Final Report: Neuroevolution of Task Specialisaiton for Multi-Agent Systems

Mostafa Rizk

February 1, 2021

1 Introduction

2 Landscape Analysis

What are the fundamental features of the fitness landscape of a task specialisation problem?

2.1 Creating a Fitness Gradient

One conclusion of the research so far has been that the landscape of task specialisation problems appears to be very flat and thus difficult to find any solution at all, let alone a specialised one. If a flat fitness landscape is inherent to task specialisation problems, then this is significant because it implies that algorithms that are highly exploratory should be used, as they are with similar problems [1]. However, it is also possible that the fitness function might be making evolution unnecessarily difficult because it only rewards task completion, treating partial success the same as failure. If partial successes are rewarded, the landscape could have a smoother gradient, making it easier for evolution to traverse. This leads to the following hypothesis:

Hypothesis 1.1: Rewarding agents for partial retrieval of the resource will create a smoother fitness gradient.

This hypothesis is answered by conducting the following experiment:

Experiment 1.1- RWG Analysis with Modified Fitness: Modify the fitness function such that agents are rewarded for partially retrieving a resource. Specifically, divide the reward for retrieving a resource by the length of the arena and give agents part of the reward if they move the resource a single tile closer to the goal. Then conduct RWG analysis as described in [1] and compare the plots to those produced for the original fitness function.

If the mean curve of the distribution has a smoother gradient, then the hypothesis is supported.

The plots for the original landscape analysis and the analysis with the new fitness function are shown in Figure 1 and Figure 2 respectively.

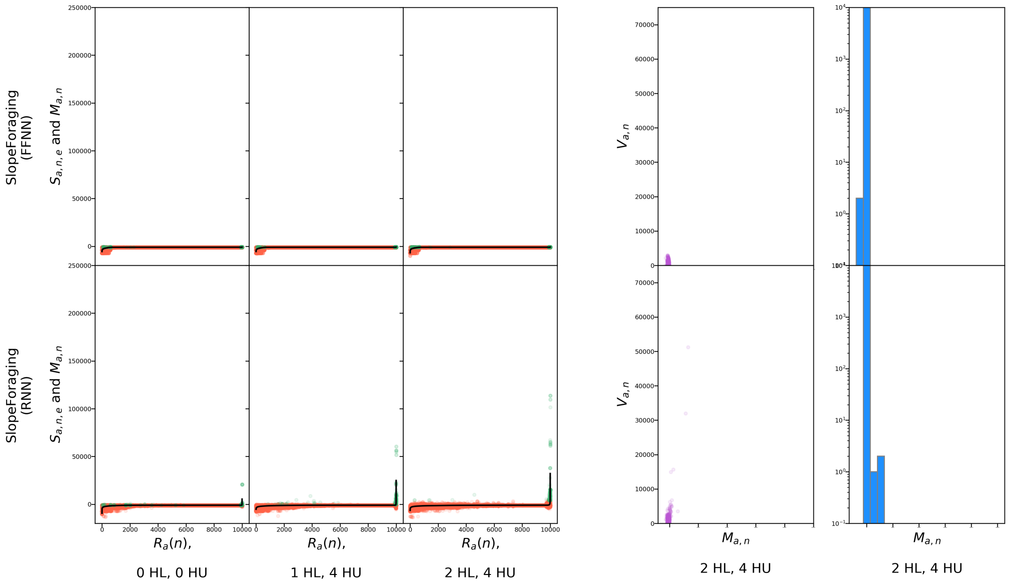

Figure 1: RWG Analysis with the original fitness function

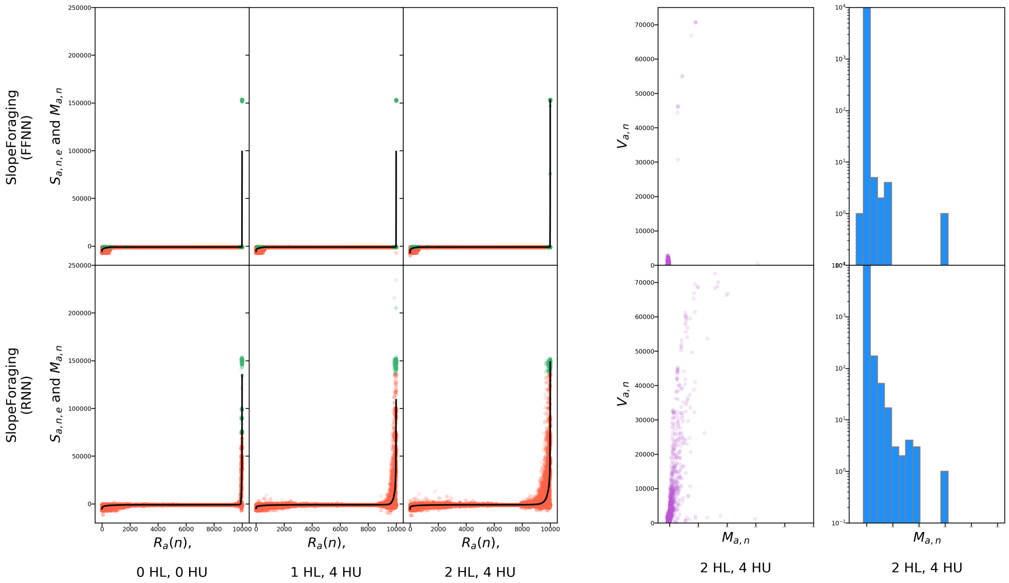

Figure 2: RWG Analysis with incremental rewards

Looking at the 6 mean plots in Figure 2 compared to Figure 1, we see that with incremental rewards, the solutions do in fact form more of a gradient towards the right end of the plot, in particular for RNNs (bottom row). This means that a lot of the solutions in the landscape are partially successful at retrieving resources. However, the landscape is still largely flat. This is presumably because most solutions in the landscape don't even get to the point of picking up a resource from the source.

Thus the answer to Hypothesis 1.1 is:

Answer 1.1: Rewarding agents for partial retrieval of the resource creates a smoother fitness gradient, but the landscape is still largely flat.

In order to further smooth the fitness landscape there would need to be rewards for partial progress up the slope, but the problem with this is that it is likely to disincentivise the collector behaviour in evolution. Why would an agent stay at the nest and collect resource when they are rewarded for going up the slope? Additionally, while this problem is simple enough to smooth the fitness gradient in this way, such smoothing may not be possible for other more complex problems. The purpose of this research is to understand how to use AI to find good solutions with minimal (if any) human intervention.

Moreover, the flatness of the landscape is, to some extent, inherent to this task. This is a problem common in evolutionary computation known as the bootstrapping problem [2, 3]. Bootstrapping is "when the task is too demanding for the fitness function to apply any meaningful selection pressure on a randomly generated population of initial candidate solutions" [2]. The bootstrapping problem often occurs when the goal behaviour is complex relative to the very simple available actions [3]. For this problem, the available actions are primitives like 'move forward one tile' and 'move backward one tile'. When putting together sequences of these actions, most sequences will obviously not be very successful.

There are many proposed solutions to solving the bootstrapping problem and they fall under the broad categories of inserting human knowledge into the learning process or increasing the diversity of solutions [2]. Smoothing the landscape falls under the former category, but this family of techniques has some shortcomings, in particular it reduces the potential for automation and risks the experimenter introducing negative biases. Using rwg to find a seed falls under the second category; it has the shortcoming of additional computation, which may not scale well for problems with larger solution spaces, but for those problems there is a wealth of alternative techniques for increasing diversity, such as novelty search. In keeping with the spirit of AI, I have chosen to continue using the more challenging fitness function.

2.2 Visualising Specialisation

I already performed the initial analysis of the landscape (Figure 1), however, the visualisation did not contain any information about how specialised each team was i.e. was a given solution a pair of generalists or a pair of specialists? Are the better solutions in the landscape usually specialists? To visualise this information I need a way of measuring the degree of specialisation of a solution, so the second task in this group is to evaluate some candidate metrics and the third is to incorporate the chosen metric into the landscape analysis. The hypothesis and experiment are:

Hypothesis 1.2: Specialist solutions outperform generalist ones.

Experiment 1.2- Specialisation Plots of RWG Analysis: Use the chosen specialisation metric to assess all the solutions found by RWG analysis and incorporate them into the plots.

The most specialised solutions are expected to be at the rightmost of the mean plot and above the generalist ones. If this is the case, then it supports the hypothesis. If there are generalist solutions that outperform specialist ones, it disproves the hypothesis.

2.2.1 Specialisation in the Literature

To quantitatively measure specialisation, it is important to first define what specialisation is. For the purpose of this research, specialisation is when a team of agents cooperate, by performing different roles, to accomplish the task more effectively than a team of generalist (i.e. independent) agents. Agents cooperating by doing the same role, e.g. carrying a resource together, is not considered specialisation. Agents performing different roles that do not provide an advantage over a generalist team (e.g. one agent doing a generalist strategy and another agent being stationary) is also not considered specialisation.

This research is only concerned with tasks where specialisation improves performance and how to effectively evolve specialisation for those tasks. In real-world applications, we may not know a priori what the specialist solution to the problem is, but for the slope foraging problem, we know that it involves a subset of agents dropping resources from the top of the slope and the rest of the agents collecting the dropped resources from the cache. Therefore, our metric must be one that indicates these dropper and collector behaviours are being enacted to retrieve resources.

The way specialisation is measured in the literature uses one or more of the following:

Strategy switches

Nitschke et al [4] count times strategy was switched over all possible switches. Less switches means more specialised. Gautrais et al [5] do the same but take the complement. Gigliotta et al [6] counts time spent changing strategies. Pini et al [7, 8] use an analog of Ferrante et al's slope foraging [9], where there is a source and a nest and agents can either take a long corridor to the nest or place items in a cache that transports them to the other side, where their team-mates take them to the nest. They count the number of times the cache was chosen instead of the corridor as an indicator of the degree of specialisation.

These are not easily applicable to this slope foraging implementation because it uses low-level actions, whereas they assume agents can directly choose a high-level strategy. Agents changing from "go forward" to "go backward" doesn't tell much about specialisation. But if agent actions were just "do dropper behaviour", "do collector behaviour" and "do generalist behaviour", it would be more feasible to use measures like these works. Some measures also look at specialisation of the individual rather than the team, which is not as meaningful for slope foraging.

Diversity and Entropy

Li et al [10] group agent genotypes into clusters and use Shannon's entropy to measure the diversity of the team. Each cluster is believed to be a specialist strategy. Specialisation is defined as "the part of diversity that is demanded for better performance" so a diverse team is only specialised if their diversity is correlated to improved performance. Li et al then look at diversity and performance through the learning process and calculate their correlation coefficient as a measure for the degree of specialisation of the final strategy(?)

This is computationally expensive because it requires data on team performance throughout all of the learning process and requires integration over all algorithm configurations (?). Additionally, since the authors use a PFSM, genotypic diversity implies phenotypic diversity (?). Using a neural network, different genotypes might still have the same behaviour. Since then, work has been done on diversity-based search algorithms. That literature uses behaviour characterisation to describe the behaviour being done. Agents could be clustered according to this, but diversity does not necessarily indicate specialisation or cooperation. More recent work by Gomes et al [11] use co-evolution to evolve heterogeneous cooperative teams, they count the number of different agents or populations. Populations with similar behaviours are merged so all agents on the team are different, but again, this does not necessarily indicate that the team has specialised.

O'Donnell and Jeanne [12] calculate the entropy of an individual agent, specifically the proportions of its time spent doing different activities (i.e. the diversity of resources it gathers). They use it directly as the degree of specialisation. This runs into the same problem as before, where the metric is difficult to apply to slope foraging because agents do not choose high-level strategies. In addition the metric is for individuals and does not apply to this context because agents only gather one type of resource. Balch?

Resource retrieval

Ferrante et al [9] have a comparatively simple metric. They count the proportion of resources retrieved cooperatively over the total resources retrieved. This is independent of whether actions are high or low level and doesn't need the full evolutionary history. It is designed for slope foraging so is appropriate, though it has shortcomings. Cooperative retrieval simply means a resource was picked up by two or more agents before being delivered. An agent could pick up and drop the resource at the source and a generalist could come to collect it and that would count as a specialist strategy. Technically, this is still a dropper-collector strategy, but it does not capture the type of dropper-collector strategy that improves performance.

Additionally, it does not account for 'hitch-hikers' i.e. agents doing nothing while the rest of the team retrieves all resources cooperatively. Since the Ferrante setup is homogeneous (i.e. all agents have the same policy), it is unlikely that some would cooperate while others hitch-hike, but since this research considers heterogeneous teams, hitch-hikers are more likely to be encountered, so the specialisation metric should distinguish between teams that have a hitch-hiker and teams that don't. It also can't distinguish teams where a dropper and collector swap roles part-way. This is unlikely to occur, though. In short, the Ferrante metric's main shortcoming is that it does not distinguish *imperfect* specialists.

2.2.2 Choosing a Metric

Based on a review of techniques in the literature, the metric used by [9] is the most appropriate. However, to overcome its shortcomings, I propose to add two new terms to the formula. Ferrante et al measure a term I will refer to as the *cooperation rate* or:

$$R_{coop} = \frac{N_{coop}}{N_{total}}$$

where:

N_{coop} is The number of resources retrieved cooperatively (i.e. picked up by at least 2 agents before being delivered to the nest)

N_{total} is The total number of resources retrieved by the team

To account for instances of imperfect specialisation where agents cooperatively retrieve a resource but drop or collect it in the wrong place, I introduce the *efficient cooperation rate*:

$$R'_{coop} = \frac{N'_{coop}}{N_{total}}$$

where:

N_{spec} is The number of resources retrieved using efficient cooperation (i.e. the resource was dropped on the slope, picked up from the cache and the agents that dropped and picked up were different)

To capture the spectrum of specialisation, I take the average of the two terms:

$$R_{spec} = \frac{R_{coop} + R'_{coop}}{2}$$

To detect "hitch-hikers", I also introduce the *agent participation*:

$$P = \frac{A_{active}}{A_{total}}$$

where:

A_{active} is The number of agents that participated in the retrieval of at least one resource

A_{total} is The total number of agents on the team

The final measure of degree of specialisation is then:

$$S = R_{spec} \times P$$

One shortcoming of this metric is that it does not detect role-switching. For example, a team composed of a dropper and a collector but halfway through, the dropper becomes the collector and vice versa. Role-switching can be detected by observing the variance of an agent's y-coordinate in the arena. A dropper or collector that remains in their role throughout runtime will move at most 1 tile in either y direction, so they will have a low variance. An dropper that changes roles will have a high variance, akin to a generalist. The average y-variance of a team indicates role-switching and, to an extent, specialisation. The average variance could potentially be used to measure specialisation on its own, though it would not detect hitch-hikers or even just teams that are completely stationary. For this reason, and to avoid over-complicating the metric, I only use the variance as an additional measure to check for role-switching.

2.2.3 Results

Team Strategy	$\boldsymbol{R_{coop}}$	$\boldsymbol{R'_{coop}}$	$\boldsymbol{R_{spec}}$
Dropper + Collector	1.0	1.0	1.0
Generalist + Generalist	0.0	0.0	0.0
Inefficient Dropper + Collector			
Dropper + Collector + Hitch-hiker			
Swapper + Swapper			
Lazy Generalist + Lazy Generalist			

Team Strategy	$\boldsymbol{R_{coop} \times P}$	$\boldsymbol{R'_{coop} \times P}$	$\boldsymbol{R_{spec} \times P}$
Dropper + Collector	1.0	1.0	1.0
Generalist + Generalist	0.0	0.0	0.0
Inefficient Dropper + Collector			
Dropper + Collector + Hitch-hiker			
Swapper + Swapper			
Lazy Generalist + Lazy Generalist			

2.3 Dimensionality Reduction

The neural networks used to solve the Slope Foraging problem have hundreds of weights, meaning that the fitness landscape has just as many dimensions and is consequently very difficult to visualise. Reducing the dimensionality of the problem to two dimensions (with a third dimension for fitness values) makes visualisation possible and gives many valuable insights into the features of the landscape, especially if combined with a specialisation metric. How far apart are specialist and generalist solutions in the landscape? How many optima are there? What trajectory does evolution take through the landscape?
The hypothesis is:

Hypothesis 1.3: There are two optima in the landscape, one composed of specialist solutions and one composed of mostly generalist solutions.

Experiment 1.3- Dimensionality Reduction: Assemble a dataset of solutions, combining some random ones from rwg and some evolved ones. Apply the chosen dimensionality reduction technique to the dataset. Use the chosen specialisation metric to plot the degree of specialisation (in addition to the fitness).

If there are indeed two optima, then the hypothesis is supported, otherwise it is not supported. If, additionally, the specialist solutions are fitter than the generalist ones, Hypothesis 1.2 is further supported.

2.3.1 Choosing a Dimensionality Reduction Algorithm

In [13], the authors use local optima networks [citation for LONs] to represent optima and t-SNE algorithm [14] for dimensionality reduction. But this technique only works for discrete problems. In [15] the authors adapt LONs to continuous landscapes as a proof of concept, though they mention that the resulting visualisation is highly dependant on the sampling method.
Sammon mapping [16] has been used for genetic algorithms [17] and neuroevolution [18, 19] in the past. It has often been chosen due to its ability to preserve relative distances between genomes in the reduced representation.
More recent work in the neuroevolution literature, though, [20, 21] uses t-

SNE [14].

The t-SNE algorithm was designed, in part, to overcome the "crowding" problem often faced by "local techniques for multidimensional scaling", where "the area of the two-dimensional map that is available to accommodate moderately distant datapoints will not be nearly large enough compared with the area available to accommodate nearby datapoint" so datapoints with small distances between them, if modelled accurately, means the moderately distant points are too far away and the spaces between clusters are too small, resulting in datapoints from different clusters all crowding together [14].

Though [20] visualises the space of behaviour characterisations (i.e. the space of phenotypes) rather than the fitness landscape (the space of genotypes). Additionally, their objective is to visualise the evolutionary trajectory rather than gain an understanding of the fitness landscape. Similarly, [21] use t-SNE to compress the observation space rather than the genotype space.

Recent work recommends using Latin Hypercube Sampling (LHS) in continuous spaces "to prevent over- sampling of the centre of the search space" [22]. LHS may also be better for high-dimensional spaces [23]

LHS divides the cumulative distribution function of each variable (in this case, each neural network weight) into bins equal to the number of samples [24]. It takes one sample from each bin and does the same for each weight. The samples for each weight are then combined randomly to form a genome.

Based on the literature, I have chosen to use t-SNE with latin hypercube sampling. For the purposes of this study, I only need to identify how specialist and generalist solutions relate to one another. Local Optima Networks, adapted to continuous landscapes, could be used for a more thorough analysis of the landscape, but is not strictly necessary and given that it is an emerging area of research, it is worthy of a research project unto itself and so is left for future work.

In LHS, sampled values must be within a certain range. I chose this range by taking 60 evolved team genomes and plotting the distribution of values for each genome, shown in Figure 3. According to the plot, all weights fall in the (-6,6) range, so I chose this as the range for LHS. The importance of choosing a range is discussed in [1] where the authors state that the "the bounds of the distribution (be there hard bounds as in the case of uniform distributions, or soft bounds as with the variance of a Gaussian) define the tested value range for the weights, which may be critical depending on application and network size".

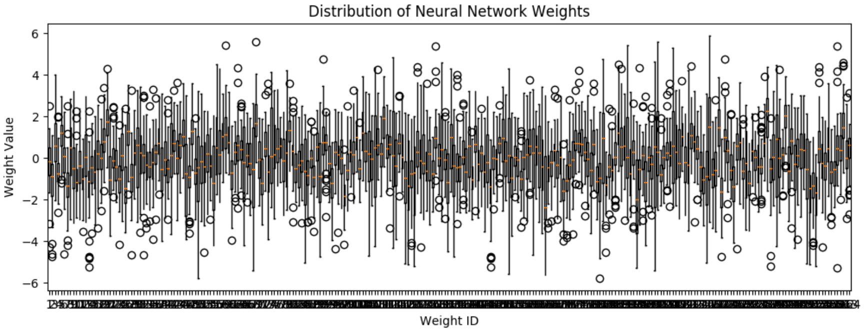

Figure 3: Distribution of weights in evolved genomes

2.3.2 Results

The algorithm I use to perform dimensionality reduction reduces the landscape to two dimensions. I then add a third dimension for visualising the fitness as height and I use colour to indicate the degree of specialisation. The plot in Figure 4 includes the best 500 and worst 500 genomes from the rwg analysis in Figure 1 along with some genomes evolved using a centralised learner and single-population coevolution. In the case of single population coevolution, only one genome is evolved, so I concatenated two identical copies of those genomes and doubled their score so that I can put them in the same landscape.

Preliminary results show that there indeed appear to be two optima, which supports Hypothesis 1.3. One of the optima has a higher score while the other has a lower score. Anecdotally, the scores observed for the green optima are usually only seen for teams of specialists, however a specialisation metric is still needed to verify this empirically.

2.4 Analysis of Different Slope Angles

Hypothesis 1.4.1: When the slope is flat, there will only be one optima of generalists and no specialist solutions.

Hypothesis 1.4.2: An intermediate slope will have two optima (one specialist and one generalist).

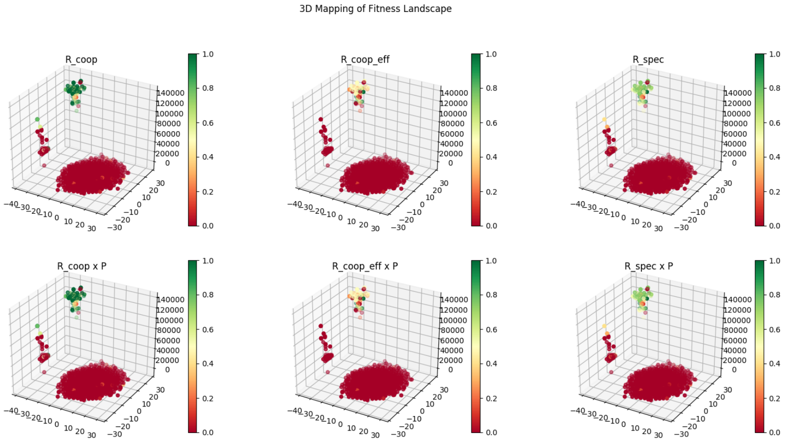

Figure 4: Reduced Fitness Landscape

Hypothesis 1.4.3: The specialist solutions will outscore the generalists, but the disparity will be smaller than the steeper slope setting.

Experiment 1.4- RWG with Different Slope Angles: Repeat RWG, as it was done for previous experiments, but with no slope and intermediate slope. Then perform 30 evolutionary runs, each with a different seed genome and random seed. Use the generated data to create RWG analysis plots and perform dimensionality reduction.

2.5 Decentralised Solutions

Hypothesis 1.5: Solutions found by the decentralised algorithm will be primarily generalist whereas solutions found by the centralised algorithm will be primarily specialist.

Experiment 1.5- Decentralised Dimensionality Reduction: Perform 30 evolutionary runs of the decentralised setup. Calculate the degree of specialisation of all evolved solutions. Concatenate the genomes of all individuals on a team and apply dimensionality reduction to the data. Combine the decentralised data with the centralised data and plot on the same plot.

3 Centralised vs Decentralised Solutions

When do centralised and decentralised solutions outperform one another in task specialisation problems?

3.1 Scalability

Hypothesis 2.1.1: If evolution is run with larger team sizes, the fitness of centralised solutions will decrease.

Hypothesis 2.1.2: If evolution is run with larger team sizes, the fitness of decentralised solutions will decrease less rapidly than centralised solutions

Hypothesis 2.1.3 If evolution is run with larger team sizes, the centralised approach will find fewer specialists.

Hypothesis 2.1.4 If evolution is run with larger team sizes, the decentralised approach will find the same number of specialists.

Experiment 2.1- Evolution with Larger Teams: Do 30 evolutionary runs (centralised and decentralised) for teams of 4, 6, 8 and 10 agents. Calculate the specialisation of all evolved solutions. Plot the average fitness per agent and team specialisation for teams in each setup.

3.2 Robustness

Hypothesis 2.2.1: When evolved teams have team-members replaced, the average fitness per agent decreases for centralised teams.

Hypothesis 2.2.2: When evolved teams have team-members replaced, the average fitness per agent decreases for decentralised teams, but by a smaller margin than centralised teams.

Hypothesis 2.2.3: When evolved teams have team-members replaced, the team specialisation decreases for centralised teams.

Hypothesis 2.2.4: When evolved teams have team-members replaced, the team specialisation remains the same for decentralised teams.

Experiment 2.2- Changing Team-mates: Take all 60 evolved 2-agent teams (30 centralised + 30 decentralised). For each evolved team, replace one agent 4 times: once with a dropper, once with a collector, once with a generalist and once with a stationary agent. Calculate the average fitness and team specialisation then repeat for the other agent on the team. That is a total of $2 \times 30 \times 4 \times 2 = 480$ fitness evaluations. Create a violin plot for the centralised setup and one for decentralised setup. Draw one violin for the original team and one for each replacement (a total of five). The data for each violin includes when each of the two agents was replaced. The y-axis shows the fitness and the plots also have a colour scale to show the degree of specialisation.

4 Theoretical Modelling

What is the price of anarchy in task specialisation problems?

Hypothesis 3.1.1: The slope foraging problem is a stag hunt with two Nash equilibria, where the payoff-dominant equilibrium is specialist solutions and the risk-dominant is generalist solutions.

Hypothesis 3.1.2: Socially optimal teams are more likely to converge to the payoff-dominant Nash equilibrum while the self-interested teams are more likely to converge to the risk-dominant Nash equilibrium.

Experiment 3.1- Game-theoretic Modelling: Formulate a game that models the experimental setup. Find the nash equilibria and study the learning dynamics.

References

[1] D. Oller, T. Glasmachers, and G. Cuccu, "Analyzing reinforcement learning benchmarks with random weight guessing," in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 975–982, 2020.

[2] F. Silva, M. Duarte, L. Correia, S. M. Oliveira, and A. L. Christensen, "Open issues in evolutionary robotics," *Evolutionary computation*, vol. 24, no. 2, pp. 205–236, 2016.

[3] Y. Wei, M. Hiraga, K. Ohkura, and Z. Car, "Autonomous task allocation by artificial evolution for robotic swarms in complex tasks," *Artificial Life and Robotics*, vol. 24, no. 1, pp. 127–134, 2019.

[4] G. S. Nitschke, M. C. Schut, and A. Eiben, "Evolving behavioral specialization in robot teams to solve a collective construction task," *Swarm and Evolutionary Computation*, vol. 2, pp. 25–38, 2012.

[5] J. Gautrais, G. Theraulaz, J.-L. Deneubourg, and C. Anderson, "Emergent polyethism as a consequence of increased colony size in insect societies," *Journal of theoretical biology*, vol. 215, no. 3, pp. 363–373, 2002.

[6] O. Gigliotta, "Equal but different: Task allocation in homogeneous communicating robots," *Neurocomputing*, vol. 272, pp. 3–9, 2018.

[7] G. Pini, A. Brutschy, M. Frison, A. Roli, M. Dorigo, and M. Birattari, "Task partitioning in swarms of robots: An adaptive method for strategy selection," *Swarm Intelligence*, vol. 5, no. 3-4, pp. 283–304, 2011.

[8] G. Pini, A. Brutschy, G. Francesca, M. Dorigo, and M. Birattari, "Multi-armed bandit formulation of the task partitioning problem in swarm robotics," in *International Conference on Swarm Intelligence*, pp. 109–120, Springer, 2012.

[9] E. Ferrante, A. E. Turgut, E. Duéñez-Guzmán, M. Dorigo, and T. Wenseleers, "Evolution of self-organized task specialization in robot swarms," *PLoS Comput Biol*, vol. 11, no. 8, p. e1004273, 2015.

[10] L. Li, A. Martinoli, and Y. S. Abu-Mostafa, "Learning and measuring specialization in collaborative swarm systems," *Adaptive Behavior*, vol. 12, no. 3-4, pp. 199–212, 2004.

[11] J. Gomes, P. Mariano, and A. L. Christensen, “Dynamic team heterogeneity in cooperative coevolutionary algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 6, pp. 934–948, 2017.

[12] S. O’Donnell and R. L. Jeanne, “Forager specialization and the control of nest repair in polybia occidentalis olivier (hymenoptera: Vespidae),” *Behavioral Ecology and Sociobiology*, vol. 27, no. 5, pp. 359–364, 1990.

[13] N. Veerapen and G. Ochoa, “Visualising the global structure of search landscapes: genetic improvement as a case study,” *Genetic programming and evolvable machines*, vol. 19, no. 3, pp. 317–349, 2018.

[14] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne.,” *Journal of machine learning research*, vol. 9, no. 11, 2008.

[15] J. Adair, G. Ochoa, and K. M. Malan, “Local optima networks for continuous fitness landscapes,” in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1407–1414, 2019.

[16] J. W. Sammon, “A nonlinear mapping for data structure analysis,” *IEEE Transactions on computers*, vol. 100, no. 5, pp. 401–409, 1969.

[17] Y.-H. Kim and B.-R. Moon, “New usage of sammon’s mapping for genetic visualization,” in *Genetic and Evolutionary Computation Conference*, pp. 1136–1147, Springer, 2003.

[18] S. Risi, C. E. Hughes, and K. O. Stanley, “Evolving plastic neural networks with novelty search,” *Adaptive Behavior*, vol. 18, no. 6, pp. 470–491, 2010.

[19] F. Silva, P. Urbano, L. Correia, and A. L. Christensen, “odneat: An algorithm for decentralised online evolution of robotic controllers,” *Evolutionary Computation*, vol. 23, no. 3, pp. 421–449, 2015.

[20] R. Wang, J. Clune, and K. O. Stanley, “Vine: an open source interactive data visualization tool for neuroevolution,” in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1562–1564, 2018.

[21] S. Risi and K. O. Stanley, “Deep neuroevolution of recurrent and discrete world models,” in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 456–462, 2019.

[22] G. Ochoa and K. Malan, "Recent advances in fitness landscape analysis," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1077–1094, 2019.

[23] M. A. Contreras-Cruz, G. Ochoa, and J. P. Ramirez-Paredes, "Synthetic vs. real-world continuous landscapes: A local optima networks view," in *International Conference on Bioinspired Methods and Their Applications*, pp. 3–16, Springer, 2020.

[24] J. C. Helton and F. J. Davis, "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems," *Reliability Engineering & System Safety*, vol. 81, no. 1, pp. 23–69, 2003.

A Summary of Results

1 What are the fundamental features of the fitness landscape of a task specialisation problem?

Answer 1.1: Rewarding agents for partial retrieval of the resource creates a smoother fitness gradient, but the landscape is still largely flat.

2 When do centralised and decentralised solutions outperform one another in task specialisation problems?

3 What is the price of anarchy in task specialisation problems?