Analyzing Ant Behavior: The Impact of Chemical Signals and Environmental Factors in a Simulation

by

Anthony Artino

Research Project

Submitted in fulfillment of the requirements for the Modeling Projects Seminar Course (MATH 42039), under the supervision of Prof. Jing Li, as part of the Experiential Learning and Writing Intensive Course components.

Kent State University

April 18, 2025

Contents

Abstract

This project investigates the wonders of ant behavioral responses to chemical signals by using the Ant's simulation module as we focus on how varying environmental and biological parameters affect aggregation behavior. We ran these simulations under controlled conditions, repeating each scenario at least ten times to ensure statistical accuracy. Certain key variables included the number of chemical concentration areas (m), the probability of ants moving toward the chemical (*probAnt*), total ant population (n), chemical diffusion rate, and even the maximum pheromone concentration (`MAXPHER`). Furthermore, these results revealed that the increased chemical sources and higher *probAnt* values happened to enhance ant clustering, while the greater population densities ended up amplifying the overall responsiveness. Notably, these variations in diffusion rate influenced the aggregation spread, as well as `MAXPHER` revealing a nonlinear dose-response pattern, with a peak attraction at low concentrations and repulsion at much higher levels. Thus, these findings best highlight the nuanced, threshold-based nature of ant chemotaxis as well as provide us with a deeper understanding of collective movement behaviors throughout social organisms.

Background

Have you ever wondered why ants typically move in a random set or are usually grouped? In the world of ants, these small, little insects exhibit complex movement patterns that are primarily driven by both random exploration and chemical signaling. For instance, in the absence of external stimuli, ants typically move in a seemingly random manner that is undirected, a behavior known as random walk or Brownian motion, which practically allows them to freely explore their environment efficiently [1]. This random act of movement is not purely haphazard, as several studies have specifically shown that it precisely provides ants with the ability to explore large areas without any sort of predetermined path, as this helps optimize their search for sources like food as well as searching for any resource in an uncertain environment [3]. However, their movement is not just purely random, as studies have indicated that ants use pheromones to help communicate with each other and to coordinate collective behavior.

Furthermore, when a foraging ant discovers a food source, it deposits a chemical trail on its return to the colony as this would influence other groups of ants to follow along the scent and reinforcing the trail through positive feedback [1]. Over time, the pheromone concentration will determine the probability of ants to select a particular path, leading to many different emergent patterns such as trail formation and even clustering around high pheromone concentrations [1]. The diffusion and evaporation rates of pheromones can further affect ant behavior, as some highly volatile trails may quickly dissipate which would force some of the ants to go back to their daily, random exploration [1]. On the other hand, having strong or persistent pheromone trails can simply establish dominant foraging routes as this would minimize the randomness in movement [1]. Furthermore, these specific principles can make ant behavior an excellent model for studying self-organization, optimization, or even collective intelligence in biological systems. Thus, having a well understanding on how ants balance random movement with pheromone where having guided decision-making helps provide valuable information into swarm intelligence as well as real-world applications like robotic path finding or network optimization [1].

Introduction

It is a known fact that ant colonies exhibit remarkable collective behavior, largely facilitated by their use of pheromone trails to help communicate with each other. These sorts of chemical signals help enable these ants to navigate efficiently between food sources as well as their nest through optimizing foraging efforts and colony survival. Furthermore, the main purpose of this study is to precisely simulate ant movement all in a controlled environment and to even quantitatively analyze their total interaction with the pheromone trails over time.

For this simulation, we are modeling the behavior of ants by using a cellular automaton framework where we can visualize the individual ants moving across a discrete grid based on local pheromone concentrations. These pheromones will diffuse and evaporate over time, as this will replicate real-world dynamics, while groups of ants will deposit pheromones to reinforce random trails, as this would lead to several emergent patterns of aggregation. By running multiple simulation trials, this study aims to determine the average number of ants within the two units of a chemical trail at each time step, as we can then analyze the required implications of their movement patterns. Understanding these interactions will help us understand self-organizing systems, which have applications in optimization algorithms, robotics, and swarm intelligence. Furthermore, the findings from this study will help contribute to a broader concept of decentralized decision-making and cooperative behavior in biological as well as artificial systems [2].

Model

To incorporate a cellular automaton framework into our ant simulation, in which ants would move on a discrete $n \times n$ grid and interact with an evolving pheromone field, we would have to apply stochastic movement rules as well as pheromone-based decision-making to simply replicate the foraging behavior observed in real ant colonies.

Grid Representation and Initialization

Here, our simulation environment consists of two primary grids:

- **Ant Grid:** This specific grid helps represent the positions as well as the orientations of ants. For instance, each cell may contain an ant facing one of four cardinal directions (north, south, east, or west) or be empty, meaning it does not move at all. On the other hand, the borders of the grid act as impassable boundaries, as this prevents ants from exiting the boundaries. The initial distribution of ants follows a probability parameter P_{ant}, which helps determine the total likelihood of an ant occupying any given cell at the given start of the simulation.
- **Pheromone Grid:** This grid simply stores the concentration of pheromones in each cell. Initially, a localized pheromone source is placed at the very center of the grid, consisting of concentrations that are decreasing radially outward. Thus, this setup mimics natural pheromone deposition, providing an attractant for ants to follow.

Movement and Decision Rules

Specifically, at each time step, the ants will update their precise positions according to local pheromone concentrations and environmental constraints. This decision-making process will follow these key steps:

1. **Sensing:** Each ant will evaluate the pheromone levels in its immediate neighborhood, typically the three front-adjacent cells. Furthermore, if a neighboring cell has a higher pheromone concentration than the current cell, then the ant will move in that direction. If multiple directions end up having equal pheromone levels, then the ant will randomly select one.
2. **Movement:** If the selected direction leads to an unoccupied cell, the ant will move forward by only just one unit. If the target cell is already occupied by another ant or is a boundary, the ant will remain in its current position.
3. **Pheromone Deposition:** After the ant moves, it will deposit a fixed amount of pheromone in the cell that it occupies, reinforcing the trail for subsequent ants to become attracted to it.
4. **Pheromone Diffusion and Evaporation:** The pheromones will eventually diffuse over time at a rate D according to a standard diffusion equation. Concurrently, the evaporation will occur at a rate E, as this causes the pheromones to completely dissipate over time, preventing unlimited accumulation.

Simulation Execution and Data Collection

When it comes to running this simulation, it will run for T time steps as it consistently repeats over multiple trials to take account of the variability in stochastic movement. Some key performance metrics will include:

- The mean number of ants within two units of the pheromone trail at each time step.
- The rate at which ants successfully follow the trail.
- The distribution of ants relative to high-concentration pheromone regions.

These metrics will provide a strong insight into the ant colony's collective response to chemical signals as well as the efficiency of pheromone-mediated navigation strategies. Furthermore, by systematically observing the ants' behavior under these controlled conditions, this model will help enable a quantitative assessment of decentralized swarm intelligence as well as the emergent patterns that form through simple local interactions.

Mathematical Representation

The core mechanics of the simulation can be represented mathematically. For instance, we know that the pheromone concentration at a given cell $\rho(i,j)$ will evolve as this is all due to the following diffusion equation:

$$\frac{\rho(i,j)}{\Delta t} = D \sum_{(m,n)\in N(i,j)} (\rho(m,n) - \rho(i,j)) - \lambda \rho(i,j), \tag{1}$$

in which:

- D represents the diffusion coefficient as it controls how quickly the pheromones will spread across the neighboring cells.
- λ is the evaporation rate at which it reduces the pheromone concentration over time.
- $N(i,j)$ represents the set of neighboring cells of (i,j), which include cells in the cardinal directions of north, south, east, and west.
- Δt is the time step that is between the updates of the pheromone concentration grid.

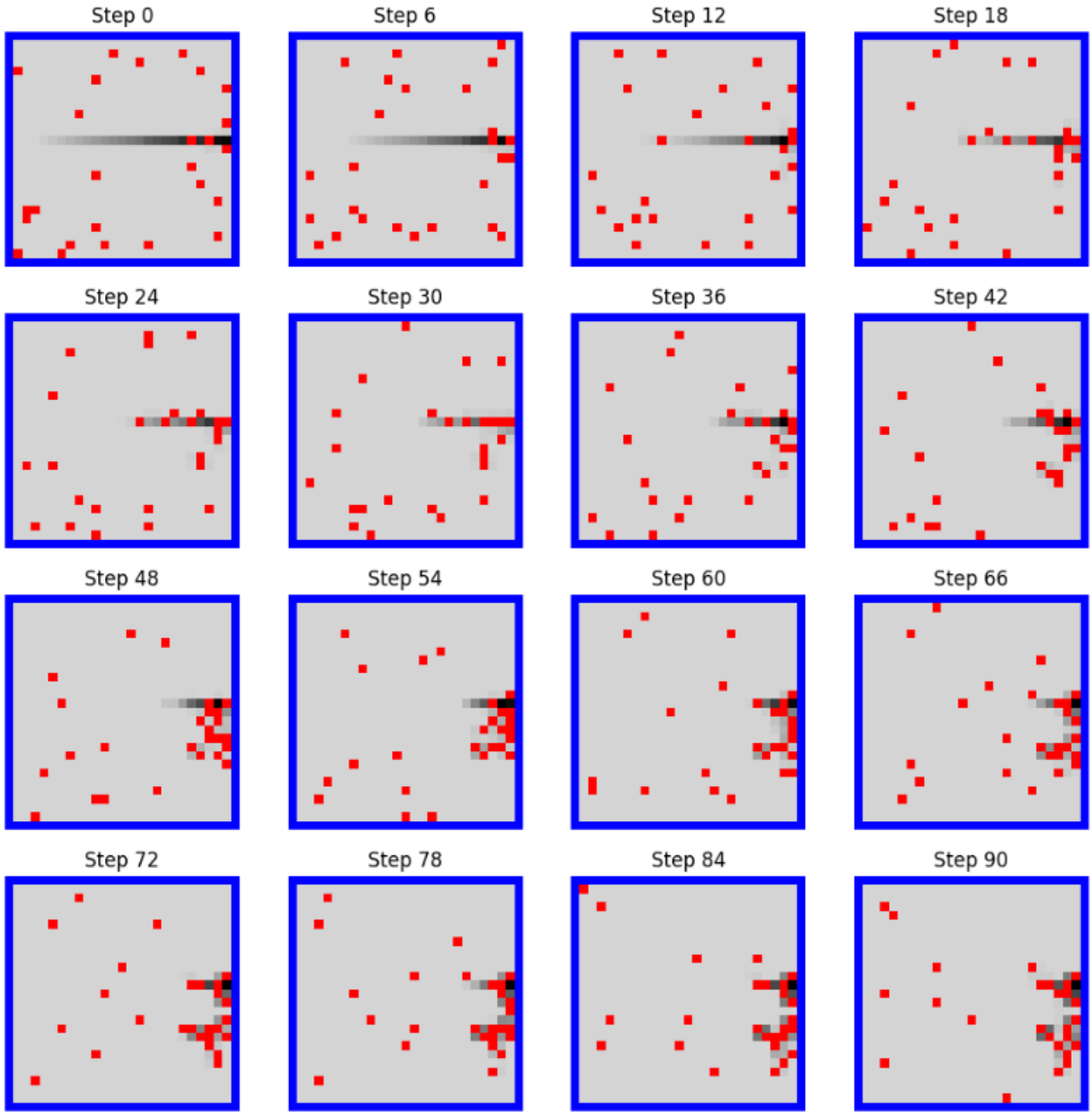
Furthermore, the ants' movement is determined through probability, as this is solely based on the pheromone concentrations in the neighboring cells. Given an ant at position (i,j), it will evaluate the pheromone values in its neighborhood as well as move. This is all presented through the following formula:

$$P(d) = \frac{\rho(i',j')}{\sum_{d'} \rho(i',j')}, \tag{2}$$

In this case, we have d representing the possible movement direction (north, south, east, or west), and $\rho(i',j')$ being the pheromone concentration at the cell (i',j') in the direction d. Thus, by simply iterating this process over multiple time steps and across numerous trials, we can collect statistical data based on the ants' aggregation behavior. This analysis will focus on the average number of ants that are primarily located within two units of the pheromone trail

over time and across their repeated runs, which helps provide a stronger understanding of the efficiency of the simulated foraging strategy. In the next section, we will present the results obtained from simulating fixed conditions, followed by a discussion on their implications.

Figure 1: Illustration of Random Ant Movements



Chapter 1

The Impact of Chemical Concentration Areas on Ant Behavior

Analysis

In this chapter, we will investigate the behavior of ants in response to the varying numbers of chemical concentration areas, as this is denoted by m, where we will set $m = 1, 2, 3, 4, 5$. This model will simulate ant movements and pheromone diffusion while tracking the number of ants in proximity to the chemical areas over time. First off, we will need to initialize the grid system, where the ants are randomly distributed with a probability of 0.04 in each cell, while the chemical concentrations are introduced into the grid with the five different values of m. For each value of m, we can conduct multiple simulations (ten in this case) to help ensure that the results are statistically robust. For each simulation, we will track the number of ants that come near the chemical areas at each time step, iterating for approximately 100 steps.

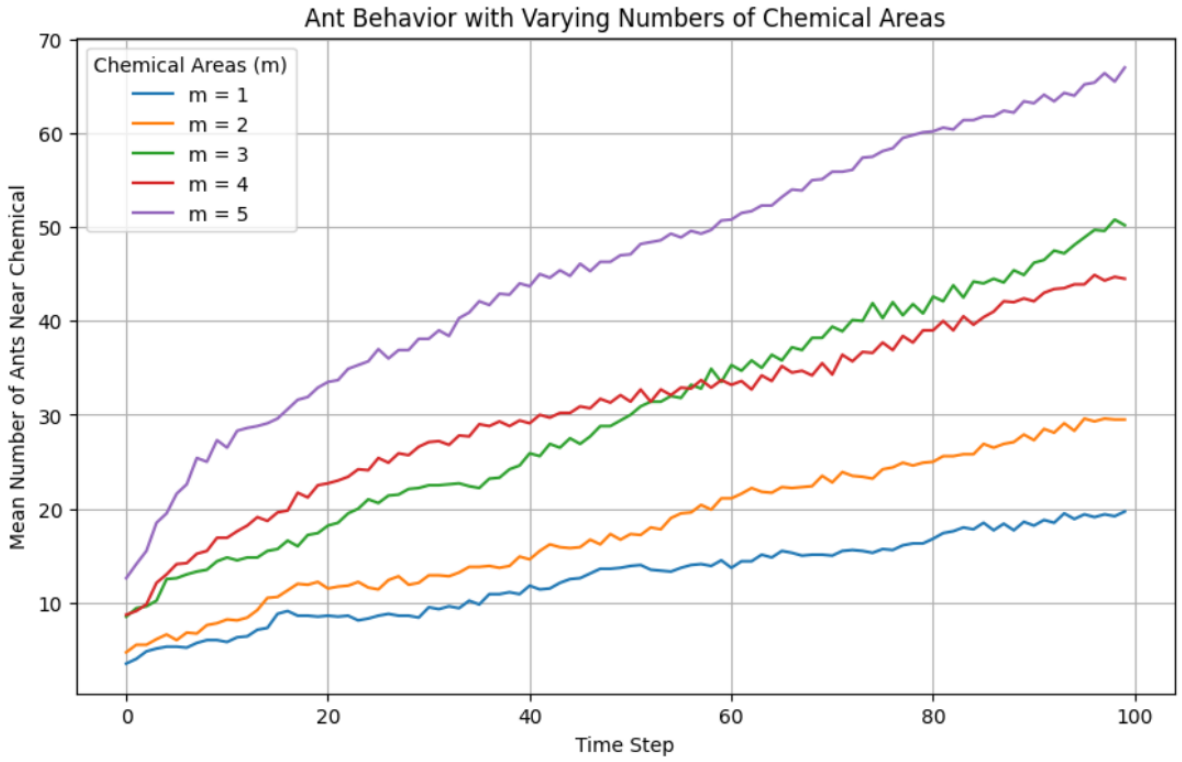
Our results will reveal a clear and concise pattern as the number of chemical areas increases. For instance, when m increases, we will see that the number of ants near chemical concentrations will generally rise, especially during their earlier steps of the simulation. This is all because the ants are more likely to encounter one of the chemical areas when there are more of them distributed on the grid. The presence of multiple chemical areas will offer more potential "targets" for the ants to follow, as this would lead them to a much higher overall interaction with these areas. From the results, we can observe that the response of ants to chemical concentrations follows a dynamic, time-dependent pattern. At lower values of m, the number of ants near chemical areas stabilizes after a few time steps as the ants either explore other areas or encounter pheromone trails that guide them away from the chemical zones. On the other hand, with larger m, the increased number of chemical areas creates more pheromone gradients, which continuously attract ants toward these regions throughout the simulation.

The maximum number of ants near the chemicals occurs when $m = 5$, indicating that a higher concentration of chemical areas leads to greater interaction between ants and the pheromone gradients. As the simulation progresses, the mean number of ants near chemical areas fluctuates due to the combined effects of pheromone diffusion and evaporation, which balance out the attraction forces over time.

Results and Discussion

Below is the graph generated from the simulation, which illustrates the mean number of ants near chemical areas over time for varying numbers of chemical areas, $m = 1, 2, 3, 4, 5$.

Figure 1.1: Graph of Varying Numbers of Chemical Areas



We can see here that the graph shows the mean number of ants near chemical areas over 100 time steps for varying numbers of chemical areas m. Firstly, when $m = 1$, the mean is consistently the lowest out of all five of them. This may be likely due to the limited as well as localized pheromone gradient, as this makes it quite hard for ants to locate and gather near a single source. On the other hand, at $m = 5$, we can see that the mean is consistently the highest. Thus, by having more chemical areas, the ants will have more pheromone trails to follow, as this results in a stronger attraction as well as a broader engagement across the grid. This specifically creates a more effective and distributed chemical field, drawing in more ants overall. For the intermediate values such as $(m = 2,3,4)$, the mean number of ants increases with each added chemical area, but with diminishing returns. Here, the differences between these values are much less pronounced, as this suggests the idea that beyond a certain point, any additional chemical area offers limited gains in attraction. Hence, these findings highlight the idea that while increasing chemical density improves the ant response, there is a threshold beyond which the impact levels off.

Chapter 2

Exploring the Effect of Varying Ant Probability on Simulation Behavior

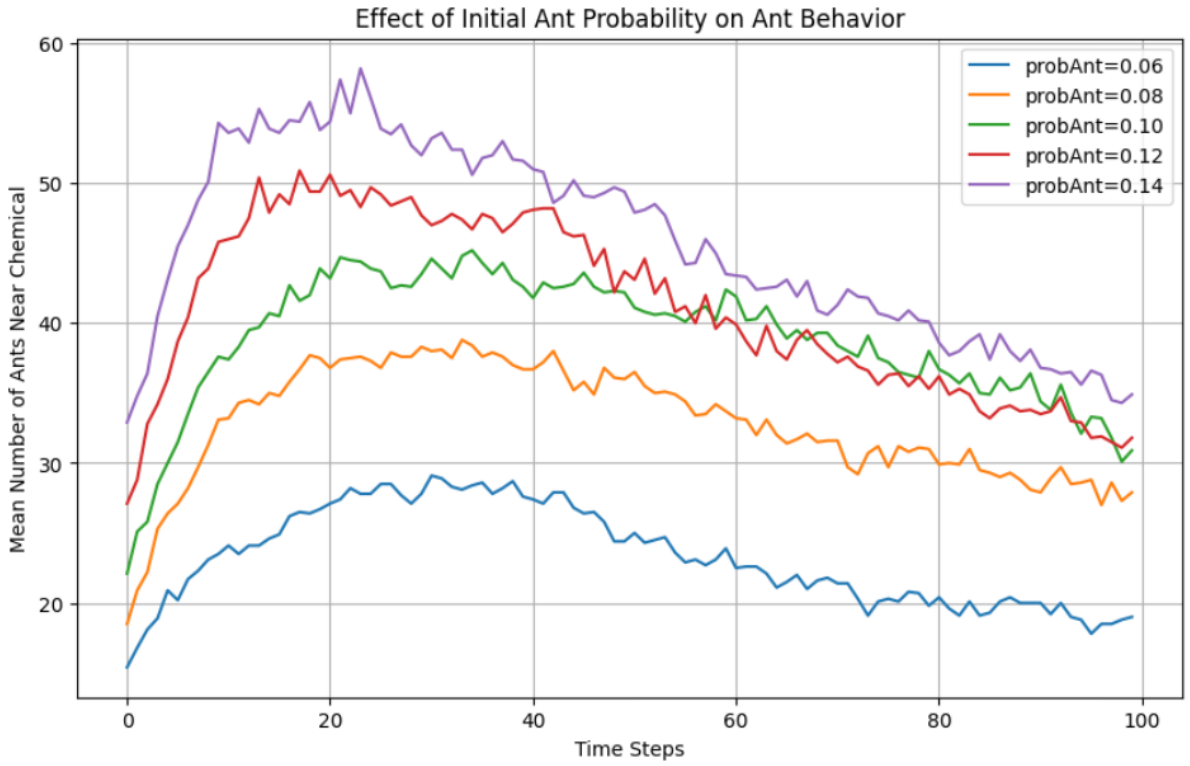
Analysis

In this chapter, we aim to explore the total effect of varying the initial probability of ants (probAnt) on the specific mean number of ants near a chemical source over time. For instance, the probability of ants is varied from 0.06 to 0.14 in increments of 0.02 with the assumption that there would be a higher probAnt that will lead to a much higher initial concentration of ants in the grid. This experiment uses a simulation framework in which the ants are distributed across a 50×50 grid, and their movement, as well as their interaction with a pheromone field, are modeled over 100 time steps. Thus, the primary goal is to determine how the changes in the initial probability of ants will impact the number of ants that are present near a chemical source, which is tracked over multiple trials.

Results and Discussion

Below is the graph displaying the effect of varying the initial ant probability (probAnt) on the mean number of ants near the chemical source over time:

Figure 2.1: Graph of Varying Numbers of Chemical Areas



The graph above helps show how the mean number of ants near the chemical source changes over time during different initial ant probabilities (probAnt), where each line represents the average across 10 trials. Here, the x-axis represents time, while the y-axis shows the mean number of ants near the chemical source. Furthermore, for a higher initial probability, such as probAnt $= 0.12$ and probAnt $= 0.14$, the mean number of ants near the chemical source will remain consistently higher throughout the simulation. This is mainly because a larger initial population will increase the likelihood of ants encountering the pheromone field early on. For instance, as more ants are drawn to the source, the pheromone trail is reinforced, as this creates quite a positive feedback loop that further attracts the ants.

In contrast, the intermediate values of probAnt, such as 0.08 and 0.10, result in a more moderate number of ants near the chemical source. However, we can see from the graph that some attraction still occurs, as the overall population is consistently smaller than in the high-probability cases, leading to fewer interactions with the pheromone field and a weaker reinforcing effect. Also, at the lowest initial probability, probAnt $= 0.06$, it is displayed that the mean number of ants near the chemical source will remain significantly the lowest throughout the entire simulation. Precisely, with fewer ants on the grid, we can say that the chance of encountering and following pheromone trails is completely reduced, which limits the accumulation of ants near the chemical source over time.

Furthermore, these sorts of trends help demonstrate the main critical role of the initial ant density as it shapes the outcome of pheromone-based navigation. With a higher initial probability, it can be seen that it practically leads to a greater interaction with the pheromone field as well as a stronger collective movement toward the chemical source. On the other hand, the lower initial probabilities happen to reduce these interactions, resulting in a smaller and less organized response. Hence, this suggests that the initial population density is a key factor when it comes to determining the effectiveness of resource-seeking behavior in ants.

Chapter 3

Exploring the Effects of Varying the Number of Ants

Analysis

In this chapter, we aim to investigate how the number of ants, n, precisely affects the concentration of ants near a chemical source over time. The variable n is the number of ants initially distributed across the grid, as we will also vary n from 10 to 50 in increments of 10. The main purpose of this simulation is to determine how the different numbers of ants impact the number of ants that remain near the chemical source during the simulation. Thus, we will need to perform the following steps:

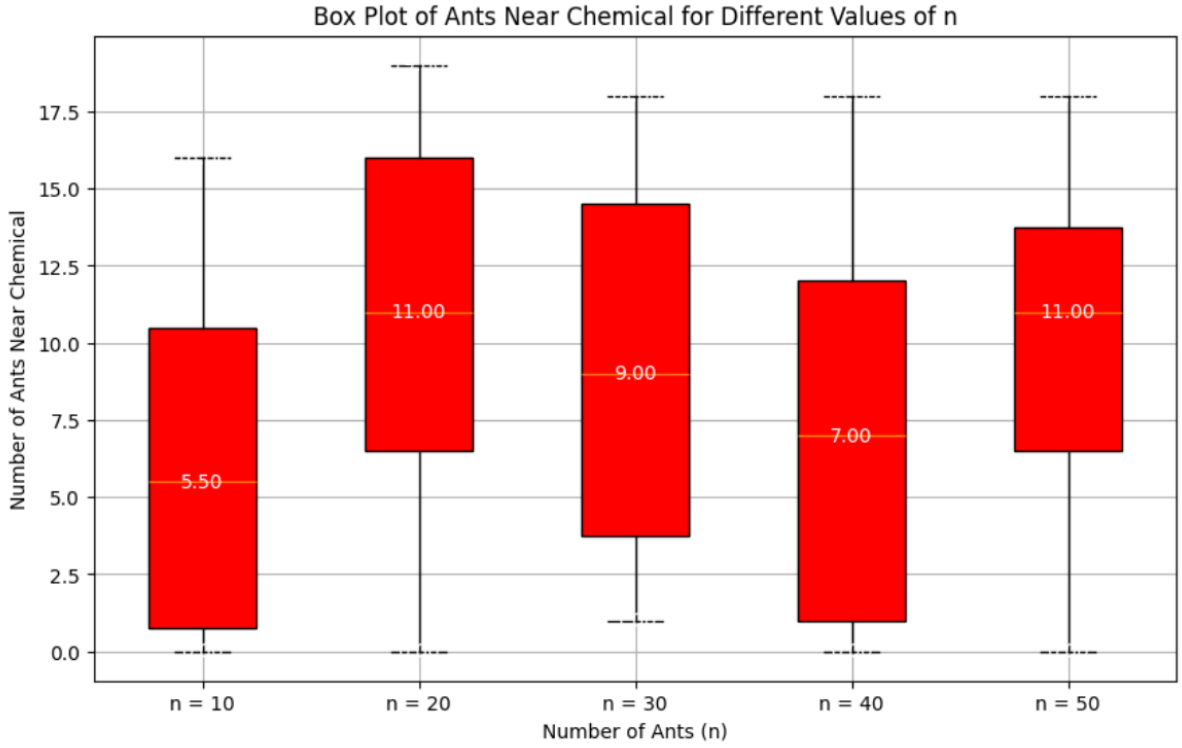
1. Run the simulation 10 times for each value of n (i.e., $n = 10, 20, 30, 40, 50$).
2. For each run, we will calculate the number of ants near the chemical source at each time step. Specifically, we are interested in measuring the mean number of ants that fall within two units of the chemical.
3. The simulation will be repeated for each of the five different values of n, and a box plot will be generated to compare the distribution of ants near the chemical for each n.

This approach will help us understand the relationship between the number of ants n as well as the resulting concentration of ants near the chemical source, where we will be able to see whether a higher n leads to a higher concentration or if the relationship is more complex.

Results and Discussion

The following graph shows the box plot of the mean number of ants near the chemical source for different values of n (10, 20, 30, 40, 50) across 10 trials. Each box plot represents the distribution of ant counts near the chemical source for a specific value of n, with the median, upper, and lower quartiles displayed.

Figure 3.1: Box Plot of the Ants near Chemical Areas



The box plot illustrates how varying the number of ants n will influence the total concentration of ants near the chemical source. As expected, increasing n generally leads to a much higher number of ants near the chemical. For instance, at $n = 10$, the concentration is relatively low, as most of the ants are not coming anywhere near the chemical, which is indicated by the lower median as well as the narrow spread of values. However, as n increases to 20, there is a much more noticeable rise in the concentration near the chemical, where more ants are being drawn towards the source, as this is reflected by a higher median as well as a broader distribution.

At $n = 30$, the concentration continues to increase, but the growth rate begins to slow compared to $n = 20$. The spread of the values is still wider, indicating that some ants are more effective at reaching the chemical than others, potentially due to factors like competition or pheromone interference. At $n = 40$, although the number of ants is higher, the concentration near the chemical plateaus slightly, with the median decreasing and the range narrowing again. This suggests that the system may be approaching a point of diminishing returns, where additional ants do not significantly increase the number of ants near the chemical.

Finally, at $n = 50$, the trend is similar to $n = 40$, where the concentration near the chemical does not significantly increase despite the higher number of ants. This plateau indicates that once a certain threshold is reached, adding more ants leads to only marginal increases in the concentration of ants near the chemical, likely due to factors such as overcrowding or limited chemical attraction capacity. Overall, the highest concentrations are seen at $n = 20$ and $n = 30$, with diminishing increases at higher values of n.

Chapter 4

Analyzing Ant Behavior in Response to Varying Diffusion Rates

Analysis

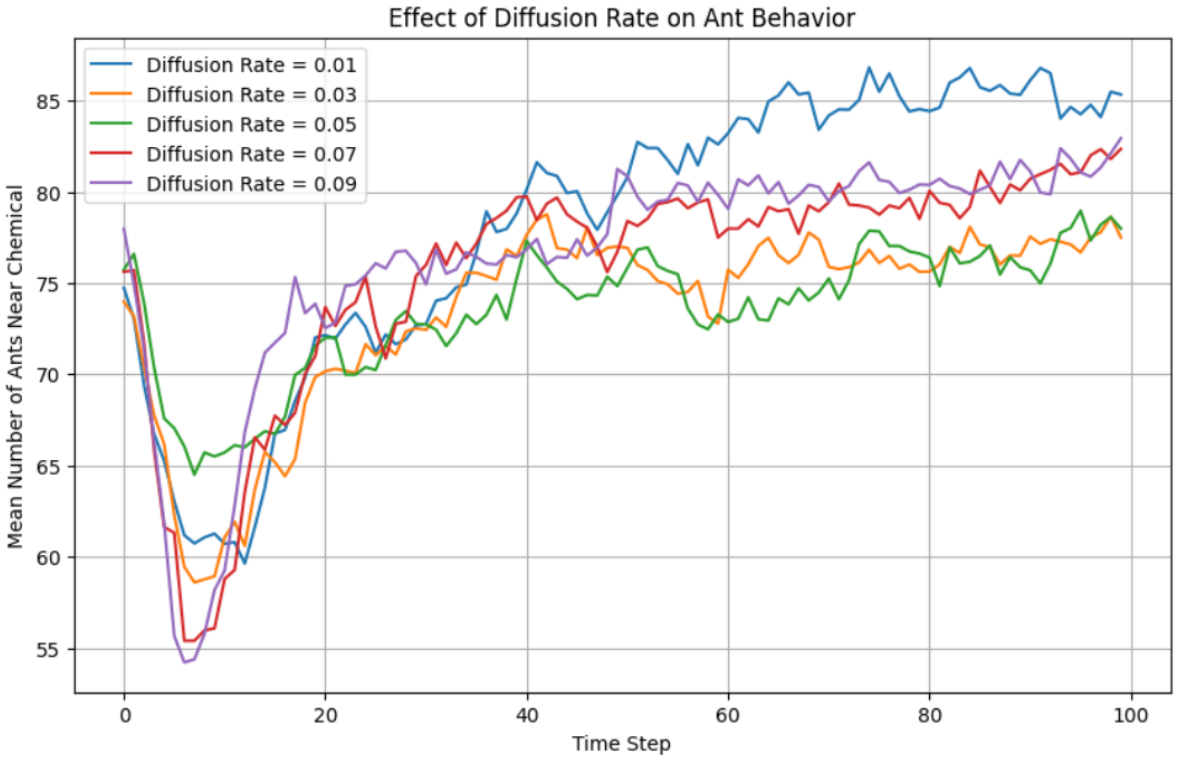
In this chapter, we investigate the effect of varying the `diffusionRate` on ant behavior, specifically the mean number of ants near the chemical at each time step. Precisely, the `diffusionRate` helps influence how the chemical spreads throughout the environment, and the fact that our goal is to examine how changes in the `diffusionRate` impact the ants' proximity to the chemical over time. Here, this simulation was run with a `diffusionRate` varying from 0.01 to 0.09 in increments of 0.02. For each `diffusionRate`, the simulation was executed approximately 10 times, as the mean number of ants near the chemical was calculated at each time step and averaged over the runs. The probability of an ant being present (`probAnt`) was set to 0.04, and other parameters such as the `MAXPHER`, `EVAPORATE`, and `DEPOSIT` rates were specifically fixed as specified earlier. Hence, the total simulation time was set to 100 steps.

Furthermore, each simulation involves the initialization of two grids—one for the ants and one for the pheromones—which is followed by the application of sensory, walking, and diffusion processes. On the other hand, the diffusion process spreads the pheromone's chemical over time, as the primary focus here is on how different rates of diffusion from 0.01 to 0.09 happen to alter the number of ants that are within 2 units of the chemical at any given moment. The number of ants within that proximity was then averaged over multiple simulation runs and time steps. Here, we assumed that the only variable being altered across simulations was the `diffusionRate`, and all other parameters remained constant throughout the analysis. This, of course, will allow us to isolate the effect of the diffusion rate on ant behavior as well as infer how diffusion affects the ants' ability to locate and follow the chemical trail.

Results and Discussion

The graph analyzing the **"Effect of Diffusion Rate on Ant Behavior"** reveals nuanced patterns across different diffusion rates over time. Initially, all rates start with approximately 75 ants near the chemical, showcasing a strong initial response.

Figure 4.1: Plot of Diffusion Rate on Ants



Interestingly, we can see here from the graph above that the slower diffusion rate of (`0.01`) will ultimately outperform the faster rates by sustaining a higher ant concentration near the chemical over time. This is mainly because a slower diffusion rate happens to cause the chemical to spread less rapidly, as this precisely allows ants to stay much closer to the source as well as reinforce the pheromone trail over an extended period. Also, the slower diffusion ensures that the ants' movement is practically guided much more persistently by the chemical, as this keeps a higher concentration near the chemical source. Thus, the slower spread of the chemical would also mean that the ants will have more time to respond as well as adjust to the trail, resulting in better retention over time.

On the other hand, we can see that the faster diffusion rates (`0.03, 0.05, 0.07, 0.09`) happen to show less retention overall, specifically with ants not staying near the chemical as long. This is because the chemical spreads too quickly, as this leads to a rapid dispersal of the pheromone. As a result, the ants will lose the concentrated pheromone trail and will end up less likely to remain near that source. The rapid spreading of the pheromone will make it even harder for ants to track it over time, as the signal they are following starts to become more diluted way too quickly. Furthermore, this causes a quicker loss of attraction as well as less overall retention near the chemical source.

To be more descriptive, the subtle differences between the `green (0.05)` and `orange (0.03)` lines are most evident in the early phases as they highlight the diverging recovery trends before converging later on. Initially, the `orange line (0.03)` happens to show a slightly slower recovery in the end due to a moderate diffusion rate in which it is slow enough to retain some pheromone, but not enough to sustain a stronger concentration over time. On the other hand, the `green line (0.05)` recovers just a little faster, likely due to a more optimal balance between evaporation as well as deposition. Specifically, this allows the pheromone

to practically remain dense enough to potentially guide the ants effectively without dispersing widely. By step 100, both lines stabilize between 75 to 80 ants, as this suggests that the spatial patterns will eventually converge as the diffusion effects level out. With the higher diffusion rates, such as `red line (0.07)` and `purple line (0.09)`, we can see that in the beginning, the pheromone will spread way too rapidly, as this causes there to be a sharp decline in ant concentration with quite a sluggish recovery. This, of course, reinforces the idea that having excessive diffusion will dilute the chemical trail, making it much harder for ants to follow.

Overall, these highlights help show how the `diffusion rate` critically shapes the ant behavior and spatial dynamics. By having slower rates such as `(0.01)`, they preserve the pheromone trail longer, allowing ants to remain near the source, while the faster rates will lead to rapid dissipation as well as a much weaker trail. Even though the diffusion effects converge over time, the optimal trail-following occurs when there's a balance between spread and retention. Thus, these particular findings help highlight the idea that the trail's effectiveness hinges not just on how far the chemical spreads, but also on how well these ants can stay oriented to it over time.

Chapter 5

Exploring the Impact of Varying MAXPHER

Analysis

In this chapter, we investigated the effect of varying the MAXPHER parameter on the behavior of ants throughout a simulated environment. Here, MAXPHER represents the maximum pheromone concentration that ants can detect, as this influences their total movement as well as their interaction with chemical trails. By systematically adjusting MAXPHER from 10 to 80 in increments of 10 while keeping all other parameters constant, we aimed to determine its impact on the mean number of ants within two units of the chemical trail over time. This experiment consisted of running the simulation ten times for each MAXPHER value as well as calculating the mean number of ants near the chemical trail at each time step. Each simulation was initialized with a 50×50 grid, where ants were placed with a probability of 0.04, and pheromone diffusion, evaporation, and deposition mechanisms were implemented. Thus, the ants' movement was influenced by pheromone concentration, with a preference for directions where the higher pheromone levels were specifically detected.

Methodology

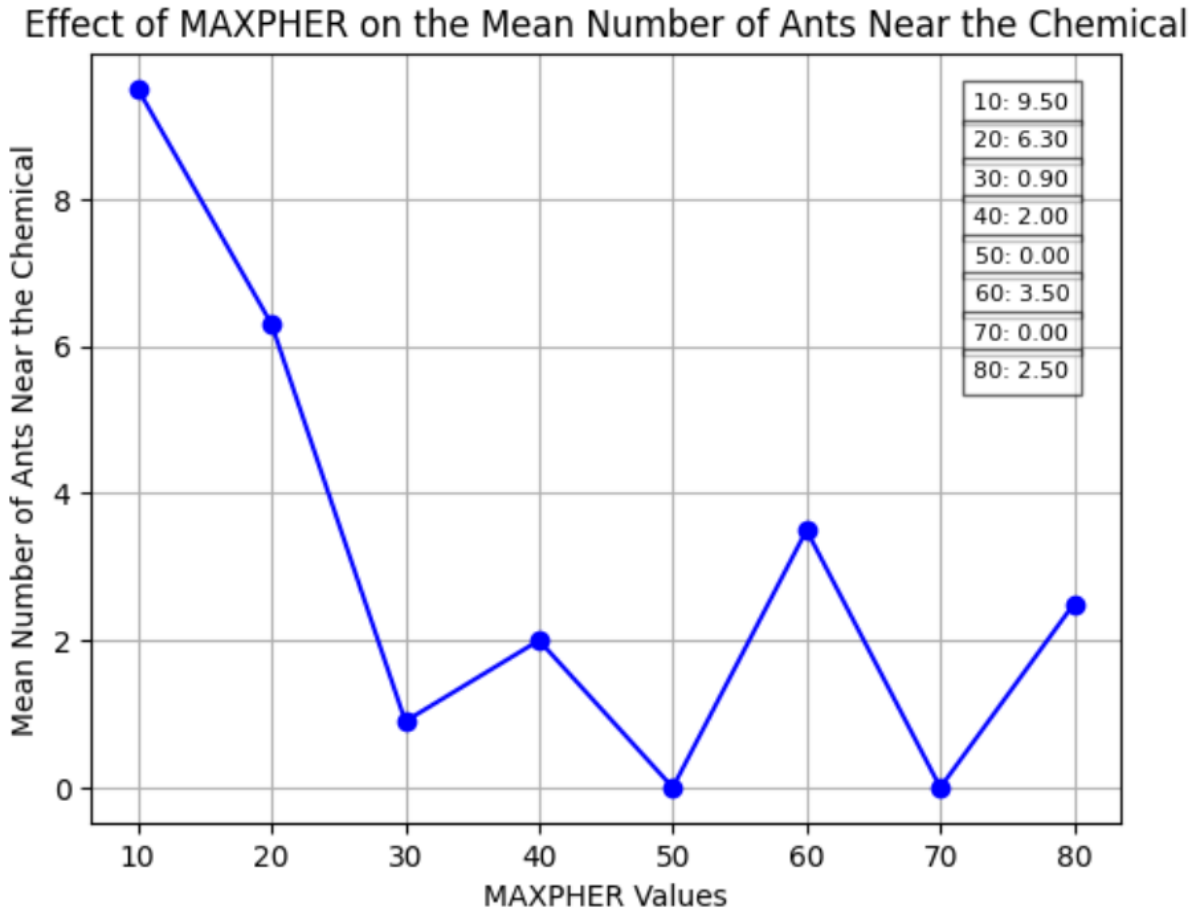
The methodology involved:

- Initializing the ant and pheromone grids for each MAXPHER value.
- Running the simulation for 100 time steps.
- Tracking and averaging the number of ants near the chemical source over ten trials.
- Plotting the results to visualize the relationship between MAXPHER and ant behavior.

Results and Discussion

The results indicate a nonlinear relationship between the MAXPHER concentration as well as the number of ants that are present near the chemical, where a pattern is commonly observed in biological and chemical attraction studies. The data specifically suggest that ants are more attracted to a low concentration of MAXPHER, but as the concentration increases, their response diminishes, ultimately leading to avoidance at certain levels.

Figure 5.1: Plot of MAXPHER values



By observing the plot, we can see that at `10 MAXPHER`, the number of ants near the chemical peaked at an average of 9.50 ants. This suggests that a low concentration is the most effective when it comes to attracting ants. Furthermore, it aligns with the known pheromone communication behaviors in ants, where a moderate chemical presence helps enhance detection and attraction. However, as we increase the concentration, we can see that the number of ants has steadily declined. For instance, at `20 MAXPHER`, the mean number of ants dropped to approximately 6.30, showing a complete, reduced attraction. On the other hand, by `30 MAXPHER`, the attraction happens to nearly vanish as it has an average of only 0.90 ants present.

Interestingly enough, when you look at `40 MAXPHER`, there was visually a small increase where it went up to 2.00 ants, as this could be attributed to random fluctuations or even a secondary effect of the chemical. However, at the points `50 MAXPHER` and `70 MAXPHER`, we can see that the mean number of ants happens to drop to 0.00, as this suggests to us that there was a complete repulsion at these precise concentrations. Furthermore, this could indicate that the chemical may have been overwhelming or even toxic at higher doses, which would have disrupted the ants' ability to process it effectively. However, despite these minor drop-offs, some slight recoveries were starting at `60 MAXPHERS`, recorded with 3.50 ants, as well as `80 MAXPHER`, recorded at 2.50 ants. Precisely, these minor resurgences could potentially imply that there was a threshold effect, in which the chemical happens to become less overwhelming beyond a certain concentration.

Conclusion

In this project, we were able to conduct a comprehensive investigation of ant behavior in response to chemical concentration by using the Ants simulation module. Furthermore, our primary objective was to analyze how varying parameters influenced the mean number of ants within two units of the chemical over time. To ensure that there is accuracy, each simulation would need to run at least 10 times, where the mean values are computed for each condition. We explored multiple experimental conditions by keeping most parameters fixed while systematically adjusting specific variables.

First, we examined the impact of different numbers of chemical concentration areas ($m = 1, 2, 3, 4, 5$) to simply observe whether increasing the number of attractant sources would alter ant movement patterns. Next, we varied the probability of an ant moving toward the chemical (*probAnt*) from 0.06 to 0.14, all in increments of 0.02, to help determine the total sensitivity of ants to the chemical signal. Furthermore, we then adjusted the total number of ants (n) from 10 to 50 in increments of 10 to practically assess how the population density influenced attraction behavior. Additionally, the diffusion rate of the chemical was modified from a range of 0.01 to 0.09, all in increments of 0.02, as this allowed us to examine how the spread of pheromone affected ant aggregation.

Finally, we were able to vary the `MAXPHER` concentration from 10 to 80 in increments of 10, which happened to reveal a nonlinear dose-response relationship in which we witnessed that the ants were most attracted at low concentrations, but the attraction declined as the concentration increased, with some levels even leading to repulsion. These specific findings suggest that ants exhibit an optimal attraction threshold, beyond which the chemical may overwhelm their sensory receptors or even act as a deterrent. The results provide valuable insight into ant foraging behavior and demonstrate how chemical signals regulate movement in collective biological systems.

Bibliography

[1] Musco, C., et al. *Ant-inspired density estimation via random walks*. *Proceedings of the National Academy of Sciences*, vol. 114, no. 40, 19 Sept. 2017, pp. 10534–10541. Available at: `https://doi.org/10.1073/pnas.1706439114`.

[2] Shiftlet, A. B., & Shiftlet, G. W. *Introduction to Computational Science*. 2018.

[3] Zhang, N. & Yong, E. H. *NTU*. Available at: `https://dr.ntu.edu.sg/bitstream/10356/173905/2/PhysRevE.108.054306.pdf`.

Python Code

Cellular Automation of Model of Ants randomly moving Code

```
# Cellular Automation of Model of Ants randomly moving

import numpy as np
import matplotlib.pyplot as plt
from random import randint, random, seed
from copy import deepcopy
import time
from IPython.display import clear_output, display

## Constants
EMPTY = 0
NORTH = 1
EAST = 2
SOUTH = 3
WEST = 4
STAY = 5
BORDER = 6

# Initialize antGrid with random ants
def initAntGrid(n, probAnt):
grid = BORDER * np.ones((n + 2, n + 2))
for i in range(1, n + 1):
for j in range(1, n + 1):
if random() < probAnt:
grid[i, j] = randint(1, 5)
else:
grid[i, j] = EMPTY
return grid

# Initialize pheromoneGrid with a trail that increases in
   strength
def initPherGrid(n, m, MAXPHER):
grid = np.zeros((n + 2, n + 2))

# Borders have a value of -0.01
grid[:, 0] = grid[:, -1] = -0.01
grid[0, :] = grid[-1, :] = -0.01

mid = n // 2
for i in range(1, n + 1):
for j in range(m):
grid[mid, i] = MAXPHER * i / n
return grid

# Function to apply diffusion to the pheromone grid
```

```
def applyDiffusion(pherGrid, diffusionRate):
n = len(pherGrid) - 2
newPherGrid = deepcopy(pherGrid)
for i in range(1, n + 1):
for j in range(1, n + 1):
newPherGrid[i, j] = max(
pherGrid[i, j] + diffusionRate / 4 * (
pherGrid[i - 1, j] + pherGrid[i, j + 1] + pherGrid[i + 1, j] +
   pherGrid[i, j - 1] - 4 * pherGrid[i, j]
),
0
)
return newPherGrid

# Function to apply sensing rules to orient ants
def applySense(antGrid, pherGrid):
n = len(antGrid) - 2
newAntGrid = deepcopy(antGrid)

for i in range(1, n + 1):
for j in range(1, n + 1):
if antGrid[i, j] == EMPTY:
newAntGrid[i, j] = EMPTY
else:
neighbors = [antGrid[i - 1, j], antGrid[i, j + 1], antGrid[i + 1,
    j], antGrid[i, j - 1]]
lst = [pherGrid[i - 1, j], pherGrid[i, j + 1], pherGrid[i + 1, j
   ], pherGrid[i, j - 1]]

if antGrid[i, j] < STAY:
lst[int(antGrid[i, j]) - 1] = -2  # Prevent immediate return

for inx in range(4):
if neighbors[inx] > 0:
lst[inx] = -2

mx = max(lst)  # Max pheromone direction
if mx < 0:
newAntGrid[i, j] = STAY
else:
posList = [ind for ind in range(4) if lst[ind] == mx]
newAntGrid[i, j] = posList[randint(0, len(posList) - 1)] + 1  #
   Choose a direction

return newAntGrid

# Function for ant movement
def walk(antGrid, pherGrid, EVAPORATE, DEPOSIT, THRESHOLD):
n = len(antGrid) - 2
newAntGrid = deepcopy(antGrid)
newPherGrid = deepcopy(pherGrid)
```

```
92
93 for i in range(1, n + 1):
94 for j in range(1, n + 1):
95 if antGrid[i, j] == EMPTY:
96 newPherGrid[i, j] = max(newPherGrid[i, j] - EVAPORATE, 0)
97 elif antGrid[i, j] == NORTH and newAntGrid[i - 1, j] == EMPTY:
98 if newPherGrid[i, j] > THRESHOLD:
99 newPherGrid[i, j] += DEPOSIT
100 newAntGrid[i, j], newAntGrid[i - 1, j] = EMPTY, SOUTH
101 elif antGrid[i, j] == EAST and newAntGrid[i, j + 1] == EMPTY:
102 if newPherGrid[i, j] > THRESHOLD:
103 newPherGrid[i, j] += DEPOSIT
104 newAntGrid[i, j], newAntGrid[i, j + 1] = EMPTY, WEST
105 elif antGrid[i, j] == SOUTH and newAntGrid[i + 1, j] == EMPTY:
106 if newPherGrid[i, j] > THRESHOLD:
107 newPherGrid[i, j] += DEPOSIT
108 newAntGrid[i, j], newAntGrid[i + 1, j] = EMPTY, NORTH
109 elif antGrid[i, j] == WEST and newAntGrid[i, j - 1] == EMPTY:
110 if newPherGrid[i, j] > THRESHOLD:
111 newPherGrid[i, j] += DEPOSIT
112 newAntGrid[i, j], newAntGrid[i, j - 1] = EMPTY, EAST
113 else:
114 newAntGrid[i, j] = STAY  # Can't move
115
116 return newAntGrid, newPherGrid
117
118 # Generating a color grid for visualization
119 def makeColorGridFloat(antGrid, pherGrid, MAXPHER):
120 n = len(antGrid) - 2
121 colorGrid = np.zeros((n + 2, n + 2, 3))
122
123 # Pheromone values are within the [0, MAXPHER] range
124 firstEls = np.clip(1 - deepcopy(pherGrid) / MAXPHER, 0, 1)
125 colorGrid[:, :, 0] = colorGrid[:, :, 1] = colorGrid[:, :, 2] =
    firstEls / 1.2
126
127 for i in range(n + 2):
128 for j in range(n + 2):
129 if antGrid[i, j] == BORDER:
130 colorGrid[i, j] = [0, 0, 1]  # Blue border
131 elif antGrid[i, j] > 0:
132 colorGrid[i, j] = [1, 0, 0]  # Red ants
133
134 # All values are within [0, 1] before returning the color grid
135 colorGrid = np.clip(colorGrid, 0, 1)  # Clamp the RGB values to
    the valid range
136 return colorGrid
137
138 # Ant simulation setup
139 seed(5)
140 n = 50
```

```
probAnt = 0.04
diffusionRate = 0.01
MAXPHER = 50.0
EVAPORATE = 1
DEPOSIT = 2
THRESHOLD = 0
m = 1  # Number of areas of chemical concentrations
t = 100
num_trials = 10

for trial in range(1, num_trials + 1):
print(f"Trial␣{trial}/{num_trials}")

antGrid = initAntGrid(n, probAnt)
pherGrid = initPherGrid(n, m, MAXPHER)

fig = plt.figure()
plt.imshow(makeColorGridFloat(antGrid, pherGrid, MAXPHER),
   animated=True, interpolation='nearest')
plt.title(f"Trial␣{trial}/{num_trials}")
plt.show()

# Simulation loop
for i in range(t):
antGrid = applySense(antGrid, pherGrid)
antGrid, pherGrid = walk(antGrid, pherGrid, EVAPORATE, DEPOSIT,
   THRESHOLD)
pherGrid = applyDiffusion(pherGrid, diffusionRate)

plt.clf()
plt.imshow(makeColorGridFloat(antGrid, pherGrid, MAXPHER))
plt.title(f"Trial␣{trial}/{num_trials}")
plt.show()
clear_output(wait=True)
```