Epistemic Landscape Results - Spring 2025

Our previous work in this network epistemology research project was essentially structured as a multi-agent bandit problem. In these results we explore the concept of epistemic landscapes which instead models the simulation as a multi-agent hill climbing problem. The goal is to find the global peak in an unexplored landscape, and to see how restricting dissemination affects agents' ability to do so. When modeling topics such as scientific research, this better captures the idea that agents will make slight (or large) modifications to their approaches that should affect the results, rather than repeatedly choosing from the same 2 (or more) probability distributions. We start with 2d landscapes then move to 3+ dimensional landscapes.

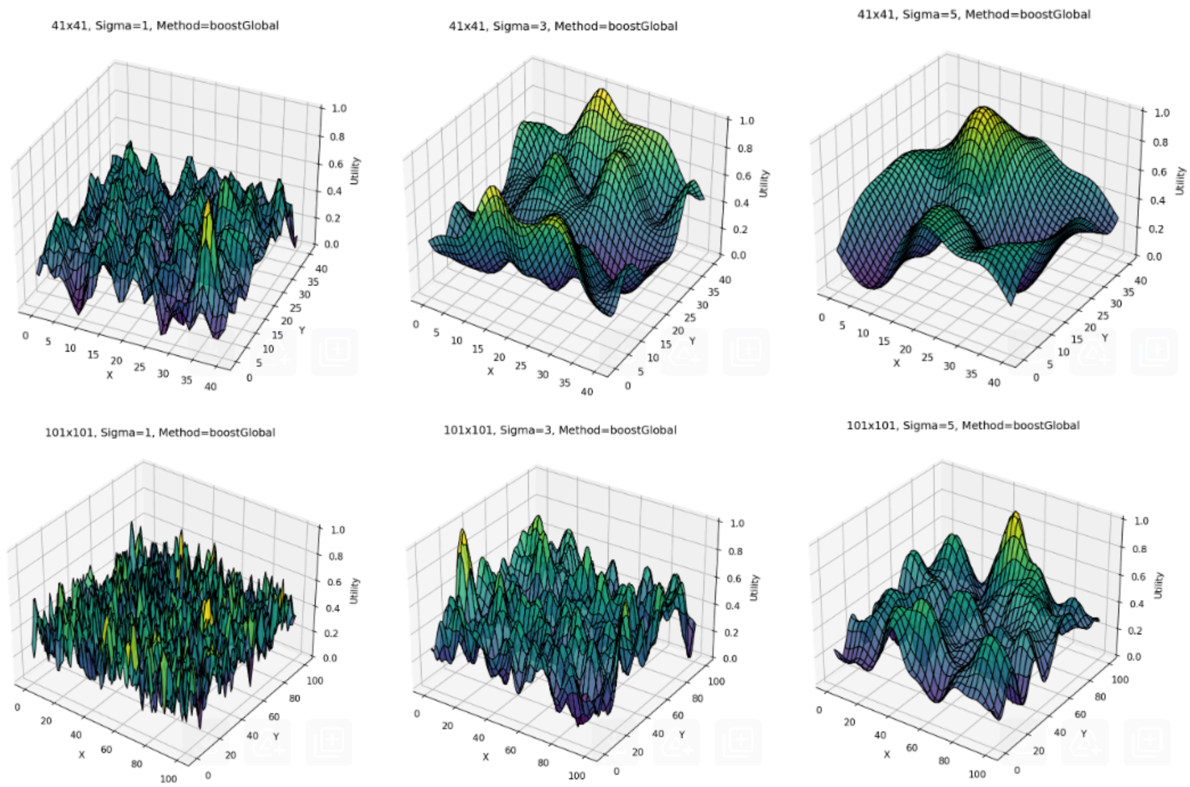
Creating the Landscape

Landscape Generation

The 2d landscapes were created by assigning uniformly random values between 0 and 1 to an n x n grid then smoothing with a gaussian filter and normalizing again so the min value is 0 and the max is 1. The landscapes wrap around, so in a 10x10 grid the coordinate (0, 0) is adjacent to (9, 0).

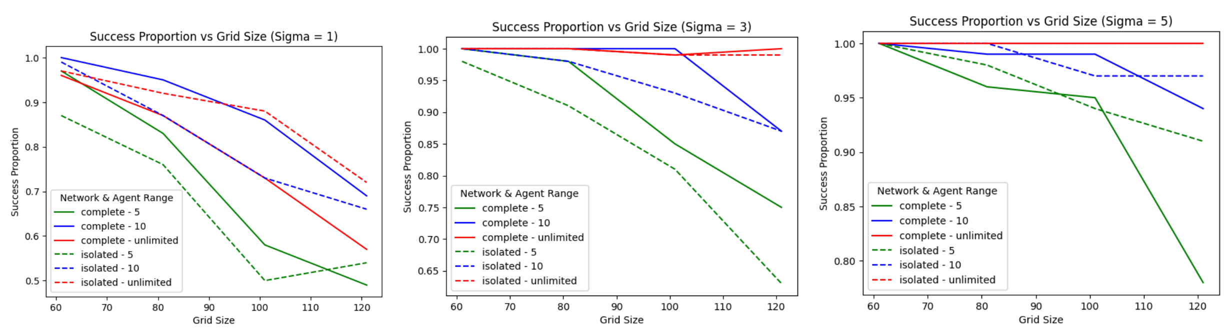
We wanted to make sure there was only one global hill. There were a few methods to achieve this, but the one used in all of the following simulations ("boostGlobal" is what it's called in the codebase) involved finding the global peak, increasing the original value of the point to 2, then reapplying the gaussian filter. In a test of 1000 landscape generations, the second tallest peak was less than 0.9 in 996/1000 trials and averaged a height of 0.62.

The gaussian filter parameter, sigma, affects how rugged or smooth the landscape is. The following chart gives an idea of what this looks like:

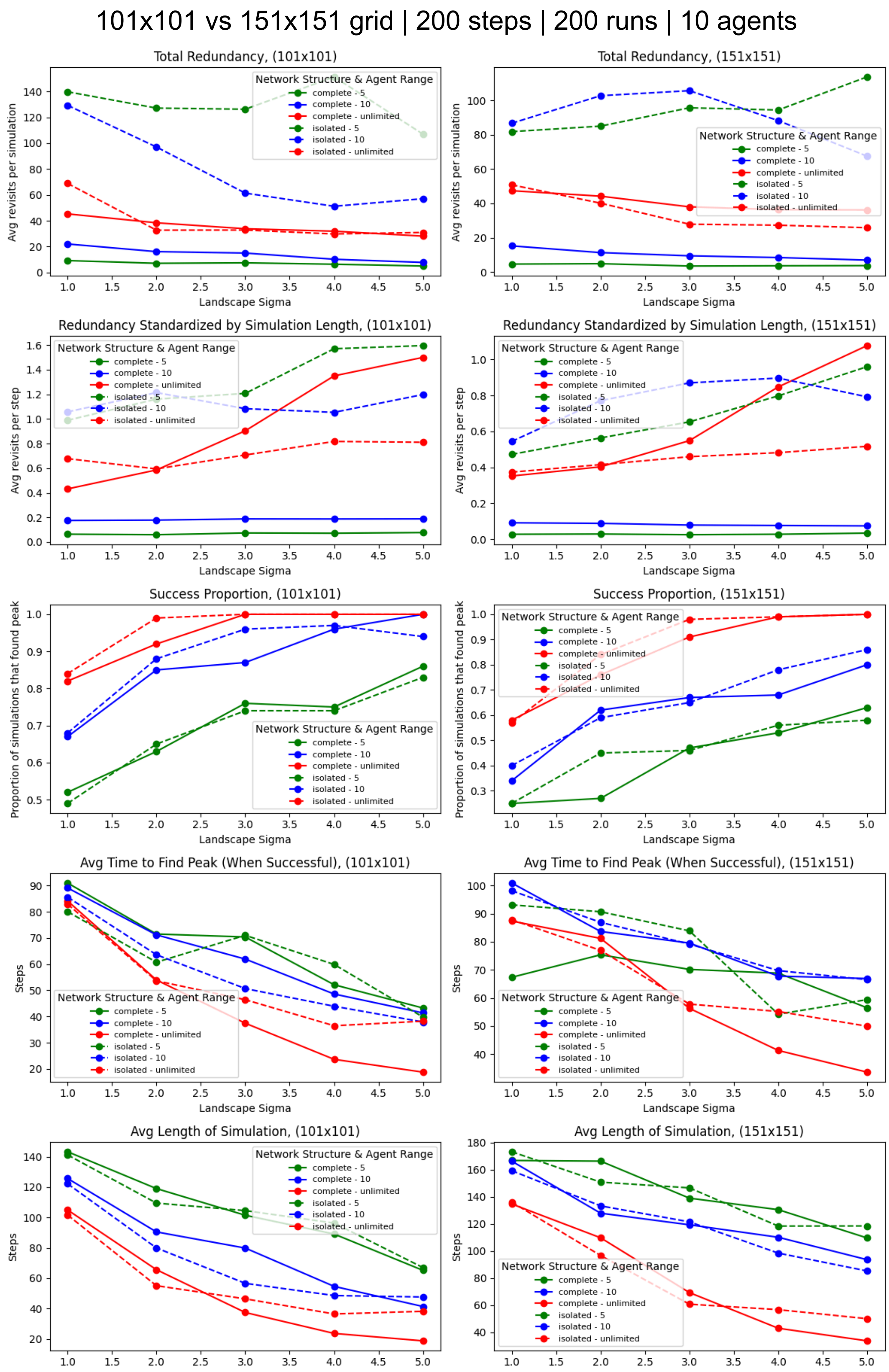


Determining the ideal grid size for simulations

We originally planned on a 41x41 grid, but this made it too easy to find the global peak. Various sizes were tested which led us to use a 101x101 grid for most of the simulations.



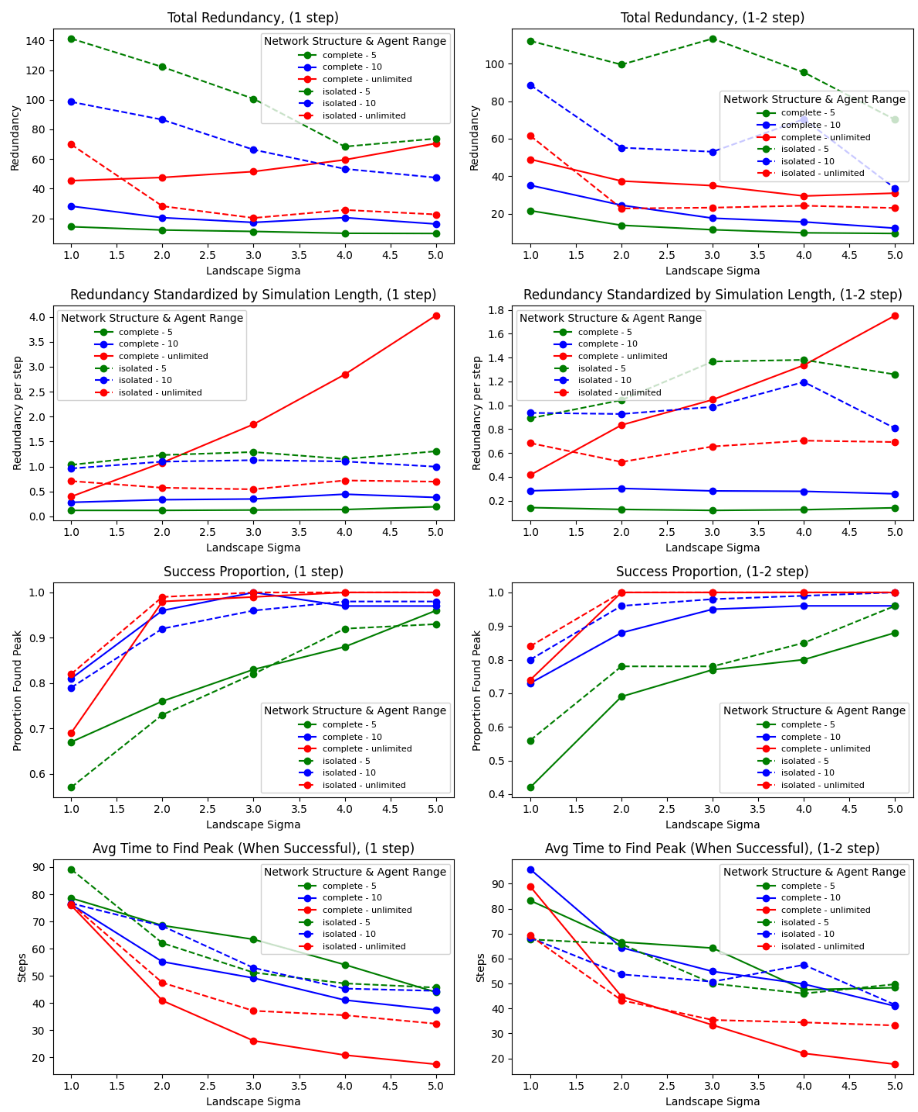
Later on, we also looked at a 151x151 grid, although we still ended up using 101x101 for the other sections of this document.


101x101 vs 151x151 grid | 200 steps | 200 runs | 10 agents
Total Redundancy, (101x101)
Total Redundancy, (151x151)
Avg revisits per simulation
Redundancy Standardized by Simulation Length, (101x101)
Redundancy Standardized by Simulation Length, (151x151)
Avg revisits per step
Success Proportion, (101x101)
Success Proportion, (151x151)
Proportion of simulations that found peak
Avg Time to Find Peak (When Successful), (101x101)
Avg Time to Find Peak (When Successful), (151x151)
Avg Length of Simulation, (101x101)
Avg Length of Simulation, (151x151)
Steps
Landscape Sigma
Network Structure & Agent Range
complete - 5
complete - 10
complete - unlimited
isolated - 5
isolated - 10
isolated - unlimited

Preliminary Results and Updating Agent Algorithm

Originally, agents would choose a new patch to explore by moving one step in an unexplored direction of the current best known location. In the complete network, this resulted in a lot of redundancy as there are only so many adjacent tiles for agents to move to without exploring the same one. It makes more sense that whenever an agent observes a new best tile that was discovered by somebody else to move 1-2 steps away, reducing redundancy and seeming more realistic. All future simulations in this document (as well as the 101x101 vs 151x151 results above) use this strategy instead.

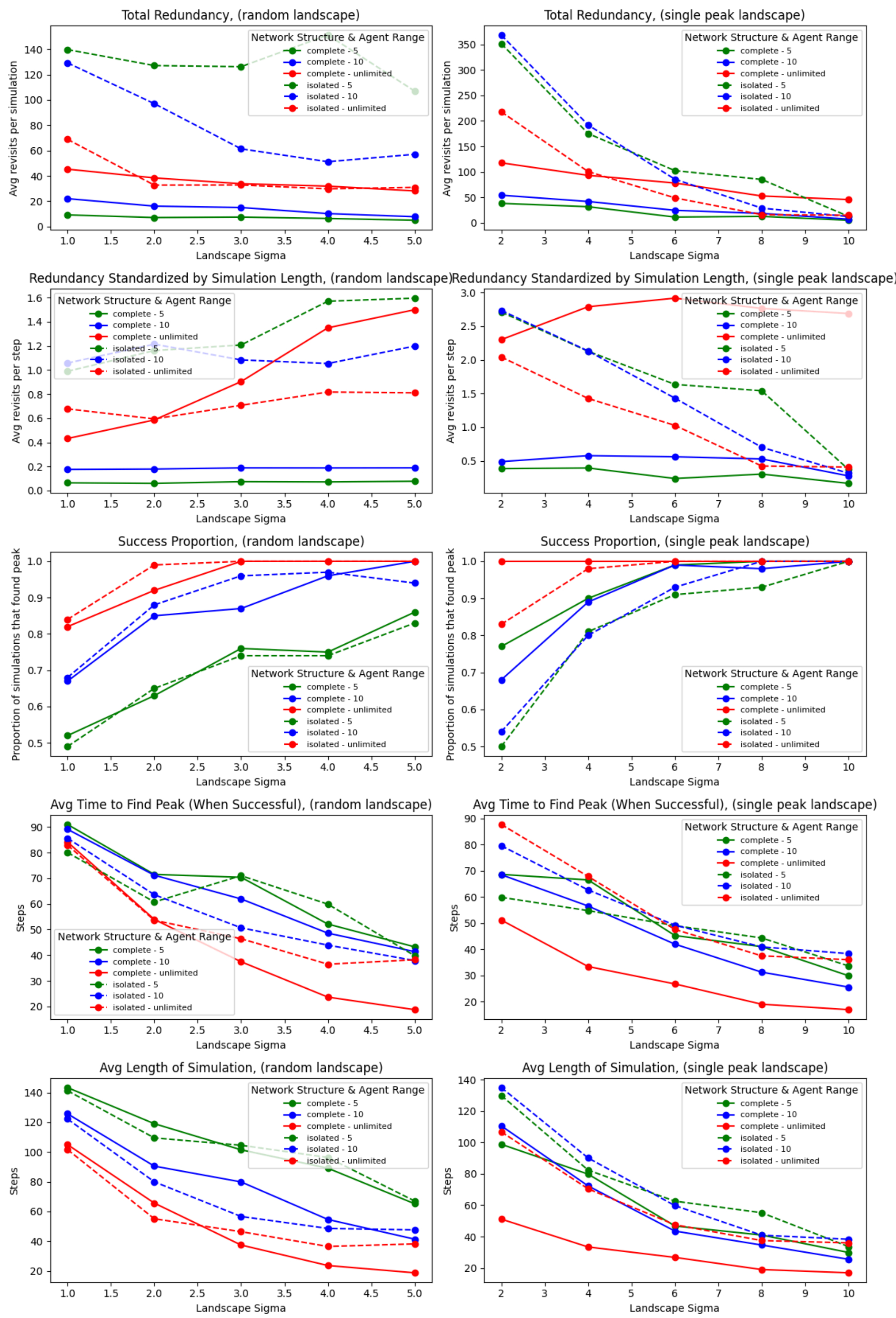
1 step vs 1-2 step algorithm | 200 steps | 200 runs | 10 agents



Single Peak Landscape

To mirror some work in the literature, we also explore a new type of landscape that is mostly flat except for a singular large hill.

101x101 grid | 200 steps | 200 runs | 10 agents

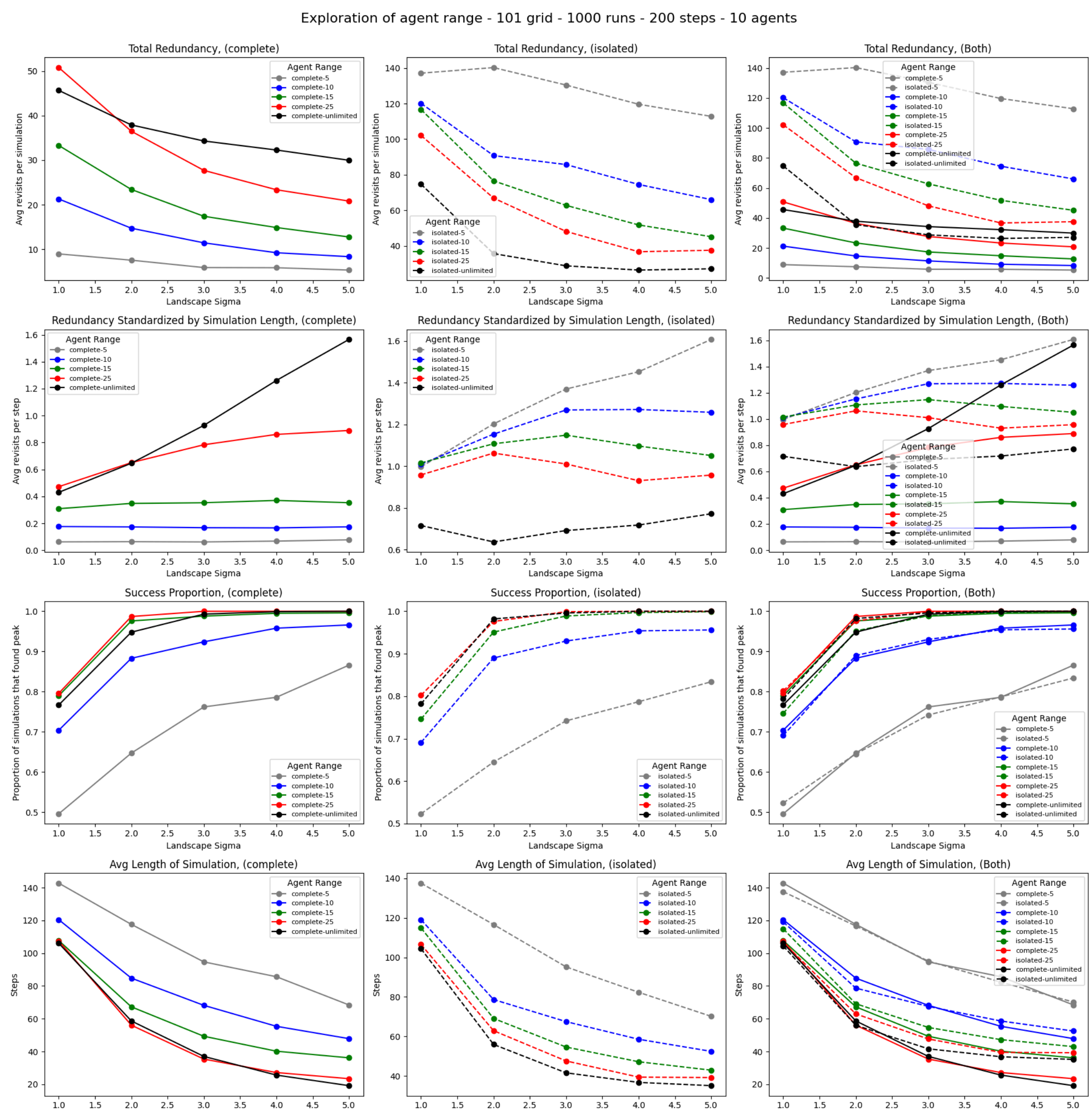


The following 3 sections (exploring agent range, network structure, and epsilon greedy strategy) will look at both the single peak landscape and the traditional random landscape). Notably, the landscape type has a significant effect on the results and conclusions one might draw from them.

Exploring Agent Range

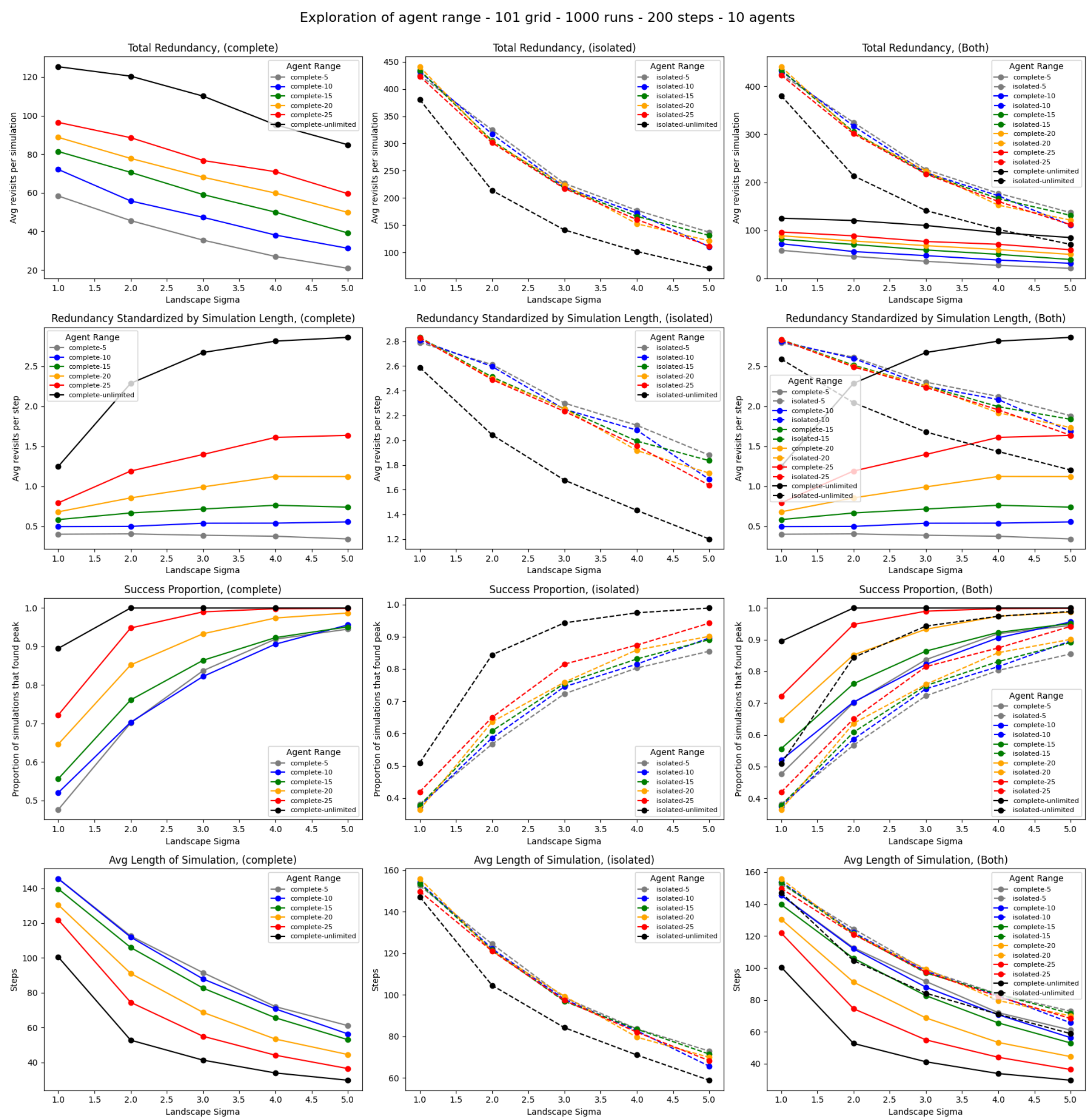
Random Landscape Results:

<u>Key findings:</u> In general, increasing agent range improves success rate, although in the complete network it looks like a range of 25 (in either direction, so the range encompasses 25+1+25=51 x 51 grid) performs better than the unlimited network. Whether the agents are connected or isolated doesn't seem to have a huge affect.



Single Peak Landscape Results:

<u>Key Findings:</u> In general, larger agent range results in higher success proportion. Although, in the complete network it is also accompanied by an increase in redundancy. The connected networks outperform the isolated networks.

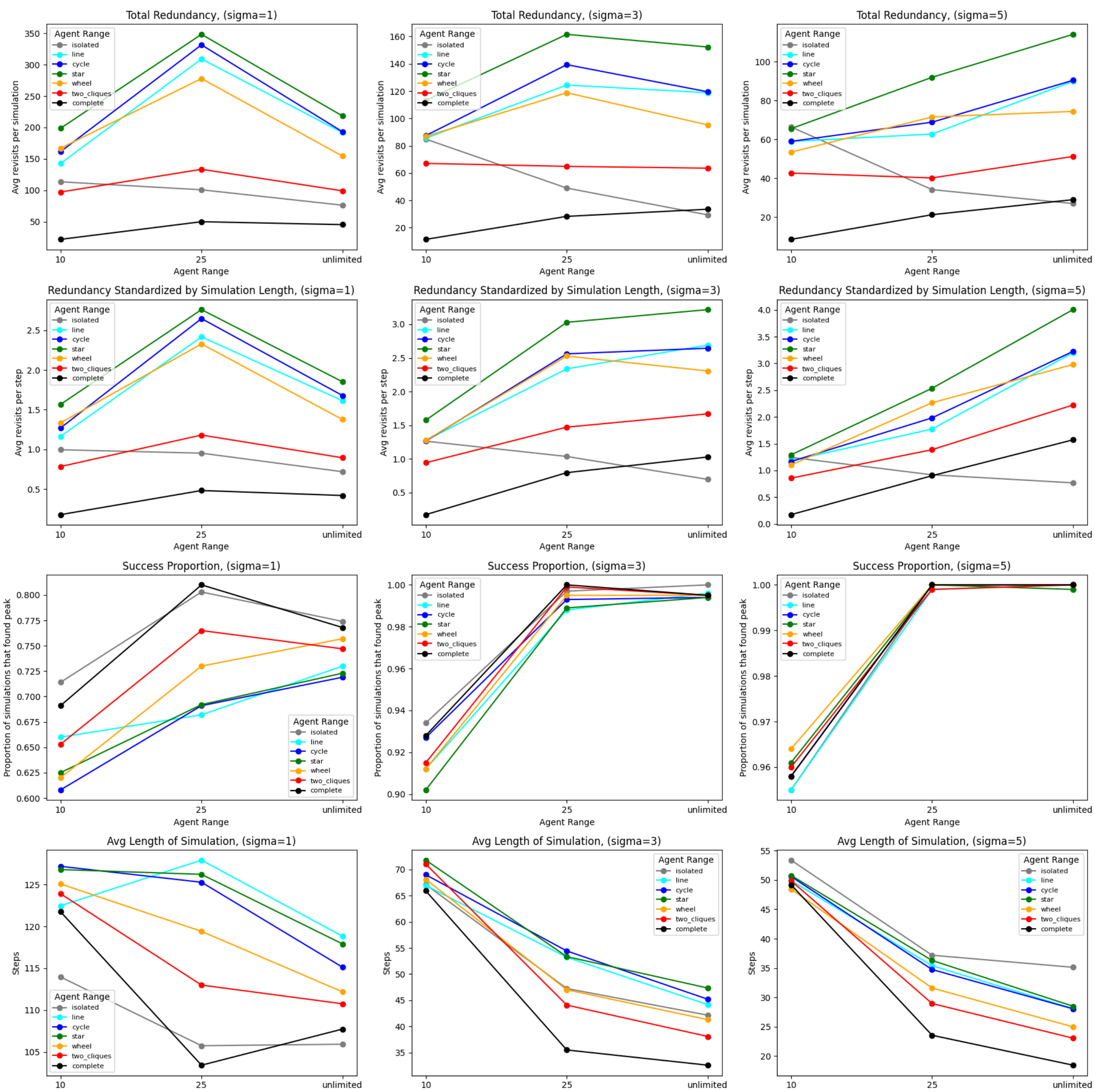


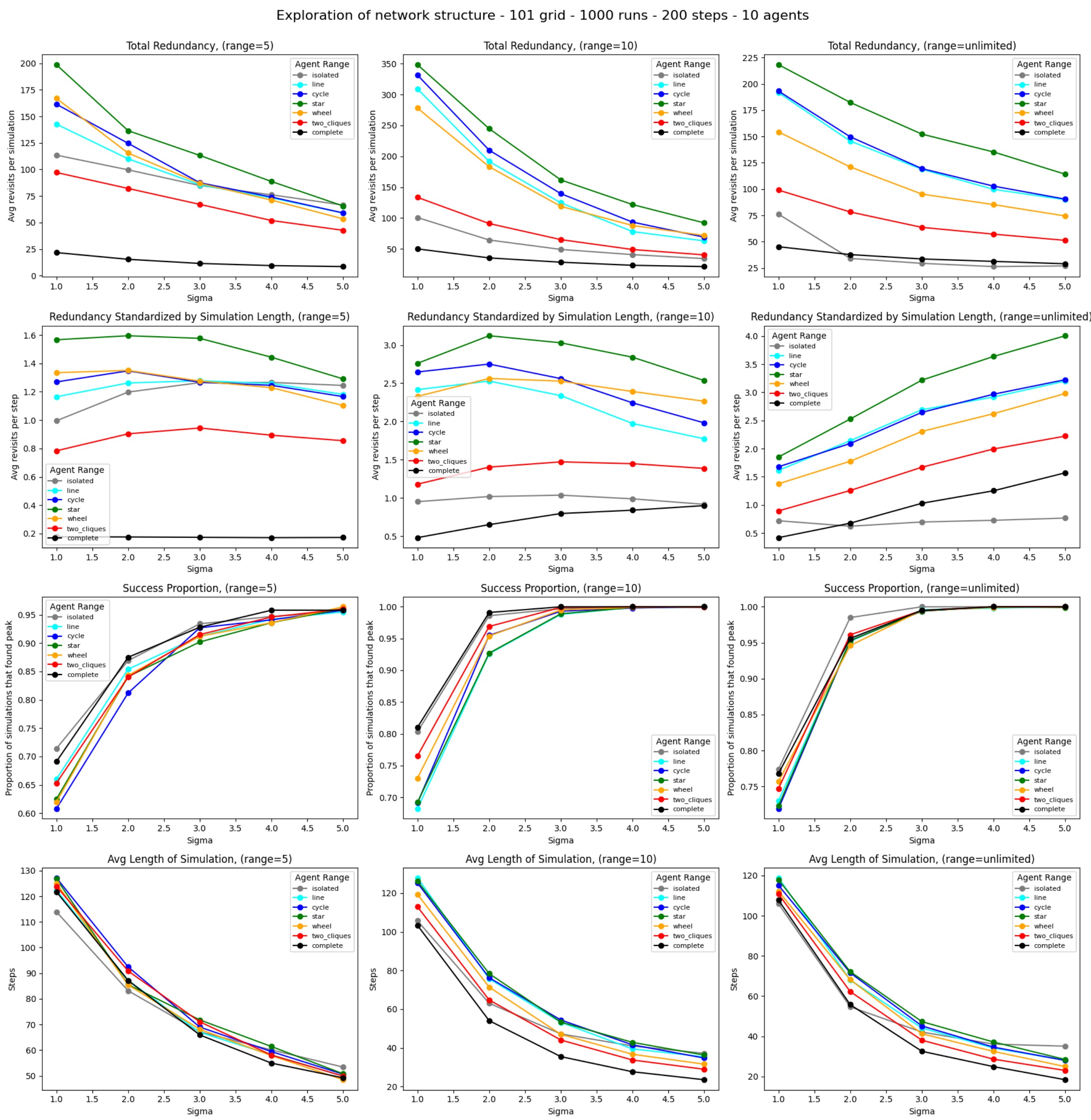
Exploring Different Network Structures

Random Landscape Results:

<u>Key Findings</u>: All networks easily succeed with little differentiation at higher values of sigma (landscape smoothness). Interestingly, the most successful networks are the two extreme cases: completely connected and completely isolated which perform about the same, while all the other partially connected networks perform worse to varying degrees.

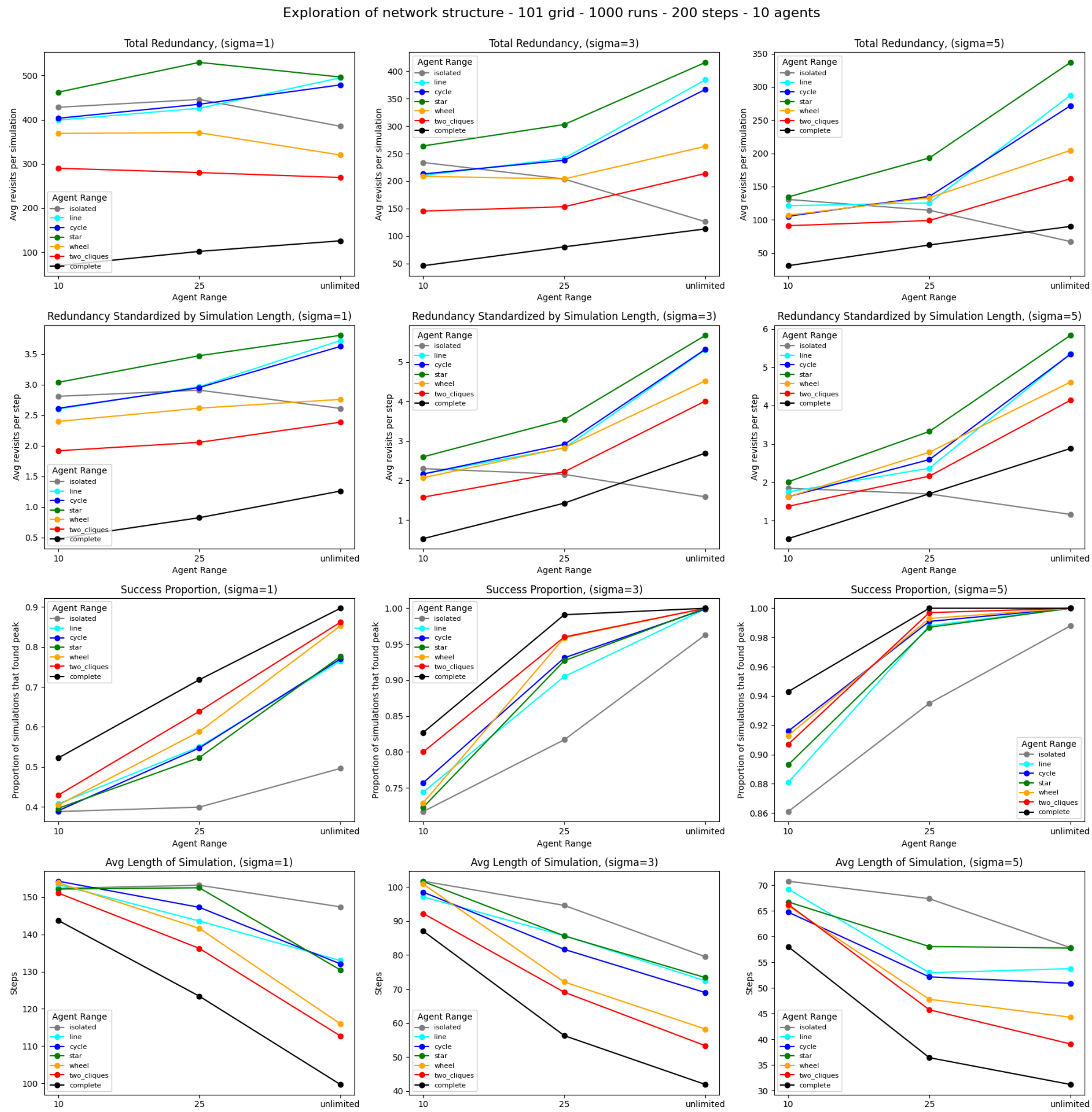
Exploration of network structure - 101 grid - 1000 runs - 200 steps - 10 agents

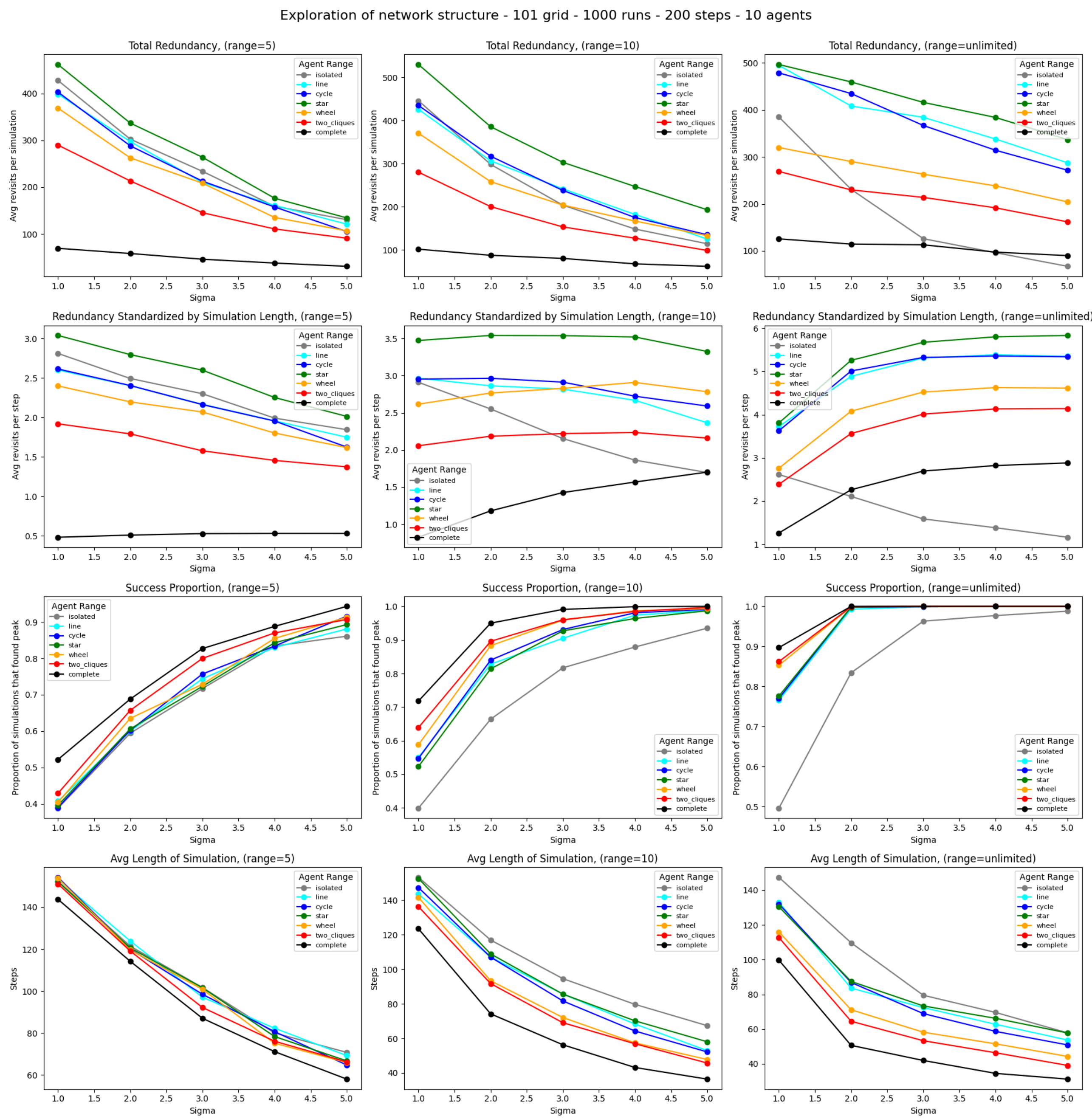



Exploration of network structure - 101 grid - 1000 runs - 200 steps - 10 agents
Total Redundancy, (range=5)
Total Redundancy, (range=10)
Total Redundancy, (range=unlimited)
Avg revisits per simulation
Redundancy Standardized by Simulation Length, (range=5)
Redundancy Standardized by Simulation Length, (range=10)
Redundancy Standardized by Simulation Length, (range=unlimited)
Avg revisits per step
Success Proportion, (range=5)
Success Proportion, (range=10)
Success Proportion, (range=unlimited)
Proportion of simulations that found peak
Avg Length of Simulation, (range=5)
Avg Length of Simulation, (range=10)
Avg Length of Simulation, (range=unlimited)
Steps
Sigma
Agent Range
isolated
line
cycle
star
wheel
two_cliques
complete

Single Peak Landscape Results:

Key Findings: In general, more connected networks have a higher success rate, less redundancy, and find the peak faster.



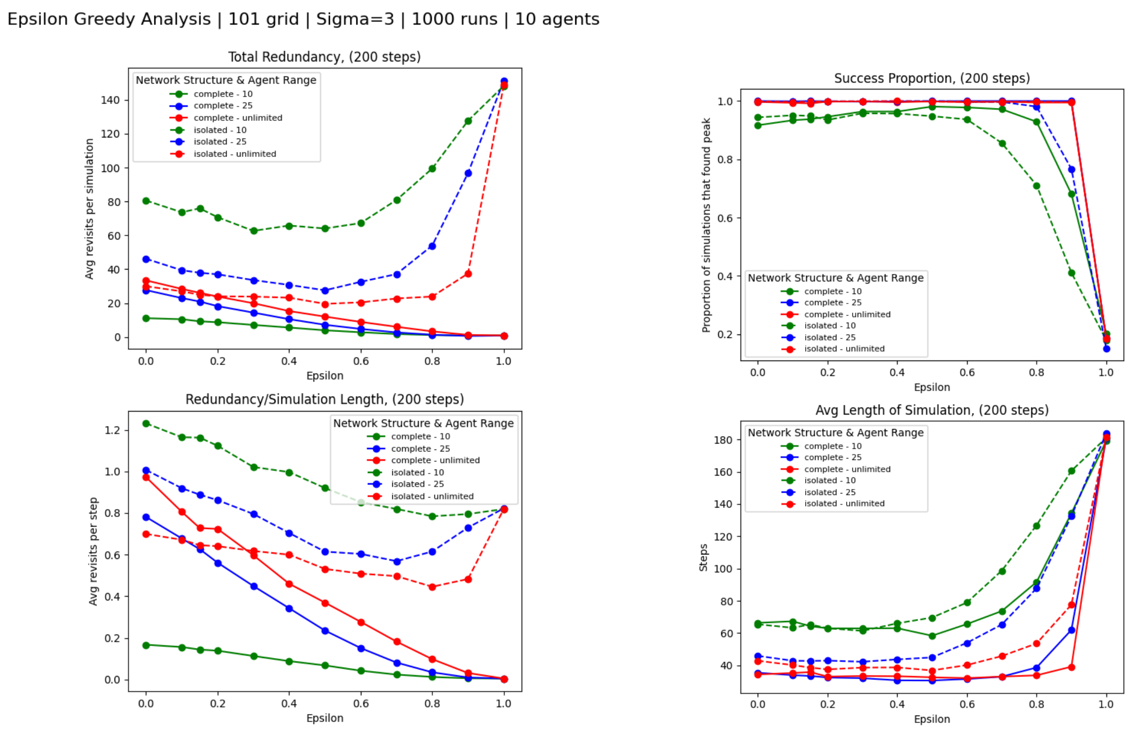

Exploration of network structure - 101 grid - 1000 runs - 200 steps - 10 agents
Total Redundancy, (range=5)
Total Redundancy, (range=10)
Total Redundancy, (range=unlimited)
Redundancy Standardized by Simulation Length, (range=5)
Redundancy Standardized by Simulation Length, (range=10)
Redundancy Standardized by Simulation Length, (range=unlimited)
Success Proportion, (range=5)
Success Proportion, (range=10)
Success Proportion, (range=unlimited)
Avg Length of Simulation, (range=5)
Avg Length of Simulation, (range=10)
Avg Length of Simulation, (range=unlimited)
Avg revisits per simulation
Avg revisits per step
Proportion of simulations that found peak
Steps
Sigma
Agent Range
isolated
line
cycle
star
wheel
two_cliques
complete

Epsilon Greedy

In these simulations, agents explore a random tile in range with probability epsilon and do their normal hill climbing algorithm otherwise.

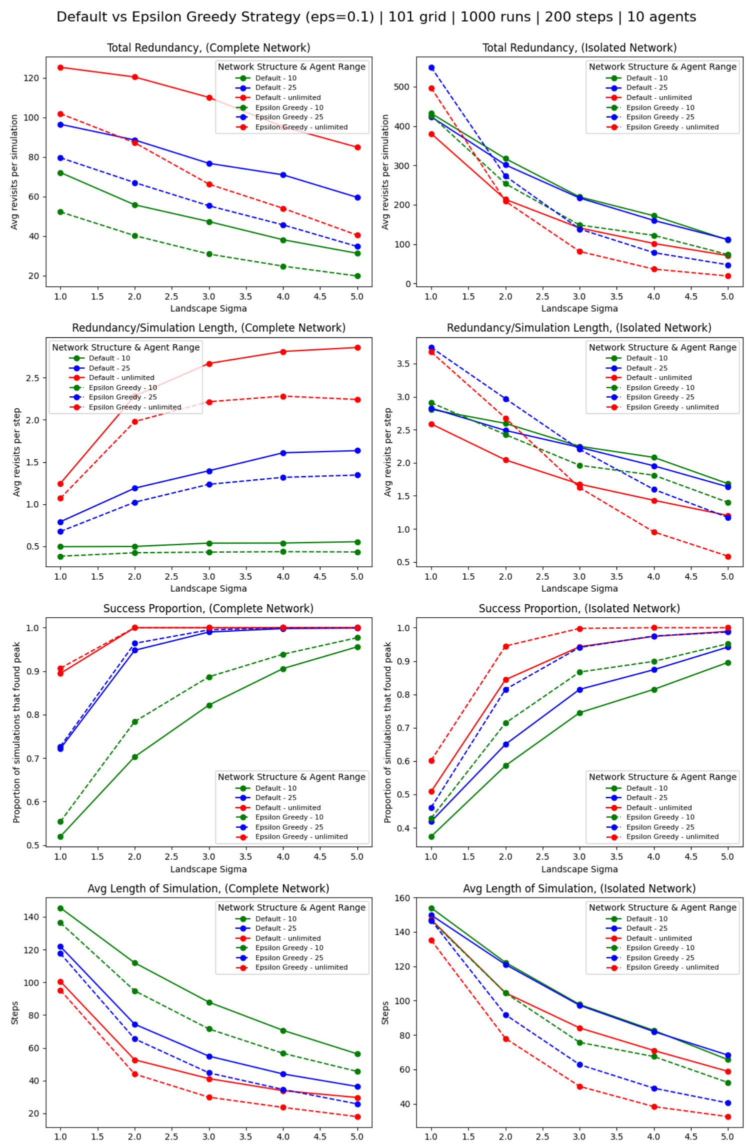
Random Landscape Results:

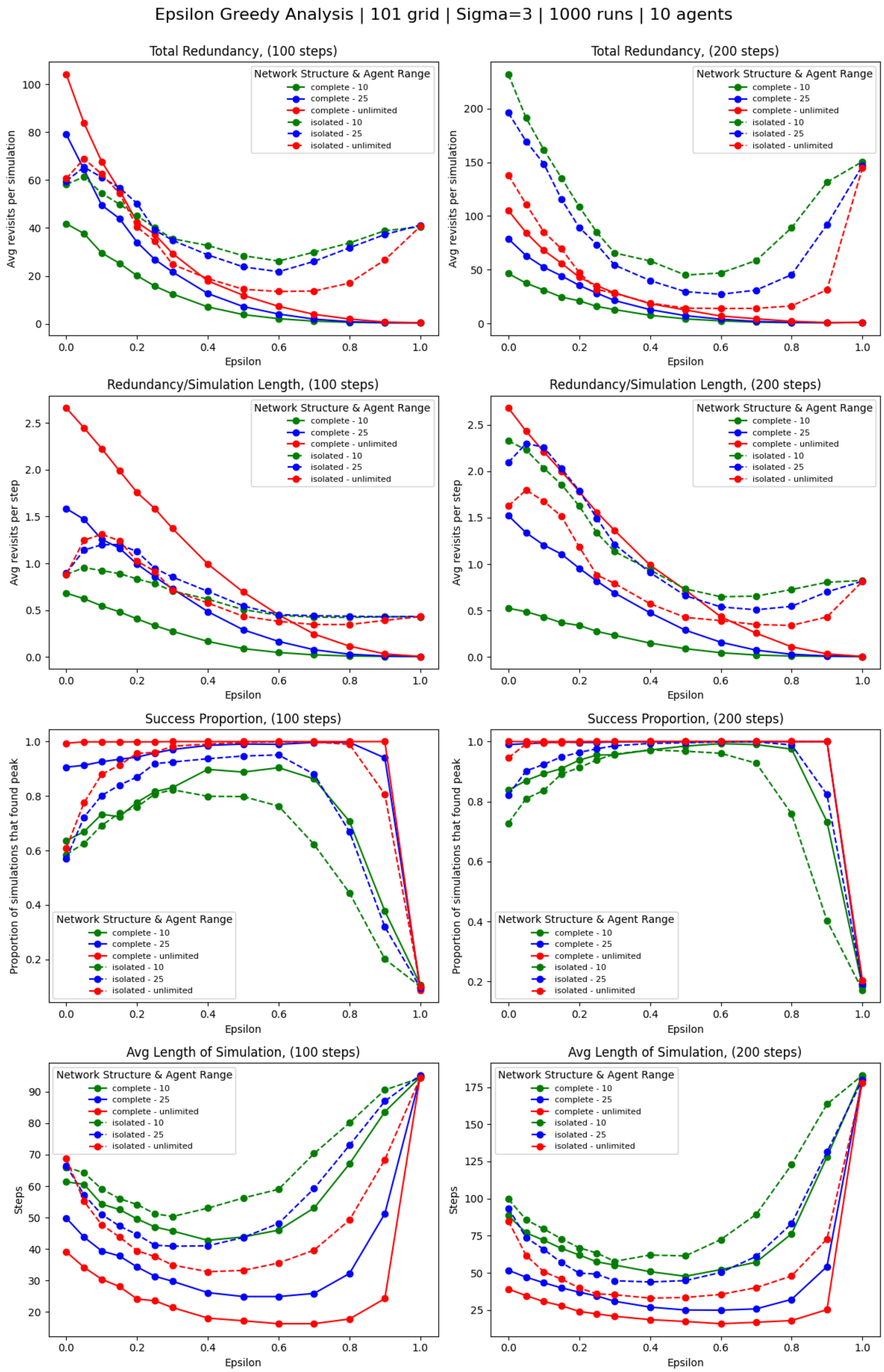
<u>Key Findings:</u> Strategy is very successful for all values of epsilon, although need to look at more sigma values. Using simulation length as a secondary metric of success, it looks like the connected networks perform better (find the peak faster) than the isolated ones, especially as the range gets larger.



Single Peak Landscape Results:
<u>Key Findings:</u> Compared to the default strategy, the epsilon greedy strategy performs about the same and perhaps slightly better in connected networks, and signing=ficantly improves the success rate in the isolated networks. When following epsilon greedy strategy, greater range improves success, and connected networks perform better than isolated ones for values of epsilon.

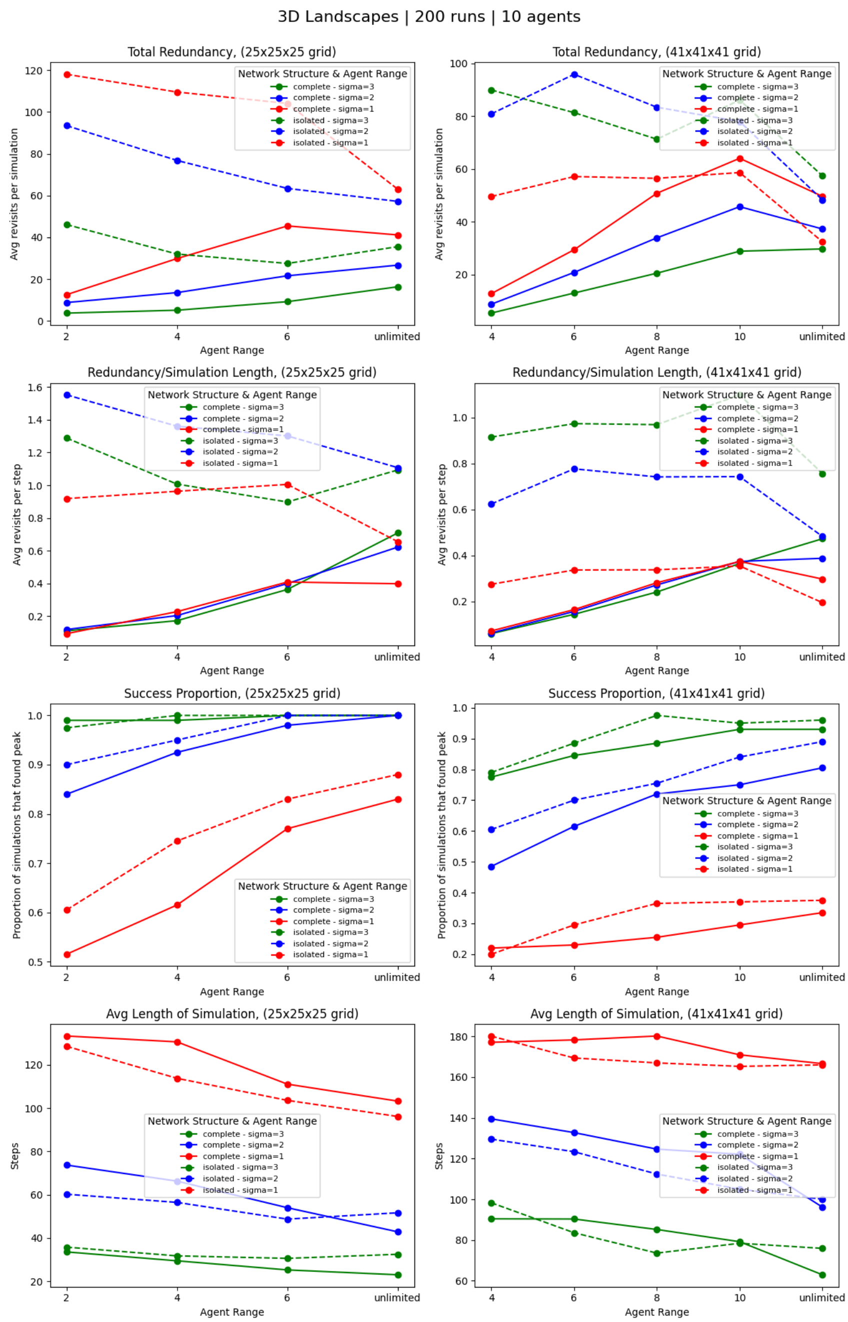
Note that this graph does not follow the convention of solid line = complete network and dashed line = isolated network that you are probably used to from looking at the other graphs. Here is showcasing default vs epsilon greedy strategy.




Epsilon Greedy Analysis | 101 grid | Sigma=3 | 1000 runs | 10 agents
Total Redundancy, (100 steps)
Total Redundancy, (200 steps)
Avg revisits per simulation
Redundancy/Simulation Length, (100 steps)
Redundancy/Simulation Length, (200 steps)
Avg revisits per step
Success Proportion, (100 steps)
Success Proportion, (200 steps)
Proportion of simulations that found peak
Avg Length of Simulation, (100 steps)
Avg Length of Simulation, (200 steps)
Steps
Epsilon
Network Structure & Agent Range
complete - 10
complete - 25
complete - unlimited
isolated - 10
isolated - 25
isolated - unlimited

3d Landscapes

In this first graph the agents follow the original strategy of moving 1-2 steps (in any direction, so 2+1+2 ^ 3 = 125 options) from the current best whenever it's discovered by a different agent. In general the isolated networks have a higher success rate despite having more redundancy. This is thought to potentially result from the large option space making it difficult for the complete networks to methodically climb a hill and hone in on the peak.



In this next graph the complete agents now once again only move 1 step in any direction (so 1+1+1 ^ 3 = 27 options, similar to the 2+1+2 ^ 2 = 25 options in the 2d landscape). Here we see that the complete and isolated networks perform similarly, just like in the 2d landscape.