Active particles using reinforcement learning to navigate in complex motility landscapes

MACHINE
LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
7 September 2022

REVISED
11 November 2022

ACCEPTED FOR PUBLICATION
30 November 2022

PUBLISHED
16 December 2022

PAPER

Active particles using reinforcement learning to navigate in complex motility landscapes

Paul A Monderkamp[*], **Fabian Jan Schwarzendahl**, **Michael A Klatt** and **Hartmut Löwen**

Institut für Theoretische Physik II: Weiche Materie, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany

[*] Author to whom any correspondence should be addressed.

E-mail: paul.monderkamp@hhu.de

Keywords: active Brownian particles, reinforcement learning, active matter, soft condensed matter, active colloids, Q-learning

Supplementary material for this article is available online

Abstract

As the length scales of the smallest technology continue to advance beyond the micron scale it becomes increasingly important to equip robotic components with the means for intelligent and autonomous decision making with limited information. With the help of a tabular Q-learning algorithm, we design a model for training a microswimmer, to navigate quickly through an environment given by various different scalar motility fields, while receiving a limited amount of local information. We compare the performances of the microswimmer, defined via time of first passage to a target, with performances of suitable reference cases. We show that the strategy obtained with our reinforcement learning model indeed represents an efficient navigation strategy, that outperforms the reference cases. By confronting the swimmer with a variety of unfamiliar environments after the finalised training, we show that the obtained strategy generalises to different classes of random fields.

1. Introduction

Technological advances in producing micron sized swimmers and robots give hope for applications to minimal invasive medicine [1]. The possibilities reach from targeted drug delivery, over material removal in minimal invasive surgery, to telemetric applications where microrobots transmit information that is otherwise hard to obtain. In all of these examples, microrobots need to find a specific target, e.g. the location to which a drug needs to be delivered, or an infected piece of tissue that needs to be surgically extracted. In order to find these targets, usually only local information about the surrounding environment of the robot is given. The robots might need to travel through a complex network of veines or pass through mucus, which makes navigation challenging. Hence, smart navigation strategies for microswimmers need to be found. Here, we develop intelligent strategies, that utilise only limited local information, for microswimmers in a complex motility field by employing reinforcement machine learning techniques.

Recently, machine learning techniques have been applied to active and soft matter systems [2–4]. Specifically, artificial microswimmers, which have been studied intensely [5], might be used for technological applications such as decontamination of polluted water [6], or minimal invasive surgery [1, 7–9]. Active particles have been taught to navigate in different environments, for example optimal paths in force fields [10–12] or flow [13–16] have been computed. Related to the latter, gliders have learned to navigate in a turbulent flow [17, 18] and microswimmers learned a complex flow field [19–24]. In experiments, reinforcement learning has been applied to microswimmers [25] and artificial visual perception has been given to active colloids [26].

The motility of active particles is strongly influenced by the surrounding medium [27], and in particular, viscous landscapes have been studied [28–33], giving rise to viscotaxis. Furthermore, active particles can be steered with an orientation dependent motility [34].

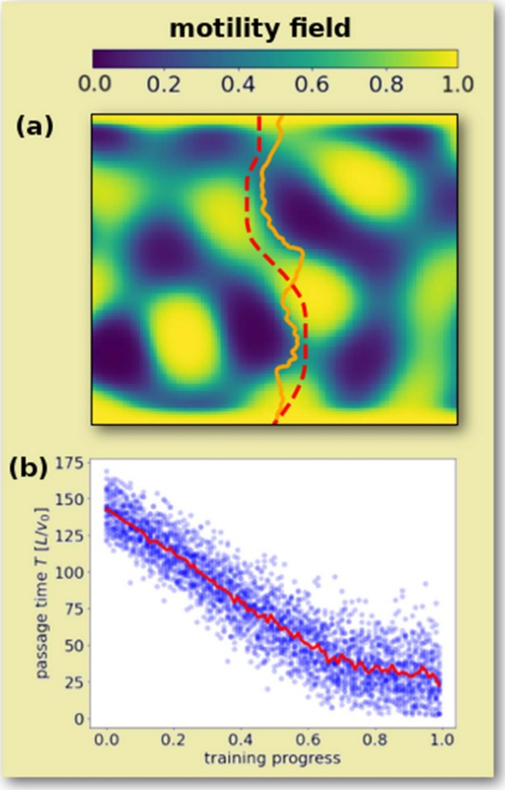


Figure 1. Training an ABP to navigate, with limited local information, through a complex random environment given by a scalar motility field. (a) Typical trajectory after the finalised training procedure (solid orange line) and optimal trajectory obtained with the help of *Dijkstras (shortest path) algorithm* (dashed red line). (b) Average passage time through the simulation box (from bottom to top) at each percent of the training procedure (red line), sampled from 30 independent training procedures. Each blue dot denotes an average over 10^4 trajectories.

In this paper, we teach an active particle that has only local information about its environment to navigate through a complex motility field. A reinforcement learning technique (Q-learning, see section 2.2) that requires a limited amount of data storage is used, making it usable for real life applications. Over the training time, the Q-learning active Brownian particle (QABP) learns to solve different realisations of a random environment, with increasing success (figure 1(b)). At the end of training, the particle outperforms a simple active Brownian particle (ABP), and comes close to the globally optimal path (figure 1(a)) with only local information. Furthermore, once the particle has learned a strategy, we place it in qualitatively different environments, in which it still finds an almost optimal path.

2. Methods

2.1. Equations of motion

We model the swimmer as an overdamped ABP in two dimensions with position $\mathbf{r}(t)$ and orientation $\hat{\mathbf{u}}(t) = (\cos\phi(t), \sin\phi(t))$. It exerts a space-dependent self-propulsion velocity $v_0\mu(\mathbf{r})$ along its orientation. $\mu(\mathbf{r}) \in (0, 1]$ represents the motility field around the particle, such that the particle velocity is bound between 0 and the self-propulsion velocity v_0. In order to perform intelligent navigation, the QABP is capable of either rotating itself with an angular velocity $\omega_Q(\mathbf{r}(t), t) = \pm\,\omega_0$ in either direction or retaining it is orientation such that $\omega_Q(\mathbf{r}(t), t) = 0$. Accordingly, the equations of motion are

$$\dot{\mathbf{r}}(t) = v_0\mu(\boldsymbol{r}(t))\hat{\mathbf{u}}(t), \tag{1}$$

$$\dot{\phi}(t) = \omega_Q(\boldsymbol{r}(t), t) + \sqrt{2D_r}\xi, \tag{2}$$

where ξ represents Gaussian white noise exerted from the solvent environment on the orientation of the particle, with $\langle\xi(t)\rangle = 0$ and $\langle\xi(t)\xi(t')\rangle = \delta(t - t')$.

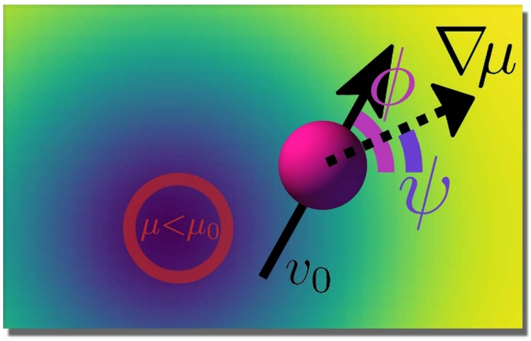


Figure 2. Schematic of the local information, that the QABP receives for the decision making. The swimmer knows the discretised polar angle of its own orientation $\hat{\mathbf{u}} = (\cos(\phi), \sin(\phi))$, and the polar angle ψ of the local gradient $\nabla\mu$ of the motility field. Further, it has information about low motility zones, defined by $\mu < \mu_0$, as indicated by the red circle.

The swimmer moves within a box of side lengths $L_x = 100v_0\tau_Q$ and $L_y = 85v_0\tau_Q$, where τ_Q denotes the characteristic time scale of an intelligent decision (see section 2.2). The reinforcement learning problem is defined by navigating as quickly as possible from the bottom hard wall to the top hard wall of the box. Depending on the specific model of $\mu(\mathbf{r})$, either reflecting or periodic boundary conditions in horizontal direction are used. The QABP and the reference case of the ABP start their trajectories at $\mathbf{r}(t=0)= (0.5L_x, 0)$, oriented upwards ($\hat{\mathbf{u}}(t=0) = (0, 1)$). The swimmers are trained in a motility field that is generated with a Gaussian random wave model (GRW), with wave vectors $\boldsymbol{k}_n$ of wavelength $\|\mathbf{k}_n\| = 20/L_y$ (for details see supplementary material section 1). The GRW gives isotropic non-periodic random waves, a typical example is shown figure 1(a). A remarkable property of this model is an optimal suppression of density fluctuations above a certain wavelength, a property known as stealthy hyperuniformity [35–38]. The use of stealthy hyperuniform models as randomly generated motility fields has the advantage that the formation of large clusters of low motility zones is suppressed. This typically results in the presence of several global paths through the environment without an imminent danger of getting stuck in a dead end at a non-convex low motility zone, which greatly facilitates the learning of the QABP.

2.2. Q-learning algorithm

To enable swimmer navigation within the simulated physical environment, a tabular Q-learning algorithm [39] is superimposed on the Brownian dynamics simulation, giving the QABP the ability for self rotation through the torque expressed by ω_Q. Such an algorithm is characterised by a matrix table $\mathcal{Q}$, which encompasses the strategy and learned experience by the active agent. This matrix functions as a decision matrix, where the rows represent all possible discrete states, in which the swimmer can reside and the columns represent all possible discrete actions. At any time t of action, the swimmer checks its current state i, and performs the action A_i corresponding to the highest value within row i of the matrix:

$$A_i = \arg\max_j \mathcal{Q}_{ij}(t). \tag{3}$$

In our model, the swimmer has information (see figure 2) about its own orientation $\phi(t)$ and the polar angle $\psi(\mathbf{r}(t))$ of the local gradient of the motility field $\nabla\mu(\mathbf{r}(t))$. Furthermore, it knows whether $\mu(\mathbf{r}(t))$ is above or below a threshold value $\mu_0 = 0.25$ (see supplementary material section 4).

The orientational dynamics of the QABP (see equation (2)) are approximated by run and tumble dynamics, where the swimmer tumbles in each integration time step $t_n = n\Delta t$ with a probability $P_{\text{tumble}} = 2D_r\Delta t/(\Delta\phi)^2$, where $\Delta\phi = 2\pi/M_\phi$. The local gradient direction and swimmer orientation are discretised on the unit circle with $M_\phi = M_\psi = 12$ (see supplementary material section 3). All possible combinations of discrete orientations and gradient directions as well as the binary information about the velocity form the complete state space of the Q-learning algorithm. With a given periodicity $\tau_Q = 10\Delta t$, the swimmer takes action, by rotating itself in either direction by $\Delta\phi$, or not rotating, depending on the decision matrix $\mathcal{Q}$. This rotation defines an effective angular velocity $\omega_0 = \Delta\phi/\tau_Q = 2\pi/(N_\phi\tau_Q)$.

In order to obtain a decision matrix, which represents a good strategy for navigating through the complex environment given by $\mu(\mathbf{r})$, $\mathcal{Q}$ is optimised over the course of $N_{epi} = 10^6$ episodes, i.e. trajectories. Before the training procedure, $\mathcal{Q}$ is initialised with zero values. For each episode in the learning phase, the trajectory of the swimmer is simulated until it either reaches the top of the box, swims into a region with

$\mu < 0.5\mu_0 = 0.125$ or the travel time surpasses an upper bound $T_{\max}$, obtained by 100 times the time of a comparable optimal trajectory, obtained with *Dijkstra's algorithm* (see supplementary material section 2). To obtain a navigation strategy, as general as possible, 10^3 realisations of the random environment are used over of $N_{epi} = 10^6$ episodes. The results are sampled, with the trained QABP, on 10^3 new environments.

During training, when an action j is performed, the QABP transitions from state i to i'. Then $\mathcal{Q}$ is updated, following the update formula

$$\mathcal{Q}_{ij}^{new} = \mathcal{Q}_{ij} + \alpha\left(R + \gamma \max_k(Q_{i'k}) - \mathcal{Q}_{ij}\right). \tag{4}$$

Here, $\alpha, \gamma \in [0,1]$ denote hyperparameters of the learning algorithm and R denotes the sum of the specific numeric rewards, that the active agent obtained through performing the current action j. The learning rate α is initialised at 10^{-4} and linearly decreases to 10^{-5} at the end of the training, reinforcing the reliability of $\mathcal{Q}$ with proceeding learning. The term $\gamma \max_j(Q_{i'j})$ incorporates the highest entry in a row from the following state into the current $\mathcal{Q}_{ij}$, estimating the future reward. Since a reasonably different behaviour in neighbouring states is expected across the swimmers state space, $\gamma = 0.3$ is used. During training an ϵ-greedy policy is used. Here, random actions are chosen with probability $\epsilon = 1$ at the beginning of training, then during training ϵ is decreased linearly to 0 such that equation (3) is used for any decision at the end of training (see supplementary material section 5).

In order to navigate efficiently the swimmer is rewarded once it reaches the top of the simulation box. Further, it is punished when it enters a low motility region, or if its displacement is very small (for reward details supplementary material section 4.).

3. Results

To give an intuition on the development of the strategy, three characteristic trajectories from different stages of the learning process are shown in figure 3. For visual reference, each panel additionally shows a globally optimal trajectory obtained via *Dijkstra's algorithm*. In figure 3(a) we show a trajectory from an episode early in the training procedure. Since the QABP has yet to learn about its environment, the probability ϵ to perform a random rotation in either direction is close to 1. Accordingly, the trajectory is similar to that of a common ABP. Due to the indecisiveness of the QABP at this explorative stage, the episode terminates eventually by entering a low motility zone.

Figure 3(b) depicts a trajectory from an episode halfway through the training procedure. The corresponding probability to perform random actions ϵ is approximately 0.45. The trajectory shows randomness, through rotational diffusion as well as random active rotation. Despite the fact that more than half of the actions are randomly chosen, it is visible, how the QABP displays noticeable competence of avoiding the regions with $\mu \ll 1$ to reach the finish line. Finally, the trajectory in figure 3(c) shows the dynamics of the QABP after the learning procedure when $\epsilon = 0$. The QABP swims decisively in vertical direction, such that the trajectory exhibits little dents, thereby maneuvering around the low motility zone in its path.

3.1. Quantitative performance

Optimising the navigation through the environment, given by the motility field $\mu(\mathbf{r})$, relies on the balancing between two opposing principles: minimising the length of the path while simultaneously maximising the instantaneous velocity $v_0\mu(\mathbf{r})$ [40, 41]. Formally, this problem is solved by the solution that minimises the following functional

$$T[\mathbf{c}] = \int_{\mathbf{c}_0}^{\mathbf{c}_1} \frac{||\dot{\mathbf{c}}(t)||}{v_0\mu(\mathbf{c}(t))}\mathrm{d}t, \tag{5}$$

which is the passage time T from the starting point $\mathbf{c}_0$ to any point on the finish line $\mathbf{c}_1$. Here, $\boldsymbol{c}(t)$ is a curve through the environment, parameterised by t. A convenient figure of merit for the performance of the swimmer along any trajectory $\mathbf{r}(t)$ is defined as

$$\frac{v_y}{v_0} := \frac{L_y}{T[\mathbf{r}(t)]v_0}. \tag{6}$$

This quantity is in the interval $(0,1]$ for any trajectory. After each independent training procedure, the resulting $\mathcal{Q}$ is tested in 10^3 trajectories on every independent realisation $\mu(\mathbf{r})$ respectively. Performance data

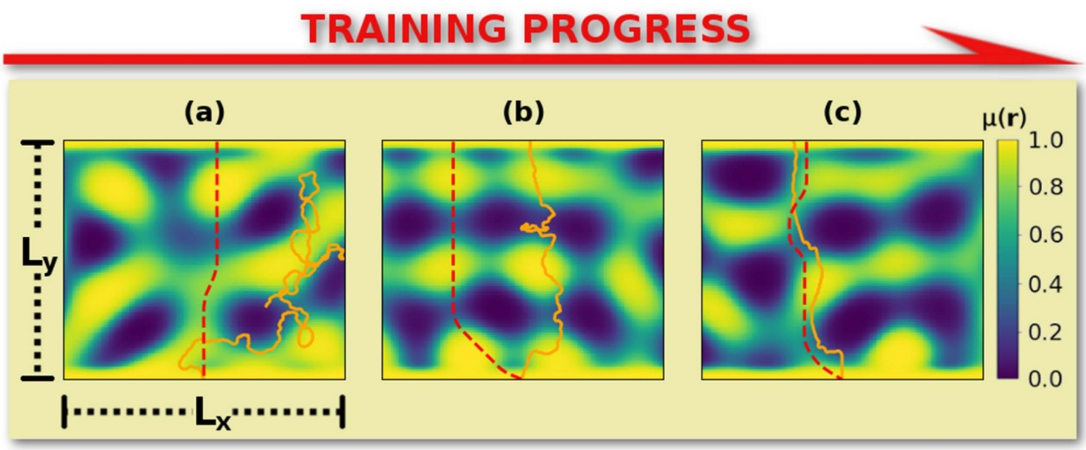


Figure 3. Three typical trajectories (orange solid lines) in three motility fields $\mu(\mathbf{r})$ over the course of the training procedure of the QABP. The particle's objective is to cross the box from bottom to top as fast as possible. The environments are generated with the help of modified isotropic Gaussian random waves. The probablities to perform random actions are (a) $\epsilon = 1.0$, (b) $\epsilon \approx 0.58$, (c) $\epsilon \approx 0$. The red dashed line highlight the optimal trajectories obtained with *Dijkstras algorithm* (see supplementary material section 2). Motility fields are generated as modified isotropic Gaussian random waves and in the horizontal direction reflecting boundary conditions are employed.

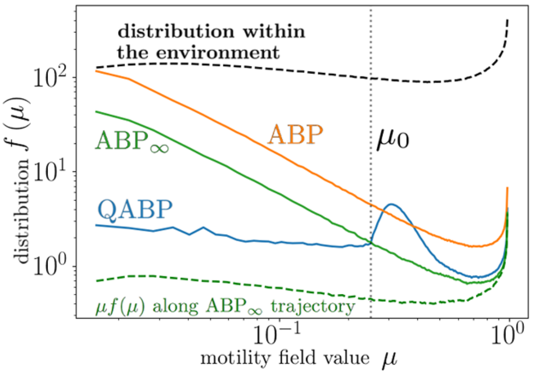


Figure 4. Distributions of scalar values $\mu(\mathbf{r}(t)) = \|\dot{\mathbf{r}}\|/v_0$ along trajectories with $\langle\mu\rangle \approx 0.557$ and $\mathcal{P}_{\mathrm{rot}} \approx 46$, sampled from 15 simulations with 1000 trajectories each. The lines indicate ABP (solid orange), straight trajectory, i.e. ABP_∞ (solid green), and intelligent swimmer (QABP, solid blue). The dashed black line shows the distribution of the $\mu(\mathbf{r})$-values within the environment, averaged from 1000 independent $\mu(\mathbf{r})$. The dashed green line corresponds to $\mu f(\mu)$ of the ABP_∞ trajectory. The dotted vertical line denotes the velocity state threshold $\mu_0 = 0.25$.

was additionally gathered from multiple independent training procedures for each set a parameters. The general performance of the resulting strategy, encoded in $\mathcal{Q}$, is determined by averaging over the 10^3 trajectories, giving $\langle v_y\rangle/v_0$. The performance will depend on the average motility $\langle\mu\rangle$ and the rotational Péclet number, which is defined as

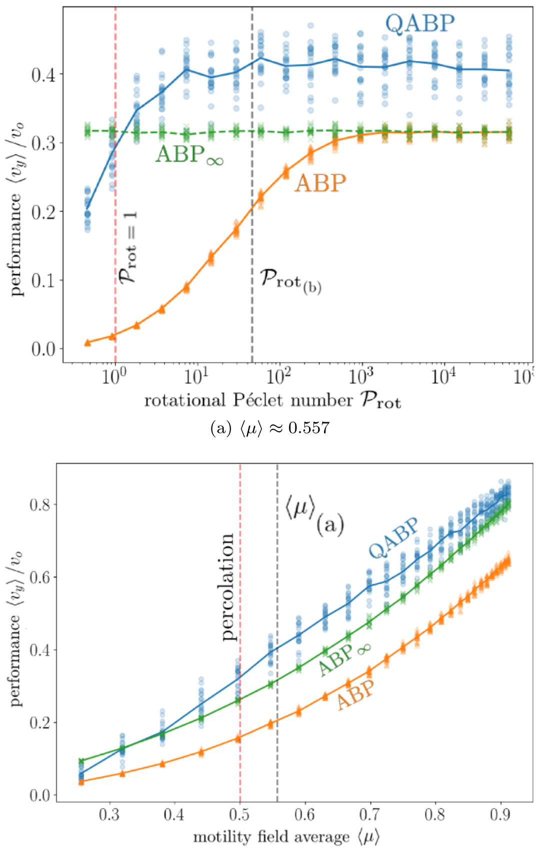
$$\mathcal{P}_{\mathrm{rot}} = (\Delta\phi)^2/2D_{\mathrm{rot}}\tau_Q \tag{7}$$

comparing the typical time scale of the rotational diffusion to that of the intelligent active rotation.

The total passage time T (see equation (5)) can be calculated from the non-normalised distribution of velocities $f(\mu)$, sampled with a time step of τ_Q, along a given trajectory as

$$T = \tau_Q \int_0^1 f(\mu)d\mu. \tag{8}$$

To quantify the behaviour of the individual swimmers, the frequencies $f(\mu)$ of the scalar values of $\mu(\mathbf{r})$ are shown in figure 4. The data is averaged from 15 independent simulation runs with 10^3 trajectories each. Additionally $\mu f(\mu)$ is shown for the ABP, for a straight line (i.e. an ABP in the limit of vanishing diffusion denoted by ABP_∞), and for the distribution of function values $\mu(\mathbf{r})$ within the environment, averaged from 10^3 independent motility fields. More specifically, the distribution of the ABP_∞ trajectory (green solid line)



(a) $\langle \mu \rangle \approx 0.557$

(b) $\mathcal{P}_{\rm rot} \approx 46$

Figure 5. Performances $\langle v_y \rangle / v_0$ of the QABP as well as an ABP and a particle swimming in a straight line without diffusion (ABP$_\infty$) as reference swimmers. The scattered data data points indicate a simulation with 10^3 trajectories each. The solid lines denote the averages over the respective ten data points per parameter set. (a) shows the performance for varying rotational Péclet number $\mathcal{P}_{\rm rot}$ and (b) for varying environment $\langle \mu \rangle$.

counts the frequencies of function values of $\mu(\mathbf{r})$ along a vertical line, sampled in time. Due to the longer retention time at slower velocities, we obtain the distribution in space, sampled at constant distances, as $\mu f(\mu)$. Since the data is averaged over 1.5×10^4 different environments, the corresponding $\mu f(\mu)$ (green dashed line) is proportional to the distribution of the values in the total field (black dashed line). On the other hand, the ABP$_\infty$ reflects the limit of the ABP, for vanishing diffusion. As visible in the plot, the respective curves are proportional to each other, emphasising that the performances of the ABP is tangible through consideration of the motility field alone. Observing the QABP case, it is visible that the distribution in the lower velocities is approximately constant and lies orders of magnitude below both ABP cases. Instead, $f(\mu)$ displays a significant peak above the velocity state threshold indicated by the vertical line at $\mu_0 = 0.25$. This exemplifies how the navigation strategy of the QABP relies on circumvention of the low motility zones, through higher motility regions, thereby elongating the trajectory, but saving time through the faster swimming.

The performances $\langle v_y \rangle / v_0$ of the QABP, and the two reference cases (ABP, ABP$_\infty$) are shown as a function of $\mathcal{P}_{\rm rot}$ in figure 5(a), where $\langle \mu \rangle \approx 0.557$ is chosen. For large $\mathcal{P}_{\rm rot}$, the $\langle v_y \rangle / v_0$ of the ABP approaches the performance of the ABP$_\infty$. More specifically, the ABP is bounded by the ABP$_\infty$, and $\langle v_y \rangle / v_0$ is monotonous in $\mathcal{P}_{\rm rot}$. For small $\mathcal{P}_{\rm rot}$, the performance of the ABP approaches 0. For almost the whole parameter range the QABP is faster than both reference cases. Additionally $\langle v_y \rangle / v_0$ seems to be independent from $\mathcal{P}_{\rm rot}$ for $\mathcal{P}_{\rm rot} \gtrsim 1$. This demonstrates the QABP's ability to steer against the kicks from rotational diffusion. Only for $\mathcal{P}_{\rm rot} \lesssim 1$, the QABP loses its ability to correct for rotational noise, and hence the performance declines with decreasing values of $\mathcal{P}_{\rm rot}$. Figure 5(b) shows the respective performances as a function the environment parameter $\langle \mu \rangle$ for a constant $\mathcal{P}_{\rm rot} \approx 46$. For all three cases, $\langle v_y \rangle / v_0$ increases with μ. Once again, the QABP surpasses the ABP across all $\langle \mu \rangle$, and the ABP$_\infty$ for most of the shown parameter range. For $\langle \mu \rangle \approx 1$, the performances $\langle v_y \rangle / v_0$ of both QABP and ABP$_\infty$ approach 1, since the optimal trajectory becomes a straight line. For $\langle \mu \rangle = 0.5$, the motility field $\mu(\mathbf{r})$ displays a percolation transition of

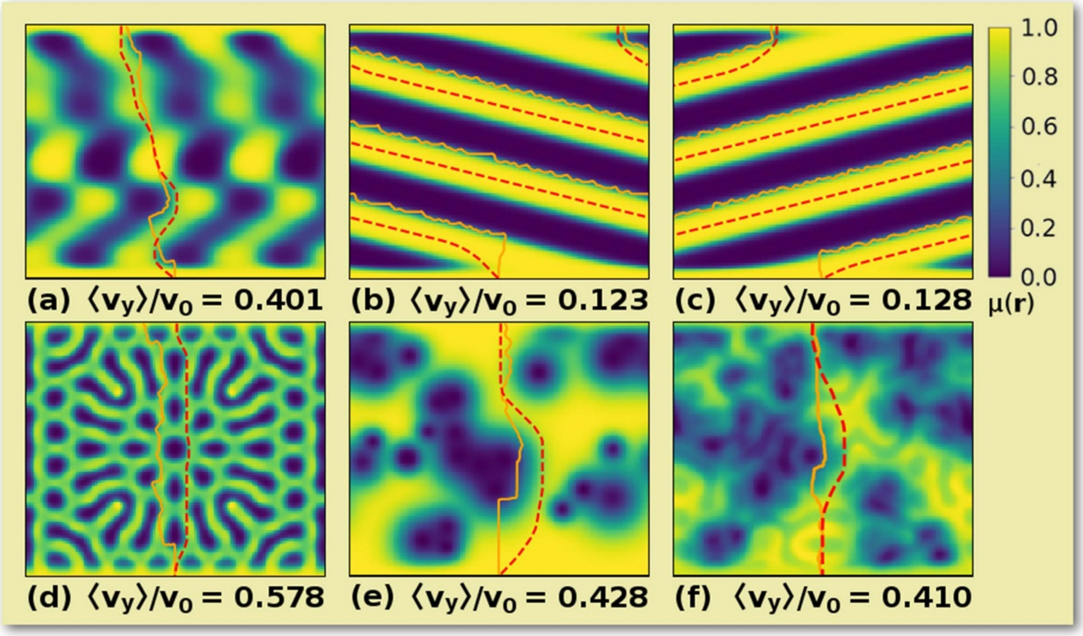


Figure 6. Trajectories of QABPs, trained in non-periodic Gaussian random wave environments (see figure 3), placed in unknown environments, after the finalised learning process ($\mathcal{P}_{\mathrm{rot}} \approx 46$, $\langle\mu\rangle \approx 0.557$). The globally optimised trajectories, generated with the help of *Dijkstra's algorithm*, are shown as dashed red lines. The motility fields $\mu(\mathbf{r})$ are generated with the help of (a) Gaussian random waves (see. Supplemental Material section 1) periodic in horizontal direction, (b), (c) Gaussian random waves periodic in both Carthesian directions, (d) Gray–Scott model for reaction-diffusion (e) 30 Gauss peaks scattered randomly across the simulation box and (f) Eight Gauss peaks scattered randomly across the simulation box, superimposed with solutions to the randomly initialised Gray–Scott model.

the low motility zones, and therefore the Q-learning model of avoiding low motility zones becomes less viable. It is visible in the figure, however, that the QABP surpasses the ABP_∞ down to $\langle\mu\rangle \approx 0.35$.

3.2. Generalisation to unfamiliar environments

In order to demonstrate the generality of the learned strategy, i.e. to show that the final result $\mathcal{Q}$ of the learning process transcends the specific implementation of our learning environment, we test the QABP's ability to navigate in other motility fields that represent different types of long- and short-order order in space. More specifically, the swimmer is first trained on the previously used non-periodic Gaussian random fields with reflecting boundary conditions. After the learning procedure is finalised, the swimmer is placed in the respective unfamiliar environment.

A selection of the emerging trajectories ($\mathcal{P}_{\mathrm{rot}} \approx 46$) is shown in figure 6. All fields are generated such that $\langle\mu\rangle \approx 0.557$. Additionally, a performance $\langle v_y\rangle/v_0$ is given averaged from ten independent training procedures with 10^3 trajectories each. For all fields, the QABP outperforms the references cases and displays performances reasonably close to the globally optimal solutions (the exact numerical values of the different performances can be found in the supplementary material section 9).

3.2.1. Directional patterns

The first instance in figure 6(a) displays an environment that is periodic in horizontal direction. The horizontal components of the random wave vectors fulfill $|\mathbf{k}_n \cdot \hat{\mathbf{e}}_x| = 6\pi/L$. The absolute value of the vertical components $|\mathbf{k}_n \cdot \hat{\mathbf{e}}_y|$ are randomised uniformly in $[0, 12\pi/L]$. The QABP visually displays competence of maneuvering through the new environment. Furthermore, $\langle v_y\rangle/v_0$ shows similar values as presented for the same parameters ($\mathcal{P}_{\mathrm{rot}} \approx 46$, $\langle\mu\rangle \approx 0.557$) in section 3.1. This likely stems from the similarity of the wave vectors $\mathbf{k}_n$ and the resulting characteristic length scales in the environments $\mu(\mathbf{r})$.

Figures 6(b) and (c) show trajectories through periodic Gaussian random fields with $\mathbf{k}_n \cdot \hat{\mathbf{e}}_x = 2\pi/L$ and $\mathbf{k}_n \cdot \hat{\mathbf{e}}_y = \pm 8\pi/L$, respectively. It is visible through the global solutions, that the optimal paths are obtained by avoiding the low motility stripes through the periodicity of the box, persistently swimming in the same slanted direction. The performance of both environments with opposite parity are approximately equal. We thereby show, that the machine learning model is on average not subject to unexpected symmetry breaking.

Furthermore, the swimmers exhibit a significant tendency to slide across the boundary of the low motility zones. This behaviour exemplifies the strategy of the QABP to avoid motilities below a certain threshold rather than optimising the instantaneous velocity at all times.

3.2.2. Gray-Scott field

Figure 6(d) shows an environment obtained through integration of the Gray–Scott equations for reaction-diffusion [42, 43] (see supplementary material section 1.A). The typical length scales of the low motility zones obtained through this model are drastically smaller, than the previous examples. Furthermore, due to the entirely different algorithm, the functional form at the edges of the low motility zones is different. Despite these drastic differences to the original training data, the QABP displays the capability of efficiently maneuvering through the environment.

3.2.3. Strongly clustering patterns

All of the previously shown examples of scalar fields are based on hyperuniform models with suppressed density fluctuations. Next, we show that after training, the QABP also successfully navigates in random motility fields with strong heterogeneities (based on non-hyperuniform models).

Figure 6(e) shows an environment, which is generated by scattering 30 points $\mathbf{q}_k$ uniformly in the simulation box, and assigning each a characteristic randomly chosen length scale κ_k. Explicitly, the motility field is given by

$$\mu(\mathbf{r}) = \prod_{k=1}^{30} \left[1 - \exp\left(\frac{||\mathbf{r} - \mathbf{q}_k||^2}{{\kappa_k}^2}\right)\right], \tag{9}$$

with periodic boundary conditions. The environment in figure 6(e) displays several clusters of peaks. Nevertheless, we observe that the QABP finds a quick path through the environment by evading low motility zones of any size (for which it was efficiently trained in the stealthy hyperuniform GRW).

Finally, figure 6(f) shows a trajectory through a motility field that is given by a sum of equation (9) and a randomly initialised solution to the Gray–Scott model with amplitude 0.1 (see supplementary material section 1.A). This approach yields a motility field, with large low motility zones throughout the environment, overlaid with a more regular perturbation, that causes local gradients, which drastically influence local information. Even though, not having learned about the local structure, our results show, that the QABP noticeably interacts with the respective local gradients only at low swimming velocities. This result emphasises the significance of the inclusion of the swimming velocity in the QABPs state space.

4. Conclusions

In this work, we used a reinforcement learning algorithm to teach a microswimmer (QABP) to navigate through complex environments, given by scalar motility fields, that determine the local swimming velocity of the particle. Brownian dynamics simulations were used to investigate the dynamics of the QABP in this two-dimensional physical environment. To enable smart navigation, a tabular Q-learning algorithm was superimposed. The swimmer receives the ability to perform deterministic rotations, while only receiving local information about its environment.

First, modified Gaussian random waves were employed as motility fields. Two reference cases of an ABP and a particle swimming in a straight line (ABP$_\infty$) were simulated and it was shown that the time of first passage of the ABP to a given target can be inferred from the motility field. The performance, i.e. the speed of finding the target, of the ABP, is bounded by the ABP$_\infty$, as the limit of low diffusion. We demonstrate, that our intelligent QABP outperforms both the ABP and ABP$_\infty$. To demonstrate the applicability of the resulting strategy, we test the ability of the QABP to solve different environments, generated with various algorithms, though only having learned the Gaussian random wave environment. The swimmers display competence of maneuvering through all the displayed examples of motility fields and again outperforms the ABP and ABP$_\infty$. Our stealthy hyperuniform model provides a random yet relatively homogeneous environment that is well suited for the initial training of the QABP. In our observation, QABP provides competitive results in non-hyperuniform fields unseen during the training phase. Furthermore, due to the translational symmetries of our scalar random fields, we expect that the strategy, which has been learned in the training on relatively small box sizes, can automatically be transferred to applications, which feature meaningfully large environments.

Throughout this paper, we lay emphasis on the fact, that the final decision matrix only requires the microswimmer to know little local information about its environment. This will be of particular relevance to future microrobotic applications, where individual autonomous agents rarely possess the ability to capture information about the whole environment at once, and the amount of data storage is dictated by the size of the technical components. Future studies of this local algorithm can be extended to more complex problems such as the inclusion of hydrodynamic force fields, or more general vectorial fields. [13, 16]. The explicit inclusion of cargo uptake and delivery [44, 45], as well as the consumption of fuel, into the machine learning model, may be of interest to medical applications [1]. Swimming strategies, which combine a deterministic approach to the decision making, such as through our reinforcement learning, with undeterministic approaches, e.g. random actions (cf figure 3(b)), may yield insight as models for biological microswimmers, that are motivated, for instance by the search for nutrients.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgment

The authors like to thank Jens Grauer for sharing his machine learning expertise and methodic advice. Without his impactful feeback, this paper would not exist in the current form. This work was supported by the German Research Foundation (DFG) within project LO 418/25-1.

Conflict of Interest declaration

The authors declare, that there are no conflicts of interest to disclose.

ORCID iDs

Paul A Monderkamp https://orcid.org/0000-0001-5244-1586
Fabian Jan Schwarzendahl https://orcid.org/0000-0002-5779-3772
Michael A Klatt https://orcid.org/0000-0002-1029-5960
Hartmut Löwen https://orcid.org/0000-0001-5376-8062

References

[1] Nelson B J, Kaliakatsos I K and Abbott J J 2010 *Annu. Rev. Biomed. Eng.* **12** 55
[2] Cichos F, Gustavsson K, Mehlig B and Volpe G 2020 *Nat. Mach. Intell.* **2** 94
[3] Clegg P S 2021 *Soft Matter* **17** 3991
[4] Falk M J, Alizadehyazdi V, Jaeger H and Murugan A 2021 *Phys. Rev. Res.* **3** 033291
[5] Bechinger C, Di Leonardo R, Löwen H, Reichhardt C, Volpe G and Volpe G 2016 *Rev. Mod. Phys.* **88** 045006
[6] Gao W and Wang J 2014 *ACS Nano* **8** 3170
[7] Abdelmohsen L K, Peng F, Tu Y and Wilson D A 2014 *J. Mater. Chem.* B **2** 2395
[8] Patra D, Sengupta S, Duan W, Zhang H, Pavlick R and Sen A 2013 *Nanoscale* **5** 1273
[9] You M, Chen C, Xu L, Mou F and Guan J 2018 *Acc. Chem. Res.* **51** 3006
[10] Schneider E and Stark H 2019 *Europhys. Lett.* **127** 64003
[11] Yang Y and Bevan M A 2018 *ACS Nano* **12** 10712
[12] La H M, Lim R and Sheng W 2014 *IEEE Trans. Control Syst. Technol.* **23** 52
[13] Liebchen B and Löwen H 2019 *Europhys. Lett.* **127** 34003
[14] Daddi-Moussa-Ider A, Löwen H and Liebchen B 2021 *Commun. Phys.* **4** 1
[15] Zanovello L, Caraglio M, Franosch T and Faccioli P 2021 *Phys. Rev. Lett.* **126** 018001
[16] Nasiri M and Liebchen B 2022 *New J. Phys.* **24** 073042
[17] Reddy G, Celani A, Sejnowski T J and Vergassola M 2016 *Proc. Natl Acad. Sci.* **113** E4877
[18] Reddy G, Wong-Ng J, Celani A, Sejnowski T J and Vergassola M 2018 *Nature* **562** 236
[19] Colabrese S, Gustavsson K, Celani A and Biferale L 2017 *Phys. Rev. Lett.* **118** 158004
[20] Colabrese S, Gustavsson K, Celani A and Biferale L 2018 *Phys. Rev. Fluids* **3** 084301
[21] Gustavsson K, Biferale L, Celani A and Colabrese S 2017 *Eur. Phys. J.* E **40** 110
[22] Alageshan J K, Verma A K, Bec J and Pandit R 2020 *Phys. Rev.* E **101** 043110
[23] Qiu J, Huang W, Xu C and Zhao L 2020 *Sci. China Phys. Mech. Astron.* **63** 284711
[24] Biferale L, Bonaccorso F, Buzzicotti M, Clark P Leoni Di and Gustavsson K 2019 *Chaos* **29** 103138
[25] Muiños-Landin S, Fischer A, Holubec V and Cichos F 2021 *Sci. Robot.* **6** eabd9285
[26] Lavergne F A, Wendehenne H, Bäuerle T and Bechinger C 2019 *Science* **364** 70
[27] Breoni D, Schmiedeberg M and Löwen H 2020 *Phys. Rev.* E **102** 062604
[28] Datt C and Elfring G J 2019 *Phys. Rev. Lett.* **123** 158006
[29] Liebchen B, Monderkamp P, ten Hagen B and Löwen H 2018 *Phys. Rev. Lett.* **120** 208002
[30] Daniels M J, Longland J M and Gilbart J 1980 *Microbiology* **118** 429
[31] Kaiser G and Doetsch R 1975 *Nature* **255** 656
[32] Petrino M G and Doetsch R 1978 *Microbiology* **109** 113
[33] Takabe K, Tahara H, Islam M S, Affroze S, Kudo S and Nakamura S 2017 *Microbiology* **163** 153
[34] Sprenger A R, Fernandez-Rodriguez M A, Alvarez L, Isa L, Wittkowski R and Löwen H 2020 *Langmuir* **36** 7066

[35] Torquato S 2018 *Phys. Rep.* **745** 1–95
[36] Ma Z and Torquato S 2017 *J. Appl. Phys.* **121** 244904
[37] Chen Y, Britton W A and Dal Negro L 2021 *Opt. Lett.* **46** 5360
[38] Klatt M A, Hörmann M and Mecke K 2022 *J. Stat. Mech.: Theory Exp.* **2022** 043301
[39] Sutton R S and Barto A G 2018 *Reinforcement Learning: An Introduction* (Cambridge, MA: MIT Press)
[40] Louste C and Liégeois A 2000 *JINT* **27** 99
[41] Reynoso-Mora P, Chen W and Tomizuka M 2016 *Optim. Control Appl. Methods* **37** 1263
[42] McGough J S and Riley K 2004 *Nonlinear Anal. Real World Appl.* **5** 105
[43] Gray P and Scott S K 1990 *Chemical Oscillations and Instabilities: Non-Linear Chemical Kinetics* (Oxford: Clarendon)
[44] Ma X, Hahn K and Sanchez S 2015 *J. Am. Chem. Soc.* **137** 4976
[45] Demirörs A F, Akan M T, Poloni E and Studart A R 2018 *Soft Matter* **14** 4741