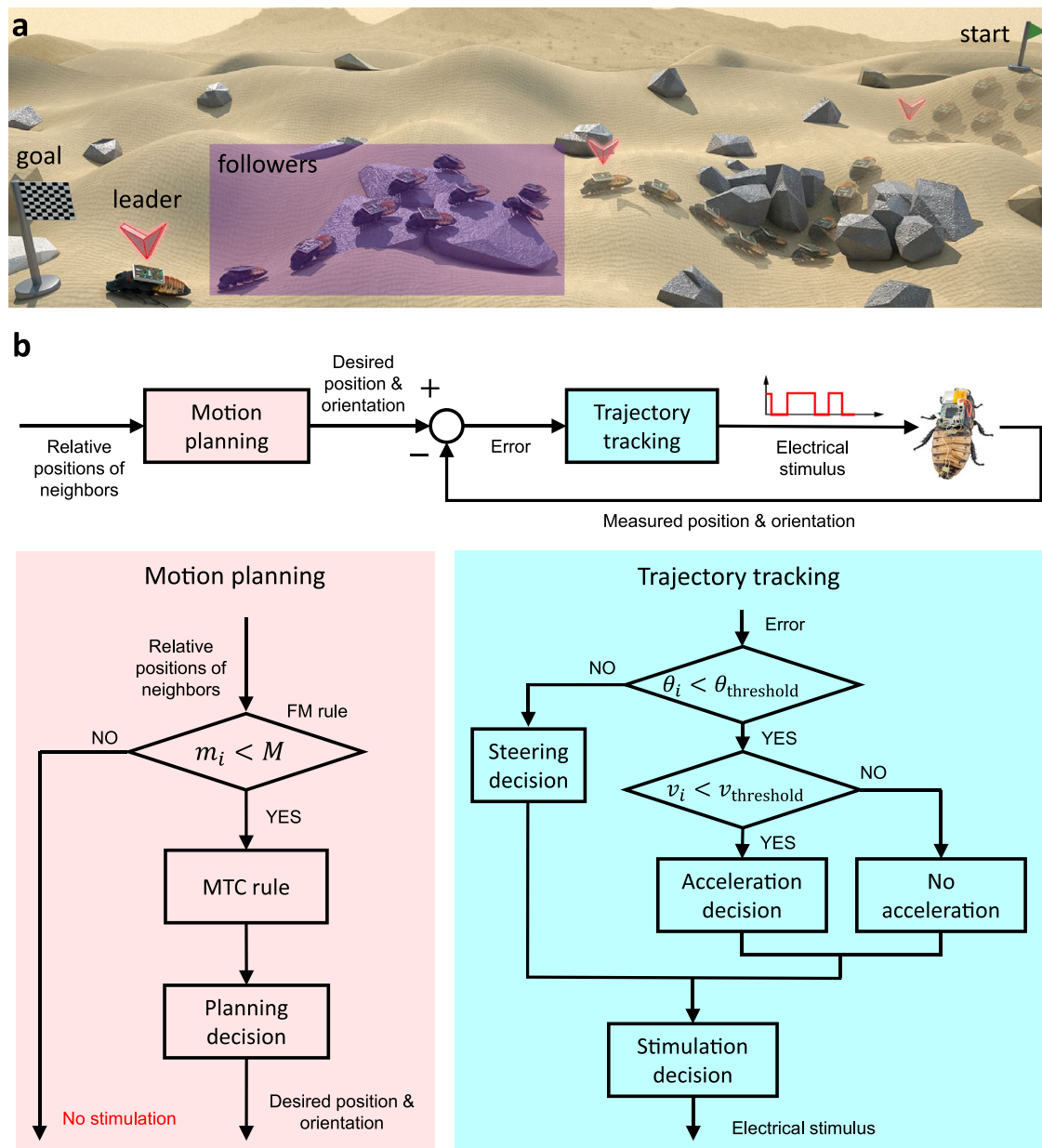


Fig. 2 | Navigation algorithm design for a cyborg swarm. a The cyborg swarm consists of a leader and several followers. Each cyborg can detect neighbors within a limited sensing range, and distinguish the leader and followers. Only the leader is given the position of the goal. **b** An illustration of the proposed navigation algorithm. It consists of two parts: motion planning and trajectory tracking. The motion

planning algorithm provides the desired positions of cyborgs for the next step based on their local information and passes it to the trajectory-tracking algorithm. The trajectory tracking algorithm computes the corresponding amplitude and types of stimulation (left, right, or acceleration) to be applied to the insects.

habituation, thereby extending operational time. Additionally, our control strategy reduces the risk of entanglements among cyborgs, enhancing overall swarm safety. Experimental validation has confirmed the effectiveness of the proposed algorithm. Unlike previous studies that tested control algorithms in idealized settings, such as flat plains, our research demonstrates that both the control algorithm and the design of cyborg insects are adaptive and robust in navigating complex obstructed environments.

Results

Swarm navigation algorithm design

The proposed algorithm in this paper can navigate a swarm of cyborgs from the start to a predetermined goal in an unknown sandy terrain in the presence of obstacles and hills (Fig. 2a). The multi-cyborg system

to be controlled is composed of a leader and several followers, with agents capable of basic directional steering and forward motion. Each agent can detect neighbors within a limited sensing range and distinguish the leader from the followers. Only the leader is given the position of the goal. The proposed algorithm consists of two main components: motion planning and trajectory tracking (Fig. 2b). The motion planning algorithm provides the desired positions of cyborgs for the next time step based on their local information. The trajectory tracking algorithm then receives this information and computes the corresponding amplitude and types of stimulation (left, right, or acceleration) to be applied to the insects.

Our motion planning algorithm was developed based on the observed behavior of tourists who follow a tour leader. Specifically, the algorithm consists of the following two clear and simple rules. The first

rule is the free motion (FM) rule, which grants followers the freedom to move either when the leader is visible or when they remain close to the crowd. Otherwise, they adhere to the move-toward-crowd (MTC) rule, which guides followers toward the leader when visible and toward the direction of the crowd when not visible. Combining these rules, we propose the Tour Group Inspired (TGI) control algorithm. It can be applied to the multi-cyborg system: if the number of neighbors m_i within follower i 's free range (i.e., a circular area around it whose radius is smaller than its sensing range) is less than a threshold M , follower i follows the MTC rule; otherwise, it makes free motions. When the followers calculate m_i , the leader is given a higher weight than an individual follower. As a result, when the leader moves out of follower i 's free range, the follower perceives the loss as equivalent to several neighbors, often falling below the threshold M . This triggers follower i to follow the leader, maintaining group cohesion and preventing the condition that the leader walks away while the group stays behind.

Note that an individual in free motion mode may not actively follow the swarm at all times. However, the control algorithm ensures that if the number of neighbors of a “free” follower falls under the threshold, it will be re-stimulated to rejoin the group. This mechanism allows the individual to temporarily engage in free motion while still maintaining overall cohesion with the swarm. Moreover, this mechanism also mitigates the risk of entanglements that occur when cyborgs are stimulated in close proximity. If a cyborg insect is in an area with a sufficient number of neighboring insects, it is allowed to move freely without stimulation. In this unstimulated state, cyborgs rely on their natural instincts to maneuver around nearby insects, thereby avoiding entanglements.

We then present the details of our trajectory tracking algorithm design. We note that two types of stimulation, steering (left or right) and acceleration, can be applied to the insects. To determine the type of stimulation, the algorithm operates as follows. For each follower i , the algorithm first divides its free range into several sectors. Then, the algorithm selects an arbitrary cyborg from the target sector (the one with the most neighbors) as the target cyborg, and steers the follower toward it. If angle θ_i between the current moving direction of follower i and a line connecting itself with the target cyborg is less than a threshold $\theta_{\text{threshold}}$, steering is not applied, and the algorithm proceeds to the acceleration decision. The rule that governs the acceleration is designed as follows. No acceleration will be applied if follower i 's speed v_i exceeds a certain threshold $v_{\text{threshold}}$. Otherwise, acceleration will be applied, and its magnitude is proportional to the distance between the follower and the target cyborg. This magnitude is bounded by the maximum voltage (2.5V) a cyborg can bear. Details of the control algorithm design are given in the Supplementary Materials.

Our TGI control algorithm capitalizes on the free motions of insects, potentially enhancing efficiency in obstacle and terrain negotiation. Additionally, the utilization of free motions decreases the frequency of electric stimulations on insects. Consequently, it reduces the likelihood of habituation of insects, saves the battery power of the backpack, and, in turn, prolongs the utilization of the cyborgs. Moreover, the FM rule in the TGI algorithm mitigates the impact of insects' individual variability on control performance. Specifically, the FM rule enables free motions of cyborgs in densely packed parts of the swarm, effectively averting entanglements and thereby bolstering the robustness of the multi-cyborg system.

Experimental verification of the proposed algorithm

The feasibility of the proposed swarm navigation algorithm is tested under real-world experiments in a proof-of-concept but challenging scenario. Ten trials of experiments were conducted to demonstrate the reproducibility of experiments.

The experiments used 20 cyborgs as the robot platform in a 3.5m by 3.5m sandy field with rocks and hills (Supplementary Fig. 1c). Among the 20 cyborgs, there is one leader, and the other 19 are the

followers. The leader agent was assigned a designated goal position, whereas the 19 follower agents were only provided the relative positions of their intermediate neighboring agents. As aforementioned, the followers can distinguish between the leader and the followers. All the cyborgs have no information about obstacles and hills in advance. The leader was guided to the target using the algorithm proposed in our previous study¹⁸, while the followers were controlled using the TGI algorithm developed in this work. Details of the experimental setup are given in the Supplementary Materials.

The proposed control strategy has been successfully implemented on the multi-cyborg system, effectively guiding the swarm to the designated goal area. To demonstrate the reproducibility of experiments, ten trials of experiments were conducted. We summarize the corresponding results in Fig. 3. An illustration of the cyborg swarm navigation experiments in unknown obstructed terrain is provided in Fig. 3a. The trajectories of the cyborgs are presented in Fig. 3b, where the leader's path, the followers' trajectories, the positions where cyborgs were stimulated, and the final positions of all 20 cyborgs are presented with a yellow curve, blue curves, red circles, and black dots, respectively. Figure 3c shows the corresponding top view of Exp 1. The results of Exp 2 to 10 are illustrated in Fig. 3d. In Fig. 3d, a noticeable deviation of the leader in Exp 2 is observed, where it appears to jump to a different position and then return to its designated trajectory. This is due to the leader's marker, which is utilized for localization within the VICON motion capture system, becoming contaminated and temporarily untrackable for >5 s. During this period, the control algorithm was unable to monitor the leader's precise position. The following mechanism was designed to mitigate such disruptions. When the leader's marker becomes untrackable, the followers navigate toward the leader's last known position, ensuring group cohesion and preventing the group from dispersing. Once the VICON system reacquires the leader's marker, the control algorithm re-engages, guiding the leader back to its intended trajectory, at which point the follower agents resume real-time tracking.

The degree of autonomy depicted in Fig. 3e illustrates the efficacy of the proposed algorithm in leveraging insects' instincts. In this study, we define the degree of autonomy for the multi-cyborg system as follows: For each experimental trial, we calculate the ratio A_k^l for the leader and average ratio A_k^f for the followers in trial k , between the time without stimulation and the whole experimental period. Then, the degree of autonomy of the total 10 trials of experiments is defined as $\sum_{k=1}^{10} A_k^l / 10$ for the leader, and $\sum_{k=1}^{10} A_k^f / 10$ for the followers. As illustrated in Fig. 3e, the blue bar indicates the degree of autonomy of the followers, while the yellow bar represents the degree of autonomy of the leaders. The error bars denote the standard deviation. Under conventional control, the leader's degree of autonomy averaged 0.26 across 10 experimental trials, while the followers, under the proposed control algorithm, attained an average degree of autonomy of 0.5. Our approach nearly doubles the cyborgs' free-motion time, effectively mitigating the likelihood of habituation of insects, saving the battery power of the backpack, and in turn, prolonging the utilization of the swarm system.

Discussion

During the experiments, several other intriguing phenomena demonstrated the features of the proposed TGI control algorithm:

- The employment of free motions reduces entanglements among cyborgs, thus leading to a higher safety of the multi-cyborg system.
- The interactions among neighbors may facilitate a trapped cyborg to escape a difficult situation.
- Neighbors can help a cyborg recover from an overturn, which, in turn, enhances the robustness of a cyborg swarm.

These features are detailed as follows.

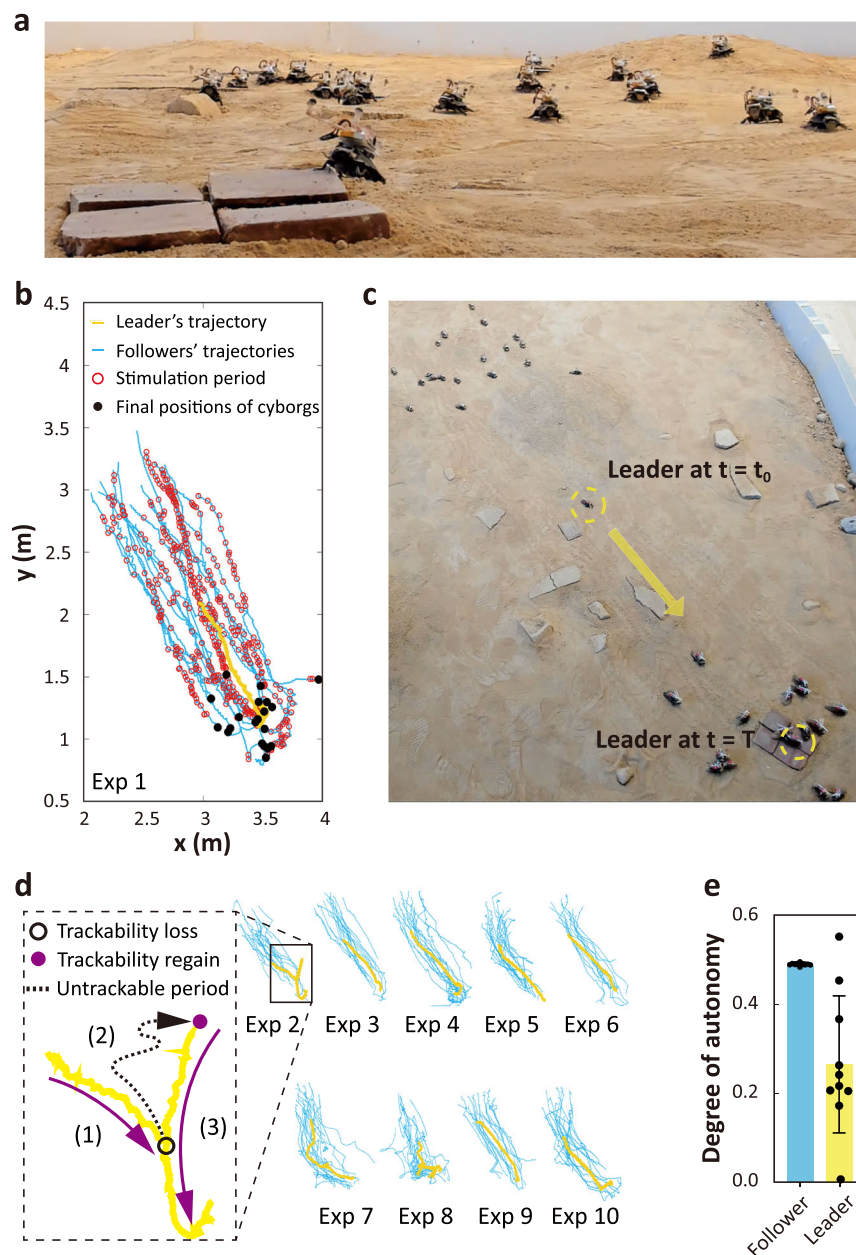


Fig. 3 | Summary of swarm navigation experiments. **a** An illustration of experiments in an obstructed soft terrain (front view). **b** Path plot of Exp 1. **c** The top view of Exp 1. The video of Exp 1 is Supplementary Video 2. **d** Path plots of Exp 2 to 10. In Exp 2, the leader's marker was untrackable for >5s, but the followers maintained group cohesion by moving to the leader's last known position. They resumed tracking the leader after it gained its trackability. **e** The degree of autonomy for the

leader under conventional control (yellow) and the followers under our proposed control algorithm (blue) in 10 trials of experiments. The error bars denote the standard deviation. Our approach nearly doubles the cyborgs' free-motion time, effectively mitigating the likelihood of habituation of insects. Source data are provided as a Source Data file.

First, the TGI control algorithm improves the safety of the multi-cyborg system by reducing entanglement between cyborgs. Our experiments showed that cyborgs instinctively avoid collisions with neighbors during free movement. However, when two cyborgs approached each other too closely and the separation rule from the conventional BOIDS algorithm³⁰ was applied to stimulate the cyborgs to move away from each other, they failed to separate and instead became entangled. Entanglement can lead to undesirable results, such as damage to the cyborgs. This problem was often encountered with the conventional control based on the BOIDS algorithm when neighbors became too close. As depicted in Fig. 4a, the central line within the box indicates the median entanglement

number, while the boundaries represent the 25th and 75th percentiles (lower and upper quartiles). The whiskers extend to the smallest and largest entanglement numbers within 1.5 times the interquartile range from the quartiles. This figure demonstrates that the proposed TGI algorithm reduces entanglements compared to the BOIDS algorithm. Figure 4b depicts entanglements where multiple cyborgs overlap, whereas Fig. 4c shows closely positioned but non-entangling cyborgs. The feasibility of using free motion to prevent entanglement is demonstrated in Fig. 4d and e. When the cyborgs approached each other, the BOIDS method tried to separate them by applying intensive stimulation, which proved to be unsuccessful (Fig. 4d). Conversely, the proposed TGI control exploited the

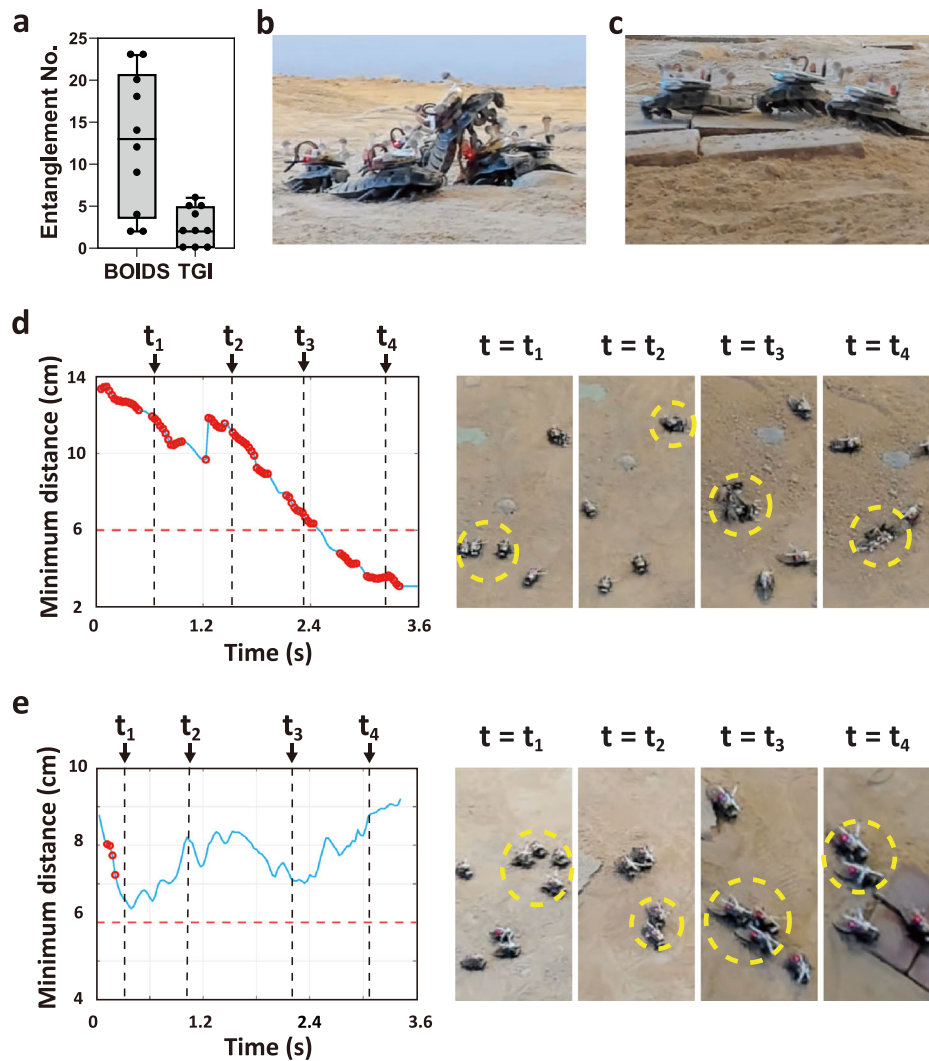


Fig. 4 | Comparison between the experimental results under the conventional BOIDS and the proposed TGI algorithms. a The frequency of entanglements in 10 trials of experiments, respectively, under two control algorithms. The central line within the box indicates the median entanglement number, while the boundaries represent the 25th and 75th percentiles (lower and upper quartiles). The whiskers extend to the smallest and largest entanglement numbers within 1.5 times the interquartile range from the quartiles. The number of entanglements under the proposed TGI algorithm is notably lower than that under the BOIDS algorithm. **b** An illustration of entanglements: when parts of two or more cyborgs overlap with each other. Entanglement can lead to undesirable results, such as damage to the

cyborgs. **c** A group of close but non-entangling cyborgs. **d** The minimum distance among all pairs of cyborgs circled out in experimental snapshots in one trial of experiments conducted with the BOIDS algorithm. Red points indicate instances where at least one of those cyborgs received electrical stimulation. The frequent stimulations on cyborgs that were too close to each other led to entanglements (distance < 6 cm). **e** The minimum distance in one trial of experiments under TGI control. The TGI algorithm leverages insects' instincts to prevent entanglements. The corresponding snapshots reveal that cyborgs can move closely without entanglements under TGI control. The experiment results are shown by Supplementary Video 3. Source data are provided as a Source Data file.

free motion of the cyborgs, taking advantage of their innate behaviors to avoid entanglement (Fig. 4e).

Second, the interactions among neighbors can help a trapped cyborg escape a difficult situation. As shown in Fig. 5a, in an experiment, a cyborg's "Y" shape marker became wedged on the edge of an obstacle. Meanwhile, following the proposed control algorithm, other cyborgs were navigated around the cyborg, which indirectly let the swarm bypass the obstacle. Furthermore, it is noteworthy that, in line with the MTC rule of the proposed control algorithm, cyborgs bypassing challenging zones can assist trapped cyborgs, facilitating the passage of the entire swarm. As illustrated in the snapshots, when the cyborg marked by a yellow circle initially became wedged on an obstacle, it remained unstimulated due to its proximity to the neighbors. However, when the neighbors were bypassing the stuck one ($t = t_2$), they started to "attract" it (stimulations were reapplied to the cyborg), pulling it away from the obstacle.

Third, neighbors can aid a overturned cyborg for more efficient uprighting. Although neighboring cyborgs may navigate around an overturned cyborg in a manner similar to how they avoid obstacles, they also play a passive yet crucial role in facilitating its recovery. The overturned cyborg can utilize interactions with nearby teammates, such as physical contact or grasping, to aid in uprighting itself. This cooperative interaction enhances the overall resilience of the swarm without requiring direct, algorithmic intervention. The bar chart in Fig. 5b illustrates three cases of self-attempted and neighbor-aided recovery from overturns. The discrepancy in recovery performance is particularly evident in Case 3, where a cyborg struggled for 26.5 s before grasping a passing neighbor and then successfully recovered in only 4.5 s. Snapshots of Cases 1 and 3, where cyborgs recovered with the help of neighbors, are also provided in Fig. 5b, respectively. The proposed TGI control algorithm ensures the presence of neighbors around each cyborg and, in turn, enhances the robustness of the multi-cyborg system.

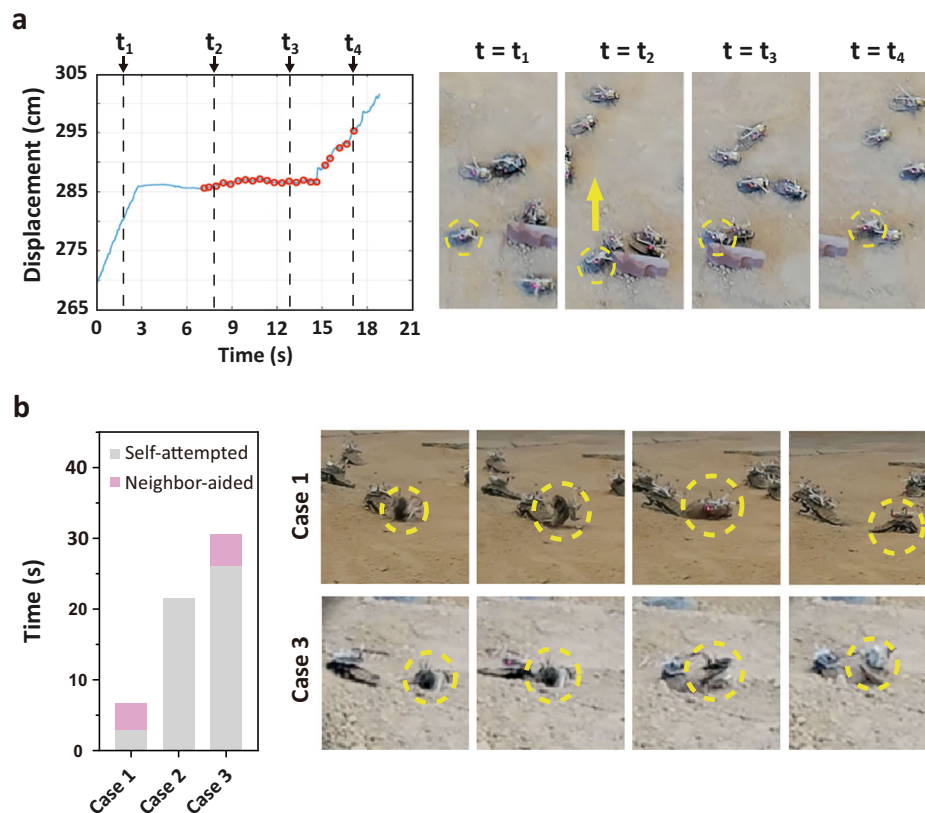


Fig. 5 | The process of an immobilized cyborg being saved by neighbors.

a Entangle and detach from obstacles. A cyborg's "Y" shape marker became wedged on an obstacle ($t = t_2$), and the corresponding displacement almost remained the same during this period. Then the surrounding cyborgs navigated around the trapped one and the obstacle ($t = t_3$). Through our algorithm, the trapped cyborg gradually overcame the obstacle with the "attractive force" from other cyborgs, and the displacement started changing again. The red dots in the displacement plot represent electrical stimulations applied to the cyborgs. Finally, the cyborg detached from the obstacle ($t = t_4$). See Supplementary video 4. **b** Recover from an

overturn. The gray bar in the bar chart denotes the time interval of a self-attempted period for recovery of an overturned cyborg. The bar's bottom and top indicate the start and ending moments. Similarly, the purple bar denotes the time taken for recovery with help from a neighbor. In Cases 1 and 3, the overturned cyborg first attempted and failed to recover by itself for a while, then successfully recovered by grabbing a passing neighbor. The corresponding recovery processes are illustrated by the snapshots. The experiment results are shown by Supplementary Video 5. Source data are provided as a Source Data file.

In summary, our approach offers advantages over conventional control methods in terms of safety, endurance, and adaptability when applied to cyborg swarms. Without relying on precise locomotion control, our algorithm achieves a compact swarm through adjustable parameter design and effectively harnesses insects' instincts for efficient obstacle negotiation in complex, obstructed terrains. Additionally, our algorithm mitigates the risk of habituation and extends operational time by reducing the need for control inputs by an average of 50% compared to the individual control. This reduction is accomplished through the free-motion and move-toward-crowd rules, which leverage the insects' natural motion while maintaining swarm cohesion. Furthermore, our algorithm mitigates the risk of entanglements that can destabilize the swarm and damage the cyborgs, using a switching mechanism between stimulated and free motions based on local swarm compactness. Experiments demonstrate that this design reduces the number of entanglements by over 85% on average compared to the BOIDS method, enhancing the overall safety of the swarm system.

Note that although the control algorithm proposed in this work is theoretically decentralized, the cyborgs in our experiments were provided data from a centralized motion capture system. To achieve decentralized sensing, enabling cyborgs to directly measure distances between each other, we propose a hybrid approach for future work that combines Inertial Measurement Unit (IMU) and Ultra-Wideband (UWB) technologies. Specifically, each cyborg insect will be

equipped with a low-power IMU to measure yaw, pitch, and roll angles. This data will be used to estimate position by analyzing body vibrations during locomotion, with speed derived from step frequency and position calculated through time integration. To mitigate IMU drift, UWB modules will be deployed to periodically update each insect's position using precise distance measurements. This decentralized hybrid IMU-UWB system can provide robust, accurate localization for the cyborgs without a motion capture system. It can be a promising direction for the practical utilization of the proposed control strategy.

Methods

Control algorithm design

Let us define the position of a cyborg by $\mathbf{q}_i(t) \in \mathbb{R}^2$, velocity by $\mathbf{V}_i(t) \in \mathbb{R}^2$, and orientation by a unit vector $\mathbf{o}_i(t) \in \mathbb{R}^2$, at time t . Without loss of generality, we assign $i = 0$ for the leader and $i \neq 0$ for followers. For cyborg i at time t , we define the sets of its neighbors by

$$\mathcal{N}_i(t) = \{j \in \{0, \dots, N\} \setminus \{i\} \mid 0 < \|\mathbf{q}_j(t) - \mathbf{q}_i(t)\| \leq r_s\}, \quad (1)$$

where N is a positive integer and r_s is a positive constant, denoting the radius of the sensing range. Similarly, we define

$$\mathcal{M}_i(t) = \{j \in \{0, \dots, N\} \setminus \{i\} \mid 0 < \|\mathbf{q}_j(t) - \mathbf{q}_i(t)\| \leq r_{\text{free}}\}, \quad (2)$$

where positive constant r_{free} denotes the radius of the free range. Thus

$$m_i = |\mathcal{M}_i|, \quad (3)$$

is the number of neighbors in the free range of cyborg i . Note that $r_{\text{free}} < r_s$.

Based on these definitions, the swarm control algorithm is detailed as follows. It consists of two rules:

- FM rule: for cyborg $i \in \{1, \dots, N\}$ (followers), if the number of neighbors within the free range $m_i > M$ where M is a positive integer, it engages in free motions; otherwise, it follows the rule below.
- MTC rule: for cyborg $i \in \{1, \dots, N\}$, its sensing range is divided into s sectors, and steered to the target sector (the one with the largest number of neighbors where the leader is counted as N cyborgs). On the other hand, the leader $i = 0$ is driven to the goal position. The aforesaid functions are realized by

$$\mathbf{f}_{m,i}(t) = \begin{cases} \mathbf{p}^{\text{goal}}(t) - \mathbf{q}_i(t), & i = 0, \\ \mathbf{p}^{\text{target}}(t) - \mathbf{q}_i(t), & i \in \{1, \dots, N\}, \end{cases} \quad (4)$$

where $\mathbf{p}^{\text{goal}}(t)$ and $\mathbf{p}^{\text{target}}(t)$ denotes the position of a target cyborg (an arbitrary cyborg in the target sector) and the goal for the leader, respectively. Then, the desired orientation of cyborg i can be computed by

$$\mathbf{o}_i^{\text{des}}(t) = \phi(\mathbf{f}_{m,i}(t)), \quad (5)$$

where $\phi: \mathbb{R}^2 \setminus \{\mathbf{0}\} \rightarrow \mathbb{R}^2 \setminus \{\mathbf{0}\}: \mathbf{x} \mapsto \frac{\mathbf{x}}{\|\mathbf{x}\|}$.

Note that the following repulsion function can also be integrated into the MTC rule to achieve a rather sparse swarm: cyborg i is steered away from a neighbor j at time t , according to

$$\mathbf{f}_{r,i}(t) = \begin{cases} \mathbf{0}, & i = 0, \\ -\sum_{j \in \mathcal{N}_i(t)} \frac{1}{\|\mathbf{q}_j(t) - \mathbf{q}_i(t)\|^2} (\mathbf{q}_j(t) - \mathbf{q}_i(t)), & i \in \{1, \dots, N\}. \end{cases} \quad (6)$$

Combining these functions, the desired velocity of follower i is computed according to

$$\mathbf{V}_i^{\text{des}}(t) = k_m \mathbf{f}_{m,i}(t) + k_r \mathbf{f}_{r,i}(t), \quad (7)$$

where k_m, k_r are constant weights. Then, the desired position $\mathbf{q}_i^{\text{des}}(t)$ of cyborg i can be computed by

$$\mathbf{q}_i^{\text{des}}(t) = \mathbf{q}_i(t) + \mathbf{V}_i^{\text{des}}(t) \Delta t, \quad (8)$$

where Δt denotes the time step.

Thus, one defines the position error of cyborg i by $\mathbf{e}_{q,i}(t) = \mathbf{q}_i^{\text{des}}(t) - \mathbf{q}_i(t)$ and the orientation error by $\mathbf{e}_{o,i}(t) = \mathbf{o}_i^{\text{des}}(t) - \mathbf{o}_i(t)$. The angle between $\mathbf{o}_i^{\text{des}}(t)$ and $\mathbf{o}_i(t)$ is denoted by θ_i , and the speed of agent i is represented by v_i . The computed $\mathbf{e}_{q,i}(t)$, $\mathbf{e}_{o,i}(t)$ and the measured θ_i, v_i are passed to the trajectory tracking algorithm summarized in **Algorithm 1** in the Supplementary Information.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The data that support the findings of this study are available within the paper and Supplementary Information files. Source data are provided in this paper. Source data are provided with this paper.

Code availability

The code that supports the findings of this study is available within the Supplementary Information files.

References

1. Yang, L. et al. Autonomous environment-adaptive microrobot swarm navigation enabled by deep learning-based real-time distribution planning. *Nat. Mach. Intell.* **4**, 480–493 (2022).
2. Zhou, X. et al. Swarm of micro flying robots in the wild. *Sci. Rob.* **7**, eabm5954 (2022).
3. Ferrer, E. C., Hardjono, T., Pentland, A. & Dorigo, M. Secure and secret cooperation in robot swarms. *Sci. Rob.* **6**, eabf1538 (2021).
4. Soria, E., Schiano, F. & Floreano, D. Predictive control of aerial swarms in cluttered environments. *Nat. Mach. Intell.* **3**, 545–554 (2021).
5. Jaffe, J. S. et al. A swarm of autonomous miniature underwater robot drifters for exploring submesoscale ocean dynamics. *Nat. Commun.* **8**, 14189 (2017).
6. Berlinger, F., Gauci, M. & Nagpal, R. Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm. *Sci. Rob.* **6**, eabd8668 (2021).
7. Tran, V. P., Garratt, M. A., Kasmarik, K., Anavatti, S. G. & Abpeikar, S. Frontier-led swarming: Robust multi-robot coverage of unknown environments. *Swarm Evol. Comput.* **75**, 101171 (2022).
8. Tang, H., Sun, W., Lin, A., Xue, M. & Zhang, X. A GWO-based multi-robot cooperation method for target searching in unknown environments. *Expert Syst. Appl.* **186**, 115795 (2021).
9. Xue, Y. & Chen, W. Multi-agent deep reinforcement learning for uavs navigation in unknown complex environment. *IEEE Trans. Intell. Veh.* **9**, 2290–2303 (2023).
10. Shibata, K., Jimbo, T. & Matsubara, T. Deep reinforcement learning of event-triggered communication and control for multi-agent cooperative transport. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)* 8671–8677 (IEEE, 2021).
11. Hegde, A. & Ghose, D. Multi-UAV collaborative transportation of payloads with obstacle avoidance. *IEEE Control Syst. Lett.* **6**, 926–931 (2021).
12. Ghamry, K. A., Kamel, M. A. & Zhang, Y. Multiple UAVs in forest fire fighting mission using particle swarm optimization. In *Proc. International Conference on Unmanned Aircraft Systems (ICUAS)* 1404–1409 (IEEE, 2017).
13. Albiero, D., Garcia, A. P., Umez, C. K. & de Paulo, R. L. Swarm robots in mechanized agricultural operations: a review about challenges for research. *Comput. Electron. Agric.* **193**, 106608 (2022).
14. Vo-Doan, T. T., Dung, V. T. & Sato, H. A cyborg insect reveals a function of a muscle in free flight. *Cyborg Bionic Syst.* <https://doi.org/10.34133/2022/9780504> (2022).
15. Nguyen, H. D., Tan, P. Z., Sato, H. & Vo-Doan, T. T. Sideways walking control of a cyborg beetle. *IEEE Trans. Med. Rob. Bionics* **2**, 331–337 (2020).
16. Li, R., Lin, Q., Kai, K., Nguyen, H. D. & Sato, H. A navigation algorithm to enable sustainable control of insect-computer hybrid robot with stimulus signal regulator and habituation-breaking function. *Soft Rob.* **11**, 473–483 (2024).
17. Liu, P., Ma, S., Liu, S., Li, Y. & Li, B. Collaborative turning and jumping control of a cyborg locust via sensory stimulation. In *Proc. International Conference on Electrical, Control and Instrumentation Engineering (ICECIE)* 1–7 (IEEE, 2021).
18. Tran-Ngoc, P. T. et al. Intelligent insect–computer hybrid robot: Installing innate obstacle negotiation and onboard human detection onto cyborg insect. *Adv. Intell. Syst.* **5**, 2200319 (2023).
19. Whitmire, E., Latif, T. & Bozkurt, A. Kinect-based system for automated control of terrestrial insect biobots. In *Proc. Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* 1470–1473 (IEEE, 2013).

20. Nguyen, H. D., Dung, V. T., Sato, H. & Vo-Doan, T. T. Efficient autonomous navigation for terrestrial insect-machine hybrid systems. *Sens. Actuators, B* **376**, 132988 (2023).
 21. Xiong, H., Latif, T., Lobaton, E., Bozkurt, A. & Sichitiu, M. L. Characterization of RSS variability for biobot localization using 802.15.4 Radios. In *Proc. IEEE Topical Conference on Wireless Sensors and Sensor Networks (WiSNet)* 1–3 (IEEE, 2016).
 22. Ariyanto, M. et al. Teleoperated locomotion for biobot between Japan and bangladesh. *Computation* **10**, 179 (2022).
 23. Tsukuda, Y. et al. Calmbots: Exploring possibilities of multiple insects with on-hand devices and flexible controls as creation interfaces. In *Proc. CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI EA)* 1–13 (ACM, 2022).
 24. Ju, Y. et al. Joint secure offloading and resource allocation for vehicular edge computing network: A multi-agent deep reinforcement learning approach. *IEEE Trans. Intell. Transp. Syst.* **24**, 5555–5569 (2023).
 25. Mo, X., Huang, Z., Xing, Y. & Lv, C. Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network. *IEEE Trans. Intell. Transp. Syst.* **23**, 9554–9567 (2022).
 26. Hu, J., Zhang, H., Song, L., Schober, R. & Poor, H. V. Cooperative internet of UAVs: distributed trajectory design by multi-agent deep reinforcement learning. *IEEE Trans. Commun.* **68**, 6807–6821 (2020).
 27. Rubenstein, M., Cornejo, A. & Nagpal, R. Programmable self-assembly in a thousand-robot swarm. *Science* **345**, 795–799 (2014).
 28. Werfel, J., Petersen, K. & Nagpal, R. Designing collective behavior in a termite-inspired robot construction team. *Science* **343**, 754–758 (2014).
 29. Holzer, R. & Shimoyama, I. Locomotion control of a bio-robotic system via electric stimulation. In *Proc. IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)* 1514–1519 (IEEE, 1997).
 30. Reynolds, C. W. Flocks, herds and schools: a distributed behavioral model. In *Proc. Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)* 25–34 (ACM, 1987).
- K.K., and Y.X.S.T. prepared the cyborg insects. Y.B., P.T.T.N., H.D.N., Q.H.H., Y.D., and Y.X.S.T. collected the raw data for the cyborg experiments. Y.B., J.S., and Y.D. conducted the data analysis. Y.B., J.S., P.T.T.N., M.O., N.W., and H.S. contributed to the preparation of the manuscript. N.W., H.S., and M.O. supervised the research.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41467-024-55197-8>.

Correspondence and requests for materials should be addressed to Naoki Wakamiya, Hirotaka Sato or Masaki Ogura.

Peer review information *Nature Communications* thanks Claudio Rossi, and the other, anonymous, reviewers for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025

Acknowledgements

The authors received funding from JST (Moonshot R&D Program, Grant Number JPMJMS223A).

Author contributions

N.W., H.S., and M.O. conceived and designed the research. Y.B., M.O., and N.W. developed the swarm control algorithm. P.T.T.N., H.D.N., D.L.L., and K.K. built the experimental platform. Y.B., P.T.T.N., Q.H.H.,