

Learning protocols for the fast and efficient control of active matter

Corneel Casert*

*Molecular Foundry, Lawrence Berkeley National Laboratory,
1 Cyclotron Road, Berkeley, California 94720, USA and
Department of Physics and Astronomy, Ghent University, 9000 Ghent, Belgium*

Stephen Whitelam†

*Molecular Foundry, Lawrence Berkeley National Laboratory,
1 Cyclotron Road, Berkeley, California 94720, USA*

We show that it is possible to learn protocols that effect fast and efficient state-to-state transformations in simulation models of active particles. By encoding the protocol in the form of a neural network we use evolutionary methods to identify protocols that take active particles from one steady state to another, as quickly as possible or with as little energy expended as possible. Our results show that protocols identified by a flexible neural-network ansatz, which allows the optimization of multiple control parameters and the emergence of sharp features, are more efficient than protocols derived recently by constrained analytical methods. Our learning scheme is straightforward to use in experiment, suggesting a way of designing protocols for the efficient manipulation of active matter in the laboratory.

I. INTRODUCTION

Active particles extract energy from their surroundings to produce directed motion [1–4]. Natural active particles include groups of animals and assemblies of cells and bacteria [5–7]; synthetic active particles include active colloids and Janus particles [8, 9]. Active matter, collections of active particles, displays emergent behavior that includes motility-induced phase separation [10, 11], flocking [12, 13], swarming [14], pattern formation [15, 16], and the formation of living crystals [17].

Recent work has focused on controlling such behavior by creating active engines [18–27], controllably clogging and unclogging microchannels [28], doing drug delivery in a targeted way [29, 30], and creating microrobotic swarms with controllable collective behavior [31–33]. For such applications, efficient time-dependent protocols are important [34–38]. Methods for identifying efficient protocols, such as reinforcement learning, have been used to optimize the navigation of active particles in complex environments [39–41] and induce transport in self-propelled disks using a controllable spotlight [42].

For purely diffusive (passive) molecular systems, analytic methods allow the identification of optimal time-dependent protocols for a range of model systems [43–46]. These results establish that rapidly-varying and discontinuous features are common components of optimal protocols, and are useful for benchmarking numerical approaches [47, 48]. However, active-matter systems are more complicated to treat analytically than passive systems, requiring the imposition of protocol constraints in order to make optimization calculations feasible for even the simplest model systems. Two recent papers derive

control protocols for confined active overdamped particles by assuming that protocols are slowly varying and smooth [49] or have a specific functional form [50]. In this paper we show numerically that relaxing these assumptions leads to more efficient control protocols for those systems. The protocols we identify can be rapidly varying and are not smooth, showing jump discontinuities similar to those seen in overdamped passive systems.

To learn protocols to control active matter we use the neuroevolutionary method described in Refs. [47, 51–53], which we adapted from the computer science literature [54–56]. Briefly, we encode a system’s time-dependent protocol [57] in the form $\mathbf{g}_\theta(t/t_f)$. Here \mathbf{g} is the output vector of a deep neural network, corresponding to the control parameters of the system (which in this paper consist of the activity of the particles and the spring constant of their confining potential), θ is the set of neural-network weights, t is the elapsed time of the protocol, and t_f is the total protocol time. We apply the protocol to the system in question, and compute an order parameter ϕ that is minimized when it achieves our desired objective (such as inducing a state-to-state transformation while emitting as little heat as possible). The neural-network weights θ are then iteratively adjusted by a genetic algorithm in order to identify the protocol whose associated value of ϕ is as small as possible.

This approach is a form of deep learning – in the limit of small mutations and a genetic population of size 2 it is equivalent to noisy gradient descent on the objective ϕ [58] – and so comes with the benefits and drawbacks of deep learning generally. Neural networks are very expressive, and if trained well can identify “good” solutions to a problem, but these solutions are not guaranteed to be optimal [59, 60]. We must therefore be pragmatic, and (as with other forms of sampling) verify that protocols obtained from different starting conditions and from independent runs of the learning algorithm are consistent. Consequently, we call the protocols identified by the al-

* ccasert@lbl.gov

† swhitelam@lbl.gov

gorithm “learned” rather than “optimal”. In general, we have found the method to be easy to apply and to solve the problems we have set it: we have benchmarked the method – see Refs. [47, 53] and Fig. S1 in the Supplementary Information (SI) – against exact solutions [43] and other numerical methods [46, 48, 61, 62]. In this paper we use it to produce protocols that are closer to optimal than the protocols obtained by other methods [49, 50].

The neuroevolutionary learning algorithm uses information that is experimentally accessible. And so while in this paper we have learned protocols for the control of simulation models (these protocols could then be applied to experiment if the simulation model is a good enough representation of the experiment), the learning algorithm can also be applied directly to experiment. The following results therefore indicate the potential of this method for learning control protocols for active systems generally.

II. ACTIVE PARTICLE IN A TRAP OF VARIABLE STIFFNESS

In this section we consider the problem of Section IIIA of Ref. [49], a single active Ornstein-Uhlenbeck particle [63–65] in a one-dimensional harmonic trap of stiffness $\alpha(t)$. A schematic of this model is shown in Fig. 1(a). The particle has position r and self-propulsion velocity v . It experiences overdamped Brownian motion with diffusion constant D and mobility μ , such that

$$\dot{r}(t) = v(t) - \mu\alpha r(t) + \sqrt{2D}\eta(t). \quad (1)$$

Here η is a Gaussian white noise term with zero mean and unit variance. The self-propulsion velocity v follows an Ornstein-Uhlenbeck process with persistence time τ and amplitude D_1 , such that $\langle v \rangle = 0$ and $\langle v(t)v(t') \rangle = D_1\tau^{-1}e^{-|t-t'|/\tau}$. The parameter D_1 is zero in the passive limit.

The trajectory-averaged heat associated with varying $\alpha(t)$ from α_i to α_f in time t_f is [19, 49, 64]

$$\begin{aligned} \langle Q \rangle = & \frac{1}{2}(\alpha_i x_i - \alpha_f x_f) + \frac{1}{2} \int_0^{t_f} dt \dot{\alpha}(t)x(t) \\ & + \frac{D_1 t_f}{\tau \mu} - \int_0^{t_f} dt \alpha(t)y(t). \end{aligned} \quad (2)$$

Here $\langle \cdot \rangle$ denotes an average over dynamical trajectories, and we have defined $x \equiv \langle r^2 \rangle$ and $y \equiv \langle rv \rangle$. For time-dependent quantities $q(t)$ we use the notation $q_i \equiv q(0)$ and $q_f \equiv q(t_f)$ to denote initial and final values. The first line of Eq. (2) is the passive heat (minus the change in energy plus the work done by changing the trap stiffness), and the second line is the active contribution to the heat. The first term on the second line is constant for fixed t_f (describing the heat dissipated to sustain the self-propelled motion), and plays no role in selecting the protocol.

For a given protocol $\alpha(t)$, the time evolution of x and y is given by the equations [49]

$$\begin{aligned} \frac{1}{2}\dot{x}(t) + \mu\alpha(t)x(t) &= y(t) + D, \text{ and} \\ \tau\dot{y}(t) + \gamma(t)y(t) &= D_1, \end{aligned} \quad (3)$$

where $\gamma(t) \equiv 1 + \mu\tau\alpha(t)$. The system starts in the steady state associated with the trap stiffness α_i , and so its initial coordinates are

$$x_i = \frac{1}{\alpha_i \mu} \left(\frac{D_1}{\gamma_i} + D \right) \text{ and } y_i = \frac{D_1}{\gamma_i}. \quad (4)$$

Ref. [49] sought protocols that carry out the change of trap stiffness $\alpha_i = 1 \rightarrow \alpha_f = 5$ with minimum mean heat, Eq. (2). The theoretical framework used in that work assumes that protocols $\alpha(t)$ are smooth and are not rapidly varying (see Section S2 for a discussion of this point). Here we revisit this problem using neuroevolution. We find that heat-minimizing protocols are not in general slowly varying or smooth, but can vary rapidly and can display jump discontinuities. The protocols we identify produce considerably less heat than do the protocols identified in Ref. [49] (see Fig. S2).

To learn a protocol $\alpha(t)$ that minimizes heat, we encode a general time-dependent protocol using a deep neural network. We choose the parameterization

$$\alpha_{\theta}(t) = \alpha_i + (\alpha_f - \alpha_i)(t/t_f) + g_{\theta}(t/t_f), \quad (5)$$

where g is the output of a neural network whose input is t/t_f . We constrain the neural network so that $\alpha_i \leq \alpha_{\theta}(t) \leq \alpha_f$, meaning that it cannot access values of α outside the range studied in Ref. [49] [66]. Initially the weights and output of the neural network are zero, and so we start by assuming a protocol that interpolates linearly with time between the initial and final values of α . We train the neural network by genetic algorithm to minimize the order parameter $\phi = \langle Q \rangle$, given by Eq. (2), which we calculate for a given protocol by propagating (3) for time t_f , using a forward Euler discretization with step $\Delta t = 10^{-3}$. An example of the learning process is shown in Fig. S2(a).

In Fig. 1(b) we show, for the choice $D_1 = 2$, that heat-minimizing protocols learned by the neural network vary between a step-like jump at the final time, for small values of t_f , and a step-like jump at the initial time, for large values of t_f . For intermediate values of t_f we observe a range of protocol types. These protocols include non-monotonic and rapidly-varying forms, and show jump discontinuities at initial and final times.

The heat associated with the final-time step protocol is just that associated with the initial steady state, and is

$$Q_1 = \frac{D_1 t_f}{\mu\tau(1 + \alpha_i \mu\tau)}. \quad (6)$$

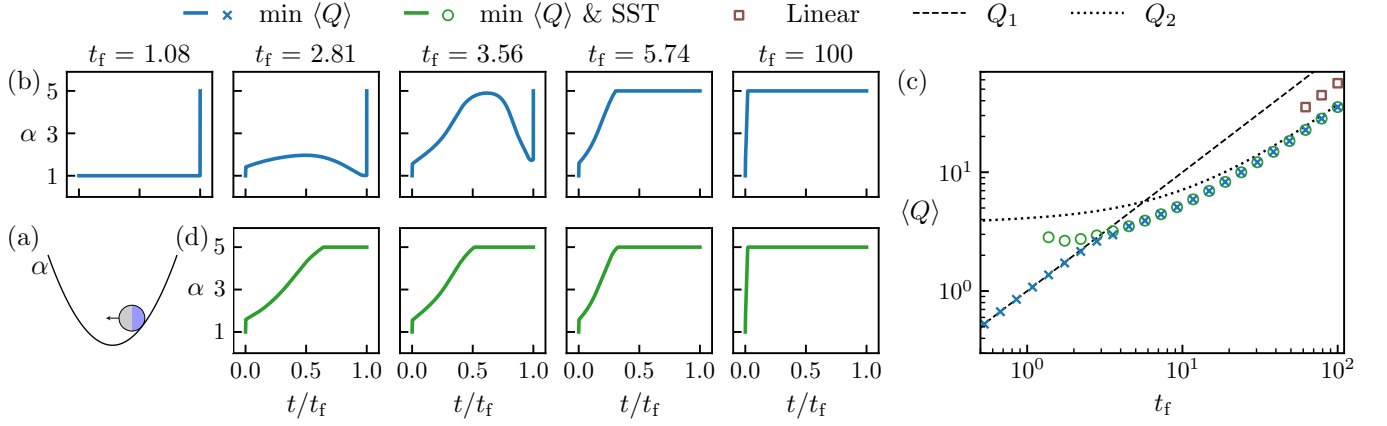


FIG. 1. Protocols $\alpha(t)$ controlling the stiffness of a trap confining an active Ornstein-Uhlenbeck particle. (a) Schematic of the model. (b) Neural-network protocols $\alpha(t)$ that minimize $\langle Q \rangle$, Eq. (2), for different protocol lengths t_f . (c) Heat associated with the protocols. The dotted and dashed lines correspond to the values Q_1 , Eq. (6), and Q_2 , Eq. (7), respectively. (d) As (b), but requiring minimum heat while enacting a state-to-state transformation (SST); see Eq. (11). Model parameters: $D = 1, \mu = 1, \tau = 1, D_1 = 2$.

The heat associated with the initial-time jump protocol can be calculated from Eqs. (2) and (3), and is

$$Q_2 = \frac{\alpha_f}{2} (x_i - x_2(t_f)) + \frac{D_1 t_f}{\mu \tau (1 + \alpha_f \mu \tau)} - \frac{D_1 \tau \alpha_f}{\gamma_f} \left(\frac{1}{\gamma_i} - \frac{1}{\gamma_f} \right) \left(1 - e^{-\gamma_f t_f / \tau} \right), \quad (7)$$

where

$$x_2(t) \equiv (x_i - x_f) e^{-2\mu \alpha_f t} + x_{ss} + 2D_1 \left(\frac{1}{\gamma_i} - \frac{1}{\gamma_f} \right) \left(2\mu \alpha_f - \frac{\gamma_f}{\tau} \right)^{-1} \times \left(e^{-\gamma_f t / \mu} - e^{-2\mu \alpha_f t} \right). \quad (8)$$

(Note that x_{ss} is given in Eq. (10).) For large t_f we have

$$Q_2 \approx \frac{D_1 t_f}{\mu \tau (1 + \alpha_f \mu \tau)}, \quad (9)$$

which is the heat associated with the final steady state.

In Fig. 1(c) we show that the heat values associated with the trained neural-network protocols interpolate, as a function of t_f , between the values Q_1 and Q_2 . Our conclusion is that this optimization problem is solved by protocols that are rapidly varying, have a variety of functional forms, and display jump discontinuities. As shown in Fig. S2, these protocols produce values of heat considerably smaller than those associated with the protocols derived in [49]. (In that figure we also show that it is possible to construct smooth but rapidly-varying protocols that can produce values of heat arbitrarily close to the discontinuous protocols identified by the learning procedure.)

The protocols just described are valid solutions to the heat-minimization problem defined in [49]. However, some of them are not meaningful in experimental terms.

For instance, for small values of t_f , the heat-minimizing protocol is a step function at the final time. This protocol is a solution to the stated problem, but effects no change of the system's microscopic coordinates. All the heat associated with the subsequent transformation of the system is ignored, simply because we have stopped the clock.

We therefore argue that it is more meaningful to search for protocols that minimize heat subject to the requirement of a state-to-state transformation. That is, we require that a specified change in the system's state has occurred. We therefore modify the problem studied in Ref. [49] to search for protocols that minimize the mean heat (2) caused by a change of trap stiffness $\alpha_i = 1 \rightarrow \alpha_f = 5$, *subject to the completion of a state-to-state transformation* (SST) between the initial steady state (4) and that associated with the final-time value of α_f ,

$$x_{ss} = \frac{1}{\alpha_f \mu} \left(\frac{D_1}{\gamma_f} + D \right) \text{ and } y_{ss} = \frac{D_1}{\gamma_f}. \quad (10)$$

As before, we impose the experimentally-motivated constraint $\alpha_i \leq \alpha_\theta(t) \leq \alpha_f$.

To solve this dual-objective problem we choose the evolutionary order parameter

$$\phi = \Delta + c \text{ if } \Delta \geq \Delta_0 \text{ and } \phi = \langle Q \rangle \text{ otherwise.} \quad (11)$$

Here $\Delta^2 \equiv (x_f - x_{ss})^2 + (y_f - y_{ss})^2$ measures the difference between the final-time system coordinates and their values (10) in the final steady state; $\Delta_0 = 10^{-3}$ is the tolerance with which we wish to achieve this steady state [67]; and $c = 100$ is an arbitrary constant whose only role is to make the first clause of (11) always larger than the second. Minimizing (11) will minimize heat emission for a protocol $\alpha(t)$ that in time t_f effects a state-to-state transformation within the precision Δ_0 .

In Fig. 1(d) we show protocols that minimize heat while achieving SST. These protocols have a variety of forms, which involve rapidly-varying portions and jump discontinuities, and that tend, for large t_f , to the initial-time jump form. For times $t_f \lesssim 1.3$ the learning algorithm could not identify a protocol that could achieve SST.

The heat emission associated with these protocols is shown in panel (c). The time t_f for which least heat is emitted is about $t_f = 1.74$, for this choice of D_1 . (For heat optimization alone, the minimum heat is $\langle Q \rangle = 0$, and is shown by Eq. (6) to occur at time $t_f = 0$, a conclusion different to that drawn in Fig. 3 of Ref. [49]. This strange result follows from the fact that the instruction to minimize heat comes with no requirement that the system change state.)

For comparison, we show the heat emission associated with the linear protocol $\alpha_{\text{lin}}(t) = \alpha_i + (\alpha_f - \alpha_i)(t/t_f)$ (square symbols). The linear protocol emits considerably more heat than learned protocols (note the log scale of the figure), and fails to achieve SST for times $t_f \lesssim 60$.

We conclude that the model of the confined active particle studied in [49] is best controlled by protocols $\alpha(t)$ that are in general rapidly varying and exhibit jump discontinuities – similar to protocols for overdamped passive systems – whether the goal is to minimize heat or to do so while also inducing SST. We note that while the evolutionary training of the neural network is a numerical procedure, the protocols it identified allowed us to derive analytic results for the minimum heat produced for sufficiently small and large trajectory lengths, Eq. (6) and Eq. (7) respectively.

III. ACTIVE PARTICLE OF VARIABLE ACTIVITY IN A TRAP OF VARIABLE STIFFNESS

A. State-to-state transformation in least time

In this section we consider the problem of Ref. [50], an active Brownian particle confined by a two-dimensional harmonic potential $U(\boldsymbol{\rho}) = \frac{1}{2}k\boldsymbol{\rho}^2$ with stiffness k . The particle is described by the position vector $\boldsymbol{\rho} = (\rho \cos \phi, \rho \sin \phi)$ and orientation θ , and moves in the direction $\hat{e}(\theta) = (\cos \theta, \sin \theta)$ with constant speed u_0 . Its dynamics is described by the Langevin equation

$$\begin{aligned} \frac{d\boldsymbol{\rho}}{d\tau} &= u_0 \hat{e}(\theta) - \mu k \boldsymbol{\rho} + \sqrt{2D_t} \boldsymbol{\xi}_r(\tau) \\ \frac{d\theta}{d\tau} &= \sqrt{2D_\theta} \xi_\theta(\tau), \end{aligned} \quad (12)$$

where τ is the time; μ is the mobility; D_t and D_θ are translational and rotational diffusion coefficients, respectively; and $\boldsymbol{\xi}_r(\tau)$ and $\xi_\theta(\tau)$ are Gaussian white noise terms with zero mean and unit variance. Upon introducing the dimensionless variables $\mathbf{r} \equiv \boldsymbol{\rho} \sqrt{D_\theta/D_t}$ and

$t \equiv \tau D_\theta$, Eq. (12) reads

$$\begin{aligned} \frac{d\mathbf{r}}{dt} &= \lambda \hat{e}(\theta) - \kappa \mathbf{r} + \sqrt{2} \boldsymbol{\xi}_r(t) \\ \frac{d\theta}{dt} &= \sqrt{2} \xi_\theta(t), \end{aligned} \quad (13)$$

where $\kappa \equiv \mu k/D_\theta$ and $\lambda \equiv u_0/\sqrt{D_\theta D_t}$ are dimensionless versions of the spring constant and the self-propulsion speed (λ is the Péclet number). These dimensionless variables are the control parameters of the problem.

The steady-state probability distribution function $\mathcal{P}_{\text{ss}}(r, \chi)$ of the system depends only on $r \equiv |\mathbf{r}|$ and $\chi \equiv \theta - \phi$, and is known exactly [68]. The steady state associated with the control-parameter choices κ and λ can be classified as passive or active (Fig. 2(a)): in the passive phase, the radial probability distribution $P(r)$ is peaked at the trap center, while in the active phase it is peaked at $r > 0$.

This model system is motivated by experiments involving spherical Janus particles, whose self-propulsion speed can be tuned through light intensity [9], confined in a trap constructed by acoustic waves [69]. For a typical experimental setup the control parameters are bounded as $0 \leq \lambda \leq 11$ and $1 \leq \kappa \leq 7$ [9, 50, 69].

The problem described in Ref. [50] is to find a time-dependent protocol $(\lambda(t), \kappa(t))$ that obeys the bounds of the previous paragraph and that minimizes the time t_f required to transform the distribution $\mathcal{P}(r, \chi)$ from a passive steady state at $(\lambda_i, \kappa_i) = (2.5, 4)$ to an active one at $(\lambda_f, \kappa_f) = (5, 4)$. Using an ansatz constrained so that the distribution $\mathcal{P}(r(t), \chi(t))$ has at all times the form of the steady-state distribution $\mathcal{P}_{\text{ss}}(r, \chi)$ with effective control-parameter values, the authors of that paper found a protocol that completed the state-to-state transformation in time $t_f \approx 0.44$. This protocol is shown in Fig. 2(a,b).

With a neural-network ansatz for the protocol $(\lambda(t), \kappa(t))$ we find that the state-to-state transformation can be achieved about three times as rapidly; see Fig. 2(a,b). For a simulation of fixed time t_f we use a genetic algorithm to train the neural network to minimize the order parameter $\phi = \Delta$, the mean-squared error between the target distribution $\mathcal{P}_{\text{ss}}^*(r, \chi)$ associated with the control-parameter values (λ_f, κ_f) and the distribution $\mathcal{P}(r(t_f), \chi(t_f))$ obtained at the end of the simulation. The latter was calculated from 10^5 independent trajectories of (13) under a given neural-network protocol.

The protocol learned by the neural network for time $t_f = 0.16$ is shown in Fig. 2(a,b), together with the protocol of Ref. [50]. Both show sharp jumps in trap stiffness, decreasing it abruptly to its smallest possible value. The neural network protocol achieves the transformation more quickly because it also enacts a sharp jump in activity, setting it to the maximum possible value (the constraints imposed in Ref. [50] mean that if one control parameter achieves its maximum value in an abrupt way, the other is not free to do so). Near the end of the learned protocol both parameters are abruptly changed to their final values.

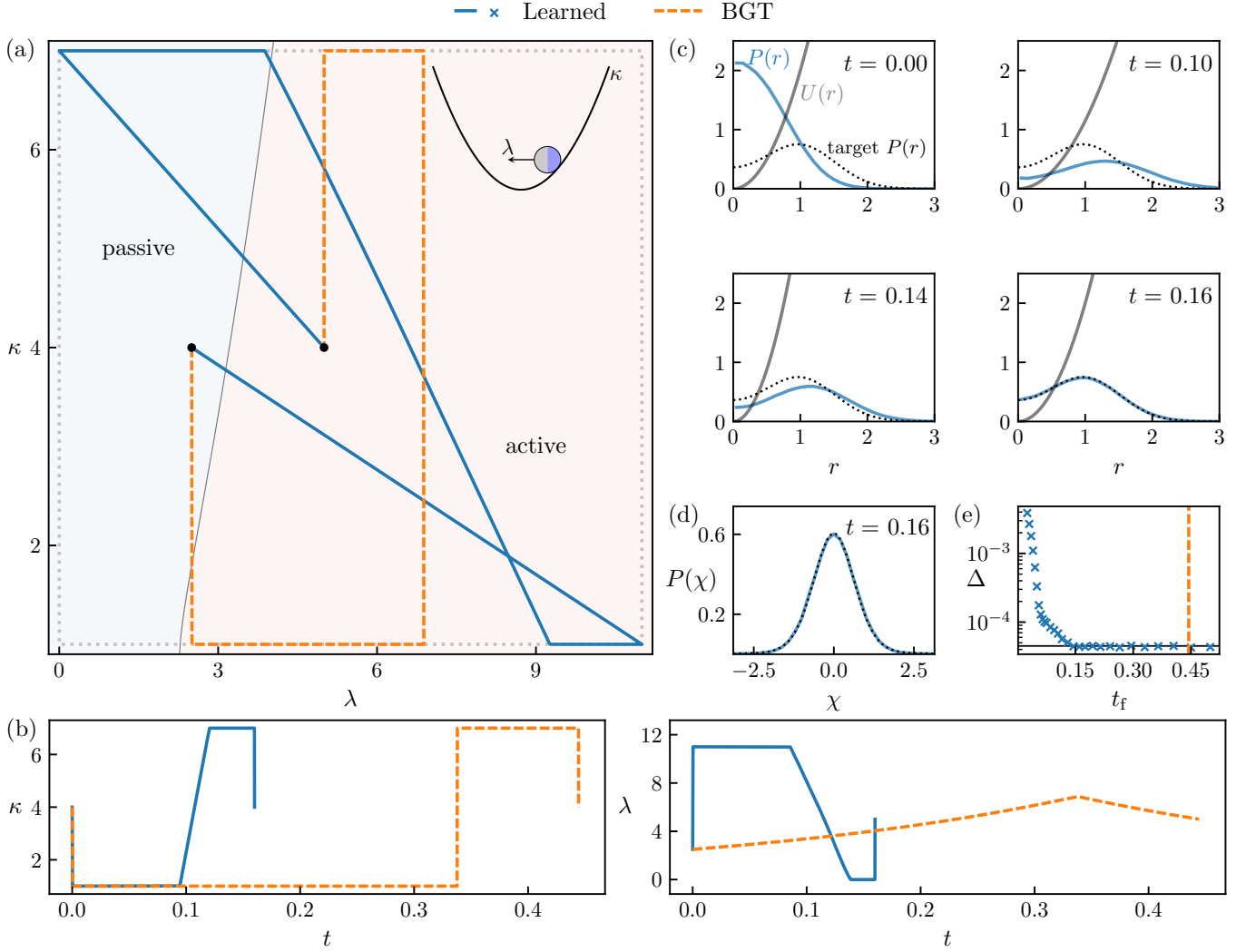


FIG. 2. State-to-state transformation for a confined active Brownian particle with control protocol $(\lambda(t), \kappa(t))$. (a) Parametric protocols from [50] (orange, “BGT”) and this work (blue). The latter achieves the transformation about three times more rapidly than the former. The black dots denote initial and final points; the dotted line denotes the bounds for the control parameter values. The schematic in the top-right corner is a one-dimensional schematic of this two-dimensional system. (b) The protocols of panel (a) as a function of time t . (c) Temporal evolution of the radial distribution function $P(r)$ together with the target distribution (dotted line) and the potential $U(r)$, for the learned protocols shown in (a,b). (d) Final-time distribution for χ . The dotted line is the target distribution. (e) Mean-squared error Δ between the final distribution and the exact solution, averaged over 10^5 trajectories. The dashed vertical line is the transformation time for the protocol of [50], and the horizontal line is the value of Δ associated with that protocol.

In Fig. 2(c) we show the temporal evolution of $P(r)$ for the learned protocol. Starting from an initial distribution peaked at the origin, the peak of $P(r)$ overshoots the peak of the target distribution (they are not at that time of the same shape). The peak of $P(r)$ is later brought back toward the target when stiffness and activity are set to their maximal and minimal values, respectively. Subsequently, both are set to their final values.

In Fig. 2(d) we show the final-time distribution of χ for the learned protocol, which matches the target distribution.

In Fig. 2(e) we show the value of Δ obtained by pro-

ocols trained at various fixed simulation times t_f . For times $t_f \gtrsim 0.15$, the learned protocol produces a small constant value of Δ consistent with the value produced by the protocol of Ref. [50] (horizontal line). For times $t_f \lesssim 0.15$ the value of Δ increases sharply with decreasing t_f , indicating that the state-to-state transformation cannot be achieved with the same precision.

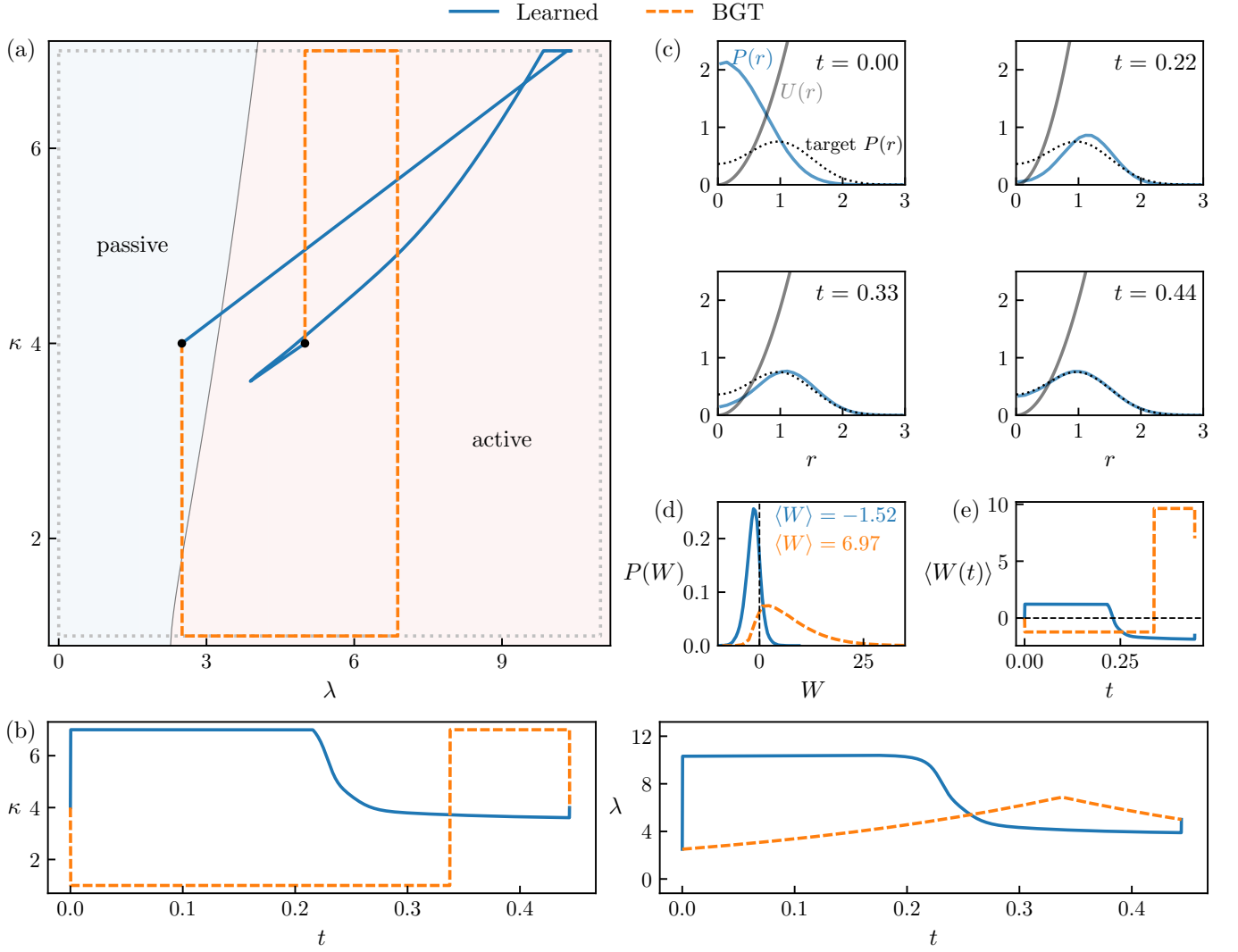


FIG. 3. Similar to Fig. 2, but now the learning algorithm is told to enact the state-to-state transformation of Fig. 2, in time $t_f = 0.44$, while minimizing work done; see Eq. (14). Panels (a-c) are analogous to those of Fig. 2. (d) Distribution of work for the two protocols. (e) Mean work as a function of time for the two protocols.

B. State-to-state transformation with work extraction

It is possible to extract work during the state-to-state transformation. Setting $t_f = 0.44$, the transformation time of the protocol of Ref. [50], we used a genetic algorithm to train a neural network to minimize the objective

$$\phi = \Delta + c \text{ if } \Delta \geq \Delta_0 \text{ and } \phi = \langle W \rangle \text{ otherwise.} \quad (14)$$

Here Δ_0 is the mean-squared error associated with the protocol of Ref. [50] (calculated using 10^5 trajectories), and $c = 100$ is an arbitrary constant whose only role is to make the first clause of (14) always larger than the second. The quantity $\langle W \rangle$ is the mean work, in units of μ/D_t , given by

$$\langle W \rangle = \int_0^{t_f} dt \dot{\kappa} \left\langle \frac{\partial U}{\partial \kappa} \right\rangle = \frac{1}{2} \int_0^{t_f} dt \dot{\kappa} \langle r^2 \rangle. \quad (15)$$

Minimizing (14) will minimize the mean work associated with a protocol $(\lambda(t), \kappa(t))$ that in time t_f effects the state-to-state transformation to a precision Δ_0 .

The protocol learned in this way is shown in Fig. 3(a,b), together with the protocol of Ref. [50]. Panels (c) show the effect of the learned protocol on the radial probability distribution. The neural-network protocol increases κ to its maximum value at the beginning of the protocol. Doing so costs work, but only small amounts because the system is initially in a passive phase and so $\langle r^2 \rangle$ is small. The protocol also increases λ to a large (but sub-maximal) value, which begins to drive the distribution into the active phase, so increasing $\langle r^2 \rangle$. Subsequently, κ is decreased to its target value, causing a decrease of energy and allowing net extraction of work.

Fig. 3(d) shows the work distributions $P(W)$ associated with the learned protocol and that of Ref. [50]. The

latter results in a broad distribution of work values, and on average requires a large input of work to enact the transformation. By contrast, the work distribution obtained using the learned protocol is sharply peaked at a negative value, and the mean work is negative.

In Fig. 3(e) we show mean work as a function of time for the two protocols. The learned protocol requires an input of work at early times in order to extract net work at later times. This solution was identified by a genetic algorithm using an order parameter (14) that depends only on quantities evaluated at the final time point. As a result, the protocol is not biased toward any particular functional form. By contrast, greedy reinforcement-learning algorithms, which at all times attempt to reduce the objective function, would (without special shaping of the reward function) be unlikely to find the solution shown here.

IV. WORK EXTRACTION FROM CONFINED, INTERACTING ACTIVE PARTICLES

We now consider the case of N interacting active Brownian particles placed within the two-dimensional harmonic trap of Section III. Particle i evolves according to the Langevin equation

$$\begin{aligned} \frac{d\mathbf{r}_i}{dt} &= \lambda \hat{\mathbf{e}}_i(\theta) - \kappa \mathbf{r}_i - \partial_{\mathbf{r}_i} \sum_{j \neq i} V(r_{ij}) + \sqrt{2} \boldsymbol{\xi}_r(t) \\ \frac{d\theta_i}{dt} &= \sqrt{2} \xi_\theta(t), \end{aligned} \quad (16)$$

whose terms are similar to those of (12) with the addition of the Weeks-Chandler-Andersen interaction

$$V(x) = \begin{cases} 4\epsilon \left[(\sigma/x)^{12} - (\sigma/x)^6 \right] + \epsilon & (x < 2^{1/6}\sigma) \\ 0 & (\text{otherwise}), \end{cases} \quad (17)$$

which takes as its argument the inter-particle separation $r_{ij} \equiv |\mathbf{r}_j - \mathbf{r}_i|$. We set σ and ϵ to 1.

We wish to learn protocols that minimize the mean work done upon reducing the trap stiffness from $\kappa_i = 5$ to $\kappa_f = 2$, in time t_f , observing the bounds on the control parameter values of Section III. Here work is

$$\langle W \rangle = \frac{N}{2} \int_0^{t_f} dt \dot{\kappa} R^2, \quad (18)$$

where $R^2 \equiv N^{-1} \sum_{i=1}^N \langle r_i^2 \rangle$. The angle brackets indicate an average over dynamical trajectories. We start from a steady state at $\lambda_i = 0$, but place no constraints (beyond those of the control-parameter bounds) on the value of λ_f . Such a transformation could be used as part of a cycle for an active engine [18–20].

No analytical solutions are known for this many-body system, but a protocol can be learned in exactly the same way as for the single-particle problems considered previously, using a genetic algorithm to train a neural network

to minimize $\phi = \langle W \rangle$. The latter was calculated from 10^3 independent trajectories.

In Fig. 4(a) we show the result of this learning procedure for trajectory time $t_f = 1$ and a number of particles between $N = 1$ and $N = 40$. In Fig. S4 and Fig. S5 we provide additional details of learned protocols for the cases $N = 12$ and $N = 40$. In all cases work can be extracted, $\langle W \rangle < 0$. However, the extracted work per particle is a non-monotonic function of N , attaining a minimum value for $N = 12$. For this particular problem, the many-body system becomes more efficient than the one-body system for $N > 25$. This finding suggests that particular cycles of many-body active engines may function more efficiently with certain particle numbers.

The learned protocols that produce the work values in Fig. 4(a) initially increase λ to its maximum value. For small N they initially increase κ to its maximum value, while for large N they initially increase κ to close to its maximum value (see Fig. S4 and Fig. S5). This initial increase of κ costs work (as with the protocols in Sec. IIIB), but the significant increase in R^2 as the activity is increased allows for net work extraction upon the subsequent reduction of κ .

As N is increased from 1, the amount of work that can be extracted per particle initially goes down. This decrease results from the fact that particles repel each other, and so R^2 in the passive initial state is significantly larger for $N > 1$ than for $N = 1$; see Fig. 4(b). Increasing κ (at early times) therefore costs more work per particle than for the case $N = 1$. Work can still be extracted from this system, but less efficiently than for the single-body system.

For N sufficiently large, however, the situation changes: R^2 for large λ and κ becomes much larger than it is for a single particle (Fig. 4(b,c)) [70]. This change allows for greater work extraction per particle when κ is decreased later on in the protocol. For $N > 25$, this effect exceeds that described in the previous paragraph, and the many-body system provides more work per particle than a one-body system.

To illustrate the origin of the non-monotonicity seen in Fig. 4(a), we consider a simplified protocol that instantaneously sets κ and λ to their maximum values, waits until the system reaches a steady state, and then sets $\kappa = \kappa_f$. The work per particle associated with this protocol is given by

$$\begin{aligned} \frac{\langle W \rangle_{\text{est}}}{N} &= \frac{1}{2} (\kappa_{\text{max}} - \kappa_i) R_{\lambda_i, \kappa_i}^2 \\ &\quad - \frac{1}{2} (\kappa_f - \kappa_{\text{max}}) R_{\lambda_{\text{max}}, \kappa_{\text{max}}}^2, \end{aligned} \quad (19)$$

where $R_{\lambda, \kappa}^2$ denotes the steady-state value of R^2 measured at (λ, κ) . In Fig. 4(c) we show that Eq. (19) is a non-monotonic function of N . It is not a quantitatively accurate model of the learned protocols, but captures one important feature of their behavior.

Extending the simulation time to $t_f = 10$ allows for even greater work extraction. This improvement is

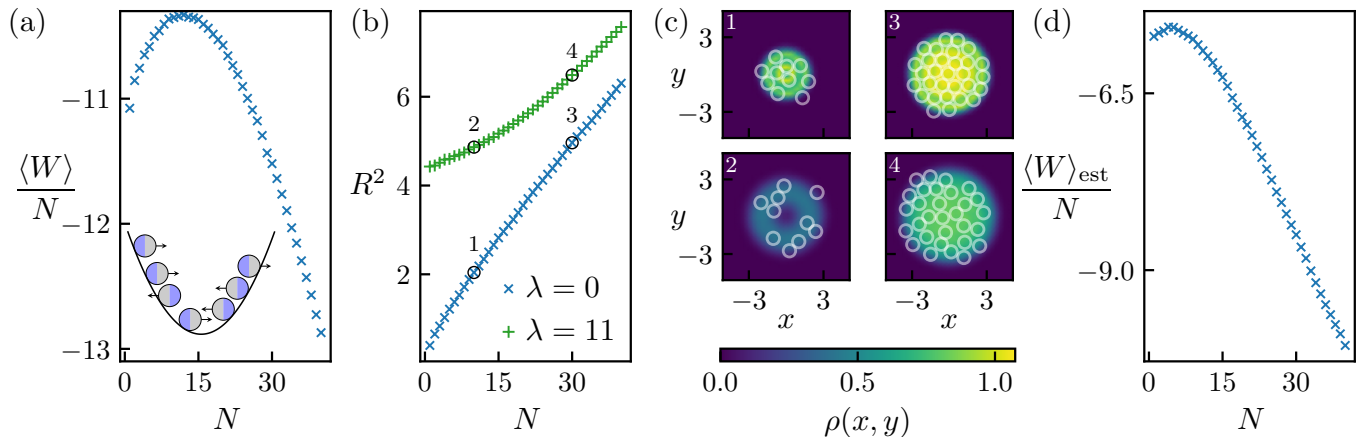


FIG. 4. Many-body active engine segment. (a) Average work $\langle W \rangle$ per particle, as a function of the number of particles N , for neural-network protocols trained to minimize work by controlling the activity and confining potential of a set of interacting active Brownian particles. The many-body system is more efficient than the one-body system for $N > 25$. (b) Steady-state values of $R^2 \equiv N^{-1} \sum_{i=1}^N \langle r_i^2 \rangle$ as a function of N . Here $\kappa = 5$, and λ is set to its minimum (blue line) and maximum (green line) values. (c) Steady-state particle density at the control parameters indicated in (b). Instantaneous particle positions for one realization of the system at steady state are shown in white. (d) A simple model of the learned protocols, Eq. (19), captures the non-monotonicity of panel (a).

achieved by a learned protocol that substantially changes the system’s activity twice, from passive to active to passive again (see Fig. S3).

V. CONCLUSIONS

We have shown that evolutionary methods can train neural networks to produce efficient protocols for state-to-state transformations in simulation models of active matter. We found protocols that were more efficient than those derived recently by constrained analytical methods, and showed that neural-network methods can aid in the design of protocols that achieve extraction of work from many-body active systems. The learning scheme used here can be applied to experiment the way it is applied to simulations, suggesting a way of designing protocols for the efficient manipulation of active matter in the laboratory.

ACKNOWLEDGMENTS

Work at the Molecular Foundry was supported by the Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231, and the Stevin Supercomputer Infrastructure, provided by the VSC (Flemish Supercomputer Center), funded by Ghent University, FWO and the Flemish Government — department EWI. C.C. was supported through a Francqui Fellowship of the Belgian American Educational Foundation, and partly through the US DOE Office of Science Scientific User Facilities AI/ML project “A digital twin for spatiotemporally resolved experiments”.

-
- [1] Sriram Ramaswamy, “The mechanics and statistics of active matter,” *The Annual Review of Condensed Matter Physics* **1**, 323–45 (2010).
 - [2] Pawel Romanczuk, Markus Bär, Werner Ebeling, Benjamin Lindner, and Lutz Schimansky-Geier, “Active brownian particles,” *The European Physical Journal Special Topics* **202**, 1–162 (2012).
 - [3] M Cristina Marchetti, JF Joanny, S Ramaswamy, TB Liverpool, J Prost, Madan Rao, and R Aditi Simha, “Hydrodynamics of soft active matter,” *Reviews of Modern Physics* **85**, 1143 (2013).
 - [4] Clemens Bechinger, Roberto Di Leonardo, Hartmut Löwen, Charles Reichhardt, Giorgio Volpe, and Giovanni Volpe, “Active particles in complex and crowded environments,” *Reviews of Modern Physics* **88**, 045006 (2016).
 - [5] Andrea Cavagna and Irene Giardinà, “Bird flocks as condensed matter,” *Annu. Rev. Condens. Matter Phys.* **5**, 183–207 (2014).
 - [6] Jens Elgeti, Roland G Winkler, and Gerhard Gompper, “Physics of microswimmers—single particle motion and collective behavior: a review,” *Reports on progress in*

- physics **78**, 056601 (2015).
- [7] Daniel Needleman and Zvonimir Dogic, “Active matter at the interface between materials science and cell biology,” *Nature reviews materials* **2**, 1–14 (2017).
 - [8] Andreas Zöttl and Holger Stark, “Emergent behavior in active colloids,” *Journal of Physics: Condensed Matter* **28**, 253001 (2016).
 - [9] Ivo Buttinoni, Giovanni Volpe, Felix Kümmel, Giorgio Volpe, and Clemens Bechinger, “Active brownian motion tunable by light,” *Journal of Physics: Condensed Matter* **24**, 284129 (2012).
 - [10] Michael E. Cates and Julien Tailleur, “Motility-induced phase separation,” *Annual Review of Condensed Matter Physics* **6**, 219 (2015).
 - [11] Jérémy O’Byrne, Alexandre Solon, Julien Tailleur, and Yongfeng Zhao, “An Introduction to Motility-induced Phase Separation,” in *Out-of-equilibrium Soft Matter* (The Royal Society of Chemistry, 2023).
 - [12] John Toner, Yuhai Tu, and Sriram Ramaswamy, “Hydrodynamics and phases of flocks,” *Annals of Physics* **318**, 170–244 (2005).
 - [13] Hugues Chaté, “Dry aligning dilute active matter,” *Annual Review of Condensed Matter Physics* **11**, 189–212 (2020).
 - [14] Avraham Be’er and Gil Ariel, “A statistical physics view of swarming bacteria,” *Movement ecology* **7**, 1–17 (2019).
 - [15] Benno Liebchen and Demian Levis, “Collective behavior of chiral active matter: Pattern formation and enhanced flocking,” *Physical review letters* **119**, 058002 (2017).
 - [16] Benno Liebchen and Hartmut Löwen, “Synthetic chemotaxis and collective behavior in active matter,” *Accounts of chemical research* **51**, 2982–2990 (2018).
 - [17] Jeremie Palacci, Stefano Sacanna, Asher Preska Steinberg, David J Pine, and Paul M Chaikin, “Living crystals of light-activated colloidal surfers,” *Science* **339**, 936–940 (2013).
 - [18] Timothy Ekeh, Michael E. Cates, and Étienne Fodor, “Thermodynamic cycles with active matter,” *Phys. Rev. E* **102**, 010101 (2020).
 - [19] Étienne Fodor and Michael E. Cates, “Active engines: Thermodynamics moves forward,” *Europhysics Letters* **134**, 10003 (2021).
 - [20] Aradhana Kumari, P. S. Pal, Arnab Saha, and Sourabh Lahiri, “Stochastic heat engine using an active particle,” *Physical Review E* **101**, 032109 (2020).
 - [21] Luca Cocconi, Jacob Knight, and Connor Roberts, “Optimal power extraction from active particles with hidden states,” *Physical Review Letters* **131**, 188301 (2023).
 - [22] Tushar K Saha, Jannik Ehrich, Momčilo Gavrilov, Susanne Still, David A Sivak, and John Bechhoefer, “Information engine in a nonequilibrium bath,” *Physical Review Letters* **131**, 057101 (2023).
 - [23] Viktor Holubec, Stefano Steffenoni, Gianmaria Falasco, and Klaus Kroy, “Active brownian heat engines,” *Physical Review Research* **2**, 043262 (2020).
 - [24] Viktor Holubec and Rahul Marathe, “Underdamped active brownian heat engine,” *Physical Review E* **102**, 060101 (2020).
 - [25] Arya Datta, Patrick Pietzonka, and Andre C Barato, “Second law for active heat engines,” *Physical Review X* **12**, 031034 (2022).
 - [26] Giulia Gronchi and Andrea Puglisi, “Optimization of an active heat engine,” *Physical Review E* **103**, 052134 (2021).
 - [27] Patrick Pietzonka, Étienne Fodor, Christoph Lohrmann, Michael E Cates, and Udo Seifert, “Autonomous engines driven by active matter: Energetics and design principles,” *Physical Review X* **9**, 041032 (2019).
 - [28] Lorenzo Caprini, Fabio Cecconi, Claudio Maggi, and U Marini Bettolo Marconi, “Activity-controlled clogging and unclogging of microchannels,” *Physical Review Research* **2**, 043359 (2020).
 - [29] Arijit Ghosh, Weinan Xu, Neha Gupta, and David H Gracias, “Active matter therapeutics,” *Nano Today* **31**, 100836 (2020).
 - [30] Ming Luo, Youzeng Feng, Tingwei Wang, and Jianguo Guan, “Micro-/nanorobots at work in active drug delivery,” *Advanced Functional Materials* **28**, 1706100 (2018).
 - [31] Michael Rubenstein, Christian Ahler, Nick Hoff, Adrian Cabrera, and Radhika Nagpal, “Kilobot: A low cost robot with scalable operations designed for collective behaviors,” *Robotics and Autonomous Systems* **62**, 966–975 (2014).
 - [32] Berk Yigit, Yunus Alapan, and Metin Sitti, “Programmable collective behavior in dynamically self-assembled mobile microrobotic swarms,” *Advanced Science* **6**, 1801837 (2019).
 - [33] Angelo Barona Balda, Aykut Argun, Agnese Callegari, and Giovanni Volpe, “Playing with active matter,” *arXiv preprint arXiv:2209.04168* (2022).
 - [34] Deepak Gupta, Sabine H. L. Klapp, and David A. Sivak, “Efficient control protocols for an active Ornstein-Uhlenbeck particle,” *Physical Review E* **108**, 024117 (2023).
 - [35] David Guéry-Odelin, Christopher Jarzynski, Carlos A. Plata, Antonio Prados, and Emmanuel Trizac, “Driving rapidly while remaining in control: Classical shortcuts from Hamiltonian to stochastic dynamics,” *Reports on Progress in Physics* **86**, 035902 (2023).
 - [36] Michael M Norton, Piyush Grover, Michael F Hagan, and Seth Fraden, “Optimal control of active nematics,” *Physical review letters* **125**, 178005 (2020).
 - [37] Shriram Chennakesavalu and Grant M Rotskoff, “Probing the theoretical and computational limits of dissipative design,” *The Journal of Chemical Physics* **155** (2021).
 - [38] Suraj Shankar, Vidya Raju, and L Mahadevan, “Optimal transport and control of active drops,” *Proceedings of the National Academy of Sciences* **119**, e2121985119 (2022).
 - [39] Paul A Monderkamp, Fabian Jan Schwarzendahl, Michael A Klatt, and Hartmut Löwen, “Active particles using reinforcement learning to navigate in complex motility landscapes,” *Machine Learning: Science and Technology* **3**, 045024 (2022).
 - [40] Mahdi Nasiri and Benno Liebchen, “Reinforcement learning of optimal active particle navigation,” *New Journal of Physics* **24**, 073042 (2022).
 - [41] Mahdi Nasiri, Hartmut Löwen, and Benno Liebchen, “Optimal active particle navigation meets machine learning,” *Europhysics Letters* **142**, 17001 (2023).
 - [42] Martin J Falk, Vahid Alizadehyazdi, Heinrich Jaeger, and Arvind Murugan, “Learning to control active matter,” *Physical Review Research* **3**, 033291 (2021).
 - [43] Tim Schmiedl and Udo Seifert, “Optimal finite-time processes in stochastic thermodynamics,” *Physical Review Letters* **98**, 108301 (2007).

- [44] Alex Gomez-Marín, Tim Schmiedl, and Udo Seifert, “Optimal protocols for minimal work processes in underdamped stochastic thermodynamics,” *The Journal of chemical physics* **129** (2008).
- [45] Steven Blaber, Miranda D Louwerse, and David A Sivak, “Steps minimize dissipation in rapidly driven stochastic systems,” *Physical Review E* **104**, L022101 (2021).
- [46] Adrienne Zhong and Michael R DeWeese, “Limited-control optimal protocols arbitrarily far from equilibrium,” *Physical Review E* **106**, 044135 (2022).
- [47] Stephen Whitelam, “Demon in the Machine: Learning to Extract Work and Absorb Entropy from Fluctuating Nanosystems,” *Physical Review X* **13**, 021005 (2023).
- [48] Megan C. Engel, Jamie A. Smith, and Michael P. Brenner, “Optimal Control of Nonequilibrium Systems through Automatic Differentiation,” *Physical Review X* **13**, 041032 (2023).
- [49] Luke K. Davis, Karel Proesmans, and Étienne Fodor, “Active matter under control: Insights from response theory,” *Phys. Rev. X* **14**, 011012 (2024).
- [50] Marco Baldovin, David Guéry-Odelin, and Emmanuel Trizac, “Control of Active Brownian Particles: An Exact Solution,” *Physical Review Letters* **131**, 118302 (2023).
- [51] Stephen Whitelam and Isaac Tamblyn, “Learning to grow: Control of material self-assembly using evolutionary reinforcement learning,” *Physical Review E* **101**, 052604 (2020).
- [52] Stephen Whitelam and Isaac Tamblyn, “Neuroevolutionary learning of particles and protocols for self-assembly,” *Physical Review Letters* **127**, 018003 (2021).
- [53] Stephen Whitelam, “How to train your demon to do fast information erasure without heat production,” *Physical Review E* **108**, 044138 (2023).
- [54] John H Holland, “Genetic algorithms,” *Scientific American* **267**, 66–73 (1992).
- [55] Melanie Mitchell, *An introduction to genetic algorithms* (MIT press, 1998).
- [56] Felipe Petroski Such, Vashisht Madhavan, Edoardo Conti, Joel Lehman, Kenneth O Stanley, and Jeff Clune, “Deep neuroevolution: genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning,” *arXiv preprint arXiv:1712.06567* (2017).
- [57] It is also straightforward within this scheme to consider a feedback-control protocol, by considering a neural network $\mathbf{g}_\theta(t/t_f, \mathbf{v})$, where \mathbf{v} is a vector of state-dependent information [47].
- [58] Stephen Whitelam, Viktor Selin, Sang-Won Park, and Isaac Tamblyn, “Correspondence between neuroevolution and gradient descent,” *Nature Communications* **12**, 1–10 (2021).
- [59] Kurt Hornik, Maxwell Stinchcombe, and Halbert White, “Multilayer feedforward networks are universal approximators,” *Neural networks* **2**, 359–366 (1989).
- [60] Yasaman Bahri, Jonathan Kadmon, Jeffrey Pennington, Sam S Schoenholz, Jascha Sohl-Dickstein, and Surya Ganguli, “Statistical mechanics of deep learning,” *Annual Review of Condensed Matter Physics* **11**, 501–528 (2020).
- [61] Grant M Rotskoff and Gavin E Crooks, “Optimal control in nonequilibrium systems: Dynamic riemannian geometry of the ising model,” *Physical Review E* **92**, 060102 (2015).
- [62] Todd R Gingrich, Grant M Rotskoff, Gavin E Crooks, and Phillip L Geissler, “Near-optimal protocols in complex nonequilibrium transformations,” *Proceedings of the National Academy of Sciences* **113**, 10263–10268 (2016).
- [63] David Martin, Jérémy O’Byrne, Michael E. Cates, Étienne Fodor, Cesare Nardini, Julien Tailleur, and Frédéric van Wijland, “Statistical mechanics of active ornstein-uhlenbeck particles,” *Phys. Rev. E* **103**, 032607 (2021).
- [64] Étienne Fodor, Robert L. Jack, and Michael E. Cates, “Irreversibility and biased ensembles in active matter: Insights from stochastic thermodynamics,” *Annual Review of Condensed Matter Physics* **13**, 215–238 (2022).
- [65] Luis L Bonilla, “Active ornstein-uhlenbeck particles,” *Physical Review E* **100**, 022601 (2019).
- [66] When we relax this constraint we find protocols that produce less heat, in general, than the protocols that observe the constraint. We impose the constraint to allow us to make contact with Ref. [49], and because experimental systems have constraints on the maximum values of their control parameters.
- [67] Protocols and heat values depend weakly on the value of this threshold, but not in a way that affects our general conclusions.
- [68] Kanaya Malakar, Arghya Das, Anupam Kundu, K. Vijay Kumar, and Abhishek Dhar, “Steady state of an active Brownian particle in a two-dimensional harmonic trap,” *Physical Review E* **101**, 022610 (2020).
- [69] Sho C Takatori, Raf De Dier, Jan Vermant, and John F Brady, “Acoustic trapping of active matter,” *Nature communications* **7**, 10694 (2016).
- [70] For small N , the value of R^2 for large λ and κ is not much larger than it is for $N = 1$, because particles spread out to form a ring and can adopt a mean radial position similar to that preferred by a single particle; see Fig. 4(c).

Supplementary Information for “Learning protocols for the fast and efficient control of active matter”

Corneel Casert*

*Molecular Foundry, Lawrence Berkeley National Laboratory,
1 Cyclotron Road, Berkeley, California 94720, USA and*

Department of Physics and Astronomy, Ghent University, 9000 Ghent, Belgium

Stephen Whitelam†

*Molecular Foundry, Lawrence Berkeley National Laboratory,
1 Cyclotron Road, Berkeley, California 94720, USA*

S1. BENCHMARKING THE LEARNING PROCEDURE

In Fig. S1 we show the protocol $\alpha(t)$ identified by the neuroevolutionary learning procedure when instructed to minimize the work

$$\langle W \rangle = \int_0^{t_f} dt \dot{\alpha}(t) x(t) \quad (\text{S1})$$

for a passive Brownian particle in a harmonic trap of stiffness $\alpha(t)$, with $\alpha_i = 1$, $\alpha_f = 5$, and $t_f = 1$. The particle is described by (3) with $D_1 = 0$. The optimal protocol for this transformation is known exactly [43], and displays jumps at both $t = 0$ and $t = t_f$. The agreement between the learned protocol and the known optimal protocol provides a benchmark for the learning algorithm when applied to the passive version of the problem studied in Section II, and provides confidence in the ability of the algorithm to identify rapidly-varying and discontinuous portions of protocols if necessary. For additional benchmarks see Refs. [47, 53].

S2. COMPARISON WITH REF. [49]

In Fig. S2, we compare a learned neural-network protocol that minimizes the heat $\langle Q \rangle$ (2) of the system of Section II with the results of protocols derived in Ref. [49], in the low-activity case $D_1 = 0.01$ shown in Fig. 2 of that reference.

Fig. S2(a) shows the mean heat $\langle Q \rangle$ produced by the neural-network protocol as a function of training time n . Initially the heat produced is that of the linear protocol $\alpha(t) = \alpha_i + (\alpha_f - \alpha_i)(t/t_f)$ from which we start our search. The linear protocol produces values of heat similar to the protocols of Ref. [49], which are approximately linear. After about 20 steps of training, the neural-network protocol converges to the final-time step form, which emits an amount of heat equal to Q_1 , given by Eq. (6).

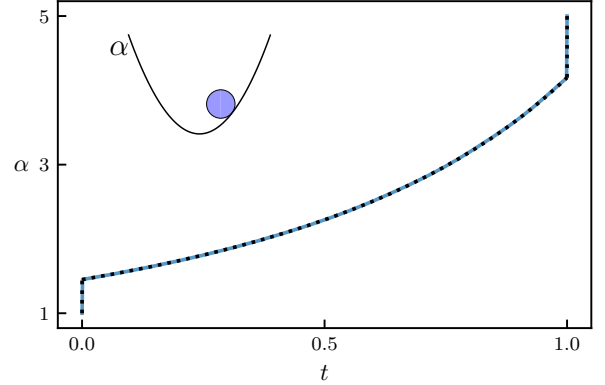


FIG. S1. Benchmarking the learning algorithm: the neuroevolution procedure used in the main text was instructed to find a protocol $\alpha(t)$ of length $t_f = 1$ that with least work changes from $\alpha_i = 1 \rightarrow \alpha_f = 5$ the stiffness α of a trap containing a passive Brownian particle. The protocol it identified is shown in blue. The black dotted line is the exact result from Ref. [43].

In Fig. S2(b) we compare, for a range of values of t_f , the heat associated with trained neural-network protocols, the protocols of Ref. [49], and the linear protocol. The latter two protocols produce similar values of heat, and both produce many times the heat emitted by the neural-network protocols. (For the neural-network protocols, the crossover between the forms producing heat Q_1 and Q_2 occurs here for a much larger t_f than is the case for the value $D_1 = 2$ considered in the main text.)

The discrepancy between the protocols derived here and the protocols of Ref. [49] probably results from the assumption made in that reference of slowly-varying driving, expressed by Eq. (7) of that reference, rather than from the assumption of smoothness. It is possible to construct smooth but rapidly-varying protocols that are arbitrarily close to discontinuous ones; for instance, the family of protocols

$$\alpha_k(t) = \alpha_i + (\alpha_f - \alpha_i) (t/t_f)^k \quad (\text{S2})$$

interpolates between the linear protocol for $k = 1$ and the final-time step protocol for $k \rightarrow \infty$. We demonstrate this interpolation in Fig. S2(c) and (d) for the trajectory

* ccasert@lbl.gov

† switelam@lbl.gov

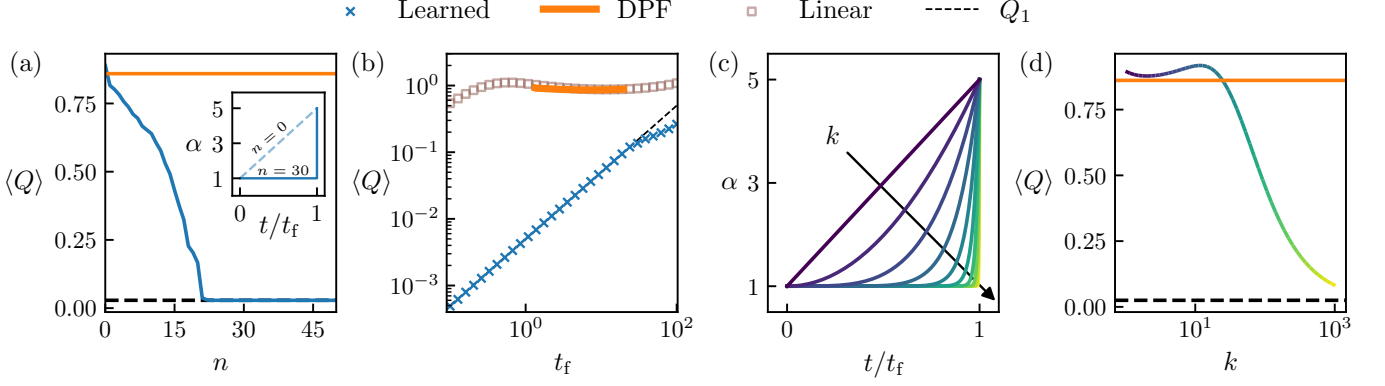


FIG. S2. Supplement to Fig. 1. (a) Heat $\langle Q \rangle$ produced by a neural-network protocol as a function of evolutionary training time n , for $D_1 = 0.01$ and $t_f = 5$ (blue). The learning algorithm is instructed to minimize heat. The orange line (“DPF”) is the result extracted from Ref. [49]; the dashed black line is Eq. (6). Inset: the neural-network protocol before training ($n = 0$) and after 30 steps of training. (b) As in Fig. 1(c), but now for $D_1 = 0.01$. (c) Smooth protocols of the family (S2) – with blue and green line shading indicating smaller or larger values of the exponent k , respectively – and (d) associated heat values for $t_f = 5$.

length $t_f = 5$. The heat produced by the protocol (S2) is not a monotonic function of k : a local minimum having heat values similar to those of Ref. [49] occurs at $k \approx 2$. For $k \gtrsim 25$ the protocol (S2) produces less heat than the protocols of Ref. [49], and for $k \gtrsim 10^3$ it produces heat similar to that produced by the final-time step function.

Eq. (7) of Ref. [49] represents a protocol $\alpha(t)$ as an expansion in terms of its time derivatives. Such an expansion is unable to represent rapidly-varying functions: for large k , near the inflection point $t \approx t_f$ of the protocol $\alpha_k(t)$, the n^{th} time derivative of (S2) (for $n \leq k$) scales as $k!/(n-k)!$, and so *increases* in size with n . An expansion such as Eq. (7) of Ref. [49], which neglects derivatives above $n = 2$, cannot represent such a protocol in a controlled way (the protocol $\alpha(t) \sim (t/t_f)^2$ is the strongest nonlinearity for which neglect of the third time derivative near $t = t_f$ is justified on the grounds of size).

Our conclusion from this section and Section II of the main text is that methods that can represent rapidly-varying (and potentially non-analytic) protocols are needed to optimally control active-matter systems, just as they are for passive-matter systems.

S3. PROTOCOLS FOR CONFINED INTERACTING ACTIVE BROWNIAN PARTICLES

In this section we supplement the results of Section IV.

Figs. S4 and S5 provide more detail of the protocols discussed in Section IV, for $N = 12$ and $N = 40$, respectively. The protocols for the control parameters λ and κ are shown in panel (a); panels (b) and (c) show the temporal evolution of the corresponding average work per particle and R^2 , respectively. In panels (d) and (e) we show the radial distribution function $P(r)$ and the

particle density at different times during the protocol.

We next consider the effect of increasing the value of t_f for the problem considered in that section. In panel (a) of Fig. S3, we show the work $\langle W \rangle$ (15) as a function of time for a system containing $N = 9$ active Brownian particles, and protocol lengths $t_f = 1$ and $t_f = 10$. The corresponding protocols for $\kappa(t)$ and $\lambda(t)$ are shown in panel (b). For $t_f = 1$, the protocol consists of increasing λ and κ to their maximum values instantly, and then reducing κ while the system is in an active state, in this way extracting work.

The protocol for $t_f = 10$ is able to extract much more work than the one for $t_f = 1$, and does so by changing the state of the system twice. The protocol starts off similar to the one for $t_f = 1$, but then decreases κ to its minimum value. Next, the system is brought back to the passive state by setting $\lambda = 0$, so that the final increase of κ to κ_f costs less work as R^2 decreases.

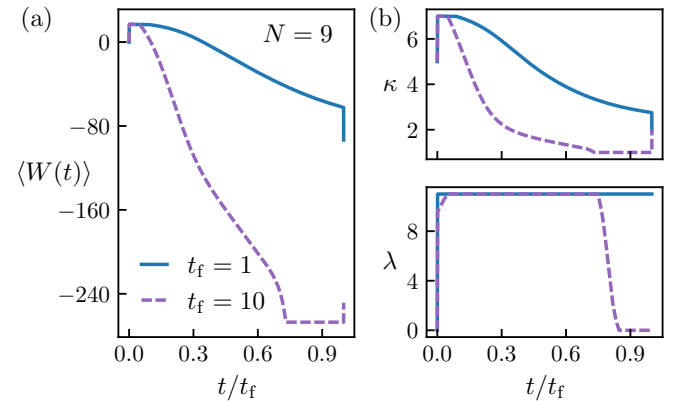


FIG. S3. (a) Work as a function of time for neural-network protocols that optimize the problem of Section IV, for $N = 9$ and two values of t_f . (b) Protocols for $\lambda(t)$ and $\kappa(t)$ corresponding to the work in panel (a).

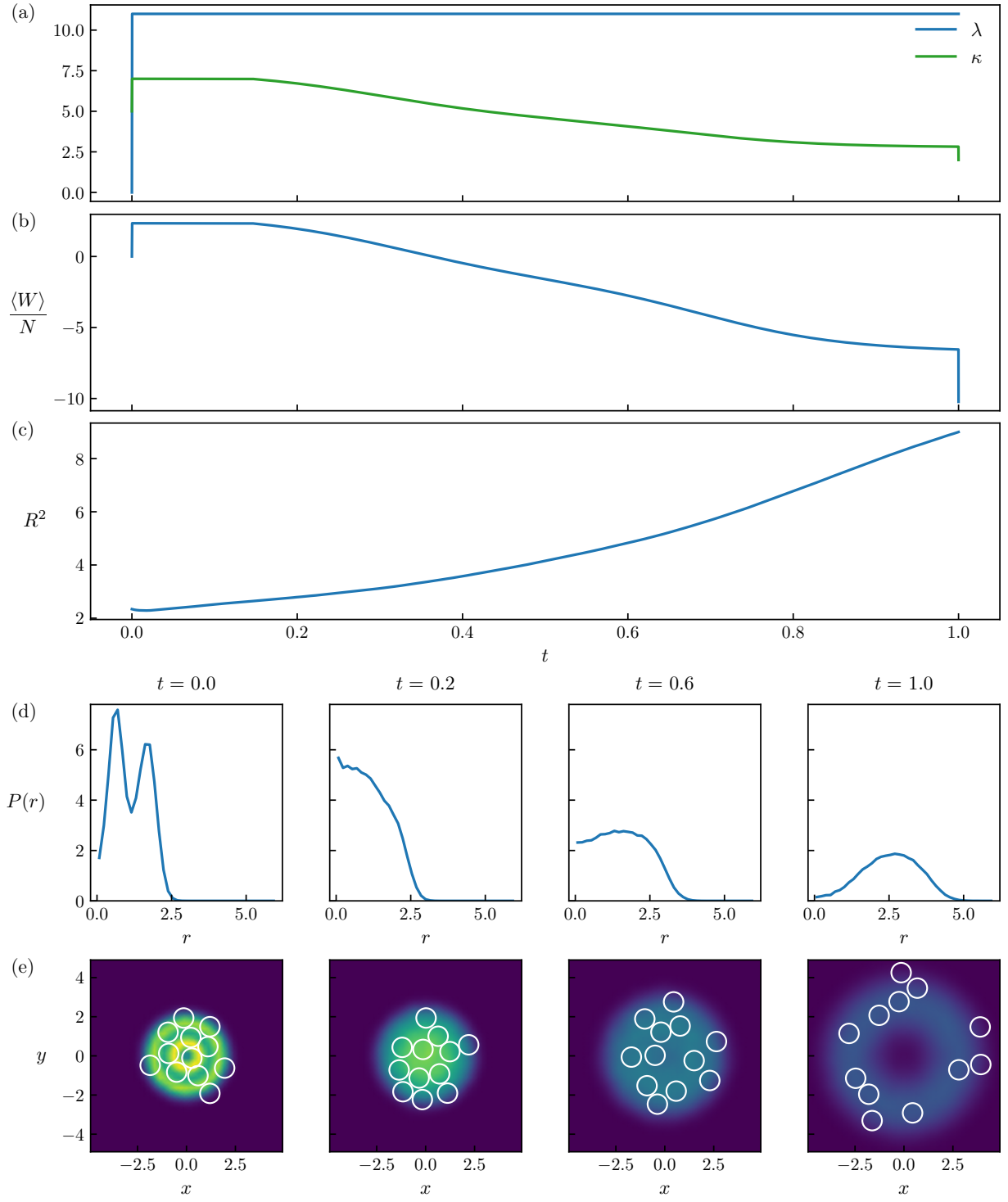
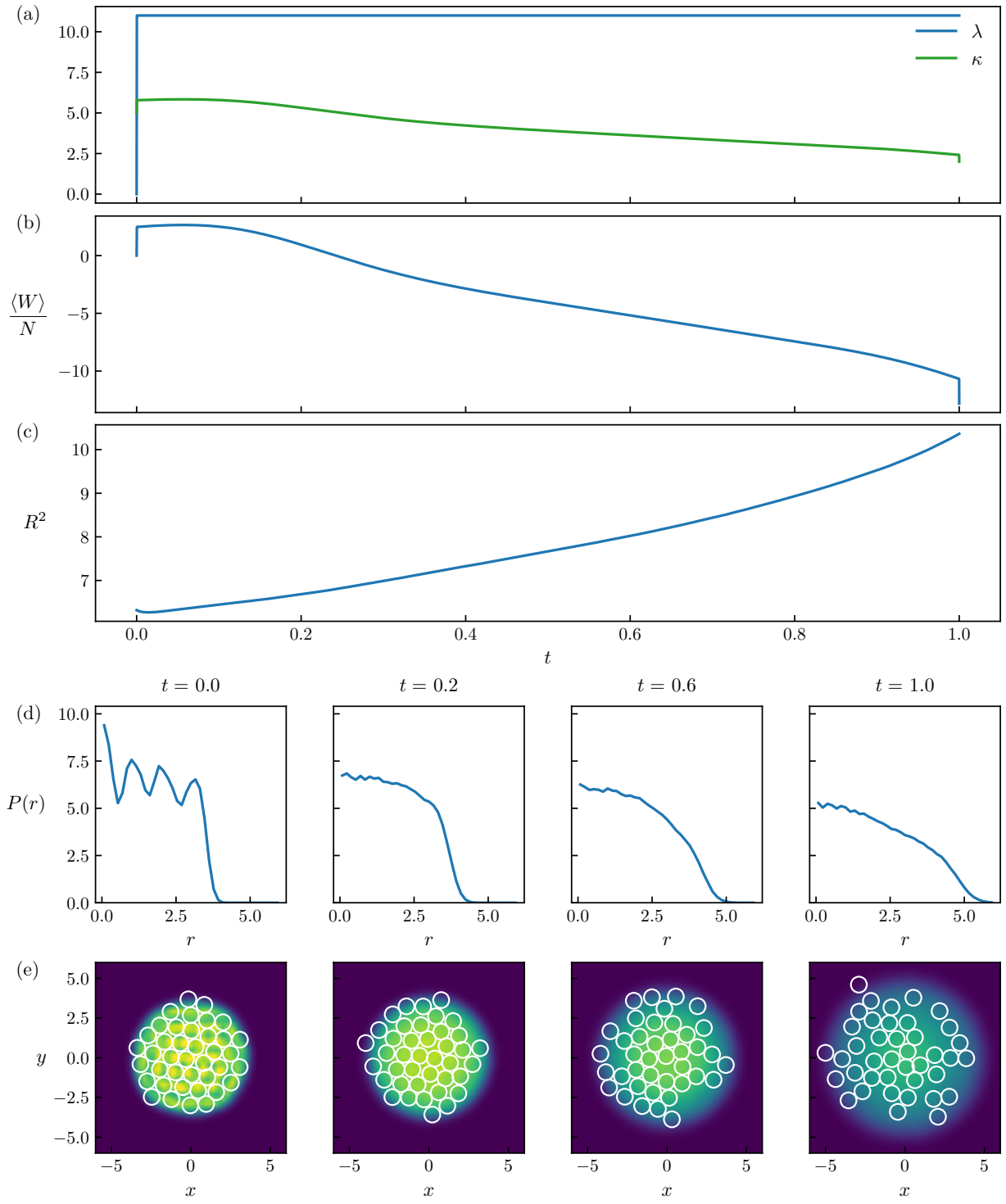


FIG. S4. (a) Control parameters, (b) average work per particle, and (c) R^2 as a function of time for the problem of Sec. IV and $N = 12$. (d) Radial distribution function and (e) particle density averaged over 10^4 realizations of the protocol. In panel (e) we show the particle positions for one of the trajectories.

FIG. S5. As Fig. S4, but now for $N = 40$.