

825G5: Adaptive Systems  
Final Project Report

**Abstract**

In the natural world, the evolutionary process provides organisms with adaptive capabilities in a completely decentralised and distributed manner; one in which the evaluation of genes is an implicit and embodied process directly affected by agents' behaviour. This is in contrast to the majority of artificial evolutionary algorithms developed for the purpose of automated design and optimisation which rely on centralised mechanisms of control. In this paper, we present an embodied evolutionary (EE) algorithm as a mechanism for the adaptive control of swarm robotics. Inspired by previous work by Watson et al. (2002), EE is presented as a cheap and effective method by which adaptation in a collective robotic system can be decentralised and autonomous, with candidate solutions evaluated in parallel. We take the work of Watson et al. (2002) further by testing the limits of the system's adaptive abilities and viewing its performance through the lens of W. Ross Ashby's early work on ultrastable systems, and exploring the potential for open-ended continuous adaptation. We find that our system can adapt to a range of environmental conditions and is moderately successful in its ability to adapt to large-scale perturbations. Our analyses provide us with preliminary insight into which parameters of the algorithm which determine its success and have pointed us towards opportunities for future research in this area.

**Introduction**

Acknowledging the role that embodiment plays in adaptive behaviour has proven to be particularly valuable to our understanding of intelligence, giving rise to significant advances in the field of robotics among other disciplines (Duffy & Joue, 2000). In this sense, the term embodiment refers to the physical instantiation of a system as situated (i.e. embedded) within an environmental context. An embodied system can therefore act in the world that it is 'structurally coupled' to, and behaviour arises as a result of a dynamic two way interaction between the agent and the environment (Ziemke, 2003). This is in contrast to more classical approaches in cognitive science which have previously

sought to analyse computation independent from the physical processes that give rise to such behaviour (Marr, 1982).

This contemporary school of thought has proven highly lucrative in the field of behaviour-based robotics, and can be seen as arguably rediscovering many of the foundational ideas in the field of ecological psychology such as ‘direct perception’ and exploitation of ‘environmental affordances’ (Gibson, 1972). In the words of Warren (2006), Gibson’s theory describes the way in which biology “*capitalizes on the regularities of the entire system as a means of ordering behaviour.*” In this way, factors other than rich computation on an individual level can produce complex behaviour by offloading much of the computational demands to the environment and utilising the inherent passive dynamics present within a system to deal with unpredictable environments (Clark, 2014).

To ‘adapt’ in this sense is defined as the ability of a system to instigate changes in response to perturbations so as to keep the essential variables of the system within a viable range. This definition is based on the body of work by W. Ross Ashby whose research, among that of other like-minded researchers in the mid-20th century, can be seen to have seeded a wide variety of disciplines including general systems theory and artificial intelligence. Inspired by the ideas relating to homeostasis, Ashby’s most famous work was his homeostat, an electronic device which explicated his theory of learning and adaptation (Cariani, 2009). The device acted to modify its own internal organisation in order to maintain specific ‘essential’ parameters within viable limits in the presence of external perturbations. He characterised such behaviour as adhering to a property known as ‘ultrastability’, and used this concept as a framework with which to view the tendency of all adaptive systems, such as biological organisms, to maintain physiological limits (like body temperature) under varying conditions (Izquierdo et al., 2013). Given this definition of adaptation, the concept of embodiment can therefore be seen as well-suited to the task of facilitating adaptive behaviour in light of the opportunity it presents for dealing with dynamic environments (i.e. amid perturbations).

In nature, we know that the ability to adapt to an environment is a central feature of successful organisms, with learning as a key underlying mechanism facilitating such a capacity. While we often think of learning as a process which instigates changes to an organism within its own lifetime, the evolutionary process can also be seen as a form of learning in which changes take effect over a larger

time scale of many generations (Watson & Szathmáry, 2016). The core principles of evolution, namely selection, variation and inheritance, combine to perform a directed search of genetic solutions which will yield optimal fitness for a population in a given environmental context. In this way, evolution can be thought of as a highly effective optimization process in which in situ evaluation of performance is performed over the course of an individual lifetime and is thereby intricately linked to the populations situated behaviour over generations (Watson et al., 2002). In the language of Ashby, this evolutionary learning can therefore be seen to modify internal organisation (e.g. genetic information) to keep the steady state of a species (i.e its survival) by maintaining its essential variables (such as food-intake of individuals). As demonstrated in the natural world, evolution as an embodied algorithm has proven able to exploit transient and often complex coupled dynamics between brain, behaviour and environment to produce adapted characteristics (Clark, 2014).

Such principles can be applied to artificial systems, and indeed evolutionary algorithms have shown to be a powerful design method to search such complex problem spaces (Harvey et al., 1997). Classical evolutionary robotics applies such principles to guide the automated design of robotic controllers (with evaluative robotic trials performed either in simulation or the real-world) through well-defined fitness functions and the control of environmental constraints (Nolfi & Floreano, 2000). However, this approach is hindered by the fact that a clear idea of goals and environmental context is needed *a priori* in order to direct the evolution of the controller. Furthermore, it is a computationally expensive and time consuming process to evaluate solutions sequentially (Alba, 2002).

A compelling alternative to these problems was presented in a seminal paper by Watson et al. (2002) in the form of an embodied evolution (EE) paradigm. EE involves the embodied distribution of the evolutionary process across a population of robots. This facilitates continuous online adaptation, meaning that the population can continue to acquire new capabilities once deployed, avoiding some of the problems associated with the reality gap (Jakobi et al., 1995). It is also an autonomous process in the sense that it does not require centralised control of the evolutionary process and is parallel in that multiple solutions are evaluated at once. It explores a novel method by which adaptive capability may be introduced to collective robotics, whereby very simple (inexpensive) robots act on local information without any need for a centralised control system coordinating complex behaviour.

Inspired by examples of collective behaviour in nature, such as the coordinated activity of social insect colonies, swarm robotic approaches offer an interesting avenue of research (Nouyan et al., 2009).

While Watson et al. (2002) presented some interesting preliminary results demonstrating the viability of this approach, the behaviour they evolved was very simple phototaxis behaviour which was then compared to a hand designed controller. The promise that EE presents in terms of offering a method by which open-ended adaptation can occur - that is the ability to *continue* to explore new behaviours - is not investigated in depth. Indeed, the literature surrounding embodied evolution does not address this ability in general (Bredecche et al., 2018). We wish to capitalise on this idea that EE presents an opportunity for open-ended adaptive capabilities, and aim to explore the limits of this adaptive system in response to disturbances. In other words, examine the algorithm's ability to facilitate adaptive behaviour in a robot collective within the framework of ultrastable systems.

## Methods

In order to test the performance of EE, we use a population of robots in a simulated environment (SituSim<sup>1</sup>) in which the task is to evolve foraging behaviours in response to food and poison sources placed within the environment. The following section details the simulation software before going on to describe the experimental set up and the specifics of the EE algorithm implementation.

### *I. Simulation environment*

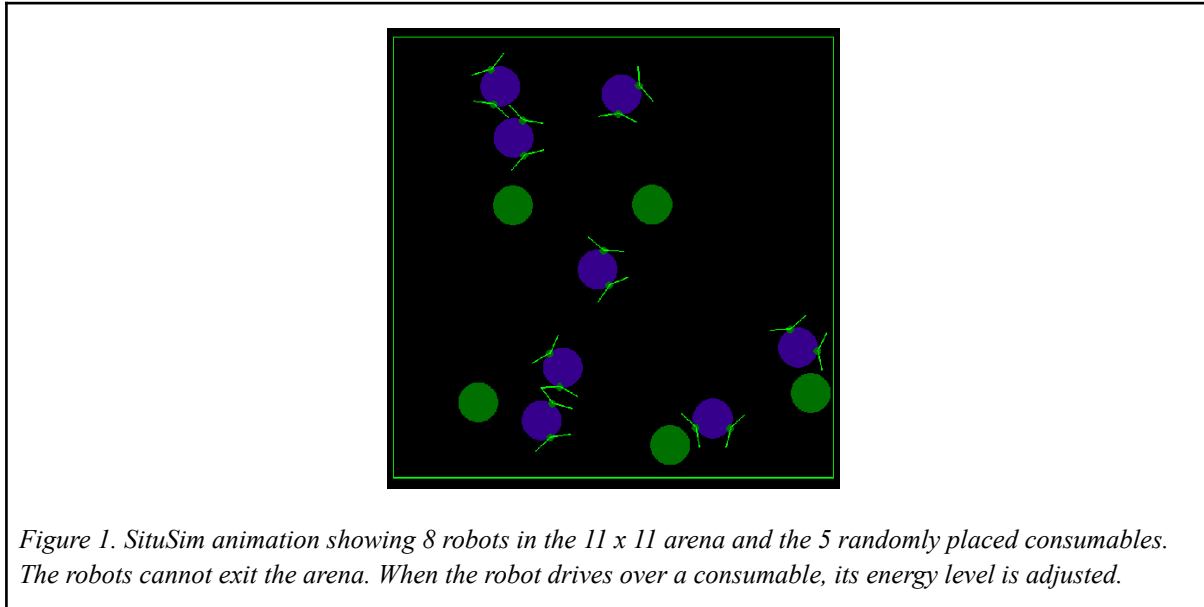
The simulation involves a 11 x 11 bounded square arena within which the robots can explore and navigate (see figure 1). Within the arena, two sources of stimulus can exist: food and poison. In all experiments, there are 5 food sources (each with radius 1) that are randomly placed within the arena where they will remain for the duration of a single simulated run. In order to provide a way in which these stimuli can be detected within the environment, they are modelled as light sources with which the strength of the light signal decays in proportion to the inverse of the squared distance to the source (see equation 1) - although this could easily be taken as modelling other sensory modalities

---

<sup>1</sup> Situated and Embodied Agent Simulation. This software is the property of the University of Sussex and is run on Python 3.

such as odour signals to be picked up by an agent's olfactory system. The food and poison sources will henceforth be described collectively as consumables.

$$\text{Signal strength} = \frac{1}{D^2} \quad (1)$$



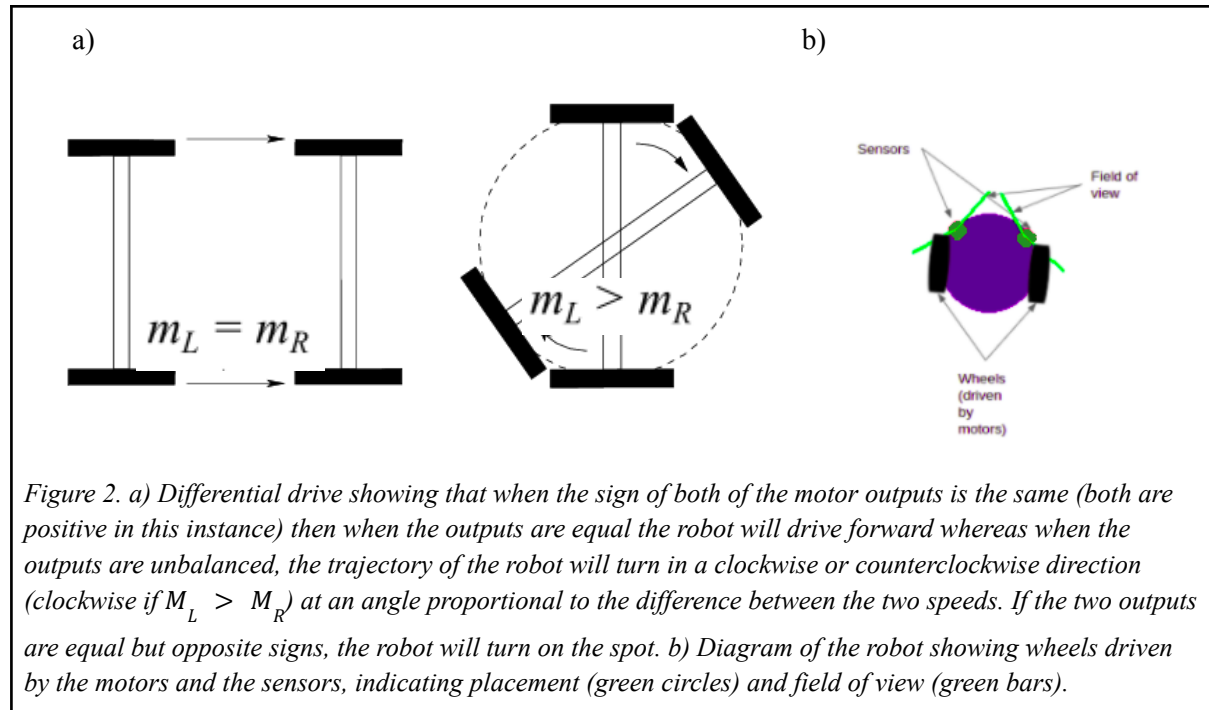
All robots in the simulation are initialised in a random place within the arena, with a starting energy value (which for all experiments is 1000). When the robot drives over a consumable stimulus its energy level is affected accordingly, with food sources acting to increase the robots energy by 10 and poison by -10. The food source is deactivated for a time of 5 discrete timesteps (which amounts to a time of 0.5 given that the system is iterated forward through time in steps of 0.1) after which it is reactivated and available to affect the energy levels of the next robot that drives over it. In Watson et al. (2002), every time the stimulus is reached, the robot independently initiates a reset procedure which moves the robot to a new random position in space. This reset procedure was much easier to implement in simulation, as once the consumable is reached, the robot is immediately transported to a new random position within the arena. If the robot's energy level drops to zero, it is unable to continue foraging (modelling death) and stops moving. An additional environmental constraint on the robot is that it expends energy at a rate proportional to the speed of its motors, therefore foraging is a necessary behaviour for survival (see equation 2). Its energy also decays at a fixed rate even when the robot does not move (0.01).

$$E_{t+1} = E_t - \lambda(v_L dt - v_R dt) - 0.01 \quad (2)$$

Where  $E$  is the energy of the robot,  $\lambda$  is the motor decay constant (0.05 for all experiments),  $v_l$  and  $v_r$  are the left and right motor speeds respectively and  $dt$  is the size of the timestep (which is 0.1).

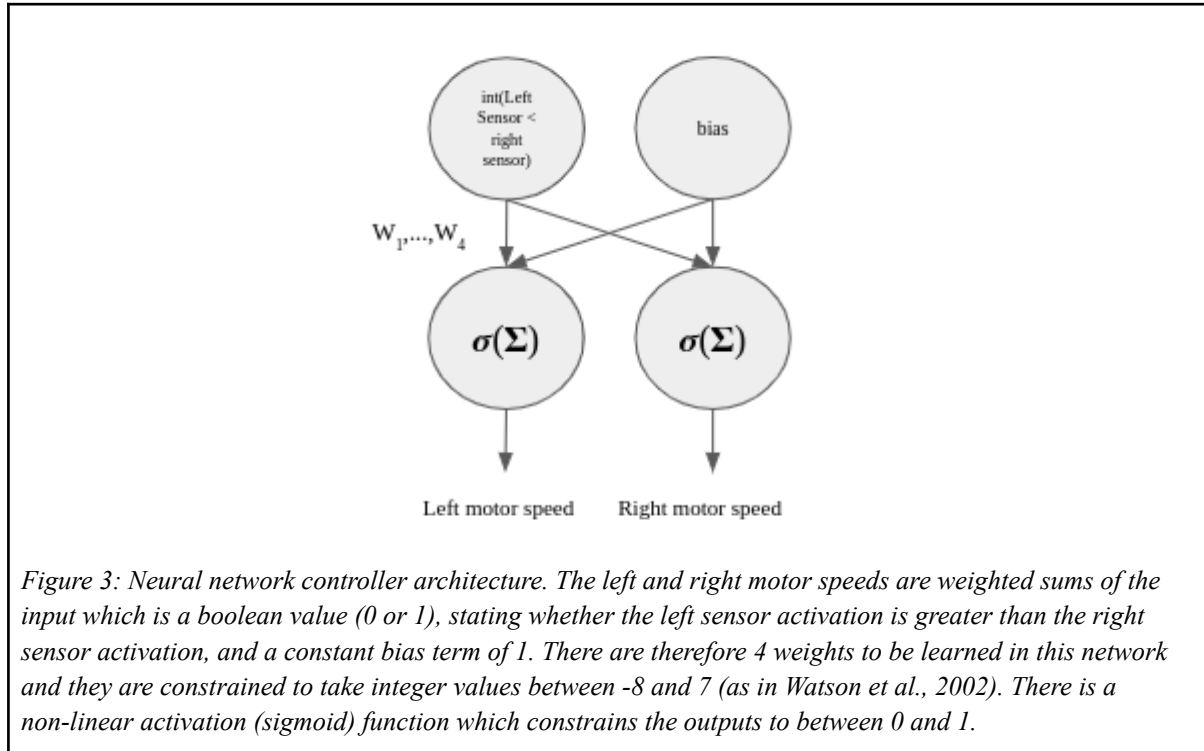
## II. Robots

The robots are circular (with a radius of 1) with differentially driven wheels on either side of the body, meaning that the direction of movement can be controlled by varying the speed of each of the wheels independently. For example, the robot can be made to turn on the spot when both motors are set to polar opposite equal speeds, while if both motors are equal it will drive in a straight line (see figure 2a). Each robot has two sensors which can pick up signals from the consumables, and they are located on the front of the robots body at an angle of  $0.33\pi$  radians to the left and right of the centre line and a field of view of  $0.8\pi$  radians for which outside this angle, they cannot sense the stimuli (see figure 2b).



The controller for each individual robot is a simple neural network in which the motor output is some function of the sensory input (see figure 3). The reason we chose a neural network for this

task is that they provide the perfect means by which the behaviour of the robot can be adjusted via a set of constrained parameters. By changing the weights within the network, the robot can be made to exhibit a wide range of behaviours in response to the food source, including the kind of phototaxis behaviours described in Braitenberg's (1987) thought experiments. The weights can therefore easily be translated into genetic information, with each weight in the network amounting to a single gene in a genotype and holding a value in a defined range.



### III. Embodied evolutionary algorithm

As in Watson et al. (2002), the evolutionary algorithm chosen to be implemented in this distributed scenario is based on the microbial genetic algorithm (MGA) developed by Harvey (2011). The reason for this is that this simplistic formulation translates well into the EE paradigm. The MGA is a variation on the classic steady-state genetic algorithm introduced by Holland (1975), and is inspired by horizontal gene transfer - a process of non-sexual reproduction that is observed in microbial organisms. Rather than the entire population being assessed for fitness and then the fittest individuals being selected as parents to create a new generation, Harvey's algorithm simply picks two individuals at random and pits them against one another (in a process known as tournament selection).

This is therefore well suited to the case of EE, given that selection can be easily implemented stochastically every time a robot falls within a given range of another robot, resulting in a selection process that is asynchronous and parallel. According to Harvey's formulation, a proportion of the genes of the less fit solution of the two is then overwritten by the fitter solution with some probability before being mutated at some predefined rate. This acts to secure in one succinct and efficient method both the idea of genetic recombination and elitism (which is the idea that the fittest solutions are retained in the population).

The following section details exactly how Harvey's MGA has been applied to the EE paradigm in our study, and this is based on the probabilistic gene transfer algorithm outlined in Watson et al (2002). Each robot in the population is initialised with a random set of 4 neural network weights with integer values between -8 and 7 (see figure 3). For the first set of experiments, this population size was 8, matching that of Watson et al. (2002). The robots move around the bounded arena on random trajectories (given that their controllers are initially badly tuned to the environment) and each time a robot collides with another there is a chance of horizontal gene transfer. We wanted the whole process of gene transfer to be completely decentralised, therefore in contrast to the MGA, when two individuals are selected (that is when a random 'collision' event occurs), the two fitnesses of the robots are not compared. Instead, each robot simply broadcasts a mutated version of its genotype with a probability proportional to its energy level. Each robot also receives a new genotype with a probability proportional to its energy level, with higher energy levels having a smaller probability of being overwritten and therefore the fitter solutions having greater resistance to being changed and remain stable in the population. This effectively implements an adaptive transmission rate in the hope this will aid convergence on optimal controller configurations and ensures that the communication requirements between agents remains minimal.

In order to translate the energy levels of the robot into a probability of gene transfer without any kind of global comparison of the overall range of fitnesses, we decided to use a sigmoid function which was tuned to the range in which we expected the robots energy levels to fall within.

$$Fitness = \frac{100}{1 + e^{(20 - 0.02x)}} \quad (1)$$



Where  $x$  is the current energy level of the robot. This puts each estimated fitness between 0 and 100 and therefore translates the energy level of the robot into a percentage chance of transmission, while  $100 - \text{fitness}$  gives the percentage chance of reception. Figure 4 shows how this sigmoid function is most sensitive to values around 1000 (which is the initial fitness level of all robots) and covers a range of around  $\pm 200$  either side of this value.

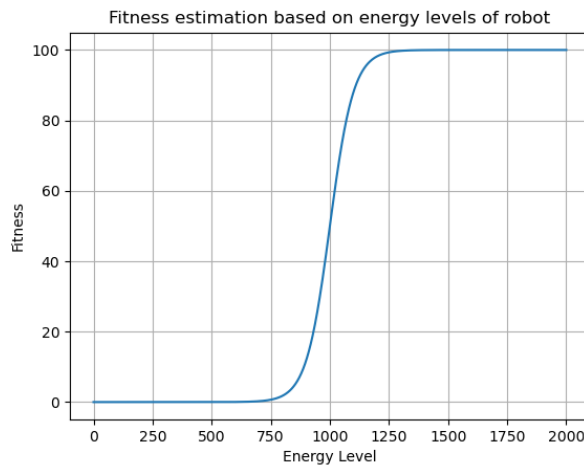


Figure 4: Sigmoid activation function which translates the robots energy levels into the range 0 to 100. Its peak sensitivity to input occurs at 1000 (which is the initial energy level of the robots). This function is described in equation 1.

## Results

### I. Adaptive capability

The first set of experiments we chose to conduct aimed to investigate the success of the system as a whole in terms of its ability to adapt to different environments.

#### A. Stimulus-seeking behaviour

The first scenario sought to investigate whether the system could adapt to produce ‘food-seeking’ behaviour. This involved randomly initialising the weights of 8 robots in the arena, each with an initial energy value of 1000, with 5 randomly placed food sources. Figure 5 shows the results of this first experiment, with figure 5b demonstrating that over an average of 10 independent trials, the system successfully adapted to the presence of food and produced food-seeking behaviour - as seen in the increase in the average energy of the population (with a final value of 2611 ( $\pm 48$ )).

SEM). For all 10 trials, the average energy of the population starts to increase before  $t = 5000$ , suggesting all trials managed to produce light seeking controllers within the allotted time.

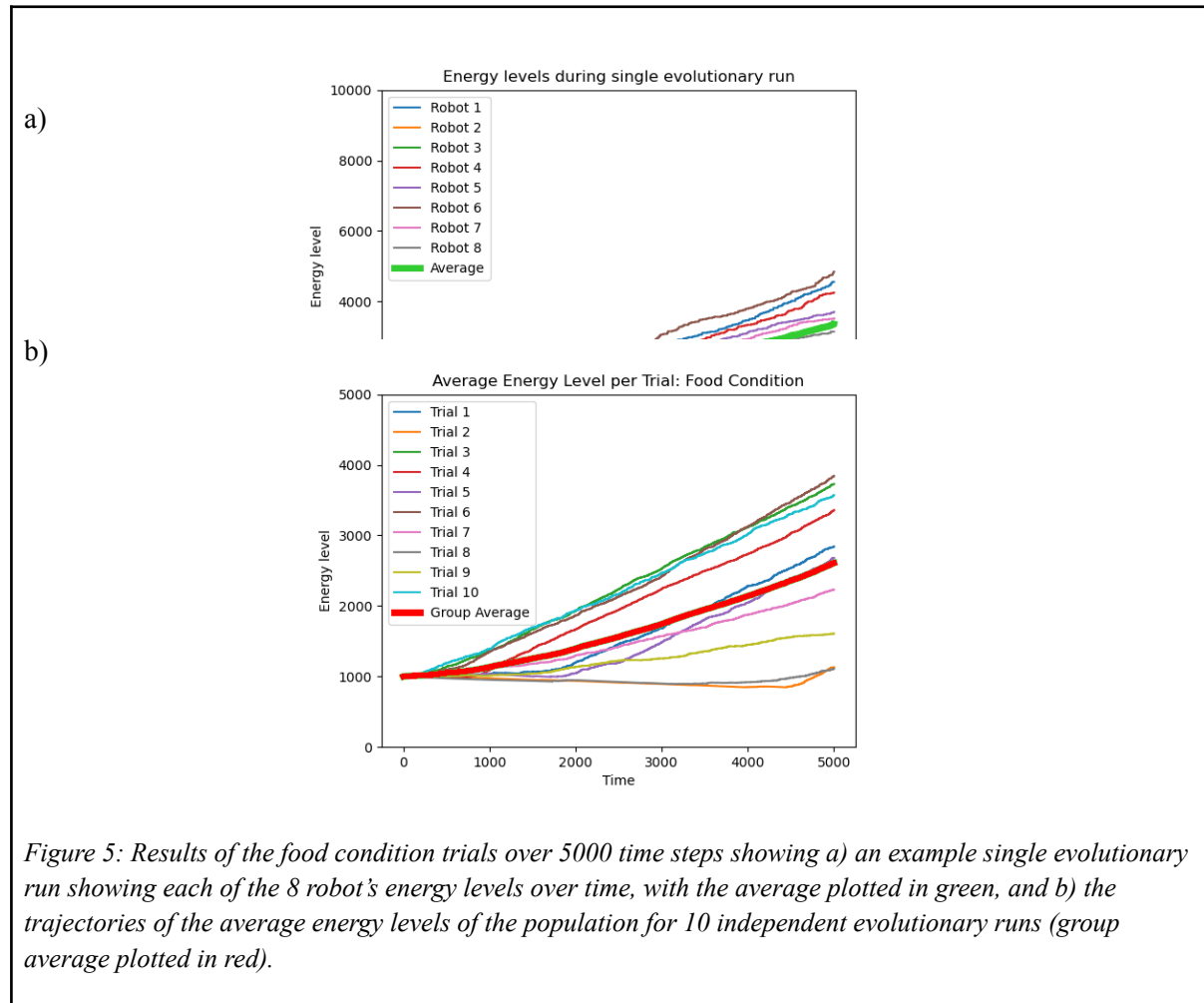
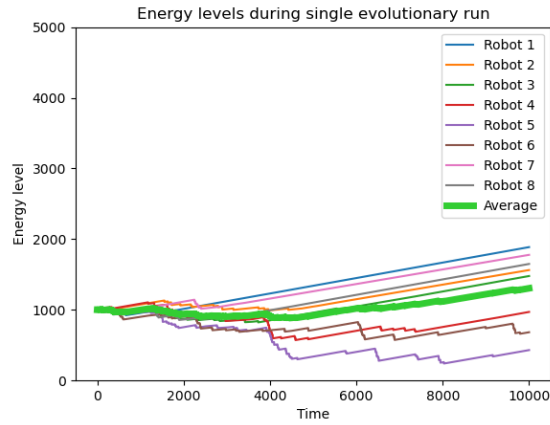


Figure 5: Results of the food condition trials over 5000 time steps showing a) an example single evolutionary run showing each of the 8 robot's energy levels over time, with the average plotted in green, and b) the trajectories of the average energy levels of the population for 10 independent evolutionary runs (group average plotted in red).

### B. Stimulus-avoiding behaviour

The second scenario investigated whether the population could evolve stimulus avoidance behaviours in the presence of poison in the environment. As with the effect on the energy levels of the robots, the decay constants were also reversed meaning that the robots energy levels recuperated for the time spent not encountering poison. This acted to incentivize the robot to keep moving, and gave us a measure of success. Figure 6 shows that while this task proved more difficult to adapt to, the population was indeed able to evolve poison-avoiding controllers, with a final average of 1126 (+/- 5) after 5000 time steps. Given the slow nature of the effect of the reversed decay constants, this took longer to evolve, and for trials 2 and 9 (see figure 6b), it seems as though the population had not evolved successful controllers given the low (and decaying) group average energy.

a)



b)

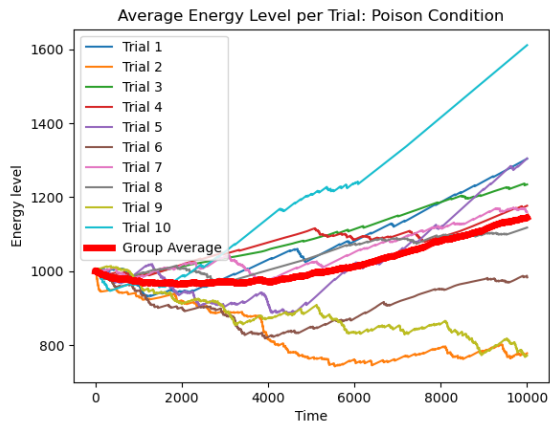
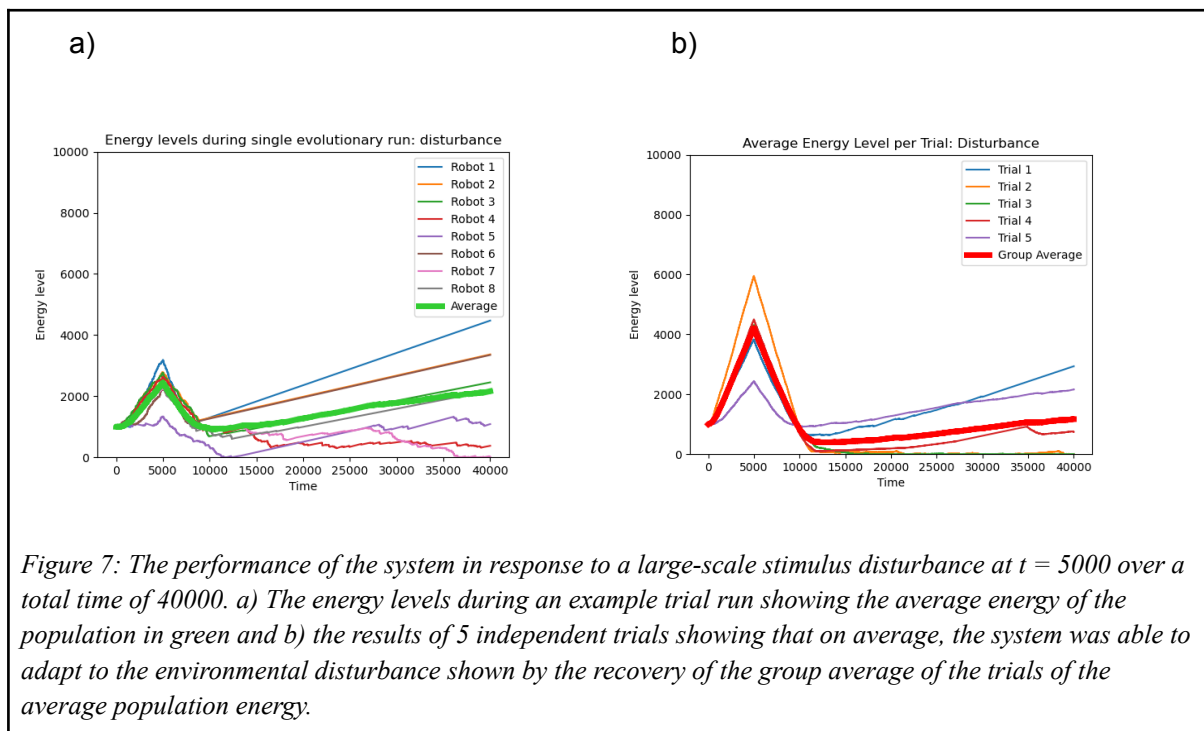


Figure 6: Results of the poison condition trials over 10000 time steps showing a) an example single evolutionary run showing each of the 8 robot's energy levels over time, with the average plotted in green, and b) the trajectories of the average energy levels of the population for 10 independent evolutionary runs (group average plotted in red). Note that the average increase in energy levels is much smaller than is seen during the food trials.

### C. Adapting to stimulus disturbances

While these first two experiments demonstrate the system is able to adapt to different environmental contexts, in order to explore the limits of this system and examine it through the lens of ultrastability, its ability to adapt to environmental disturbances during the course of a single evolutionary run must be interrogated. We therefore chose to allow the system to adapt to the food-seeking scenario, before switching the environment mid-trial, producing a large-scale disturbance of the system and forcing the population to try and adapt to the poison condition (and reversed decay rates) within a time of 40000. Any adaptation in this context can be seen as the system

trying to keep the essential variables (in this case the average energy level of the population) within viable limits (above zero and as high as possible). It is necessary to test the system in this way to ensure that the population is able to successfully adapt from a particular area occupied within weight space as opposed to just being able to evolve from randomly initialised starting weights. This presents a more challenging task given that after evolving controllers for a certain scenario, they will be informationally more similar and in a specific area of weight-space and therefore adapting ‘out’ of this area requires enough variation in the gene pool, and enough mutations in order to retain flexibility.



As figure 7 shows, the EE algorithm was on average able to successfully adapt to the disturbance. Considering the example run in figure 7a, we can see that initially, the system successfully evolves food-seeking behaviour, as seen by the increase in average energy from 1000 to 4208. The disturbance is then triggered at  $t = 5000$  and we see a sharp decrease in average energy given that the controllers are still tuned to be seeking the stimulus, yet now the stimulus acts to decrease rather than increase the robots energy level. By around  $t = 1200$ , the population seems to find a successful stimulus-avoidant controller configuration and the average energy gradually increases to a final value of 2161. Overall, figure 7b shows that from an average of 5 trials, on average the system

successfully adapts and regains energy levels (back to 1170). However, 2 out of 5 of the trials were unsuccessful (on average the robots fail to adapt and die), therefore the results are only moderately successful.

## *II. Parameter sweeps*

The second set of experiments aimed to investigate the dependence of the system's success on individual parameters in the EE algorithm, namely mutation rate, population size and the presence of an activation function (see table 1). For the following experiments, we chose to assess the effect of systematically varying each parameter by testing the effect on the success of the system in evolving food-seeking behaviours (as in figure 5). The success of the system can therefore be measured by the average maximum fitness of the population after a time of 5000.

Population size	Activation function	Mutation rate (no. genes)	Mutation size
4	Sigmoid	1	[-1,1]
8	Tanh	2	[-5,5]
16	None	3	
		4	

*Table 1. System parameters varied during the parameter sweeps experiments. Mutation rate refers to the number of genes that are mutated during broadcasting while mutation size refers to the range of magnitudes by which the gene is mutated (inclusive).*

### *A. Population size*

The first parameter we investigated was population size. Population size has been acknowledged as an important hyperparameter in evolutionary algorithms (Chen et al., 2012). Our initial experiments were carried out with a population size of 8 robots, as per Watson et al's (2002) study, as we concluded this size could be justified on the grounds that it allows for enough chance of reproduction given the arena area (and likelihood of collision) while being small enough to retain the benefits of a small population size in terms computational efficiency. The general consensus from the literature points towards a larger population enabling better exploration of the search space given the increased number of individual solutions (Chen et al., 2012). We therefore hypothesised that an

increase in the population size would positively affect the success of the EE system, and this is indeed what we found to be the case. Figure 8 shows the maximum energy values of an average of 10 independent runs for population sizes of 4, 8 and 16 robots.

While 16 robots produced the highest average maximum energy at the end of 5000 time steps as expected, it is worth considering the fact that the standard deviation was much greater, even overlapping with some of the more successful runs with a population of 8. Given that the doubling of the population size significantly increased the simulation time due to higher computational demands, one must factor in to what extent a large population is ‘worth’ this extra time. This trade-off can be likened to one that might also be made for financial factors when considering population size in real-world applications of swarm robotics.

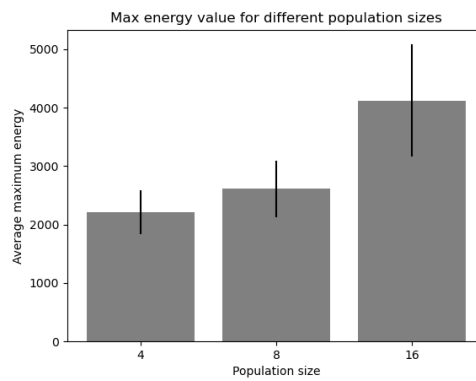


Figure 8: Different average maximum energy (+/- SD) values of the population taken as an average of 10 independent runs for 3 different population sizes (4, 8 and 16).

### B. Activation function

The second parameter we investigated was the presence of a non-linear activation function in the robots neural network controller (see figure 3). In Watson et al (2002), the authors chose not to use an activation function, however, as the results in figure 9 show, the presence of a sigmoid activation function in our case appeared to greatly increase the overall success of the adaptive ability of the system.

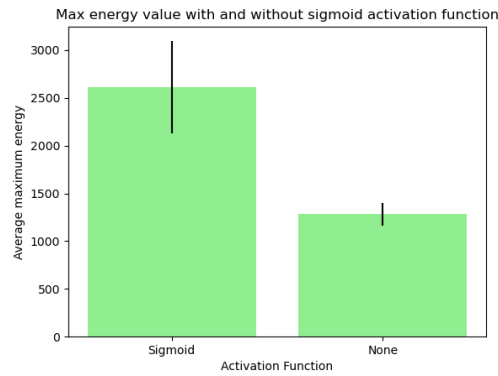
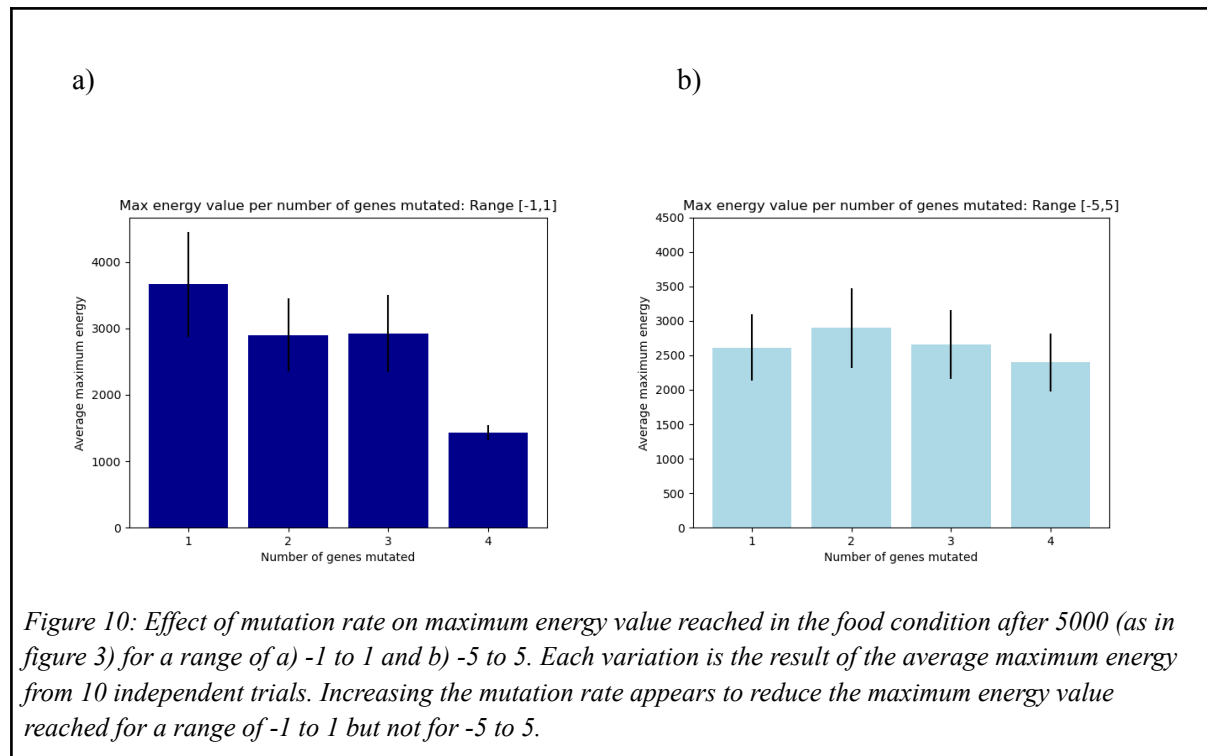


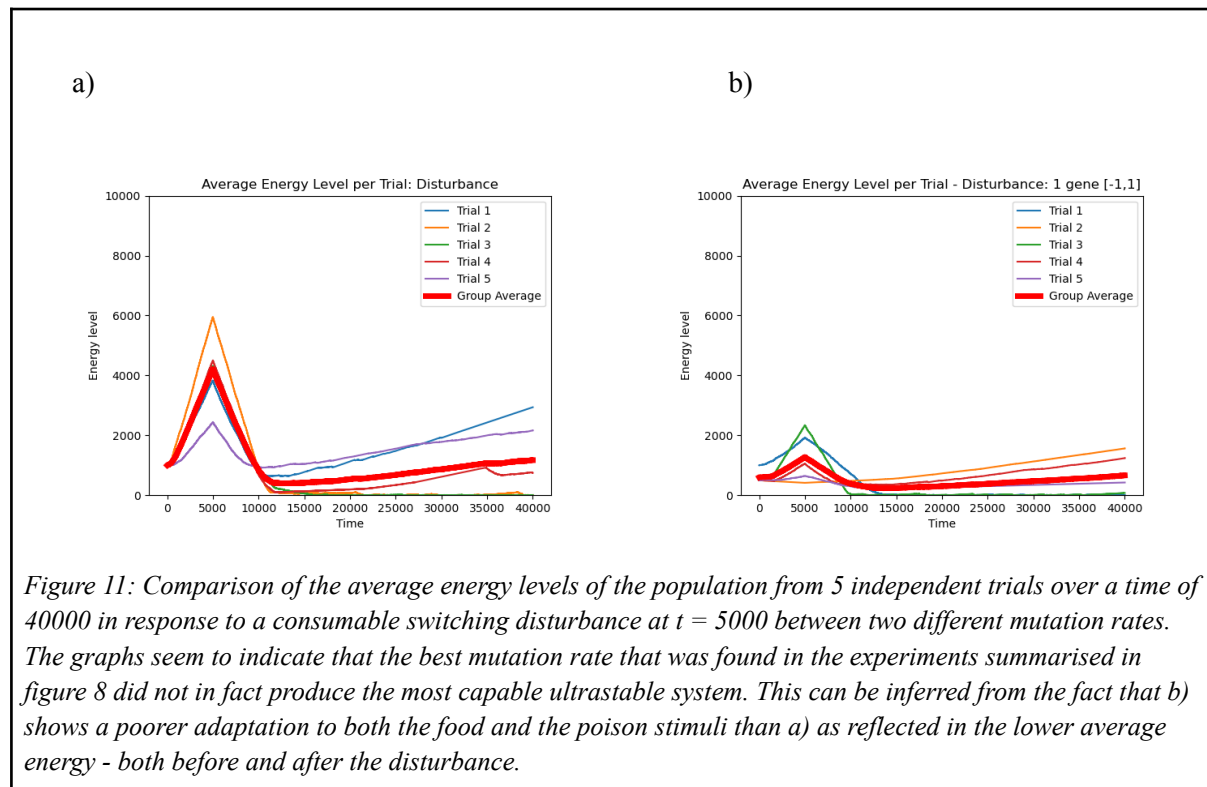
Figure 9: Different average maximum energy ( $\pm$  SD) values of the population taken as an average of 10 independent runs with and without sigmoid activation in the motor output.

### C. Mutation rate

The final parameter we investigated was mutation rate. There are two ways in which we can affect the mutation rate, the range and the number of genes that are mutated can be altered. In the initial set of experiments, one gene is picked at random and then mutated by adding a number between -5 and 5 inclusive (with uniform probability). A range of -1 to 1 was also tested. Figure 10 shows that increasing the number of genes being mutated each time during a broadcast significantly reduces the maximum energy level reached by the population, but only for those of a range -1 to 1. This variation produced an average maximum energy of 3661 ( $\pm$  780) while 4 genes led to a final value of 1426 ( $\pm$  114) after 5000 time steps. Interestingly, however, this same pattern was not observed in the average maximum energies of the mutations in the range -5 to 5 (see figure 10b).



It is worth noting that while this may produce the greatest success in the food-seeking condition, the more general adaptive capability of the system is not necessarily at its peak here. We therefore tested the most successful mutation rate (in this case 1 gene mutated within a range of -1 to 1) on the disturbance condition.





Indeed, figure 11 points towards a more complex relationship between adaptive success and mutation rate than the food-condition can inform us of given that the best mutation rate for the food-only scenario did not necessarily produce the most robust ultrastable system in response to disturbances.

## Discussion

Overall, it seems as though our EE system was successful in producing robotic controllers capable of performing foraging tasks. The system also showed moderate success in terms of its ability to perform open-ended adaptation and recover from large-scale environmental disturbances, demonstrating the characteristics of an ultrastable system. The results of our parameter sweeps appear to justify our use of an activation function in the motor outputs and provide useful insights into the choice of population size - given that the goal of EE was to produce a computationally efficient method by which continuous evolution could be implemented in a collective system (Bredeche et al., 2018).

However, the data collected from varying the mutation rate proved more difficult to interpret. It is not immediately obvious why mutating a greater proportion of genes had little effect on the overall success of the system when the magnitude of mutations were drawn from a larger range (-5 to 5). This is contrary to what we might have expected given what we know of too much variation leading to difficulty converging on optimal solutions, as the principle of elitism is not maintained (Rani et al., 2019). One potential interpretation might be due to the fitness function. If the energy levels of the robots increased rapidly enough in the beginning (because of the wider search of parameter space that larger mutations affords), the robots energy levels quickly entered the top of the sigmoid curve shown in figure 4, making each robot more resistant to receiving genes and therefore the large mutation rates did not affect the ability of the system to converge. However, this explanation is not wholly convincing and certainly needs further investigating. On top of this, the results presented in figure 11 also suggest that the selection of optimal mutation rates is dependent on the scenario the system is being assessed on, and that perhaps keeping a large mutation range is what allows for the system to be able to easily adapt to large-scale disturbances to the environment as it can perform a

wider search of the solution space. It is clear that further work exploring the effect of mutation rate and the fitness function is clearly warranted and provides an interesting opportunity for further research.

One of the most informative insights to be drawn from the analysis of our system was the importance of ensuring that agents did not become isolated from one another, thereby reducing reproduction and inhibiting the ability of the system to adapt. This criteria provides a possible explanation for why the addition of a sigmoid activation function in the robots neural network controllers improved the overall performance of the EE. We noted from qualitative observations of the simulations run with no activation function that the robots tended to get stuck in circular behavioural patterns, which reduces the chance of gene transmission as it reduces the distance covered by each robot around the arena. A sigmoid function introduces a degree of nonlinearity into the output of the system, and forces the motor output to take values between 0 and 1. The robots are therefore only ever going forward (or turning on the spot). Given that the two motor speeds can easily have opposing signs without the activation function, this increases the chance of tightly turning trajectories arising, and therefore reduces the overall success of the system given that genetic recombination is the foundation of the population's adaptive ability. However, it is worth noting that this requirement of continual contact between individuals may produce selective pressure on the system that could potentially interfere with the goal of the task (Watson et al., 2002). It also affects the kinds of disturbances the system can adapt to. For instance, in order to get our system to adapt to the poison condition, we had to also reverse the motor decay constant in order to incentivise the robots to continue moving, otherwise the robots would become reproductively isolated by evolving to remain stationary<sup>2</sup>.

A limitation of this simulation is that it does not model physical robot-robot interactions, in that the agents essentially drive over one another. This has important implications regarding the reality gap between our simulation and the behaviour of the system if the algorithm was implemented in a real-world setting. In Watson et al's (2002) study, the fact that real robots were used means that the de facto environment contains a certain degree of interference from other robots in the arena meaning

---

<sup>2</sup> See section I (B) of results

that successful adaptation involves the implicit goal of dealing with such interferences. In this respect, our simulation lacks a degree of ecological validity, therefore future efforts could work to incorporate this physical constraint into a more faithful model and analyse the effect that this has on the behaviour of the system.

## References

- Alba, E. (2002). Parallel evolutionary algorithms can achieve super-linear performance. *Inf. Process. Lett.* 82, 7–13. doi:10.1016/S0020-0190(01)00281-2
- Braitenberg, V. (1984). *Vehicles, experiments in synthetic psychology*. Cambridge, Mass.: MIT Press.
- Bredecche, N., Haasdijk, E., & Prieto, A. (2018). Embodied Evolution in Collective Robotics: A Review. *Frontiers In Robotics And AI*, 5. doi: 10.3389/frobt.2018.00012
- Cariani, P. (2009). The homeostat as embodiment of adaptive control. *International Journal Of General Systems*, 38(2), 139-154. doi: 10.1080/03081070802633593
- Clark, A. (2014). *Mindware* (2nd ed., pp. 118-141). New York: Oxford University Press.
- Chen, T., Tang, K., Chen, G., & Yao, X. (2012). A large population size can be unhelpful in evolutionary algorithms. *Theoretical Computer Science*, 436, 54-70. doi: 10.1016/j.tcs.2011.02.016
- Duffy, B. R., & Joue, G. (2000). Intelligent robots: the question of embodiment. Department of Computer Science, University College Dublin.
- Gibson, J.J. (1972). *A Theory of Direct Visual Perception*. In J. Royce, W. Rozenboom (eds.). *The Psychology of Knowing*. New York: Gordon & Breach.
- Harvey, I. (2011). The Microbial Genetic Algorithm. *Advances in Artificial Life. Darwin Meets von Neumann*, pp.126-133.
- Harvey, I., Husbands, P., Cliff, D., Thompson, A., & Jakobi, N. (1997). Evolutionary robotics: the Sussex approach. *Robotics And Autonomous Systems*, 20(2-4), 205-224. Doi: 10.1016/s0921-8890(96)00067-x
- Holland, J., (1975). *Adaptation in natural and artificial systems*. 1st ed. Ann Arbor: University of Michigan Press.

- Izquierdo, E., Aguilera, M., & Beer, R. (2013). Analysis of Ultrastability in Small Dynamical Recurrent Neural Networks. *Advances In Artificial Life, ECAL 2013*. doi: 10.7551/978-0-262-31709-2-ch008
- Jakobi, N., Husbands, P., and Harvey, I. (1995). Noise and the reality gap: the use of simulation in evolutionary robotics. *Lect. Notes Comput. Sci.* 929, 704–720.  
doi:10.1007/3-540-59496-5\_337
- Marr, D. (1982). *Vision* (p. 19). New York: Freeman.
- Nolfi, S., & Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence, and Technology*. Cambridge, MA: MIT Press.
- Nouyan, S., Groß, R., Bonani, M., Mondada, F., & Dorigo, M. (2009). Teamwork in self-organized robot colonies. *IEEE Trans. Evol. Comput.* 13, 695–711. doi: 10.1109/TEVC.2008.2011746
- Rani, S., Suri, B., & Goyal, R. (2019). On the Effectiveness of Using Elitist Genetic Algorithm in Mutation Testing. *Symmetry*, 11(9), 1145. doi: 10.3390/sym11091145
- Warren, W. (2006) The dynamics of action and perception. *Psychological Review*, 113(2), p358-389
- Watson, R., Ficici, S., & Pollack, J. (2002). Embodied Evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics And Autonomous Systems*, 39(1), 1-18. doi: 10.1016/s0921-8890(02)00170-7
- Watson, R., & Szathmáry, E. (2016). How Can Evolution Learn?. *Trends In Ecology & Evolution*, 31(2), 147-157. doi: 10.1016/j.tree.2015.11.009
- Ziemke, T. (2003). What's that Thing Called Embodiment? *Proceedings of the Annual Meeting of the Cognitive Science Society*, 25.