The Impact of Complexity Costs given Various Meta-Parameters

Project Proposal

Scott Hallauer
Department of Computer Science
University of Cape Town
Cape Town, South Africa
HLLSCO001@myuct.ac.za

Ryan Scott
Department of Computer Science
University of Cape Town
Cape Town, South Africa
SCTRYA002@myuct.ac.za

Tristan Joseph
Department of Computer Science
University of Cape Town
Cape Town, South Africa
JSPTRI002@myuct.ac.za

CCS CONCEPTS

• **Computing methodologies → Evolutionary robotics**; **Multi-agent systems**; *Neural networks.*

KEYWORDS

Collective robotics, evolutionary robotics, neuroevolution, morphological complexity, neural complexity

1 PROJECT DESCRIPTION

The field of evolutionary robotics has blossomed since its inception more than 20 years ago, with many sub-fields being created and matured. Through this research, various approaches to tackling optimisation problems have been developed.

Recently, there has been increased interest in the targeted evolution of low-complexity, high-performing robots [1, 2, 9]. The differing parameters of such experiments can be summarised into the following three categories: different definitions of robot complexity, different definitions of environment complexity, and different approaches and applications of the underlying evolutionary algorithm.

This segmentation of the problem is an issue because often an experiment's results (and, by extension, conclusion) may be greatly influenced by the choice of meta-parameters. As such, multiple studies may test similar hypotheses and yet produce varying results. These contradictory results are inconclusive and counterproductive towards the goal of providing research-based answers.

Our aim is to test three classes of meta-parameters in the question of how implementing a complexity cost on fitness affects the evolution of task-performing robots. In doing so, we evaluate whether previously reached conclusions still hold given different definitions of robot complexity, environment complexity and the evolutionary method itself.

2 PROBLEM STATEMENT

2.1 Research Hypothesis

The hypothesis being investigated in this project is phrased as follows:

> Imposing an evolutionary fitness cost on complexity (both morphological and neural) results in improved task performance in collective behaviour tasks regardless of robot (morphological and neural) and environment complexity definitions.

This hypothesis was motivated by previous results and research, as described in section 5, which suggest the claims to be generally true. In this project, we aim to extend on the results from related work in the literature, most notably those presented in the 2019 study by Nagar, Furman and Nitschke [9].

In biology, there are hypotheses which suggest that neural complexity in living organisms is a function of social and cooperative behaviour [5]. It is difficult to facilitate the study of these hypotheses as the resources required are often not feasible in the physical world. Therefore, the idea of simulating collaborative tasks in environments with evolving entities has grown in popularity. Simulations can be run at a higher magnitude and in a fraction of the time it would take to use physical variables.

In evolutionary robotics, resources are limited and the complexity of a robot body can result in significant computational overhead when task complexity becomes necessarily large. Robot configurations and evolutionary algorithms are constantly in pursuit of optimisation to ensure task completion in the most effective manner possible [4]. This naturally leads to the motivation of minimising complexity while maximising performance. Imposing a cost on complexity should provide the results necessary to make an informed conjecture about the legitimacy of its effects on both practical (collective robotics performance) and theoretical (social brain) hypotheses.

Testing this project's hypothesis will require the decomposition of these complexity definitions into three distinct categories: robot complexity, environment complexity and evolutionary algorithmic approaches. These will form three separate research objectives, which are given in section 2.2.

2.1.1 Robot Complexity. This refers to the structural intricacy of the robotic systems being placed in the simulated task environment. Two specific aspects of complexity are incorporated in this overarching term: morphological and neural complexity. Morphological complexity is related to actual configuration of a robot's physical body, including body shape as well as the number, type and arrangement of sensors on the body [1, 9]. On the other hand, neural complexity is related to the structure of a robot's controller (or, more colloquially, "brain") which is often encoded as an artificial neural network (ANN) [2]. Complexity metrics in this case include number of input/output nodes, number of hidden nodes and the connections between nodes.

2.1.2 Environment Complexity. This refers to both the design of the environment and then the interpretation of what makes an environment "complex" or difficult. There are a vast number of different environment complexity definitions utilised throughout the field of evolutionary robotics, even when taking into account that environments might be designed for differing robots. The problem here becomes that multiple experiments based on the same hypothesis can be run and reach different results, as in the example case of Nagar, Furman and Nitschke [9] reaching differing conclusions to Auerbach and Bongard [1]. It was postulated that this was due to differing definitions [9], and environment complexity could be one of them.

2.1.3 Evolutionary Algorithmic Approaches. This refers to the selection and combination of different processes and rules within the domains of evolutionary computation and artificial neural networks, specifically their intersection: neuroevolution. The idea of evolving an algorithm, or more specifically, evolving a neural network has born significant results [11]. The first approach to show real promise was NEAT, or NeuroEvolution of Augmenting Topologies. Over the years, various extensions and modifications have been created, each providing their own unique improvements [3, 7, 9]. It is clear that there are insights to be gained in the pursuit of treading new ground. The aim with this approach is to harness the benefit of exploring new methods and processes as well as to position their relative characteristics against more traditional models to further support and augment the overall hypothesis of this paper.

2.2 Research Objectives

This work aims to test if the previously obtained results achieved by Nagar, Furman and Nitschke [9] (that is, imposing an evolutionary fitness cost on complexity results in improved task performance in collective behaviour tasks) holds true under varied definitions of robot and environment complexity, as well as differing algorithmic approaches to the evolutionary process. It aims to give additional weighting to this previous conclusion, in the form of additional evidence and findings.

Therefore, this project's three research objectives are to test if imposing an evolutionary fitness cost on complexity results in improved task performance in collective behaviour tasks given:

1. Varied definitions of robot complexity.
2. Varied definitions of environment complexity.
3. Varied evolutionary algorithmic approaches.

3 PROCEDURES AND METHODS

To test our research hypothesis, we envision running multiple evolutionary robotics simulations using an extended version of the MASON multi-agent simulation library [8]. MASON is coded in Java, and any modifications to this extended version will be made by us, in Java. We will be attempting to enlist some help from the developers of the previously extended version of MASON, Danielle Nagar and Alexander Furman, in hopes that they can assist in our effective understanding of the code. The simulations will consist of collective construction tasks in which two objectives will be optimised through multi-objective neuroevolution: maximal task performance and minimal robot complexity (morphological and neural).

Each group member will analyse the evolutionary impact on task performance of altering one of three variables:

1. **Scott** (section 3.1)**:** Definitions of robot morphological and neural complexity.
2. **Ryan** (section 3.2)**:** Properties which influence environmental difficulty.
3. **Tristan** (section 3.3)**:** Multi-objective evolutionary algorithm approaches and extensions of NEAT.

We plan to agree on a standard definition of robot complexity, a standard environment set-up and a standard evolutionary algorithm. Each member will then run multiple simulations of the collective construction task – one without a complexity cost and multiple with a complexity cost where their variable of interest is varied (the other variables will be kept constant with the agreed upon standard approaches).

A possible extension to the project, should all members timeously finish their section, is running simulations of all possible combinations of the three different variables to produce a matrix of results.

3.1 Robot Complexity

Various definitions of morphological and neural complexity will be tested in the construction of a fitness function for use in the multi-objective neuroevolution algorithm. It is planned that previously tested complexity definitions from the literature will be compared in this project.

Capi and Kaneko [2] developed a definition for neural complexity (N) as being a function of the number input nodes (nr_i) and hidden nodes (nr_h) in the ANN for the robot controller. This can be simply written in an equation as follows:

$$N = nr_i + nr_h$$

For morphological complexity, Furman, Nagar and Nitschke [9] defined a function of the number of sensors (n) on a robot's body as well as the field of view (f_i) and range value (r_i) for each of the sensors. Their function can be expressed mathematically as follows:

$$M = 5 \times \sum_{i=1}^{n} \left(\frac{f_i - \wedge F_i}{\vee F_i - \wedge F_i} + \frac{r_i - \wedge R_i}{\vee R_i - \wedge R_i} \right)$$

Where $\vee F_i$ and $\wedge F_i$, and $\vee R_i$ and $\wedge R_i$, are the maximum and minimum possible values of f_i and r_i, respectively.

Therefore, as described above, the two robot complexity definitions which will be investigated in this project are:

1. **Neural complexity** measured as a function of the number of input nodes and hidden nodes in the robot controller's ANN.
2. **Morphological complexity** measured as a function of the number of sensors on the robot's body and the field of view and range value for each sensor.

3.2 Environment Complexity

The aim is to apply different definitions of environment complexity to the simulation while holding all other parameters constant. This will be done through varying the underlying simulated environment in which the robots evolve, applying two different philosophies: one utilising varying degrees of simulated friction and the other using a differently sized block approach. Both of these have been applied in the past with success [1, 9].

The main question here would be whether these differing environment designs impact the result; whether they reinforce or conflict with each other. Furman, Nagar and Nitschke [9] found that a complexity costs allows for reduced robot complexity in morphologies while retaining effective behaviour couplings, a result which implies that a complexity cost with complex environments curates reduced complexity. This directly contradicts with results obtained from Auerbach and Bongard [1] who found that a complexity cost on morphology along with increasingly complex environments results in a growing increased robot complexity.

The two environment definitions used will be as follows:

1. A simple **block and destination environment**. Each block will have differing sizes, some requiring more than one robot to move it. These will have to be moved to a destination zone. Environment complexity here is defined as a function of the number of blocks and the size of the blocks. The blocks have three sizes: small, medium and large, and this also defines the number of robots needed to push the blocks. This is the environment complexity definition used by Furman, Nagar and Nitschke [9].
2. The **inclusion of friction**, or lack thereof, in the environment. As above, but with the block sizes and numbers remaining simple while the number of patches of ice increases. When a robot rolls over a patch of ice, they will ‘slide' forward without control, and this is also in the case of moving blocks. Environment complexity here is defined as a function of the number of blocks and the number of ice patches. The block sizes will all be of size small for this definition.

3.3 Evolutionary Algorithmic Approaches

The overarching goal of this paper is to observe the result of imposing a fitness cost on complexity. It is for this reason that we will use NEAT-M (single-objective evolution) and NEAT-M-MODS (multi-objective behaviour-morphology evolution) as the baseline of our experiments. This approach was previously shown to provide a useful juxtaposition for comparing the results of imposing or not imposing a cost on complexity [9].

In the pursuit of better performance and a new approach, we propose the extension of these two algorithms in order to adapt for a rapidly moving optimum. In a previous paper, it was shown that NEAT performs poorly on problems where the optimum of the fitness function changes rapidly between generations. The researchers constructed a novel algorithm, DynNEAT (NeuroEvolution of Augmenting Topologies for Dynamic Fitness Functions), which extends NEAT by changing the size of each species based on its historical performance. The results showed that DynNEAT performs similarly to NEAT for slow or statically moving optimum, but outperforms it on problems with a rapidly moving optimum [7].

To illustrate this idea, we elaborate on how each approach chooses the size of their species. In NEAT, size is determined only by the average fitness value of individuals from the previous generation. In highly dynamic problems this value can change significantly between generations, which can be highly detrimental to progress and performance. For NEAT, species sizes in generation $i + 1$ are allocated proportionally to $s_{NEAT}(i + 1)$, an average fitness of all individuals belonging to the given species in the previous generation:

$$s_{NEAT}(i+1) = \frac{\sum_{j=1}^{N_i} f_{ij}}{N_i}$$

DynNEAT proposes that the size of a species is based not just on the previous generation, but on t previous generations. Species sizes in generation $i + 1$ are allocated proportionally to $s_{DynNEAT}(i + 1)$, the maximum average fitness of the last t generations:

$$s_{DynNEAT}(i+1) = \max_{j=i-t+1}^{i} \frac{\sum_{k=1}^{N_j} f_{jk}}{N_j}$$

With that, we propose two experiments that this study will evaluate:

1. The first experiment will compare a single-objective **NEAT-M** algorithm that will aim to maximise task performance (the number of blocks placed correctly) for a collective group of simulated robots, versus a multi-objective **NEAT-M-MODS** algorithm that will include the addition of a second objective: to minimise robot complexity (morphological or neural complexity, as defined in section 3.1).
2. A second experiment will evaluate **NEAT-M** versus **DynNEAT-M-MODS**, where a maximum average fitness is taken from t previous generations of species for the multi-objective method.

4 ETHICAL, PROFESSIONAL AND LEGAL ISSUES

We will be using a modified version of the free and open-source MASON simulator, and as such do not anticipate any legal issues. Additionally, it is for this reason that we intend to follow the rules and guidelines outlined for working with Open Source Software (OSS) [6]. This means utilising the framework in a manner that is professional and ethical. In other words, we plan to extend and apply it in such a way that integrity and performance is maintained.

5 RELATED WORK

The research field of collective robotics, and specifically the subfield of evolutionary robotics, has seen strides in progress since its inception in the early 1990s. Recently, focus has intensified in the investigation of evolutionary complexity costs and the associated effect on robot task performance. There are practical benefits to the development low-complexity robot systems, including the potential for reduced computation time [2]. Furthermore, the methodology of evolutionary simulation provides a novel approach to investigating previously inaccessible questions in evolutionary theory, such as the origin of complexity [5, 10].

A 2008 study by Capi and Kaneko [2] investigated the multi-objective neuroevolution of low-complexity neural controllers for robots required to perform multiple, simultaneous tasks. The results demonstrated that such an approach can successfully evolve simple neural controllers which are effective in performing tasks.

Similarly, Furman, Nagar and Nitschke [9] recently researched the impact of a morphological complexity cost on task performance. In this study, single- and multi-objective neuroevolution strategies were compared. The single-objective neuroevolution simulations (using NEAT-M) focussed on maximising task performance, whereas the multi-objective simulations (using NEAT-M-MODS) focussed on both maximising task performance and minimising morphological complexity. It was found that, in simpler task environments, both simulations produced robots equally effective in task performance. However, contrary to previous research [1], in difficult task environments, robots evolved with a cost on morphological complexity achieved higher task performance (with simpler morphologies) than those evolved without such a cost. These surprising results provide the key motivation behind the research objectives outlined for our project.

6 ANTICIPATED OUTCOMES

6.1 Research

We anticipate this work will produce additional evidence in line with the previously found conclusion by Furman, Nagar and Nitschke [9]. The reality is that this may not be the case and, in such a situation, we will at least demonstrate for which parameters the hypothesis (that a cost in complexity improves robot task performance) is true.

6.2 Framework

This work will produce a further extended and well-documented MASON collective robotics framework. This will hopefully be reusable for future related research projects and provide ease of use through comprehensible documentation.

6.3 Impact

We hope this project will create even greater weighting to the hypothesis previously established by Furman, Nagar and Nitschke [9]. If our findings detail otherwise, we will at least be able to show for which parameters our hypothesis is true, thereby furthering the field of evolutionary collective robotics. We will also be releasing a documented, extended version of the previously extended MASON simulator, which we hope will be reused in future research.

6.4 Key Success Factors

- Successful understanding and extension of the MASON simulator.
- Extension of the framework in a manner that allows for all individual experiments to be successfully compiled and independently run.
- Production of useful results by the experiments which are sufficient for making valid conclusions.

7 PROJECT PLAN

7.1 Risks

The risks for this project are outlined in a risk matrix that describes strategies for risk management and mitigation, as presented in appendix B. Overall, the risks observed can pose significant harm to the project, however, we do not anticipate their probability to be very likely.

7.2 Timeline

Our project runs from 13 May until 8 October, where we submit our reflection paper. See appendix A for a Gantt chart detailing the full project schedule.

7.3 Resources Required

We will utilise our own personal computers with Visual Studio Code (or a similar free IDE) for building on the extended MASON framework. If this fails, we will use the UCT computer labs with their assortment of provided IDEs. In terms of human resources, we would greatly appreciate the help of the previous extenders of the MASON code, Danielle Nagar and Alexander Furman, but will make do adequately without this assistance. In order to run the simulations themselves, we will require access to the UCT HPC cluster.

7.4 Deliverables

The most central deliverable of the project will be an academic paper presenting the results of our research. This will include detailing if, and for which parameters, imposing a cost on complexity results in improved task performance in collective robotic task environments. We will also be delivering our documented extensions to the previously extended MASON code.

A summarised list of all notable deliverables (some already completed) is presented below:

- Three literature reviews
- Project proposal
- Presentation of the project proposal
- Software feasibility demonstation
- Draft paper
- Final paper detailing the experimental findings and process
- Further extended MASON code from previous extensions, with documentation
- Project poster
- Project website
- Reflection paper

7.5 Milestones

Notable milestones include the revised project proposal to be submitted on the 10^{th} of June, preliminary code adaptation to be done by the 1^{st} of July (after which the experiments will be run) and final code and raw data to be submitted on the 2^{nd} of September. The paper should be worked on from end of June until a final draft is submitted on the 16^{th} of August. The final paper will be submitted on the 26^{th} of August. The project will then be demonstrated in the fortnight beginning 2 September and an open evening showcasing the work will take place on the 15 October. Additional milestones can be found in appendix A.

7.6 Work Allocation

As previously mentioned, work will be split amongst the team members in the following way: Scott Hallauer will focus on observing the results of applying different definitions for complexity, Ryan Scott will focus on the change in performance when altering environment complexity, and Tristan Joseph will be constructing two alternative algorithms for neuroevolution by adjusting their goals and objectives. All members will collaborate at various stages and these individual research areas will be combined to produce one conclusive observation that is augmented by the results and insights from each respective experiment.

REFERENCES

[1] Joshua E Auerbach and Joshua C Bongard. 2012. On the relationship between environmental and morphological complexity in evolved robots. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*. ACM, 521–528. https://doi.org/10.1145/2330163.2330238

[2] Genci Capi and Shin-ichiro Kaneko. 2008. Evolution of low-complexity neural controllers based on multiobjective evolution. *Artificial Life and Robotics* 12, 1-2 (2008), 53–58. https://doi.org/10.1007/s10015-007-0441-0

[3] Jeff Clune, Kenneth O Stanley, Robert T Pennock, and Charles Ofria. 2011. On the performance of indirect encoding across the continuum of regularity. *IEEE Transactions on Evolutionary Computation* 15, 3 (2011), 346–367.

[4] Stephane Doncieux, Nicolas Bredeche, Jean-Baptiste Mouret, and Agoston E Gusz Eiben. 2015. Evolutionary robotics: what, why, and where to. *Frontiers in Robotics and AI* 2 (2015), 4.

[5] Robin IM Dunbar. 1998. The social brain hypothesis. *Evolutionary Anthropology: Issues, News, and Reviews: Issues, News, and Reviews* 6, 5 (1998), 178–190.

[6] Frances Grodzinsky, Keith Miller, and Marty Wolf. 2003. Ethical issues in open source software. *Journal of Information, Communication and Ethics in Society* 1 (11 2003), 193–205. https://doi.org/10.1108/14779960380000235

[7] Peter Krčah. 2012. Effects of speciation on evolution of neural networks in highly dynamic environments. In *International Conference on Learning and Intelligent Optimization*. Springer, 425–430. https://doi.org/10.1007/978-3-642-34413-8_39

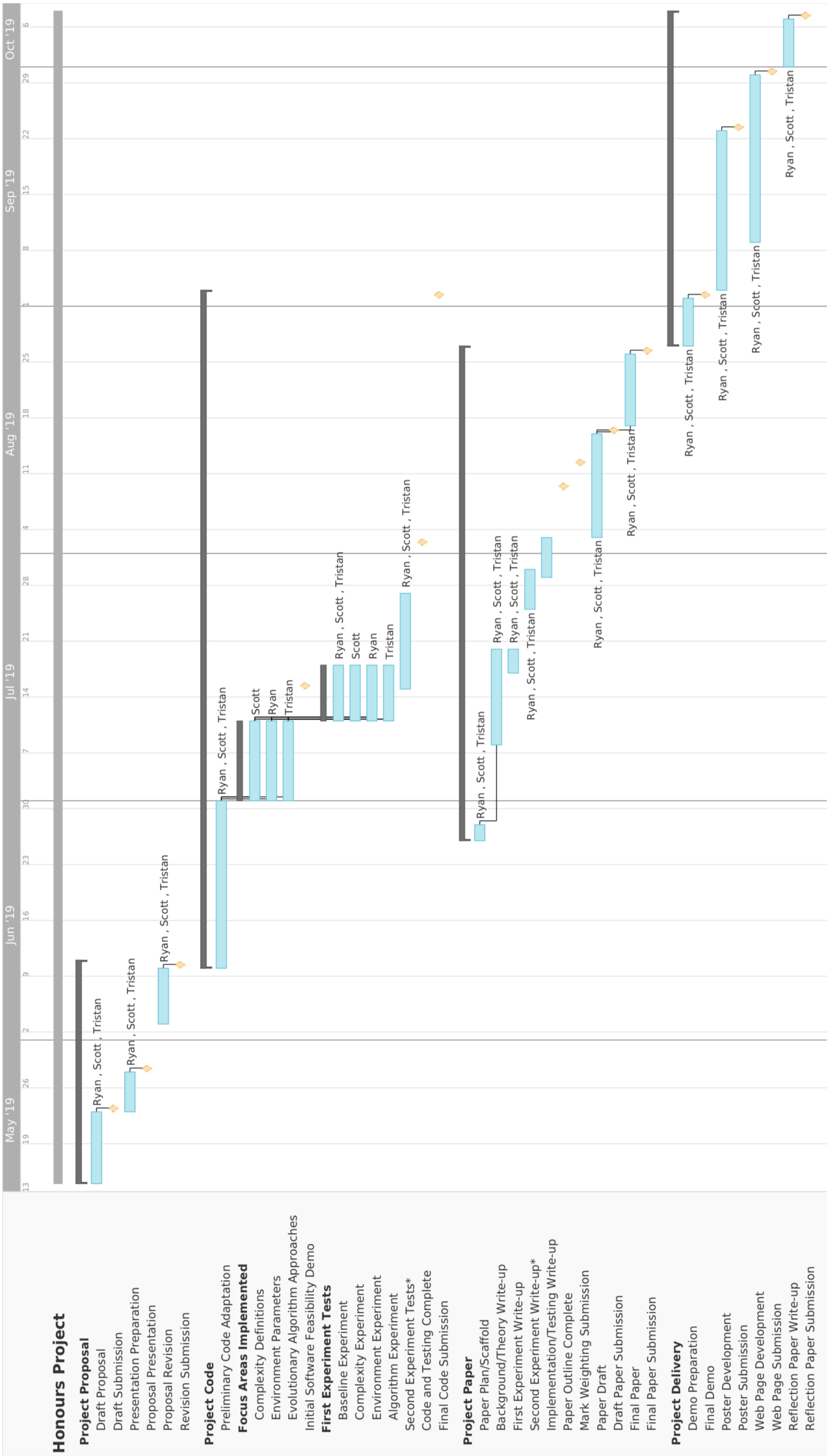
[8] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan. 2005. Mason: A multiagent simulation environment. *Simulation* 81, 7 (2005), 517–527.

[9] Danielle Nagar, Alexander Furman, and Geoff Nitschke. 2019. The Cost of Complexity in Robot Bodies. In *IEEE Congress on Evolutionary Computation, Wellington, New Zealand.*

[10] Ana Navarrete, Carel P van Schaik, and Karin Isler. 2011. Energetics and the evolution of human brain size. *Nature* 480, 7375 (2011), 91.

[11] Kenneth O Stanley and Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation* 10, 2 (2002), 99–127.

A PROJECT PLAN



B RISK MATRIX

Risk	Probability	Impact	Consequence	Mitigation	Monitoring	Management
Adapted MASON code proves too difficult or unwieldy.	4	9	Severe project disruption/delay. Cannot begin research until code is fully understood.	Perform long period of preliminary code testing and checking. Contact previous code users.	Regular meetings with supervisor and regular contact with other group members to demonstrate understanding.	Consult supervisor on steps forward. Outsource code optimisation. Look for alternative software suites.
Physical infrastructure (UCT HPC cluster) is under maintenance or cannot be accessed.	1	8	Significant project disruption, cannot run tests within a reasonable time frame.	Ensure in advance that cluster login accounts have access.	Check status of cluster regularly and check maintanance schedules. Test logins.	Seek possible alternative cluster for use (CHPC) or consult supervisor for steps forward.
Team member drops out of project.	1	3	Research objectives are largely independent with no coupling and so other members can continue with research.	Maintain good communication within team and social well-being. Attempt to maintain good mental health.	Maintain regular contact with team members and ensure regular updates of what others are working on.	Reallocate work and ensure delivery of remaining research objectives.
Research objective scope creep.	3	6	Unnecessary wasted research time causing infringement on deadlines.	Ensure regular communication of all members and supervisor to prevent scope creep.	Document and report activities of team members to measure if it is in scope.	Focus on scope and redetermine what can be accomplished before deadline.
Inability to meet final deadline.	1	10	Unable to deliver a project report, resulting in failing the course.	Ensure team workings are in scope and maintain good reporting and communication between team members.	Team members must monitor each other's progress and work to ensure everyone is on schedule.	Consult with supervisor to find way forward. Rescale project to allow for reaching deadline or cut back scope.
Furman and Nagar unreachable or unhelpful.	5	4	Increased difficulty in preliminary code adaptation leading to increased duration of this phase, possible delays.	Attempt contact through various means of communication until we get through or means are exhausted.	Monitor communication between the team and Furman, Nagar.	Consult supervisor for possible guidance in understanding the previously made code extensions. Allocate additional time to preliminary code adaptation phase of project.