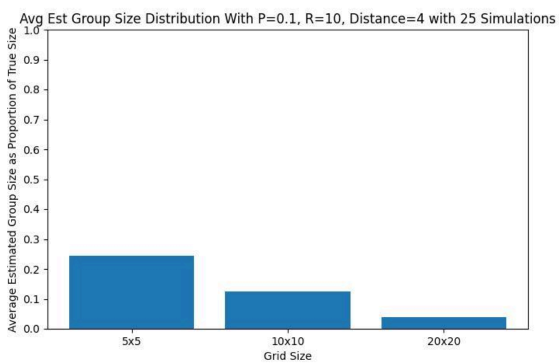
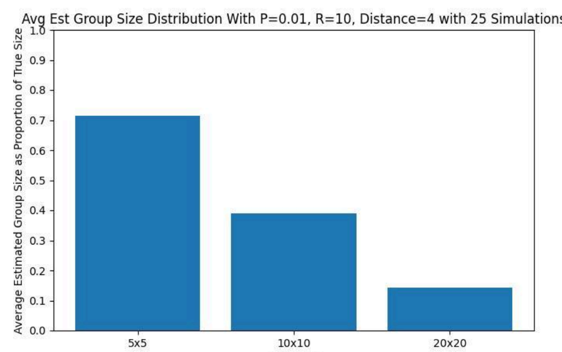
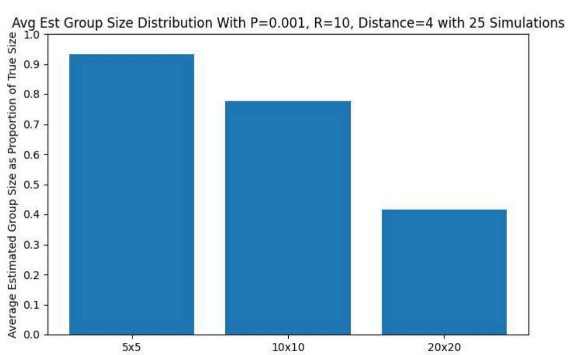
RBE511 - HW#7 Report
Christian Piper
10/13/24

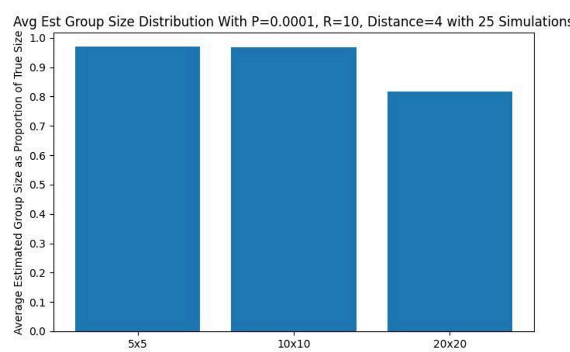
Quorum sensing is a cell-to-cell communication process that allows bacteria to coordinate their behavior in response to changes in cell density. From an algorithmic standpoint, quorum sensing can be viewed as a distributed decision-making process, where each bacterium makes decisions based on local information (the concentration of quorum-sensing molecules) and the decisions of its neighbors. This collective behavior allows bacteria to perform complex tasks, such as biofilm formation, virulence, and antibiotic resistance. In this report, I implement quorum sensing in Python. I vary the parameters of initiation probability (P), refractory time (R), and neighbor range (neighbors of 4 or 8) in small grids of 5x5, 10x10, and 20x20 and observe the effects. Each set of values is run over each group 25 times and the average taken. Initial values of P=0.01, R=10, and neighbors of 4 were chosen. These values will be adjusted one by one to find the best estimates.

By increasing the initiation probability (P value) while keeping the other values constant, the accuracy of the group size estimates improve. This is likely because with a lower probability, there is a lower chance that two initiations will happen within a refractory period of each other. If an agent receives a signal while in the refractory period, it will miss the initiating agent in its count. However, with higher P values, the time to simulate each iteration becomes extremely long. Therefore, for future testing, a compromise must be made between accuracy and simulation time. I will use a value of P=0.001 from here onwards.

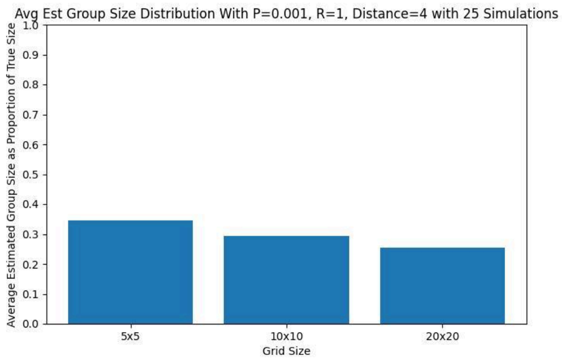


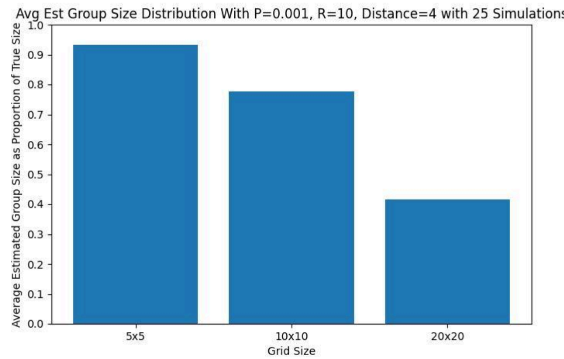


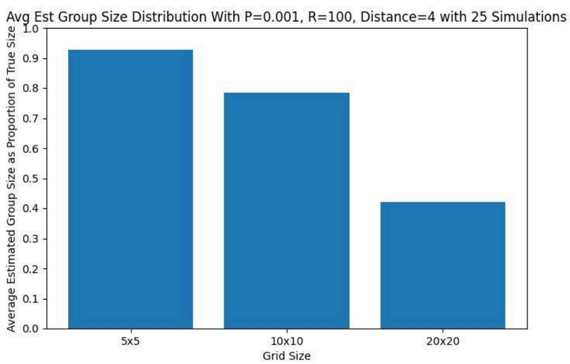


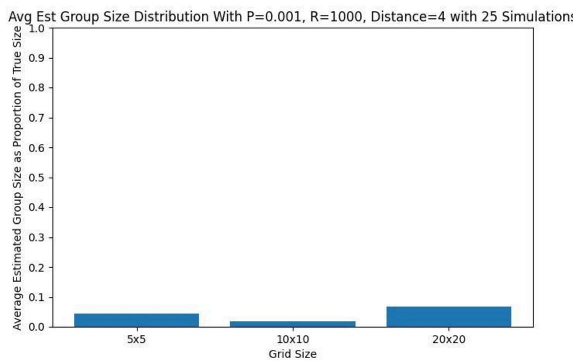


The refractory time (R value) has the opposite effect as the P value. By decreasing the R value while keeping the other values constant, the accuracy of the group size estimates improves to a point. This is likely because lower refractory periods reduce the chance that two initiations happen within a refractory period. If an agent receives a signal while in the refractory period, it will miss the initiating agent in its count. However, decreasing the R value to 1 significantly lowers the accuracy of the estimates.

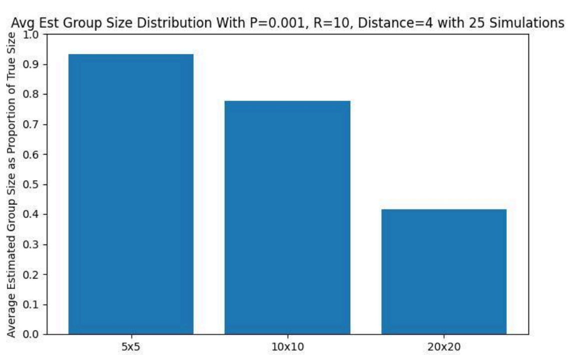


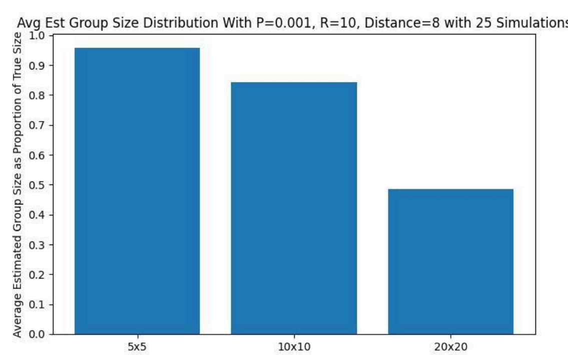






The final parameter to test is the neighbor type. Neighbors of 8 performs marginally better, especially in larger grid sizes. This is likely because the signal spreads throughout the grid faster. With a faster spread, it takes fewer steps for all of the agents to return to susceptible. This reduces the chance that two initiations happen within a refractory period. If an agent receives a signal while in the refractory period, it will miss the initiating agent in its count.





In conclusion, the best parameters I found were an initiation probability (P) of 0.0001, refractory period (R) of 10, and neighborhood of 8. To improve this algorithm, I would remove the refractory period. The agents would always be able to register a signal, ensuring they never miss a count.