Introduction

For our final project, we decided to use Emily Dolson's paper, that is about a study of how organisms evolve depending on heterogeneous environments. The environments differ based on resources, as each environment is a combination of resources that are present in that environment. They used the Avida Digital Evolution Platform.

They found that there was strong overlap between hotspots (places where a particular trait might evolve more easily) and the evolution of that trait; however, the overlaps weren't universal, indicating that other environmental factors play a role.

The below figure represents the paths taken by lineages, representing which areas were avoided even in intermediate states. Our aim is to replicate the pattern of evolutionary hotspots (locations are better suited for particular tasks) correlating with higher evolution of traits.

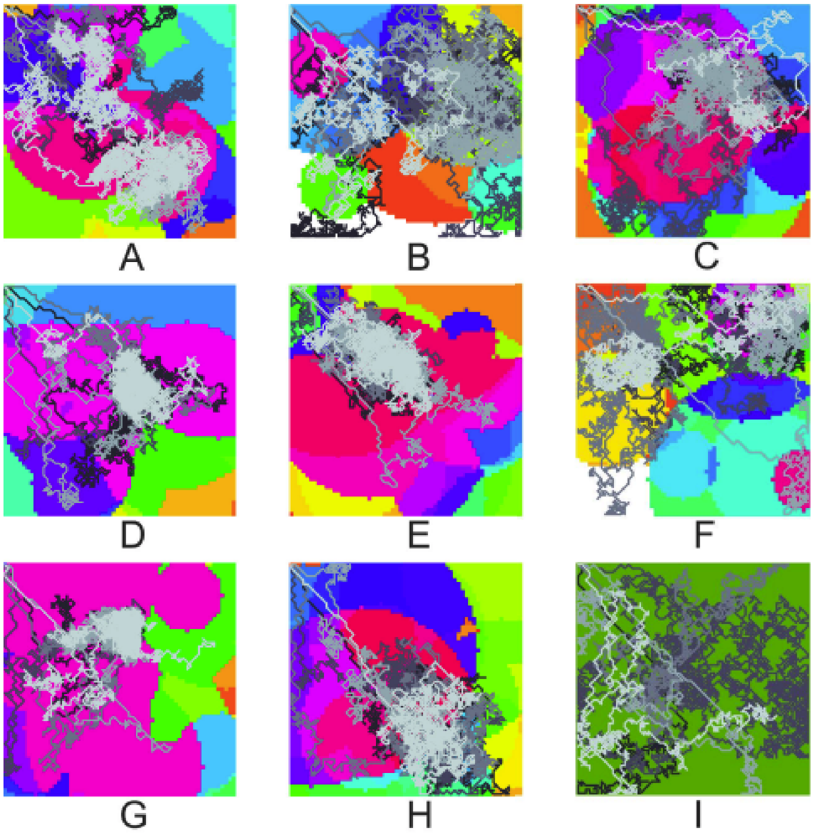


Figure 1. Spatial paths taken by lineages (Dolson et al.)

Summary of Original Paper Methodology (Dolson et al.)

1. Select eight environments for further study without examining them beforehand.
2. Perform 100 replicate runs of Avida.
3. Run each replicate for 100,000 updates, which generates approximately 2,000 generations.
4. Run 100 replicate runs of homogeneous control condition where all resources were globally available.
5. Extract the coordinates of the grid cell in which each trait first appeared within each run.
    a. Consider the first organism ever to display a given trait within a replicate because even though a trait may evolve independently multiple times within a run of Avida, including these subsequent evolutions would introduce various biases into the dataset.
6. Extracted coordinates translate into 81 separate point patterns. Each pattern contains a maximum of 100 points.
    a. There are 81 point patterns because there are patterns for 9 traits each for 8 environments and control.
    b. A lot of patterns contain fewer than 100 points, since it is not guaranteed that every trait will evolve in every replicate.
    c. Since each replicate is completely independent of each other replicate, the only relationship between points in the same pattern is the original environment that they evolved from.
7. Perform tests for spatial randomness
    a. In order to determine whether or not different regions have different evolutionary potential, we need to determine whether the point pattern is significantly different from complete spatial randomness.
    b. Test 1: A test to see whether each pattern, as a while, was random
    c. Test 2: A follow-up test to determine which regions of non-random patterns had significantly more points than expected by chance.

Test 1: A test to see whether each pattern, as a while, was random.

1. Calculate the test statistic for various distances, *r*.
2. For each point in the pattern, count the number of other points falling within *r* distance units of it
3. Average this number across all points to get out test statistic for that value of *r*.
4. For each point pattern, generate a range of expected results under the null hypothesis of complete spatial randomness.

    a. This is done by running 100 Monte Carlo simulations, the number of points randomly placed in our actual data across an area of the same size.

5. If the value of the test statistic from the observed data was greater than the highest value from the Monte Carlo simulation for more than 5% of the *r* values, conclude that the observed pattern was substantially more clumped than we would expect to see by chance.
6. If the value of the test statistic was below the range of the observed data from the Monte Carlo simulation, conclude that the observed data was more uniform than we would expect by chance. This result is unexpected for the experiments.

Test 2: A follow-up test to determine which regions of non-random patterns had significantly more points than expected by chance. (Determine the hotspot)

1. Calculate the kernel intensity surface (a heat map of point density across the environment) for each of these point patterns using the spatstat library.
2. Perform a Monte Carlo hypothesis test with 100,000 simulations to determine whether the patterns within the data were significantly more intense than expected from a random process.
3. Simulate patterns generated under the assumption of complete spatial randomness by randomly selecting points from a uniform 60x60 grid, which approximates the grid in Avida.
4. For each point pattern, select the number of points observed in the original pattern.
5. Create Kernel intensity values for each of the simulations.
6. For each cell in the grid, compare experimentally-derived intensity to the distribution of simulated intensity values.
7. Lavel cells in which our observed value was higher than 99.999% of the simulated values as hotspots of evolutionary potential for a given trait.
    a. The threshold of 0.0001 is used to ensure all results are significant after a sequential Bonferroni correction for multiple comparisons across all 3600 grid cells.

Summary of Original Paper Results (Dolson et al.)

They found that there was strong overlap between hotspots (places where a particular trait might evolve more easily, due to more resources) and the evolution of that trait; however, the overlaps weren't universal, indicating that other environmental factors play a role.

Some amount of overlap was expected for three different reasons:

1. If any trait is easier to evolve in a given region, then lineages living in that region are more likely to be successful as a result of having evolved the trait, giving them an older evolutionary history.
2. Many traits serve as building blocks for other traits, so possessing some traits is an indicator that a lineage is in a part of the fitness landscape from which other traits are easier to reach.
3. Some traits are close to each other within the fitness landscape. (For example, XOR and EQU are the two most complex traits, and they often have hotspots in nearly identical locations.

So what causes this pattern? The conclusion was that since all traits can be building blocks for each other, it would make sense that the locations with more resources have a higher evolutionary potential, making them the hotspots.

Many hotspots overlap with borders between different combinations of resources. These regions are likely to have more local diversity, as different phenotypes will adapt differently in different environments. Since the given assumption is that diversity leads to increased evolutionary potential, variation in local diversity is another possible reason for these patterns and evolutionary hotspots.

Summary of Our Methodology

Our reimplementation of the experiment is a much more simple version of the paper, as we decided to focus only on how the hotspot moves around the grid after running the tasks and show its movements on the screen. Since we aren't using Avida, we had to implement our own versions of the nine standard basic tasks. The nine tasks each had points the organisms could earn when solved. The code for the nine tasks and the corresponding points can be found in Task.h.

- The organisms gained 30 points when the **NOT** task was solved
- For all other tasks, they gain 15 points

The original paper ran for 100,000 updates, which was estimated to be 2,000 generations. In order to approximate the same number of generations over the same number of generations, we determined that organisms should be replicating around every 50 updates. Organisms very, very rarely solve the NOT task, so the functional maximum number of points they get from solving a task would be 15 points. Thus, if they solve one task per update, the "reproduction threshold" would need to be 750 points in order for them to reproduce every 50 updates. However, running with this value produced organisms that almost never completed tasks, so we decided to go in favor of maximizing new behavior, and set the resource threshold low enough that organisms could reproduce every two tasks solved. Hotspots were defined by increasing the number of cpu cycles; this is validated below (in **baseline**) to ensure it increases the point gain (selective advantage) of hotspot cells.

We implemented two extensions; the first was various reproduction point thresholds, controlled by a configuration value. This might widen the gap between adaptive and non-adaptive organisms. The second extension further expanded the results by including a moving hotspot; if hotspot benefits are localized, this might be a mechanism of increasing population-wide ability, rather than only forming a niche.

Summary of Our Results

We set out to ask two main questions:

1) Would hotspots affect which tasks are most likely to be completed?
2) Would hotspots affect the completion rate of tasks in general?

Baseline

We ran the following experiment (**baseline**):

- 100,000 updates
- All tasks (except for NOT) return 15 points upon successful completion
- All tasks are included (all 9)
- Organisms require 20 points in order to reproduce (so organisms need to successfully complete two tasks in order to reproduce)
- The hotspot is fixed, and increases the number of CPU cycles by a factor of four for organisms within it
- 11 random seeds–range(10,21)
- A full starting population (400 orgs)

Firstly, we confirm that there is a selective difference between hotspot and non-hotspot cells:

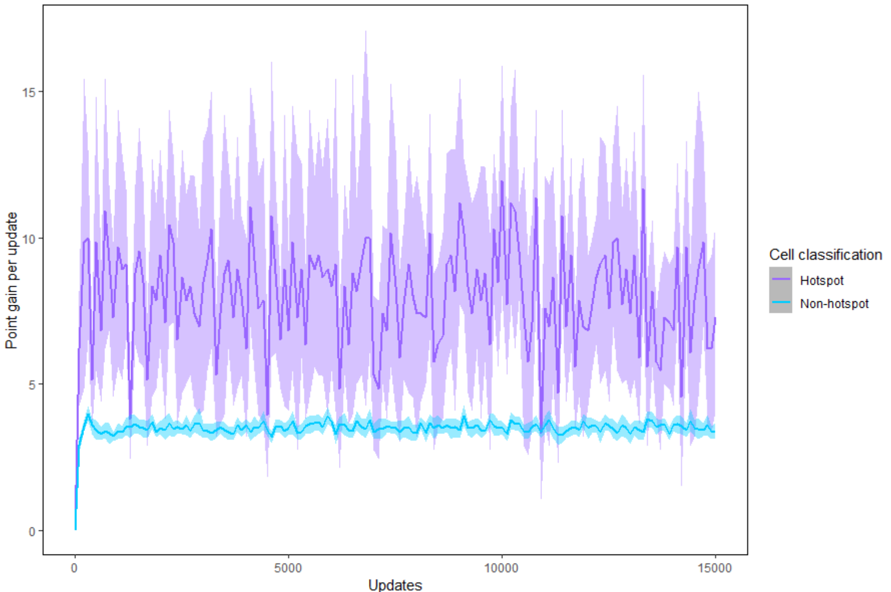


Figure 2: Points gained per organism per update. Organisms in hotspot cells, who run 4x as many cpu cycles, gain significantly more points per update than non-hotspot organisms. These lines are averaged across the 11

replicates; shading represents the confidence interval.

Evidently, organisms in the hotspot receive more points per update, on average, than their non-hotspot counterparts. By the above curve, hotspot organisms should reproduce every two to three updates, and non-hotspot cells every four to five. The next question would be which tasks are most often executed.

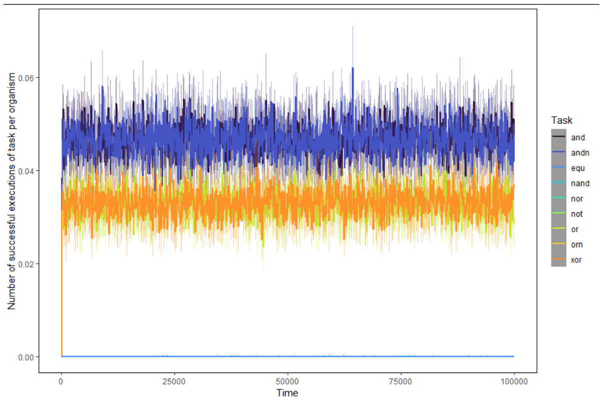
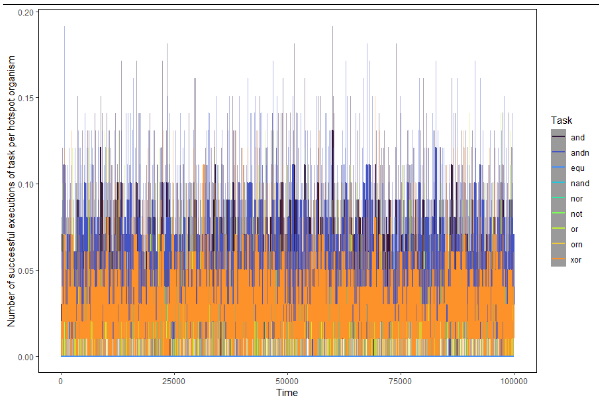
Non-hotspot cells	Hotspot cells
	

Table 1: Task completions per non-hotspot and hotspot organisms. All tasks return 15 points upon success, except for NOT, which returns 30. 20 points are required for reproduction. These lines are averaged across the 11 replicates; shading represents the confidence interval.

It appears that there are two groups of tasks, some which are more likely to be executed (e.g. AND) and some which are less likely to be executed (e.g. XOR). The hotspot didn't significantly affect which tasks were executed; the trends for which tasks were more likely to be executed were the same. However, we see a wider stratification of tasks in non-hotspot cells; the smaller distribution in hotspot cells is likely due to small sample size. Additionally, hotspot cells do indeed have higher rates of task completion. We don't see particularly different task profiles (i.e. which tasks are solved) between hotspot and non-hotspot cells; given that tasks (other than NOT) are equally rewarded, this establishes baseline behavior.

To explore whether differences in successful task distribution would develop, we tried modifying how many resources each would be returned with two tasks.

Better AND

We ran the following experiment (**better AND**):

- Identical conditions as **baseline**, except:
- Successful completion of "AND" returns 60 points, 4x as many as returned by most other tasks.

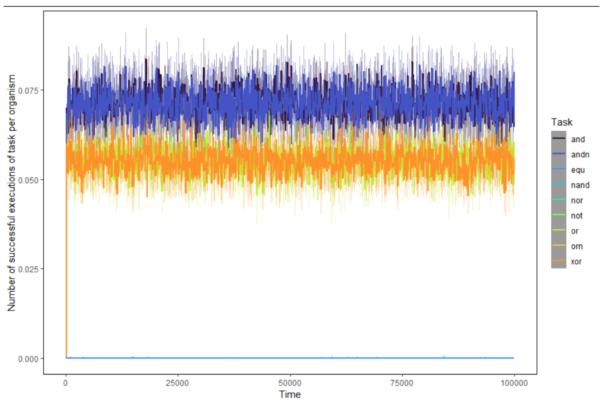
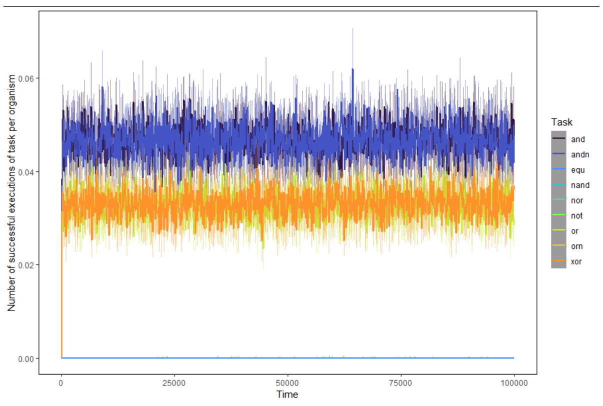
Non-hotspot cells	Hotspot cells
	

Table 2: Task completions per non-hotspot and hotspot organisms. All tasks return 15 points upon success, except for NOT, which returns 30, and AND, which returns 60. 20 points are required for reproduction. These lines are averaged across the 11 replicates; shading represents the confidence interval.

We see that the incentivized AND task produces a slightly higher rate of successful task executions compared to baseline, but perhaps slightly fewer in hotspot cells. Perhaps this is due to the significant fitness of "AND;" that task alone was sufficient for reproduction in one turn. We then tried with an "harder" (less common) task to see if it would have a different effect.

Better XOR

We ran the following experiment (**better XOR**):

- Identical conditions as **baseline**, except:
- Successful completion of “XOR” returns 60 points, 4x as many as returned by most other tasks.

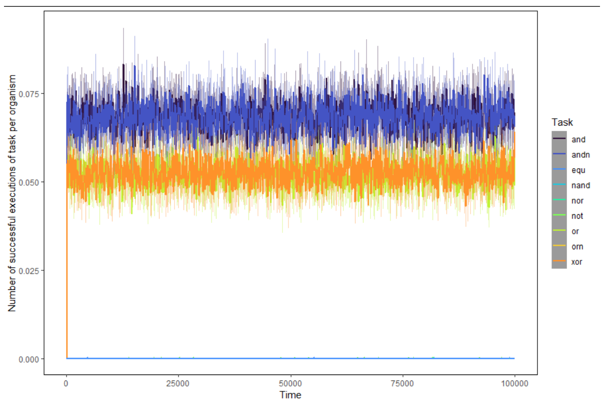
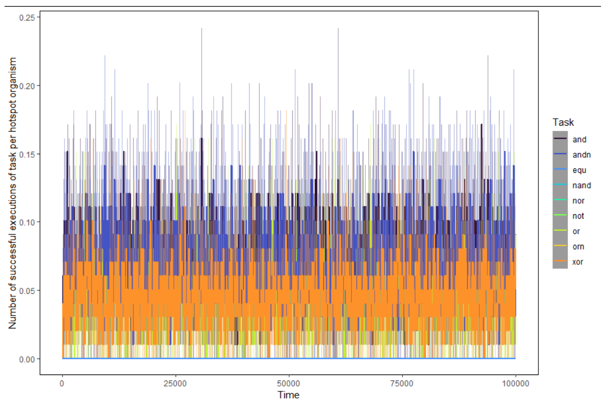
<table>
<tr><th>Non-hotspot cells</th><th>Hotspot cells</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Table 3: Task completions per non-hotspot and hotspot organisms. All tasks return 15 points upon success, except for NOT, which returns 30, and XOR, which returns 60. 20 points are required for reproduction. These lines are averaged across the 11 replicates; shading represents the confidence interval.

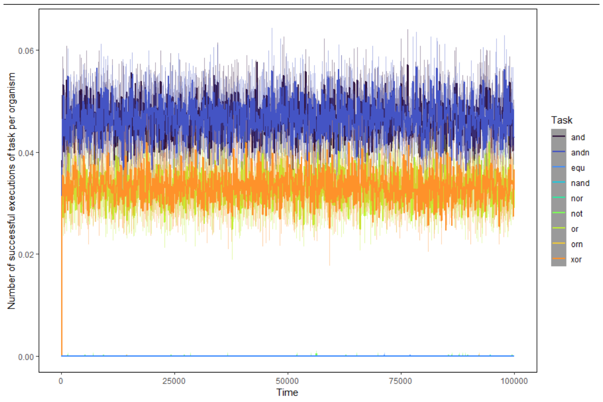
Here we see again that the overall successful completion rate improves when the rewards of a task improve, which makes logical sense. This benefit (compared to baseline) is observed in both hotspot and non-hotspot cells. The behavior may differ from the **better AND** condition because it may provide a more incremental benefit (since XOR is less likely to be successful overall).

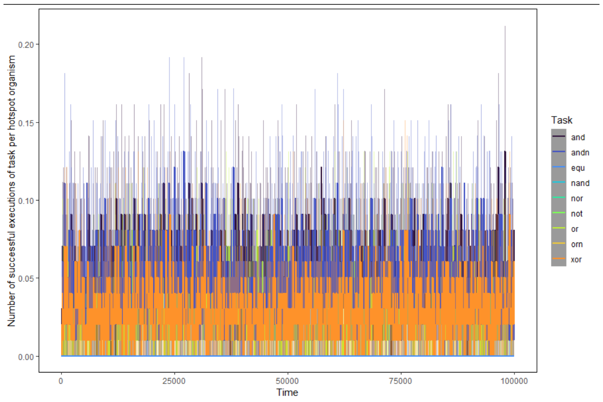
We then tried to see if a moving hotspot would affect overall ability to complete tasks; rather than being localized, task completion might spread more quickly throughout the population–or it might never arise.

Moving hotspot (1000)

We ran the following experiment (**moving hotspot**):

- Identical conditions as **baseline**, except:
- Every 1000 updates, all the cells in the hotspot would shift together one cell in some random direction.

<table>
<tr><th>Non-hotspot cells</th><th>Hotspot cells</th></tr>
<tr><td>


</td><td>


</td></tr>
</table>

Table 4: Task completions per non-hotspot and hotspot organisms. All tasks return 15 points upon success, except for NOT, which returns 30. 20 points are required for reproduction. These lines are averaged across the 11 replicates; shading represents the confidence interval. Every 1000 updates, the hotspot would move one cell in a random direction.

Evidently, a moving hotspot did not affect the overall task completion landscape; this concept ought to be tried again in a simulation where the hotspot has a stronger effect. Finally, we were using a reproductive point threshold of 20 (organisms have to solve at least two tasks to reproduce) because it produced interpretable results, but the next question would be how might different reproductive point thresholds affect the completed task landscape.

Different reproductive point thresholds

We ran the following experiment (**reproduction thresholds**):

- Identical conditions as **baseline**, except:
- The number of resources required for reproduction were set to 10, 20 (which is just **baseline**), and 30

For reference, all the y-axes in the below graph are the same range.

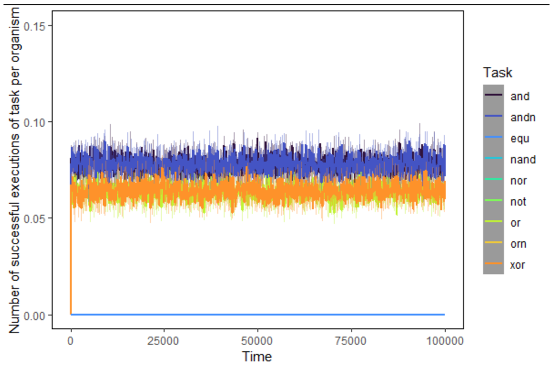
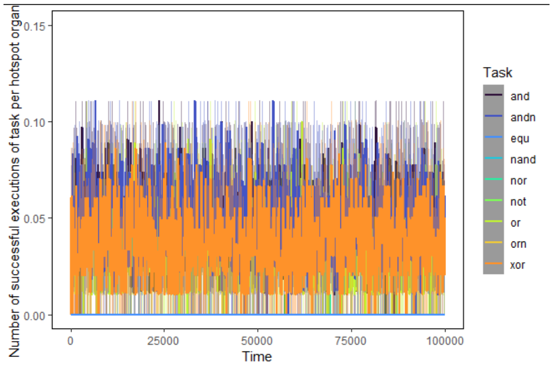
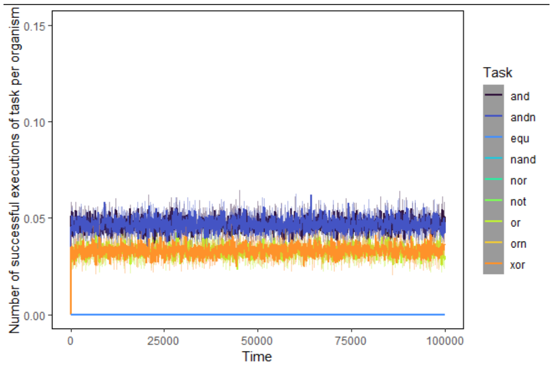
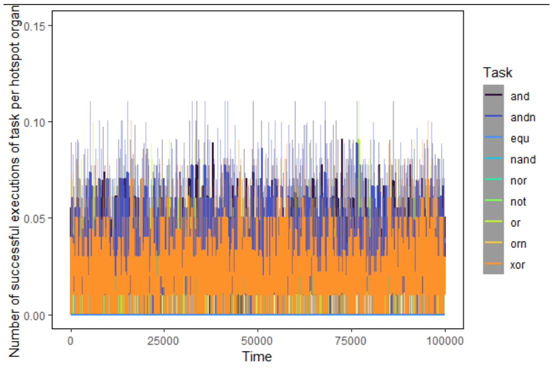
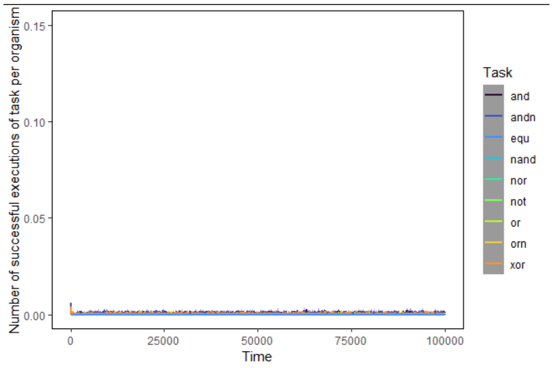
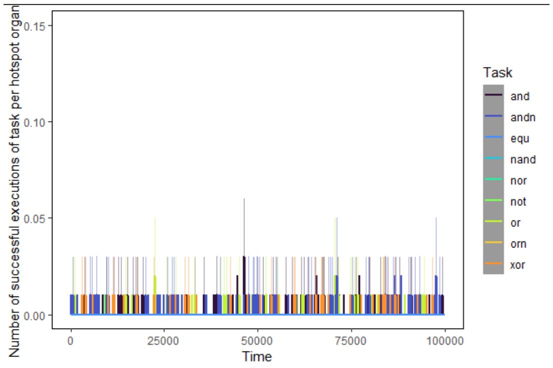
Repro points	Non-hotspot cells	Hotspot cells
10		
20		
30		

Table 5: Task completions per non-hotspot and hotspot organisms. All tasks return 15 points upon success, except for NOT, which returns 30. 20 points are required for reproduction. These lines are averaged across the 11 replicates; shading represents

the confidence interval. Rows represent the amount of resources an organism must accrue before it can replicate. At 10 resource points required for replication, an organism can reproduce following every successful task completion. At 20, it would require 2 task completions. At 30, it would require 3.

Evidently, the higher the resources required to reproduce, the lower the overall rate of completed tasks over 100,000 updates. Additionally, as demonstrated in the 30-resources-required row, hotspots do improve task-solving ability, but the data is simply much noisier (because there are only 9 possible data points–no hotspot organism completed the task, all hotspot organisms completed the task, or some in between value).

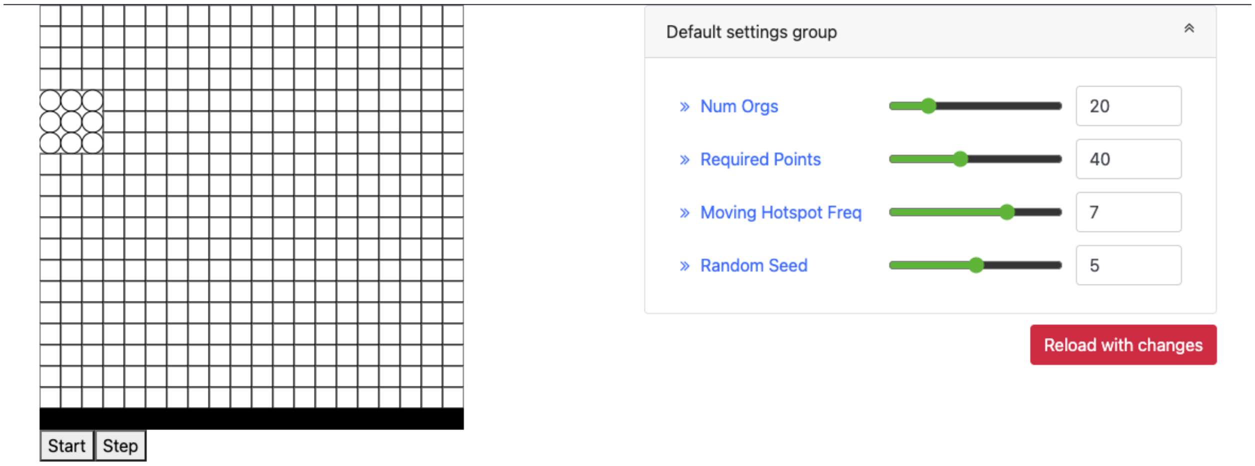
Overall, we do generally find higher rates of task completion in hotspots, much like in the paper.

Explanation of Differences

A potential explanation for the differences between our results and theirs include the fact that they seem to have tailored environments to particular instructions. We simply made it easier for organisms in hotspots overall, for all instructions; this lack of specificity may have meant there wasn't strong enough selective pressure on any single instruction for it to evolve differently between hotspot and non-hotspot conditions. Further support for this task-specificity flaw comes from the fact that we can somewhat affect which tasks are more likely to be modifying the number of resources returned; so, if future work improves the rewards system of hotspots to be task-specific, it would follow that we may see a greater range of behavior between conditions. Additionally, we used a smaller world (20x20 rather than 60x60); it could be that the hotspot was not large enough to create its own unique population, and engaged in too much gene flow with neighboring cells (and the overall world).

GUI

When we run our web.cpp file, it provides a GUI representation of a replication of "Spatial resource heterogeneity creates local hotspots of evolutionary potential" by Emily Dolson and Charles Ofria (2017). "Hotspot" cells are represented with circles, and regular cells are rectangles.



Default settings group

» Num Orgs 20

» Required Points 40

» Moving Hotspot Freq 7

» Random Seed 5

Conclusion

In our replication, a hotspot (a group of cells which improve the organism’s wellbeing, in our replication by increasing the number of CPU cycles) generally increases the per-organism per-update task completion rate, in line with expectations from the guiding paper. We extended the model to examine whether a moving hotspot might affect overall task success rates, but we found it did not have a strong effect. We also examined how different resource amounts required for reproduction might affect the task-solving landscape, and found that if organisms were required to solve three tasks or more in order to reproduce, task-solving rates were near-zero even after 100,000 updates. Generally, it appeared that the easier (fewer resources required) to reproduce, the higher the rates of successful task-solving after 100,000 updates. The lack of hotspot effects may be due to the weak selective pressure exerted by task completion and a small world. Additionally, the small hotspot size may mean that there is simply too much “gene flow,” preventing the development of unique behavior.

Work Cited

Emily Dolson, Charles Ofria; September 4–8, 2017. "Spatial resource heterogeneity creates local hotspots of evolutionary potential." Proceedings of the *ECAL 2017, the Fourteenth European Conference on Artificial Life*. *ECAL 2017, the Fourteenth European Conference on Artificial Life*. Lyon, France. (pp. pp. 122-129). ASME. https://doi.org/10.1162/isal_a_023