Simple Biologically-Inspired Embodied Agents: An Evaluation

Kristina Harper
Candidate No: 178621
K.Harper@sussex.ac.uk

6 March 2018

1 Introduction

Modern robots - from the Roomba to the Mars Rover - must master complex tasks in changing environments. Although Good-Old-Fashioned AI approaches to robotics centered on a sense-model-plan-act framework, the constant increase in complexity espoused by this framework may not be ideal, as nature tends to solve problems simply [7]. Instead, we will focus on a cybernetic approach that recognizes a dynamic relationship between a robotic agent and its environment.

The system in question is a simulated model of an agent on an infinite plane which has one light source located at Cartesian coordinates (0,10). The agent starts at (0,0) and has two actuators and a single sensor. The sensor reading and motor outputs have Weiner noise added at each iteration of the simulation. The agent has two tasks: reach the light source and return to the start point [6].

This report will analyze three biologically-inspired controllers designed to accomplish the first goal and one to accomplish the second, examine motor controller parameters, and evaluate how the parameters affect the agent's performance. Results will be analyzed within the context of the Artificial Intelligence field.

2 Methods

2.1 Method 1: Naive Sensory Perceiver

The first controller operated solely on sensory input readings. If the sensation variable read by the controller was positive, the agent moved straight. If the sensor read a negative value, the agent spun in a single direction. Different motor control values for straight and turn behavior were tested and evaluated based on whether and how quickly they accomplished each task. For each set of motor controls, the values remained consistent for straight and tumble behavior. When the agent went straight, its controls were [0.6,0.6] and when it spun, [-0.6,0.6]. Throughout the simulation, the agent has six internal states passed to the controller: boolean variables to track whether it reached the light source and start point, iteration numbers representing when the agent accomplished Task 1 and 2, the iteration number of the simulation, and a value to trigger a change in trajectory during return-home behavior.

If the agent reached approximately 1 distance unit of the light source (calculated with a sensor reading of 1 or more), it began Task 2. A simple reversal of the actuator values ([-10,10] to [10,-10] and [10,10] to [-10,-10]) predictably failed because the agent could approach the light source from any orientation. A foraging approach was then applied such that the agent went straight for 100 iterations, spun for 50 iterations, then went straight again. The pattern repeated until the simulation ended. For all methods, Task 1 must have been completed before Task 2 could be satisfied. Pseudocode for all methods can be found in Appendix A.

2.2 Method 2: Cytotaxis-based E.Coli-inspired bacterial controller

The second controller modeled the run-tumble bacterial behavior called cytotaxis, in which the organism spins to find an increasing concentration of a food source [7]. If the simulated agent detected a positive gradient in the light sensation, it went straight; if the gradient was negative, it spun. Once the agent reached approximately 1 distance unit from the light source, it transitioned to the same foraging pattern described above.

The agent utilized seven internal states in total: those discussed in Method 1 and the previous state's sensation reading to determine the gradient.

2.2.1 Method 2a: Cytotaxis with Randomized Spin

A slight variation of this light-localizing behavior was implemented. Instead of spinning in a single direction when the agent sensed a negative gradient, the agent randomized the direction of its spin with each iteration.

3 Experiments and Results

3.1 Method 1

When the agent's motor control values were less than 10, the agent rarely reached the light source as a result of interesting orbiting behavior and the simulation's time constraint. If the agent reached about 2 distance units from the light source, the agent would "get stuck" orbiting the light *(Figure 1)*.

The agent performed better when the motor controls had a high value [10,10], completing Task 1 94% and Task 2 37% of the time *(Table 1)*. For the same actuator values, the agent took on average 232 iterations to complete Task 1 and 342 for Task 2 *(Figure 5(a))*.

Method 1: Motor Controller Results		
Controller Values	Task 1 Completion	Task 2 Completion
(0.1, 0.1) (-0.1, 0.1)	4	1
(0.6, 0.6) (-0.6, 0.6)	6	0
(1, 1) (-1, 1)	51	3
(10, 10) (-10, 10)	94	37

Table 1: The success rate of the simple perceiver at each task for 100 runs of the simulation.The duration of time for each run was 500, and the agent's position was updated at each 0.1 time step, giving 5001 time steps total. The first set of controller values represents straight and the second, spin behavior.

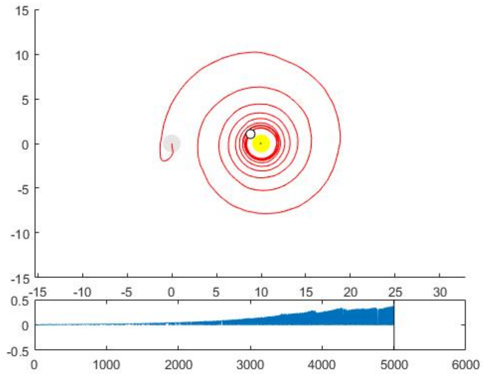


Figure 1: An example trajectory of Method 1 when motor controls were [0.6,0.6] and [-0.6,0.6]. The agent became stuck orbiting the light source.

Method 2: Motor Controller Results		
Controller Values	Task 1 Completion	Task 2 Completion
(0.1, 0.1) (-0.1, 0.1)	0	0
(0.6, 0.6) (-0.6, 0.6)	35	0
(1, 1) (-1, 1)	92	25
(10, 10) (-10, 10)	100	45

Table 2: The success rate of the cytotaxis-inspired controller for different motor controller values at each task for 100 runs.

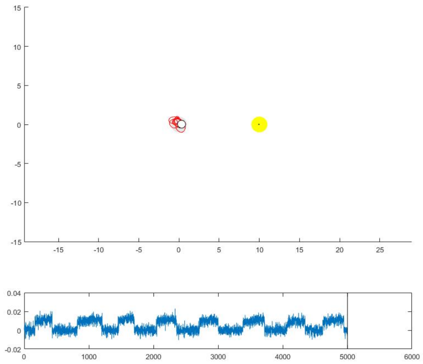
Method 2a: Motor Controller Results		
Controller Values	Task 1 Completion	Task 2 Completion
(0.1, 0.1) (-0.1, 0.1)	5	2
(0.6, 0.6) (-0.6, 0.6)	15	5
(1, 1) (-1, 1)	21	9
(10, 10) (-10, 10)	25	11

Table 3: The success rate of the cytotaxis-inspired controller for different motor controller values with randomized spin behavior for 100 runs.

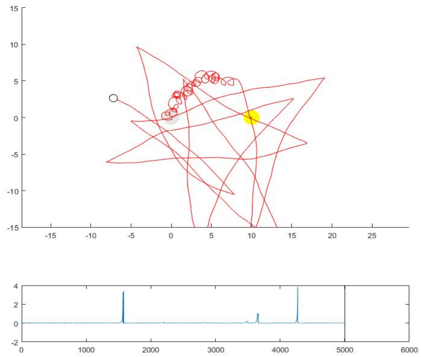
3.2 Method 2

When the motor controls were optimal out of those tested, the agent reliably reached the light 100% of the time *(Table 2)*. However, given the time constraints and the nature of the foraging behavior, it only achieved its second goal 45% of the time. When the motor controls were less than 0.5, the agent circled the start point and never completed Task 1 *(Figure 2(a), Figure 4(b))* .

With a greater motor actuator speed speed, the agent more reliably reached the light source *(Figure 2(b))*, providing time for Task 2 completion. The average time taken for the agent to accomplish Task 1 with controls [10,10] was 137 iterations; the Task 2 time was 251 *(Figure 5(b))*.



(a) Method 2: Controller Values (0.1,0.1)

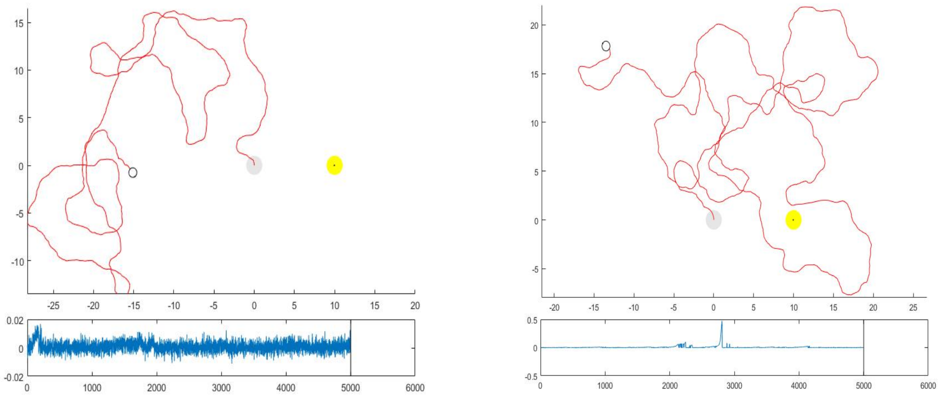


(b) Method 2: Controller Values (10,10)

Figure 2: (a) A sample trajectory of the cytotaxis-inspired controller with motor controller values [0.1,0.1] and [-0.1,0.1]. The agent never left the start point. (b) A sample trajectory of the cytotaxis-inspired controller with motor controller values [10,10] and [-10,10]. The agent successfully found the light source and start point.

3.2.1 Method 2a

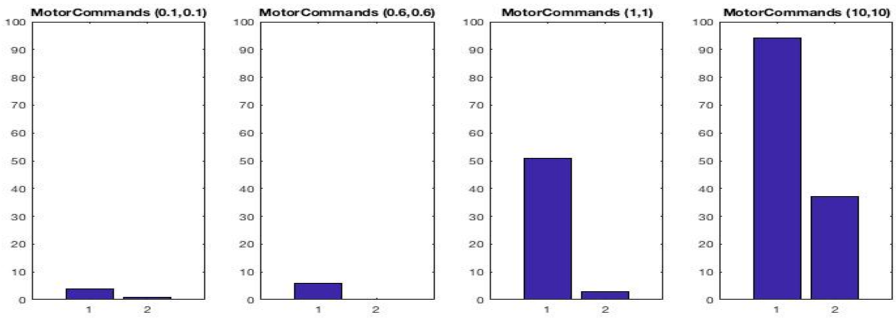
The agent rarely reached the light source even when the motor controls were set to [10,10]. Although the agent moved directly towards the light on successful runs, the agent's average Task 1 completion was only 25% at its highest *[Table 3]*. Because of the randomization in its turning direction, the agent could not reliably find a positive gradient. However, the time of its Task 1 (119) and 2 (220) completion, in the rare instances that it was successful, is comparable to the best motor control of Method 2 *(Figure 5(c))*.



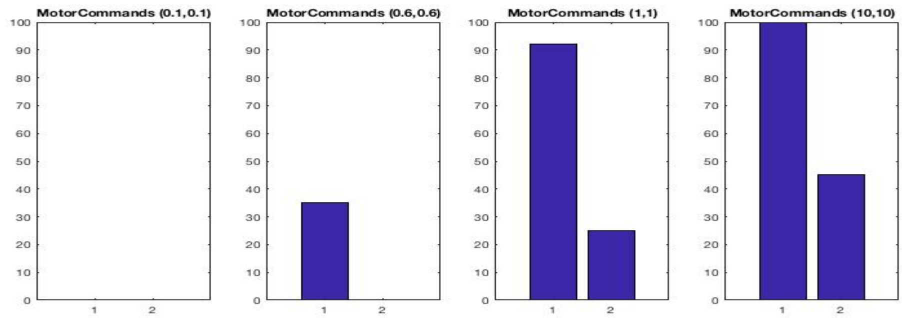
(a) Method 2a: Controller Values (1,1)

(b) Method 2a: Controller Values (10,10)

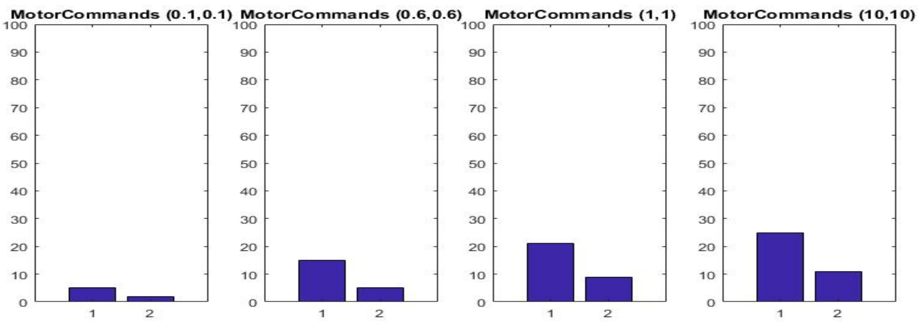
Figure 3: (a) A sample trajectory of the cytotaxis-inspired controller with randomized spin and motor controller values [1,1] and [-1,1]. (b) A sample trajectory of the randomized-spin cytotaxis controller with motor controller values [10,10] and [-10,10]. The agent seemed to randomly traverse the environment in both cases.



(a) Method 1

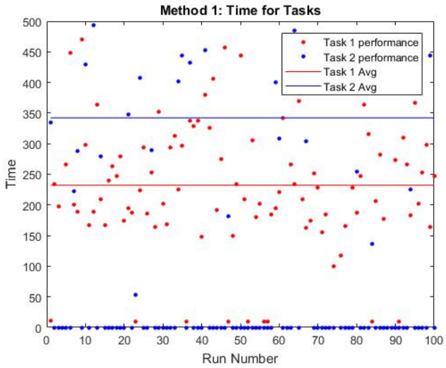


(b) Method 2

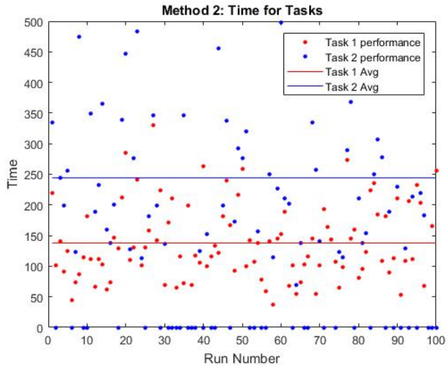


(c) Method 2a

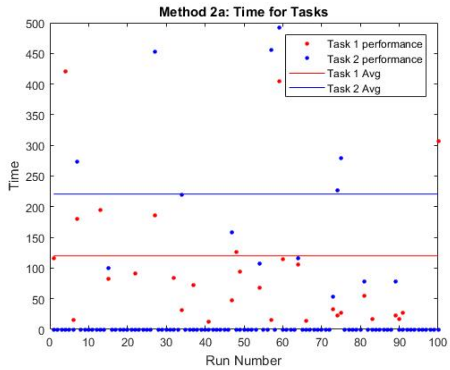
Figure 4: (a) Method 1 performance on both tasks increased as motor command values increased. (b) Method 2 reached 100% performance on Task 1. (c) Method 2a performance only slightly improved with higher motor control values. These graphs represent performance of each method over 100 runs of the simulation.



(a) Method 1: Simple Sensory Perceiver



(b) Method 2: Cytotaxis-Inspired Controller



(c) Method 2a: Cytotaxis with Randomized Spin

Figure 5: The time each task was accomplished for 100 simulation runs when each controller was set with actuator values [10,10] for straight and [-10,10] spin behavior. The average lines do not include runs when the agent did not achieve the given task. (a) Average Task 1 time was 232 and Task 2 was 342. (b) Average Task 1 time was 137 and Task 2 was 251. (c) Average Task 1 time was 119 and Task 2 was 220.

4 Discussion & Conclusions

The cytotaxis-inspired controller (Method 2) was the most successful at accomplishing both goals in the shortest time except in the [0.1,0.1] tests. Faster actuator speed improved the success rate of each task across all methods.

The interesting behavior difference between the cytotaxis-based controller and the randomized-spin implementation is representative of the difference between state determined and non-state determined systems. Method 2 can be considered state-determined because each new state is determined only by its previous state if the noise addition is ignored. If there were no noise and the controller always started at the same orientation, it would follow the same trajectory. This is not the case with Method 2a: randomness is included in the controller itself. So, the controller could approach and cross the same point from any number of pathways in the phase space [13]. From this angle of analysis we can conclude that the randomized gradient controller is a useful point of comparison, but unlikely to be appropriate for use in a real-world environment due to its unpredictability.

We can draw a broader conclusion from the experiment and simulation: even the simplest ''program'' can elicit complex, intelligent behavior. The agent displays behavior associated with living organisms, such as moving towards a food source based on high concentrations of scent in the air. This true-to-life embodied system, unlike many GOFAI symbolic reasoning implementations in abstract task domains, can be applied in the real world for localization of environmental pollution, for example [7].

This simulation can be considered intelligent under Brooks' behavior-based robotics criteria of Situatedness, Embodiment, Intelligence, and Emergence. The agent's sensations directly influence its behavior, and its body receives feedback on its sensations and therefore experiences the world through its internal state variables. Intelligence seems to emerge from the agent's successful localization of the light source and its search for its start point. Finally, the agent is emergent because it seems to be driven by goals through its interactions with the world [3].

Under the definition that ''All purposeful behavior may be considered to require negative-feedback,'' the agent does have a purpose: to find the light source using feedback. The agent also exhibits behavior consistent with goal-directed living organisms. Specifically, the storing of information from sensory input and using that information to determine action until a given goal is reached is present in all controllers developed for this report. Therefore, this simple agent could be considered intelligent [9][12].

5 Future Work

In the future, the controller could be adjusted to use an angle determined by its orientation in the world to better interact with its environment. Another potential improvement to better localize the agent is to use two sensors.

The high actuator speed which improves performance in the methodology above could create challenges in a physical robot, such as low accuracy, a high energy cost, or high likelihood of damage to the agent. These concerns lead to an important conclusion: simulations can provide a great deal of insight into the behavior of simple and complex agents, but testing the agent within a physical robot is necessary to provide essential information about environmental factors that could complicate or interfere with the agent's goal [8]. The difference between real and simulated input may limit what a hypothesis can explain [15]. The controller needs to be implemented in a real-world robot with a real-world light source to enable the properties of the environment to take effect on the agent.

References

[1] Bharat Bhushan and Madhusudan Singh. Adaptive control of nonlinear systems using bacterial foraging algorithm. *International Journal of Computer and Electrical Engineering*, 3(3):335, 2011.

[2] Valentino Braitenberg. *Vehicles: Experiments in synthetic psychology*. MIT press, 1986.

[3] Rodney A Brooks et al. Intelligence without reason. *Artificial intelligence: critical concepts*, 3:107–63, 1991.

[4] Ingemar J Cox. Blanche-an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on robotics and automation*, 7(2):193–204, 1991.

[5] H Jonas. Cybernetics and purpose: A critique. *The Phenomenon of Life: Toward a Philosophy of Biology*, pages 108–134, 2001.

[6] Simon McGregor. Adaptive systems mid-term coursework: Simulation model. *Adaptive Systems Module at University of Sussex*, 2017.

[7] John Oyekan and Huosheng Hu. Bacteria controller implementation on a physical platform for pollution monitoring. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3781–3786. IEEE, 2010.

[8] Rolf Pfeifer, Fumiya Iida, and Max Lungarella. Cognition from the bottom up: on biological inspiration, body morphology, and soft materials. *Trends in cognitive sciences*, 18(8):404–413, 2014.

[9] Arturo Rosenblueth, Norbert Wiener, and Julian Bigelow. Behavior, purpose and teleology. *Philosophy of science*, 10(1):18–24, 1943.

[10] Cezar Augusto Sierakowski and Leandro dos Santos Coelho. Path planning optimization for mobile robots based on bacteria colony approach. In *Applied soft computing technologies: The challenge of complexity*, pages 187–198. Springer, 2006.

[11] Herbert A Simon. Administrative behavior: A study of decision-making processes in administrative organization. 1957.

[12] John Thornton. Lecture notes in adaptive systems: Ashby, state-determined systems, and ultra-stability, March 2018.

[13] John Thornton. Lecture notes in adaptive systems: Early cybernetics, March 2018.

[14] Paul FMJ Verschure, Cyriel MA Pennartz, and Giovanni Pezzulo. The why, what, where, when and how of goal-directed choice: neuronal and computational principles. *Phil. Trans. R. Soc. B*, 369(1655):20130483, 2014.

[15] Barbara Webb. What does robotics offer animal behaviour? *Animal behaviour*, 60(5):545–558, 2000.

6 Appendix A

6.1 PseudoCode: Method 1

```
if sensation>0 and lightSourceReached==0
        go straight
elseif lightSourceReached==0
        spin

if lightSourceReached==1 and iterationNum+5<stopCondition
        go straight
elseif lightSourceReached==1
        spin
```

6.2 PseudoCode: Method 2

```
if lightGradient>0 and lightSourceReached==0
        go straight
elseif lightSourceReached==0
        spin

if lightSourceReached==1 and iterationNum+5>stopCondition
        go straight
elseif lightSourceReached==1
        spin
```

6.3 PseudoCode: Method 2a

```
randVar = random number between 0 and 1
if lightGradient>0 and lightSourceReached==0
        go straight
elseif lightSourceReached==0 and randVar>0.5
        spin left
elseif lightSourceReached==0 and randVar<0.5
        spin right

if lightSourceReached==1 and iterationNum+5>stopCondition
        go straight
elseif lightSourceReached==1
        spin
```