Brainbots as smart autonomous active particles with programmable motion

M. Noirhomme,[1,*] I. Mammadli,[2,†] N. Vanesse,[1] J. Pande,[3] A.-S. Smith,[2,4] and N. Vandewalle[1]

[1] *GRASP, Institute of Physics B5a, University of Liège, B-4000 Liège, Belgium*
[2] *PULS Group, Department of Physics, FAU Erlangen-Nürnberg, IZNF, 91058 Erlangen, Germany*
[3] *Department of Physical and Natural Sciences, FLAME University, Pune, India*
[4] *Group for Computational Life Sciences, Division of Physical Chemistry, Ruder Boskovic Institute, 10000 Zagreb, Croatia*

(Received 18 November 2024; revised 10 April 2025; accepted 12 May 2025; published 4 June 2025)

We present an innovative robotic device designed to provide controlled motion for studying active matter. Motion is driven by an internal vibrator powered by a small rechargeable battery. The system integrates acoustic and magnetic sensors along with a programmable microcontroller. Unlike conventional vibrobots, the motor induces horizontal vibrations, resulting in cycloidal trajectories that have been characterized and optimized. Portions of these orbits can be utilized to create specific motion patterns. As a proof of concept, we demonstrate how this versatile system can be exploited to develop active particles with varying dynamics, ranging from ballistic motion to run-and-tumble diffusive behavior.

DOI: 10.1103/pv3d-7s54

I. INTRODUCTION

Active matter consists of numerous active particles, each of which consumes energy to generate motion or exert a specific force. Consequently, these systems are inherently out of thermal equilibrium [1]. Unlike thermal systems relaxing toward equilibrium and systems with boundary conditions imposing steady currents, active matter systems break time-reversal symmetry because energy is continually dissipated by the individual constituents. Many examples of active matter are seen in biological systems and span all the scales of living organisms, from bacteria [2,3] to schools of fish [4,5]. Pioneering studies have evidenced the complex behaviors of these living systems: group formation [6,7], swarming and flocking [8–10], and transport in complex landscapes [11], to name a few.

Recent efforts have been directed toward developing synthetic self-propelled particles at different scales. Magnetic or Janus colloids are able to self-assemble into moving arrangements [12], mimicking bacteria and other tiny organisms. Mesoscopic machines can be created along liquid interfaces, imitating the locomotion of mesoscopic organisms like copepods [13] and allowing the exploration of swimming at intermediate Reynolds numbers. Asymmetric inert particles of a few millimeters in length can be excited by a vibrated plate to induce horizontal collective motions leading to vortex formation, and phase separation in crowded environments [14–17]. They are often called vibrobots and their main characteristic is their design of inclined legs, important for transferring vertical vibrations into horizontal motion.

Based on a similar vibration-driven locomotion, various commercial robotic particles are available such as hexbugs [18,19] and kilobots [20]. Hexbugs are equipped with at least one motor with a horizontal axis, inducing vertical vibrations. Their motion has been studied in harmonic traps [18], and in elastic arrays in which they exhibit synchronized motion of crystal-like structures [19]. They have also been used in realizing thermodynamic concepts like the Szilárd engine [21], and in experiments where they change their complex environments to create paths [22]. A few vibrobots have been placed in a single elastic membrane for studying the motility of a single cell passing through an aperture [23]. Multiple synthetic cells have been placed in a confined space for creating a synthetic confluent tissue [24]. This robotic class of synthetic active particles is being studied more and more these days because it is cost effective and allows the design and execution of human-scale experiments.

The key challenge concerning these robotic particles is the control of their motion [25]. Vibrobots primarily exhibit ballistic behavior. For instance, hexbugs undergo straight trajectories until they collide with other particles or the boundaries of the system. However, it can be very useful to have robotic systems capable of executing other kinds of motion, such as diffusive motion. For example,

*Contact author: mnoirhomme@uliege.be
†Contact author: isa.mammadli@fau.de

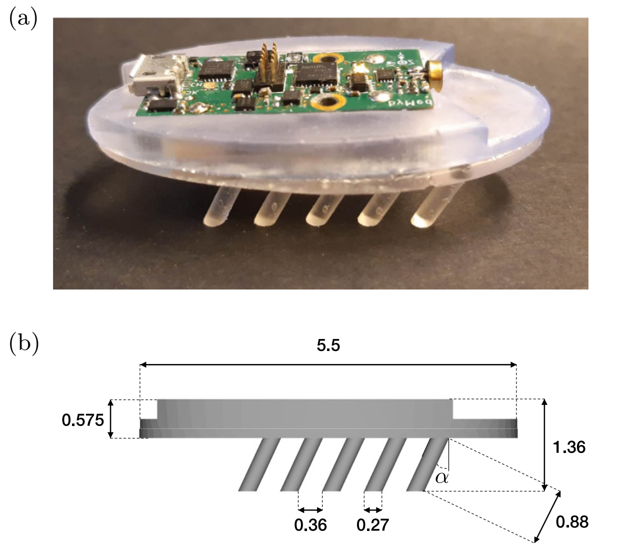

FIG. 1. (a) Picture of the autonomous brainbot. The elliptic body is 5.5-cm long and 3-cm wide. Electronic components are placed on top. (b) Side view of the 3D-printed part of the brainbot, with specific lengths in centimeters.

Brownian-like motion can help in testing physical laws, and run-and-tumble motion, which is intermediate between Brownian and ballistic motion [26], can be valuable for mimicking bacteria and other living organisms. In such motions, reorientation of the particle is essential. Therefore, an autonomous device capable of following controlled motion that includes both translation and rotation would be highly desirable.

In this paper, we address this issue by creating active, programmable vibrobot particles. The main idea is to change the vibrator orientation—and, more precisely, the motor axis—from horizontal to vertical. Horizontal vibrations are expected to induce reorientation of the device. An example of such a bot, which we call a brainbot, is illustrated in Fig. 1. The paper is arranged as follows. Section II presents the technical characteristics of the brainbot. A deep understanding of its multifaceted motion is necessary before it can be employed to execute a desired specific motion. Section III presents a complete study of its locomotion and then explains how to build a specific trajectory, from a ballistic path to a Brownian one.

II. AUTONOMOUS BRAINBOTS

A. Body

A typical 3D-printed body of a brainbot is shown in Fig. 1. In the current paper, the brainbots have five pairs of inclined legs, though it is possible to change their number as well as their tilt angle (denoted by α). The material to 3D-print the body is usually rigid acrylonitrile butadiene styrene (ABS). Another material that we have considered is flexible resin. After several tests of printing brainbot bodies in materials of different rigidity, we chose a rigid body with flexible legs, as this version maximizes the speed of the brainbot. The shape of the 3D-printed body can be modified for creating other kinds of active particles like square or disk particles, allowing for specific assemblies when they collide or pack together. In the present paper, the elliptical shape is fixed.

The leg geometry and the elasticity of the material are key ingredients for optimizing the locomotion. In the next subsections, the motion of the robot has been characterized for different tilt angles α between 5° and 25°. The .stl file of the body can be found in the Supplemental Material [27].

B. Motor

The device is a centimeter-sized structure propelled by vibrations induced by a motor placed inside the particle. The latter is made of an asymmetric mass rotating in the device along the horizontal plane. This asymmetric mass can rotate in either the clockwise (CW) or the counterclockwise (CCW) direction. As a result of the momentum balance, the brainbot rotates correspondingly in the CCW or the CW direction. The geometry of the rotator is cylindrical and it has a diameter of 10 mm and a thickness of 2.7 mm, and a mass of approximately 1 g. It is placed at the front of the robot, while the battery (the heaviest component) is placed at the rear, so as to direct the movement of the particle forward. The motor rotating the asymmetric mass is subjected to a 3-V potential difference of the pulse width modulation (PWM) type, through the modification of an effective voltage V_E. Accordingly, the rotational speed of the motor can be varied from 0 to 11 000 revolutions per minute (rpm).

When the engine is started from a state of rest, it does not reach its maximum rotation speed instantly. In the same way, when the vibrator is running and the motor is braked, there is another delay before it comes back to rest. As explained in Sec. III, these delays have a direct impact on the optimal locomotion of the brainbot.

C. Electronic components

The motor and the sensors are all connected to a programmable microcontroller placed on top of the device as shown in Fig. 1(a). Because it is fully programmable, we have named our device the brainbot.

Figure 2 shows a simplified circuit diagram of the brainbot electronics. The central part consists of an 8-bit microcontroller running dedicated firmware. The system is powered by a button lithium-ion rechargeable battery. The battery starts charging when the brainbot is connected to a micro-USB port. A circuit manages the charging process and also protects the battery against deep discharge. In the basic version of the brainbot, three sensors are connected to the microcontroller:

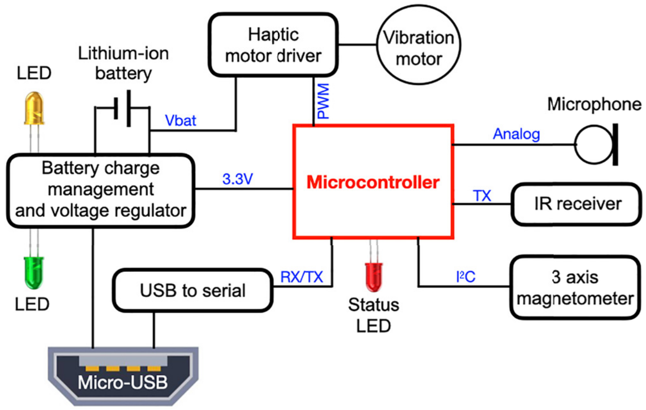

FIG. 2. Simplified circuit diagram of the brainbot electronics.

(a) An infrared (IR) receiver is used to get a signal from a remote control. A common TV remote control can be used to trigger the particle activity.

(b) A directional microphone is placed on the front of the brainbot to analyze the ambient noise. It can, for example, detect the noise produced by other neighboring brainbots.

(c) A three-axis Hall magnetometer is placed on the device to measure the magnetic field strength and orientation. Brainbots can orient along the magnetic field of the Earth or move in a magnetic landscape, as shown in Sec. III D.

Other sensors can be added in the near future, such as light sensors, accelerometers, and inclinometers. The brainbots are programmed in c, like for the usual Arduino platform. In this way, several behaviors can be implemented for piloting the brainbot, such as remote-controlled movement, random displacement, and location tracking.

D. Main characteristics

In Table I we provide the main technical characteristics of the brainbots. Some of these characteristics can be modified by 3D-printing other bodies or extended by adding different sensors.

The brainbot is placed in a rectangular arena on a planar and horizontal surface. A camera placed above tracks the brainbot position $\vec{r}$ and orientation φ by using a homemade OpenCV routine. The position $\vec{r}$ corresponds to the geometrical center of the elliptical particle and is different from the center of mass. The orientation is deduced from the semimajor axis angle φ (see Fig. 3).

III. LOCOMOTION

A. Characterization of spontaneous trajectories

Motion of the brainbot is achieved by triggering the internal motor, whose power, and the resulting intensity of the vibrations, can be controlled through the effective voltage V_E. Under constant V_E, we observe three main kinds of trajectories [blue and orange curves in Figs. 4(a)–4(d)]. At low V_E, a brainbot may spin around a point in its interior while remaining otherwise fixed in position [Figs. 4(a) and 4(b)]. At higher values of V_E, it may spin around a point in its interior while also executing translatory motion [Fig. 4(c)], or, sporadically, it may exhibit purely translatory motion without spinning [Fig. 4(d)]. The spinning component of the motion can be chosen in either the clockwise [Fig. 4(a)] or the counterclockwise [Fig. 4(b)] sense. Videos of typical motions observed can be found in the Supplemental Material [27].

The reason for this effect of V_E on the brainbot motion is as follows. Since most of the mass (due to the battery) of the brainbot is located at the rear, at low V_E the energy transmitted to the body by the vibrator allows the brainbot to jump at the level of its front legs while remaining stuck on its rear legs. By conservation of angular momentum, the brainbot's body rotates in the opposite direction of the

TABLE I. Technical characteristics of a brainbot sorted into different categories.

Bulk	Mass	13 g
Size	Length	5.5 cm
	Width	3 cm
	Height	1.5 cm
Legs	Number	2 × 5
	Inclination, α	5°–25°
Battery	Autonomy	60 min
	Charge duration	80 min
Magnetometer	Sensitivity threshold	50 mT
Microphone	Sensitivity	−42 dB
	Bandwidth	100–10 000 Hz
Motor	PWM	50%–100%
	Effective voltage, V_E	0–3 V
	Rotational speed	0–11 000 rpm

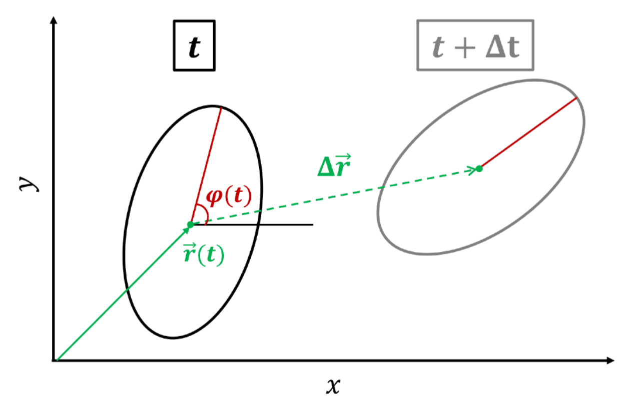

FIG. 3. The elliptical robot in planar motion. At time t, $\vec{r}(t)$ is the position vector of the geometrical center and $\varphi(t)$ is the angle of orientation between the x axis and the major axis of the ellipse.

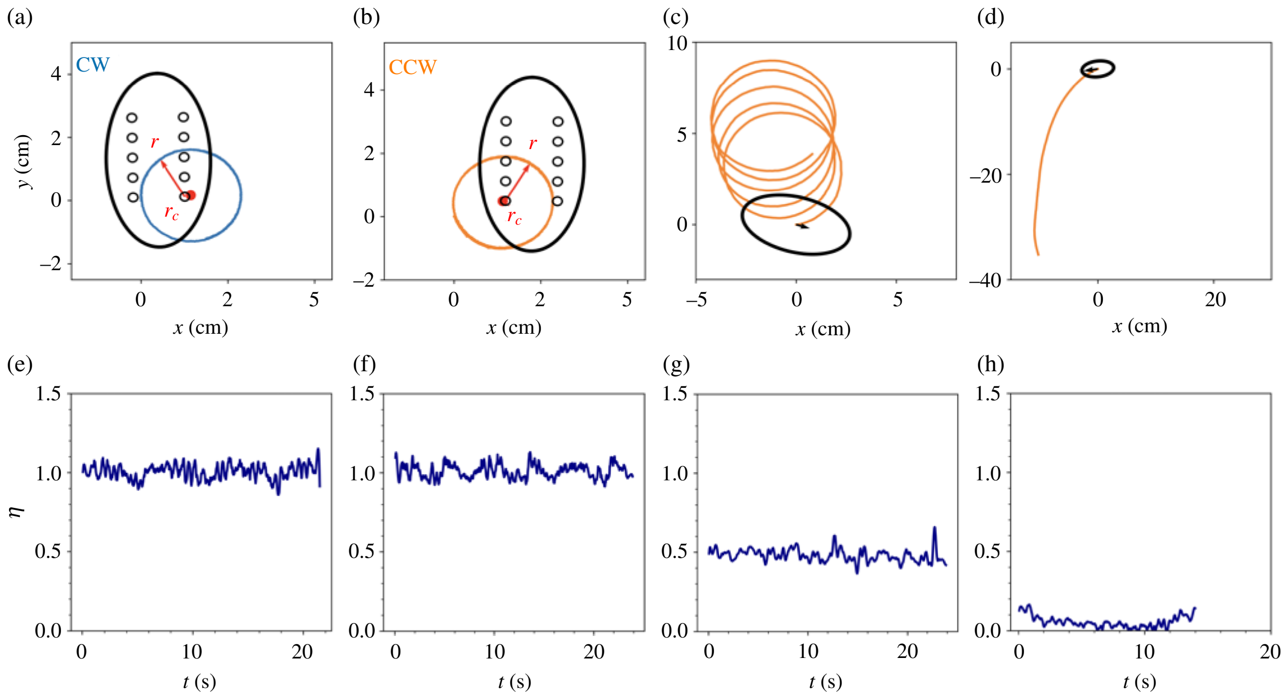

FIG. 4. Brainbot trajectories (a)–(d) with their respective η variation (e)–(h). In panels (a)–(d), the black ellipses indicate the initial position and orientation of the bots, with the small black circles within the ellipses indicating the positions of the legs. The brainbot ellipses are drawn to scale relative to the ensuing trajectories of the center of mass, marked in blue in panel (a) (for clockwise spinning) and orange in panels (b)–(d) (for counterclockwise spinning). Furthermore, in panels (a) and (b) the red dots indicate the instantaneous centers of rotation of the spinning motion, and the red arrows indicate the vector $\vec{r} - \vec{r}_c$.

motor's rotation, and thus the brainbot exhibits primarily spinning motion. At higher V_E, the brainbot still rotates to balance its angular momentum, but it jumps with its whole body since it receives more energy from the vibrator. This results in both spinning and translation.

To characterize the different kinds of motion and the effect of the effective voltage V_E and the inclination angle α more systematically, we first focus our attention on the cases where the brainbot spins around a point in its interior. As stated above, this happens for low values of V_E (similar to the case with "differential-drive-like brushbots" in Ref. [28], though the bots there are circular rather than elliptical).

We use the Savitzky-Golay [29] filter (see Appendix A for details) to clean the trajectories for noise and then calculate the velocity and the angular velocity with numerical derivatives (the trajectory and angle data are sampled at sufficiently high frequencies to allow such derivatives),

$$\vec{v} = \frac{d\vec{r}}{dt} \quad \text{and} \quad \omega = \frac{d\varphi}{dt}. \tag{1}$$

Then the well-known relationship between the two velocities, when the position vector of the instantaneous center of rotation is $\vec{r}_c$, is

$$\vec{v} = \vec{\omega} \times (\vec{r} - \vec{r}_c), \tag{2}$$

where $\vec{v} = (v_x, v_y, 0)^T$ and $\vec{\omega} = (0, 0, \omega)^T$ are the velocity and the angular velocity vectors, respectively. The analysis is limited to the two-dimensional x-y plane.

For purely spinning motion, the center of rotation $\vec{r}_c$ is fixed. Since we know $\vec{r}$, $\vec{v}$, and $\vec{\omega}$ at any given time from the observed trajectories, we can calculate $\vec{r}_c$ from Eq. (2). Figures 4(a) and 4(b) show $\vec{r}_c$ for clockwise and counterclockwise rotational motion, respectively. The results confirm the idea that, for a predominantly spinning motion, the center of rotation [marked as red dots in Figs. 4(a) and 4(b)] falls on one of the rearmost legs. In contrast, in the general case, the instantaneous center of rotation is not necessarily on a rear leg and can lie inside or outside the bot.

The distance between the geometric and rotational centers for purely spinning motion is thus

$$|\vec{r} - \vec{r}_c| = 1.45 \pm 0.3 \text{ cm}. \tag{3}$$

The range of values for $|\vec{r} - \vec{r}_c|$ comes from the fact that the distance between the point where the leg touches the ground and the geometric center of the brainbot depends on the leg angle α (see Appendix B for details). To characterize the other trajectories, we define a new parameter, η, as

$$\eta = \frac{|\vec{r} - \vec{r}_c|\,|\omega|}{|\vec{v}|}. \tag{4}$$

The parameter η is a quantitative measure of how similar the observed motion is to spinning. For purely translatory (linear) motion without any spinning, η should be 0, because ω should be 0 in this case (as φ should be constant), while higher values should indicate a contribution of spinning motion. Because of Eq. (2), for pure spinning motion, η should be around 1.

Figures 4(e)–(h) demonstrate precisely this behavior of η corresponding to the different kinds of trajectories. The value of η fluctuates over time for a given trajectory, generally between 0 and 2. Values close to 1 (in the range [0.8, 1.2]) indicate predominantly spinning motion [Figs. 4(e) and 4(f)] (see Appendix B for details). Intermediate values between 0 and 0.8, which generally occur at high motor voltages, indicate a combination of spinning and linear translatory motion [Fig. 4(g)], and values close to 0 indicate pure translation without spinning [Fig. 4(h)].

We find that the frequency distribution of all η values from all experiments [Fig. 5(a)] is similar for both clockwise (blue) and counterclockwise (orange) spins. In both cases, the most frequent modes are those of pure spin ($0.8 < \eta < 1.2$), and a combination of spinning and translatory motion ($0 < \eta < 0.8$). The third most common mode is that of pure translation without spinning ($\eta \to 0$), but it is observed much less frequently than the other two.

Since the parameter η efficiently characterizes the different kinds of trajectories observed in the experiments, we can capture the effect of the effective motor voltage V_E and the leg inclination angle α on the motion of the brainbot through their effect on the time-averaged value of η for an experiment [Fig. 5(b)]. As expected, for low motor voltages, the motion is dominated by spin (corresponding to η values close to 1), while at higher voltages, translation becomes prominent (with η close to 0). As the leg angle α increases, there is a tendency for the motion to be a combination of the spinning and translation modes, whereas for low α values one or the other of these modes, depending on the effective voltage V_E, tends to dominate.

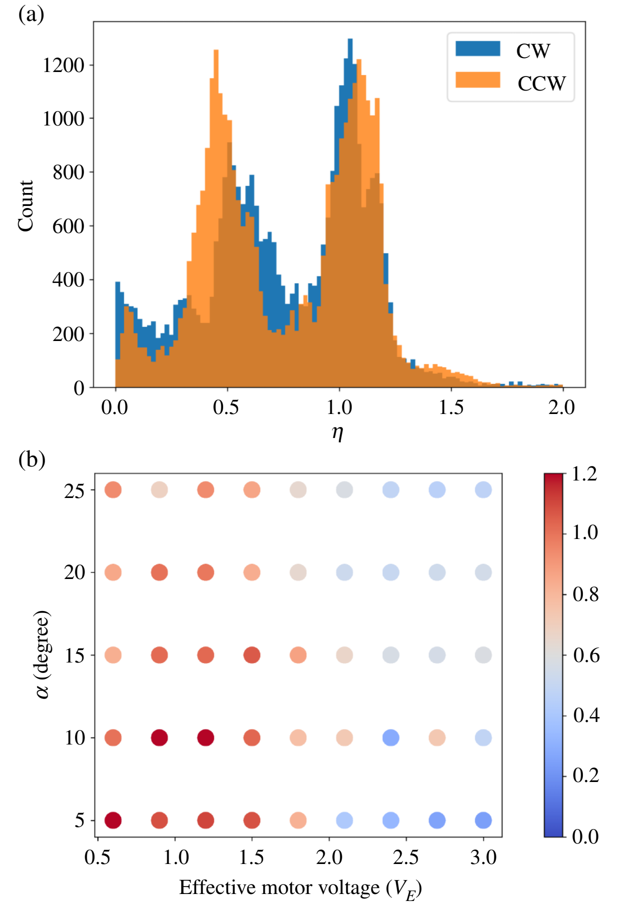

FIG. 5. (a) Frequency distribution of η values across all experiments (in which the effective motor voltage V_E and the leg angle α are varied). The distribution is similar for both clockwise and counterclockwise spins, and is dominated by $\eta \simeq 1.0$ and $\eta \simeq 0.5$, corresponding to pure spinning and a combination of spinning and translation, respectively. (b) Dependence of η on the leg angle α and the effective motor voltage V_E. For low motor voltage, η has values around 1, indicating a motion dominated by spinning. For higher motor voltages, η becomes smaller and smaller in value, indicating that the translatory component of the motion becomes more and more prominent. For small α values, the motion tends to be dominated by either spinning or translation; and when α increases, combinations of spinning and translation become more prominent.

B. Encoding ballistic motion

As discussed in Sec. III A, a brainbot can spontaneously execute straight trajectories if the effective motor voltage V_E is high and the leg angle α is small [Figs. 4(d) and 5(b)]. To obtain straight-line trajectories for a wider range of parameters, and indeed to achieve other diverse kinds of desired motion of the brainbots, one can appropriately stitch together the spontaneous trajectories described in Sec. III A.

For instance, in order to obtain an efficient straight trajectory, successive instances of clockwise and counterclockwise rotations can be performed. Figure 6(a) shows two independent trajectories formed by these alternating rotations, in which brainbots zigzag along an overall single direction. Suppose T denotes the duration of each rotation and β denotes the angle between the chords defining two successive rotations [where these chords are formed by joining the initial and final points of the curve in a single rotation, as shown in Fig. 6(a)]. Then, one can show that the translational mean speed $\bar{v}$ of the robot is

$$\bar{v} = \frac{2R}{T} \sin\left(\frac{\omega T}{2}\right) \sin\left(\frac{\beta}{2}\right), \tag{5}$$

where ω is the rotational speed and R is the radius of each circular trajectory. This speed should be considered as an upper limit, since at each rotation change, there is some delay before the brainbot reaches the maximum angular speed value. A careful observation of the trajectories of Fig. 6(a) reveals such a behavior, since the very short part of each arc has a variable radius of curvature R. We obtain the optimal mean translational speed $\bar{v} \approx 3.5$ cm/s when the duration of each sequence is $T \approx 1.5$ s.

Therefore, ballistic motion can be achieved when the brainbot performs alternating sequences of clockwise and counterclockwise rotations, and it is possible to control the translational speed by adjusting the duration T of these sequences. Videos of the experiments corresponding to the trajectories shown in Fig. 6(a) are available in the Supplemental Material [27].

C. Encoding diffusive motion

We can also make our bots undergo Brownian motion by randomizing the sequences of the spontaneous trajectories. At each step, we program the brainbot to randomly select the rotational direction (clockwise or counterclockwise), with a duration T uniformly drawn from the interval $[0.4, 1.0]$ seconds. Since two consecutive sequences may at times adopt the same rotational direction, long circular arcs can occasionally appear. As illustrated in Fig. 6(b), where two independent trajectories are represented, this type of motion appears diffusive, with trajectories filling the entire accessible space (videos corresponding to these trajectories are available in the Supplemental Material [27]). To confirm this observation, we measured the root-mean-square displacement (RMSD) of the brainbot, defined by

$$D_{\rm rms} = \sqrt{\langle[\vec{r}(t+\tau) - \vec{r}(t)]^2\rangle}, \quad (6)$$

where the angular brackets denote an average over time t for fixed τ values. A linear behavior, $D_{\rm rms} \sim \tau$, is expected for ballistic motion, while a square-root behavior, $D_{\rm rms} \sim \tau^{1/2}$, is expected for diffusive motion. A review of the RMSD method to characterize dynamical behaviors in active matter can be found in Ref. [26].

Figure 6(c) demonstrates the RMSD of our brainbots, in a log-log plot, for two cases: ballistic motion, as described in Sec. III B, and random motion, as defined above. To obtain the second curve, we conducted 12 independent experiments of random motion. A straight line with a slope of 1 is observed for ballistic motion, whereas the random motion exhibits linear behavior at short time scales followed by a square-root dependence at longer time scales. The diffusive behavior is limited at long times due to the finite size of the arena. The characteristic time separating these behaviors is approximately $\tau^\star \approx 0.9$ s, corresponding to the duration required to decorrelate the alternating ballistic sequences.

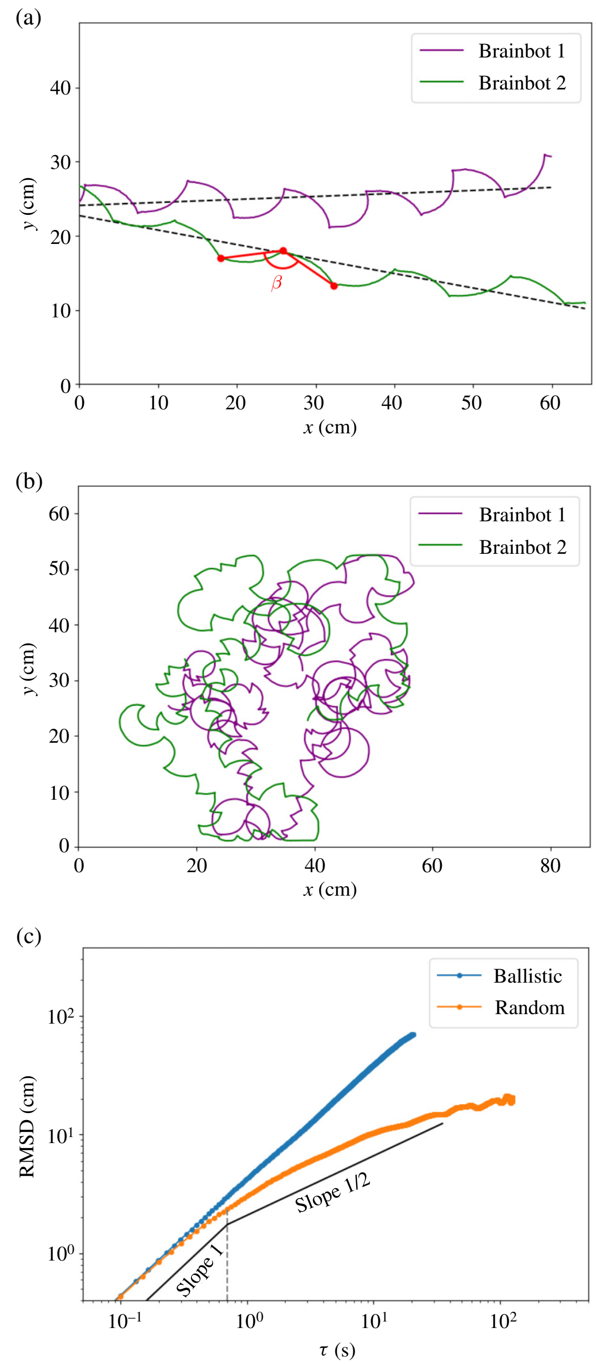

FIG. 6. (a) Two independent trajectories obtained by applying sequences of clockwise and counterclockwise rotations of the asymmetric mass in the internal motor of a brainbot. Dashed lines correspond to approximate ballistic motions. (b) Two independent trajectories obtained by randomizing the application of both clockwise and counterclockwise rotations of the asymmetric mass in the internal motor. (c) Root-mean-square displacement of a brainbot undergoing random motion (orange) or a nearly straight trajectory (blue). The black line corresponds to a double linear fit, showing a separation between a rectilinear-motion-type regime (of slope 1 on the logarithmic scale) and a diffusive-type regime (of slope 1/2). The separation time scale is approximately $\tau^\star \approx 0.9$ s.

D. Encoding interactions

In order to illustrate the capabilities of the device, we present three examples of programmed behaviors below, based on the magnetic sensors in the device and the use of programmed motion.

As a first example, we demonstrate that the orientation of the brainbot can be controlled using the Earth's magnetic field. The magnetic sensor is able to detect weak fields. To do this, the brainbot starts with a calibration procedure. It turns on itself in order to find the maximum and minimum values of the magnetic field components along its own axes (x and y). Taking these values, the brainbot acts as a compass: the robot evaluates the components of the external magnetic field along its axes, denoted B_x and B_y, and then rotates to minimize or maximize their values. When $B_x = \max\{B_x\}$, the brainbot is oriented along its long axis toward the north, while $B_x = \min\{B_x\}$ corresponds to the south. Similarly, to align itself with the east or the west, the brainbot rotates to find $\max\{B_y\}$ or $\min\{B_y\}$, respectively. These four orientations can be selected using the IR remote control.

Figure 7(a) illustrates an experiment conducted on three brainbots that receive the same instructions triggered by the IR controller. The orientation of these three brainbots is plotted as a function of time, and the orders are denoted by vertical red dashed lines. While each robot has its own behavior (trajectory), their overall behavior is similar. However, their alignment dynamics are not exactly the same, leading to a small variability in their selected orientations. The video of this experiment can be found in the Supplemental Material [27]).

As a second example, we show that the magnetic sensor can be utilized for defining interactions between brainbots and other objects. In the following, a brainbot is placed in the arena, above which a strong permanent magnet is fixed at the position $x \approx 35$ cm and $y \approx 35$ cm [denoted by a gray dot in Fig. 7(b)]. The robot executes a random search maximizing the field B following a steepest descent algorithm, wherein each motion is recorded and compared to the previous one for finding the highest value of B. This situation can be considered as a harmonic trap for the bot.

Figure 7(b) shows the trajectory of the brainbot first placed at the black dot, i.e., at $x \approx 40$ cm and $y \approx 25$ cm. As expected, the brainbot moves toward the magnet, but settles at the end on a stable circular orbit around the magnet. The latter is due to the fact that the brainbot moves slowly with a radius of curvature determined by its own motion and geometric characteristics.

In our third example, we demonstrate how attractive interactions between two identical brainbots can be programmed using the same random search algorithm as above. We attach small permanent magnets on the two brainbots, avoiding the saturation of the magnetic sensor. Each brainbot then tries to maximize B, which induces it to move toward the other one.

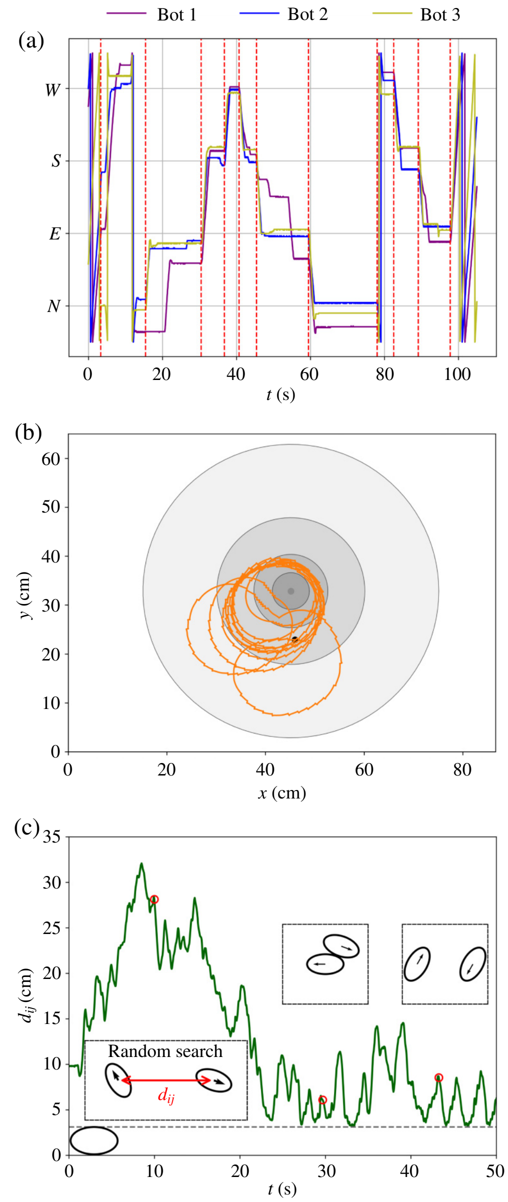

FIG. 7. (a) Orientation of three different brainbots in the Earth's magnetic field as a function of time. The red vertical dashed lines correspond to new instructions sent to the robots via IR remote control. The different brainbots orient themselves in the magnetic field according to the instructions received. (b) A magnet is placed above the arena at a position close to the coordinates (35, 35), denoted by the gray dot. The autonomous brainbot, first placed at the location of the black dot, seeks the maximum magnetic field and is trapped, due to its geometry, forming circular loops around the magnet. (c) Distance d_{ij} separating two brainbots (equipped with a magnet on their back) as a function of time. The brainbots are programmed to move toward the maximum magnetic field. After a transient regime ($t > 20$ s), the brainbots come close together. Their distance no longer exceeds 12 cm, meaning that they behave cooperatively. The gray dashed line represents the minimal distance between two brainbots, which is equal to 3 cm (twice the semiminor axis length of the brainbots). Clearly this happens when they touch while their major axes are oriented parallel to each other.

In Fig. 7(c), the center-to-center distance d_{ij} between two brainbots is plotted as a function of the time t. The two bots are initially placed face-to-face in the middle of the arena. Figure 7(c) shows that, after some transient time (about 20 s), the two brainbots come close enough for each to detect the magnet placed on the other one. Thereafter, they start to move cooperatively such that the distance separating them no longer exceeds $\sim$12 cm. This distance corresponds, more or less, to the radius of the stable orbit observed in the previous experiment. The video of this experiment can be found in the Supplemental Material [27]).

IV. CONCLUSION

In summary, we have developed a versatile and cost-effective robotic platform that can be used to build specific active systems. Our brainbot is fully programmable, and by thoroughly studying its dynamics, we identified the optimal parameters that increase its speed up to 3.5 cm/s. The natural motion tends to be either spinning or a combination of spinning and translation, and we can combine these kinds of motion appropriately to create a wide range of trajectories, from ballistic to diffusive.

The linear and diffusive behaviors can be combined to construct run-and-tumble dynamics commonly observed in bacteria and other active systems [26]. Programming run-and-tumble behavior in autonomous vibrobots represents a significant advancement for active systems, which are typically limited to ballistic motion. This can enable novel experiments to explore fundamental physical laws in active matter, particularly in systems where fluctuations are critical to the overall behavior. Future work can build on the programable capacities of brainbots, by which more complex dynamic patterns of motility can be implemented.

Hence, we now have the capacity to build a range of models for reconstituting motility patterns observed in nature for studying how individual motion gives rise to collective patterns in biological systems. In the future we plan to comprehensively explore the interactions between brainbots and various soft and hard obstacles. The data that support the findings of this article are not publicly available because they are owned by a third party and the terms of use prevent public distribution. The data are available from the authors upon reasonable request.

ACKNOWLEDGMENTS

M.N. thanks the Belgian Federal Science Policy Office (BELSPO) for the provision of financial support in the framework of the PRODEX Programme of the European Space Agency (ESA) under Contract No. 4000103267. The authors thank FNRS for financial support through Grant No. PDR T.0251.20, and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through Project-ID 416229255–SFB 1411 Design of Particulate Products (subprojects A01, C04, and D01). Special thanks go to M. Mélard for his technical support for building brainbots. L. Brennenraedts, N. Martin, and S. Noel are acknowledged for their involvement in this project.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available because they are owned by a third party and the terms of use prevent public distribution. The data are available from the authors upon reasonable request.

APPENDIX A: DATA FILTERING

Using trajectories in their raw forms results in noisy outputs. We cleaned the noise from the trajectories using the Savitzky-Golay filter [29] and obtained smooth derivatives as shown in Fig. 8. The Savitzky-Golay filter is a convolution-type filter that uses polynomial fits in given window sizes. The user is required to define the window size (S) and the degree of the polynomial (N), which results in different levels of accuracy and smoothness. We used $N = 3$ to 5 and $S = 11$ to 31 for the experimental data. Window size S must be an odd number for a well-defined central point [29]. To capture irregular variations (e.g., in angular velocities), we chose $N = 5$ and $S = 11$. Otherwise, we used lower-degree polynomials and larger window sizes. Larger windows apply more smoothing and capture the trend of the data well but can miss rapid changes like peaks. Varying N and S in the mentioned

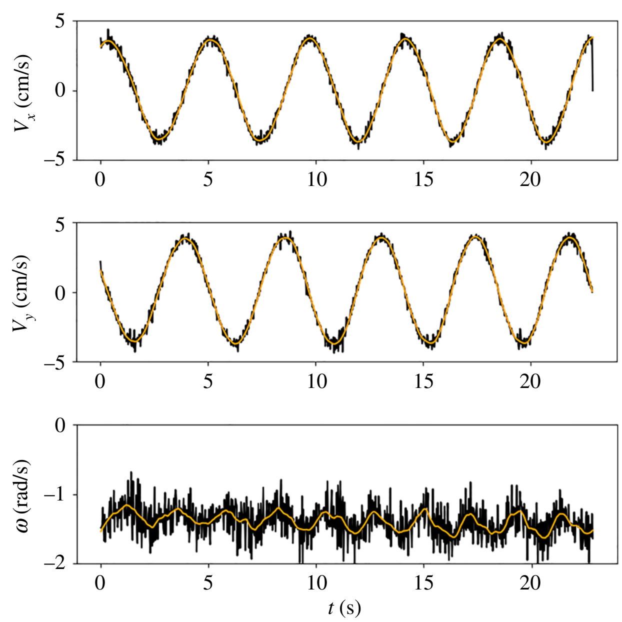

FIG. 8. The black curves show the velocities (V_x, V_y, and ω) when the relevant derivatives were computed directly from the raw data. The orange lines indicate the corresponding derivatives from data filtered using the Savitzky-Golay filter.

intervals, we assigned appropriate values for each dataset to reduce the noise while simultaneously preserving local structure of the data.

```
from scipy.signal import savgol_filter
# Load data arrays - x, y, phi as numpy.array
x_filtered = savgol_filter(x, window_length=11,
                           polyorder=3, deriv=0)

w_filtered = savgol_filter(phi, window_length=31,
                 polyorder=5, deriv=1, delta=0.03)
```

The algorithm is readily available in the *scipy* package [30], so data arrays can be filtered easily as described in the above code snippet. By setting the derivative to 1 and providing the sampling frequency (delta = 0.03 s), we obtained the velocities as well.

APPENDIX B: η FOR DIFFERENT LEG ANGLES

The magnitude of $|\vec{r}-\vec{r}_c|$ varies with different leg angles and depends on whether one takes the positions of the legs where they connect to the bot body ($\alpha = 0^\circ$) or touch the ground ($\alpha = [5^\circ, 25^\circ]$) (Fig. 9). In addition, considering the legs as cylinders of radius 0.135 cm [Fig. 1(b)] changes the value of $|\vec{r}-\vec{r}_c|$ as well. The possible values of $|\vec{r}-\vec{r}_c|$, depending on the leg angle α and including the finite size of the legs, are listed in Table II, second column.

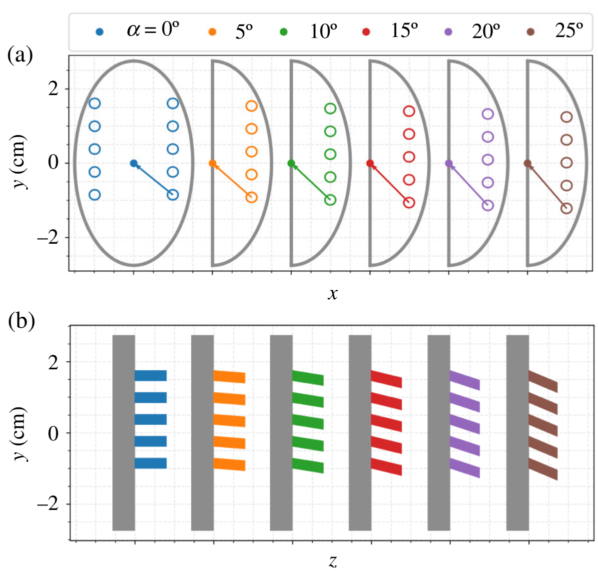

FIG. 9. For different leg inclination angles α: (a) leg locations projected on the ellipsoid body of the brainbots, and (b) side view of the brainbots. The value $\alpha = 0$ corresponds to taking the vector $|\vec{r}-\vec{r}_c|$ to be between the geometric center of the brainbot and the point where the lowermost leg is attached to the bot body. Other values of α correspond to taking the vector between the geometric center and the projection onto the brainbot body of the point where the lowermost leg touches the ground. Note the different y values of the leg locations for different α values. Only half of the bot bodies are plotted for $\alpha > 0$ due to symmetry.

TABLE II. The range of $|\vec{r}-\vec{r}_c|$ values calculated based on the actual design for each leg angle. As a result, η too attains a range defined by $[\eta_{\min}, \eta_{\max}]$.

α (deg.)	$|\vec{r}-\vec{r}_c|_{\text{spin}}$ (cm)	η	$\eta_{\min}$	$\eta_{\max}$
0.0	1.306 ± 0.135	0.90	0.81	0.99
5.0	1.352 ± 0.135	0.93	0.84	1.03
10.0	1.401 ± 0.135	0.97	0.87	1.06
15.0	1.452 ± 0.135	1.00	0.91	1.09
20.0	1.508 ± 0.135	1.04	0.95	1.13
25.0	1.570 ± 0.135	1.08	0.99	1.18

The third to fifth columns of Table II list the corresponding ranges of the values of η for purely spinning motion. We conclude that, instead of the single value $\eta = 1.0$, we must associate the interval $\eta = [0.8, 1.2]$ with spinning motion. This range coincides with $|\vec{r}-\vec{r}_c| = 1.45 \pm 0.3$ cm. Using this range allows one to use the value of η to characterize motion as spinning or otherwise, without considering the details of the leg angles. In case one wishes to know finer details about how η depends on the different leg angles, Table II provides the necessary information.

[1] S. Ramaswamy, The mechanics and statistics of active matter, Annu. Rev. Condens. Matter Phys. **1**, 323 (2010).
[2] H. Wioland, E. Lushi, and R. E. Goldstein, Directed collective motion of bacteria under channel confinement, New J. Phys. **18**, 75002 (2016).
[3] A. P. Petroff, X.-L. Wu, and A. Libchaber, Fast-moving bacteria self-organize into active two-dimensional crystals of rotating cells, Phys. Rev. Lett. **114**, 158102 (2015).
[4] C. Becco, N. Vandewalle, J. Delcourt, and P. Poncin, Experimental evidences of a structural and dynamical transition in fish school, Physica A **367**, 487 (2006).
[5] N. Miller and R. Gerlai, From schooling to shoaling: Patterns of collective motion in zebrafish (*Danio rerio*), PloS One **7**, e48865 (2012).
[6] M. Nagy, Z. Ákos, D. Biro, and T. Vicsek, Hierarchical group dynamics in pigeon flocks, Nature **464**, 890 (2010).
[7] T. Vicsek and A. Zafeiris, Collective motion, Phys. Rep. **517**, 71 (2012).
[8] D. S. Calovi, U. Lopez, S. Ngo, C. Sire, H. Chaté, and G. Theraulaz, Swarming, schooling, milling: Phase diagram of a data-driven fish school model, New J. Phys. **16**, 15026 (2014).
[9] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, Collective memory and spatial sorting in animal groups, J. Theor. Biol. **218**, 1 (2002).
[10] J. Delcourt, N. W. F. Bode, and M. Denoël, Collective vortex behaviors: Diversity, proximate, and ultimate causes of circular animal group movements, Q. Rev. Biol. **91**, 1 (2016).
[11] I. Zuriguel, D. R. Parisi, R. C. Hidalgo, C. Lozano, A. Janda, P. A. Gago, J. P. Peralta, L. M. Ferrer, L. A. Pugnaloni, E. Clément, D. Maza, I. Pagonabarraga, and A. Garcimartín, Clogging transition of many-particle systems flowing through bottlenecks, Sci. Rep. **4**, 7324 (2014).

[12] J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, and P. M. Chaikin, Living crystals of light-activated colloidal surfers, Science **339**, 936 (2013).

[13] M. Hubert, O. Trosman, Y. Collard, A. Sukhov, J. Harting, N. Vandewalle, and A.-S. Smith, Scallop theorem and swimming at the mesoscale, Phys. Rev. Lett. **126**, 224501 (2021).

[14] J. Deseigne, O. Dauchot, and H. Chaté, Collective motion of vibrated polar disks, Phys. Rev. Lett. **105**, 098001 (2010).

[15] C. A. Weber, T. Hanke, J. Deseigne, S. Léonard, O. Dauchot, E. Frey, and H. Chaté, Long-range ordering of vibrated polar disks, Phys. Rev. Lett. **110**, 208001 (2013).

[16] J. Deseigne, S. Léonard, O. Dauchot, and H. Chaté, Vibrated polar disks: Spontaneous motion, binary collisions, and collective dynamics, Soft Matter **8**, 5629 (2012).

[17] C. Scholz, M. Engel, and T. Pöschel, Rotating robots move collectively and self-organize, Nat. Commun. **9**, 931 (2018).

[18] O. Dauchot and V. Démery, Dynamics of a self-propelled particle in a harmonic trap, Phys. Rev. Lett. **122**, 068002 (2019).

[19] P. Baconnier, D. Shohat, C. Hernández López, C. Coulais, V. Démery, G. Düring, and O. Dauchot, Selective and collective actuation in active solids, Nat. Phys. **18**, 1234 (2022).

[20] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal, Kilobot: A low cost robot with scalable operations designed for collective behaviors, Soft Matter **62**, 966 (2014).

[21] O. Chor, A. Sohachi, R. Goerlich, E. Rosen, S. Rahav, and Y. Roichman, Many-body Szilárd engine with giant number fluctuations, Phys. Rev. Res. **5**, 043193 (2023).

[22] A. Altshuler, O. Lauber Bonomo, N. Gorohovsky, S. Marchini, E. Rosen, O. Tal-Friedman, S. Reuveni, and Y. Roichman, Environmental memory facilitates search with home returns, Phys. Rev. Res. **6**, 023255 (2024).

[23] J.-F. Boudet, J. Lintuvuori, C. Lacouture, T. Barois, A. Deblais, K. Xie, S. Cassagnere, B. Tregon, D. B. Brückner, J.-C. Baret, and H. Kellay, From collections of independent, mindless robots to flexible, mobile, and directional superstructures, Sci. Robot. **6**, eabd0272 (2021).

[24] P. Arora, S. Sadhukhan, S. Kumar Nandi, D. Bi, A. K. Sood, and R. Ganapathy, A shape-driven reentrant jamming transition in confluent monolayers of synthetic cell-mimics, Nat. Commun. **15**, 5645 (2024).

[25] S. Paramanick, A. Pal, H. Soni, and N. Kumar, Programming tunable active dynamics in a self-propelled robot, Eur. Phys. J. E **47**, 34 (2024).

[26] C. Bechinger, R. Di Leonardo, H. Löwen, C. Reichhardt, G. Volpe, and G. Volpe, Active particles in complex and crowded environments, Rev. Mod. Phys. **88**, 045006 (2016).

[27] See Supplemental Material at http://link.aps.org/supplemental/10.1103/pv3d-7s54 for the .stl file and the sketch of the electronics, for typical videos of our characterization experiments, the videos of the ballistic and random motions, and the videos of our experiments demonstrating the interaction between the brainbots and an external magnetic field.

[28] G. Notomista, S. Mayya, A. Mazumdar, S. Hutchinson, and M. Egerstedt, in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019* (Institute of Electrical and Electronics Engineers (IEEE), New-York, 2019), p. 5101.

[29] A. Savitzky and M. J. E. Golay, Smoothing and differentiation of data by simplified least squares procedures, Anal. Chem. **36**, 1627 (1964).

[30] P. Virtanen, *et al.*, SciPy 1.0 Contributors, SciPy 1.0: Fundamental algorithms for scientific computing in Python, Nat. Methods **17**, 261 (2020).